REVERSE: Reinforcing Evidence Verification and Search for Agentic Image geo-localization

Yong Li[1,2,3,*] **Furong Jia**[1,*] **Dacheng Yin**[3,†] **Kang Rong**[3]
Fengyun Rao[3] **Jing LYU**[3] **Fan Zhang**[2,†]
[1]Peking University
[2]The Hong Kong University of Science and Technology
[3]WeChat Vision, Tencent Inc

Abstract

Image geo-localization aims to determine where a photograph was taken, a task that often requires more than recognizing visible landmarks. Human experts typically solve it through an iterative workflow: they inspect informative regions, form location hypotheses, seek external evidence, and revise their judgments as new clues appear. Existing methods only partially capture this process: direct prediction methods bypass evidence acquisition altogether, while retrieval-augmented methods introduce external evidence but usually provide limited supervision on the intermediate decisions of where to search, how to query, and how to filter noisy results. We present REVERSE, a framework that reinforces the interplay between evidence search and verification to enable multi-turn agentic reasoning. REVERSE teaches three intermediate decisions: where to look, what to query, and what evidence to trust. To support this, we construct tool-grounded trajectories with annotated region selections, search observations, and geo-informative evidence labels, and introduce process rewards for visual grounding, query utility, and evidence discrimination. An offline search cache makes retrieval observations stable and reusable during reinforcement learning, enabling dense supervision over noisy search results. With a 4B backbone, REVERSE achieves 48.3% accuracy at 25 km on Im2GPS3k. Code is available at https://github.com/yonglleee/REVERSE.

1 Introduction

Image geo-localization estimates the geographic location of a photograph from its visual content [1, 2], with applications in navigation, urban analysis, and remote sensing. Prior methods either partition the Earth into cells and train classifiers [3–6], match the image against geo-tagged databases [7–9], or use large vision-language models to reason toward coordinate predictions [10–12]. All commit to a single prediction without querying or verifying external evidence, so performance is bounded by what the model already knows when the scene contains unfamiliar landmarks or ambiguous visual cues.

Recent agentic methods address this by allowing models to interact with tools across multiple turns [13–16], but tool access alone does not guarantee reliable geo-localization. Figure 1(A) shows two representative failures: a model that crops the wrong region submits it to image search and follows misleading results to a wrong prediction; a model that retrieves the right candidate cannot separate it from distractors and again predicts incorrectly. Both failures reveal that agentic geo-localization hinges on three intermediate decisions: *where to look* (identifying which visual cues are geographically informative), *how to query* (formulating a query that retrieves relevant external evidence), and *what to trust* (discriminating genuine evidence from visually similar distractors). Yet

*Equal contribution.
†Corresponding authors: Dacheng Yin and Fan Zhang. Fan Zhang: fanzhanggis@pku.edu.cn.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

none receives explicit supervision under standard final-coordinate training, and our approach jointly exposes them as explicit optimization targets.

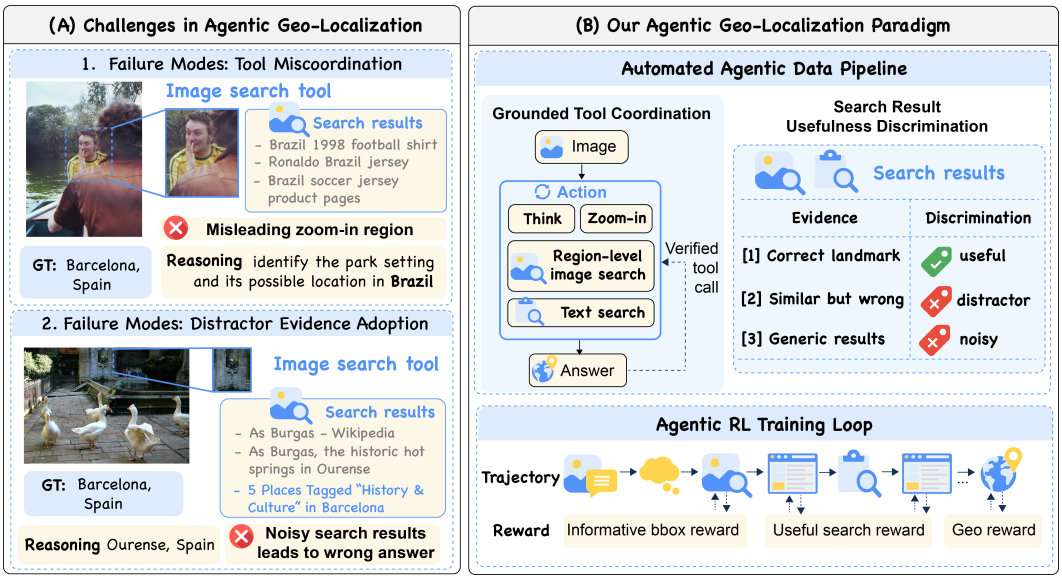


Figure 1: (A) Agentic geo-localization fails when a model crops the wrong region and follows misleading results, or retrieves the right candidate but cannot filter distractors. (B) REVERSE fixes both: spatial supervision grounds region selection, evidence labels annotate search results, and process rewards train the model via agentic RL.

We present REVERSE (**Re**inforcing **E**vidence **Ver**ification and **Se**arch for Agentic Image Geo-localization), which treats these three decisions as explicit optimization targets. If the model must make observable choices at each step (*where to look*, *how to query*, *what to trust*), each choice can be supervised directly. We build a data pipeline that generates teacher trajectories annotated with region selections and per-result teacher/reference labels, providing process supervision for decisions that final-coordinate training leaves implicit. We design process rewards for each decision and train offline against a pre-built search cache, so the full agentic reasoning loop is directly optimizable.

We evaluate on Im2GPS3k [17] and YFCC4k [18]. With a 4B model, REVERSE reaches 48.3% accuracy at 25 km on Im2GPS3k, outperforming all retrieval-augmented baselines and matching models more than an order of magnitude larger.

In summary, our contributions are threefold.

- We introduce REVERSE, an agentic image geo-localization framework that reinforces evidence search and verification by jointly aligning tool use with spatial region selection, query formulation, and evidence discrimination across multi-turn reasoning chains.
- We build a tool-grounded data generation pipeline that annotates not only final locations, but also region selections, search observations, and geo-informative evidence labels, enabling supervision over where to search, how to query, and what to trust during reasoning.
- We perform agentic reinforcement learning with a composite reward that combines geographical accuracy, format compliance, tool-use quality, and evidence discrimination, allowing a 4B vision-language model to achieve strong performance on geo-localization benchmarks.

2 Related work

2.1 Image geo-localization

Image geo-localization aims to predict the geographic location of a given image and has broad applications in urban analysis, navigation, remote sensing, and geospatial data mining. It has long been studied as a visual recognition problem [19], with early methods predicting locations

through geographic partitioning, hierarchical classification, or retrieval from geo-tagged image databases [4–6, 17]. Later work improves visual-location modeling with stronger backbones, and contrastive image-location alignment [20, 21, 7, 8]. With the emergence of large vision-language models, recent methods increasingly cast geo-localization as a language-mediated reasoning problem, where visual cues can be described, compared, retrieved, and verified using external knowledge. Retrieval-augmented approaches ground predictions with candidate images, geographic databases, or web evidence [9, 10, 22], showing the value of external database beyond closed-book visual prediction. Beyond retrieval, recent LVLM-based methods further strengthen geographic reasoning through reinforcement learning [23, 11, 12]. More recent agentic methods further explore multi-agent collaboration, map-based interaction, reinforced geographic reasoning, and tool-augmented hypothesis verification [15, 14, 13]. These works demonstrate the value of external evidence and multi-step reasoning for image geo-localization, but the search and verification process itself is rarely an explicit optimization target.

Candidate retrieval and evidence verification. Visual place recognition supplies candidate locations through visual matching against a reference database. Such retrieval can complement region-level web search by improving the candidate set available to an agent. G3 [10] emphasizes geotagged retrieval, while SpotAgent [13] and GeoVista [16] expose tool-assisted reasoning. REVERSE makes crop selection and explicit result filtering trainable through spatial and evidence-discrimination rewards. This interface also permits future integration of VPR candidate generators while retaining an explicit evidence-verification decision.

2.2 Multimodal agentic reasoning

Large vision-language models are increasingly extended from static perception to interactive reasoning with external tools. Early tool-augmented systems connect foundation models with visual modules, program execution, web search, and structured environments, showing that complex multimodal tasks often require iterative perception, action, and evidence aggregation [24–28]. More recent work further internalizes such behaviors through cold start trajectories and reinforcement learning, enabling models to decide when to invoke tools, how to combine visual operations with textual search [29–34]. These studies indicate a broader shift from static multimodal understanding toward long-horizon evidence acquisition. However, most existing settings focus on general visual question answering or open-domain research tasks, where evidence only needs to support a textual answer. Image geo-localization requires spatially and geographically discriminative evidence for coordinate prediction, making reliable search and verification particularly challenging.

3 Method

REVERSE recasts image geo-localization as an iterative process of visual inspection, evidence retrieval, and hypothesis verification. As illustrated in Figure 1, effective agentic geo-localization requires the model to make three intermediate decisions: *where to look*, *how to query*, and *what to trust*. Our method is organized around three corresponding components: an agentic geo-localization framework that defines tool-mediated reasoning (Section 3.1), a data generation pipeline that constructs trajectories and evidence labels for these decisions (Section 3.2), and a three-stage post-training procedure (SFT, Agentic Cold Start, and Agentic RL) that converts these signals into model behavior (Section 3.3).

3.1 Agentic geo-localization framework

Expert human geo-localization is rarely a single visual matching step. A human expert first inspects potentially informative visual cues, forms tentative geographic hypotheses, searches for external evidence, and then revises the hypothesis based on the reliability of the retrieved clues. We formulate image geo-localization as a multi-turn multimodal reasoning problem that follows the same cognitive loop. Given a query image I, the agent produces a trajectory

$$\tau = (I, a_1, o_1, \ldots, a_T, o_T, \hat{y}), \tag{1}$$

where a_t denotes the action at turn t, o_t denotes the corresponding observation, and $\hat{y}$ is the final predicted location. At each turn, the agent first generates a `<think>` trace and then either calls one

tool or outputs a final answer. We restrict each response to at most one tool call so that the effect of each action can be attributed clearly during training. After a bounded number of turns, the agent must produce the final prediction in the format

$$\texttt{<answer> country, city, latitude, longitude </answer>}. \tag{2}$$

A central design choice is the `<useful>` tag. Search tools usually return a ranked list of results, only some of which provide reliable geographic evidence. After receiving image search or text search results, the agent is required to output a tag such as

$$\texttt{<useful>[1, 3]</useful>},$$

where the indices refer to retrieved results that the model judges to be geo-informative. An empty tag, `<useful>[]</useful>`, indicates that none of the returned results should be trusted. This explicit evidence selection interface makes the model's judgment observable and allows the reward to evaluate whether the agent has learned *what to trust*. The agent is equipped with three tools.

Zoom tool. The zoom tool takes a normalized bounding box $b = (x_1, y_1, x_2, y_2)$ and returns a cropped, resized view of the selected region. It helps the agent inspect fine-grained visual cues that may be too small or ambiguous in the full image, such as signs, road markings, inscriptions, architectural details, and distant landmarks. Zoom exposes region selection as an explicit decision, allowing the agent to inspect geographically informative visual cues.

Image search tool. The image search tool performs region-level reverse image search. Instead of submitting the full image, the agent selects a bounding box and searches the cropped region. This design is important because full-image retrieval can be dominated by salient but geographically irrelevant content, while a carefully selected region can expose distinctive landmarks, storefronts, signs, or local visual patterns. The tool returns a ranked list of results with titles, source domains, and links, which the agent must then filter using the `<useful>` tag.

Text search tool. The text search tool accepts a natural-language query and returns web snippets. It complements visual search by verifying candidate place names, landmarks, addresses, or textual clues inferred from previous observations. Together, the three tools cover the full reasoning loop: where to look in the image, how to query external sources, and what retrieved evidence to trust.

3.2 Data generation pipeline

Training an agent for geo-localization requires more than final coordinate labels. The environment must expose learnable signals for the full cognitive loop: *where to look* through precise crop selection, *how to query* through text and image search actions, and *what to trust* through evidence discrimination. We construct such an environment using Kimi-K2.6 as a teacher model (Figure 2).

Trajectory generation and filtering. We source training images from the MP-16 Pro dataset and prompt Kimi-K2.6 to solve each image using the same agentic tool interface as the student model. The teacher receives the image and task prompt without the reference location. It produces multi-turn trajectories containing reasoning traces, zoom calls, region-level image search calls, text search queries, `<useful>` tags, and final answers. We filter out inaccessible images, failed teacher trajectories whose final prediction is too far from the ground truth, tool-free trajectories, and trajectories requiring excessive intervention. This yields a set of valid teacher trajectories that provide demonstrations of how visual inspection, search, and verification can be coordinated.

Search result and crop region annotation. Raw teacher outputs leave two annotation gaps that we close with a second pass using Kimi-K2.6. First, evidence labels are incomplete: for each cached image search call, the teacher judges whether each returned result provides concrete geographic evidence for the target image, using result indices cited in the reasoning trace as initial signals. Missing usefulness annotations are adjudicated against the returned candidates and the reference location during annotation, separately from trajectory generation. This produces per-result binary labels (`is_geo_useful`) that later drive the evidence discrimination reward. Second, a large fraction of image search calls use full-image bounding boxes, which weaken the *where to look* signal because a model can achieve high spatial overlap without localizing any specific cue. We prompt the teacher to redraw these as tighter crops around the most geo-informative region, keeping the original search results intact. The corrected boxes serve as teacher/reference spatial targets for reward computation. Appendix G separates trajectory filtering, process-label construction, and diagnostic reference labels.

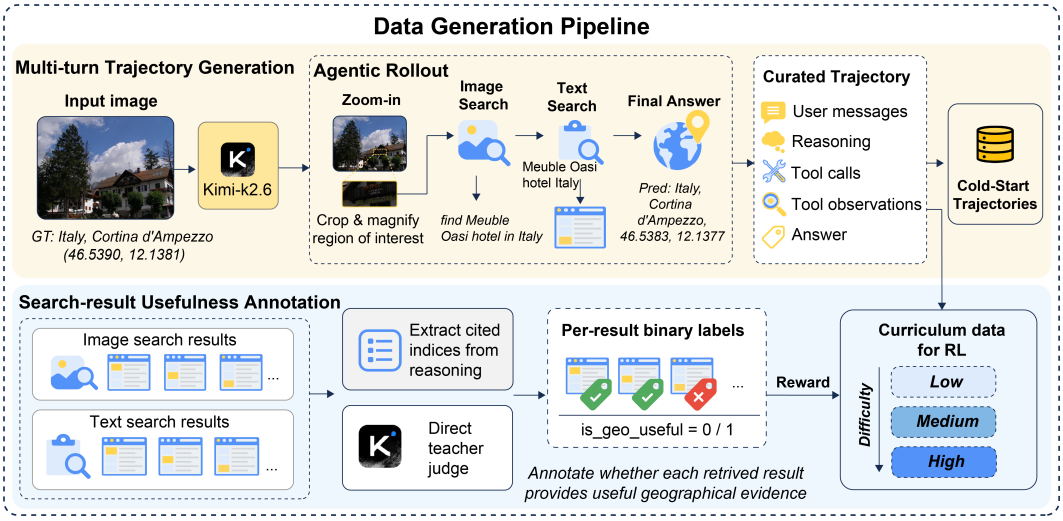


Figure 2: **Data generation pipeline.** Kimi-K2.6 generates multi-turn geo-localization trajectories over MP-16 Pro images using live search APIs. Trajectories undergo quality filtering, geo-informative label annotation, and bounding box re-annotation to correct full-image crops. The resulting dense annotations populate an offline cache that supports API-free RL training.

Offline search cache. We materialize all image search and text search observations into offline caches. Each image search entry stores the annotated bounding box, geo-informative positive results, and non-informative negatives from the same ranked list—the latter are deliberately hard since they come from the same query and are often visually or semantically related to the true location. During RL rollout, model-generated image-search boxes are matched to cached entries by IoU; text queries are matched by token-level Jaccard similarity. This keeps retrieval observations stable and reusable across training steps while preserving the noise structure of real search results.

Curriculum construction. The evidence discrimination reward requires every search call in a trajectory to carry usefulness labels. We retain trajectories meeting this condition, plus any that make no search calls at all, to form the full-curriculum set. A separate easy-curriculum subset restricts further to examples where the teacher's localization error is small. RL begins on this easier subset to stabilize tool-use behavior, then continues on the full-curriculum set for harder, more diverse samples.

3.3 Three-stage training pipeline

As illustrated in Figure 3, REVERSE trains in three stages.

Stage 1: SFT. We fine-tune Qwen3-VL-4B-Instruct [35] on 4M geo-tagged images from MP16-Pro [10], training the model to output coordinates directly from the image with no tools or reasoning chain, the goal is to build a strong geographic memory before introducing any agentic behavior.

$$\mathcal{L}_{\text{coord}} = -\sum_{t} \log p_\theta(y_t \mid I, y_{<t}). \tag{3}$$

Here y_t is a token in the coordinate target.

Stage 2: Agentic Cold Start.

To initialize the correct interaction protocol, we fine-tune Qwen3-VL-4B-Instruct on 4,427 curated teacher trajectories using standard autoregressive modeling:

$$\mathcal{L}_{\text{agent}} = -\sum_{t} \log p_\theta(z_t \mid I, z_{<t}, o_{<t}). \tag{4}$$

where z_t denotes an assistant output token and $o_{<t}$ are the tool responses available before that token. This stage aligns the model to the tool-call format, structured output tags, and basic evidence selection behavior before RL exploration begins.

Stage 3: Agentic RL. We then optimize the cold start model with GRPO [36]. For each image, the model samples multiple trajectories under the same tool environment and receives the composite reward defined in Section 3.4. The update uses group-relative advantages, which compare rollouts generated for the same image and therefore do not require a separate critic model.

$$\mathcal{L}_{\mathrm{GRPO}}(\theta) = -\mathbb{E}_{\tau\sim\pi_\theta}\left[A(\tau)\log\pi_\theta(\tau \mid I)\right] + \beta_{\mathrm{KL}}\, D_{\mathrm{KL}}(\pi_\theta\|\pi_{\mathrm{ref}}), \tag{5}$$

where $A(\tau) = (R(\tau) - \mu_g)/\sigma_g$, π_{ref} is the cold start reference policy, and β_{KL} keeps the policy close enough to maintain format and tool-use stability. RL follows the curriculum described above, moving from the easy subset to the full-coverage dataset.

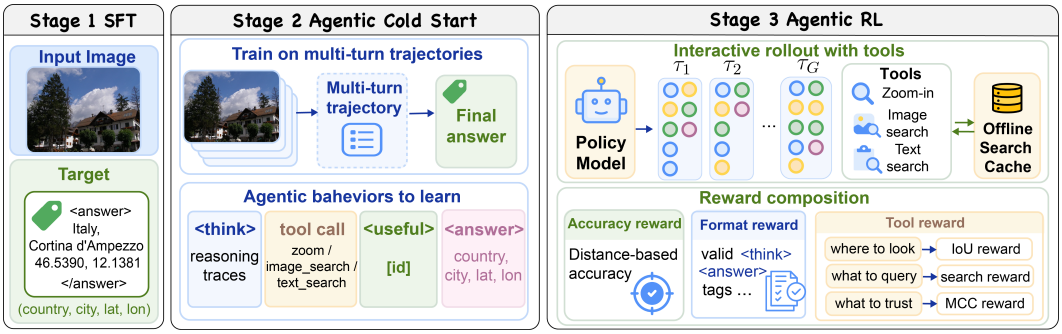


Figure 3: **REVERSE training pipeline.** The model first learns direct image geo-localization through SFT, then acquires multi-turn tool-use behavior from cold start trajectories, and finally undergoes agentic RL with offline search observations. The reward combines geographical accuracy, format compliance, and process-level signals for where to look, how to query, and what to trust.

3.4 Process reward

The reward is designed to align the model with both the final task objective and the intermediate decisions that make agentic geo-localization reliable. We decompose the trajectory reward into three terms:

$$R(\tau) = \alpha \cdot r_{\mathrm{geo}} + \beta \cdot r_{\mathrm{fmt}} + \gamma \cdot r_{\mathrm{tool}}, \tag{6}$$

where hyperparameters α, β, γ control the balance between final geo-localization accuracy, strict format compliance, and intermediate tool-use quality. Default values for all reward hyperparameters are provided in Appendix C.

Geographical reward. The geographical reward r_{geo} evaluates the final predicted location. We compute the Haversine distance $d(\hat{y}, y)$ between the predicted and ground-truth coordinates, then map it to the standard geo-localization thresholds of 1, 25, 200, 750, and 2500 km. Predictions within these thresholds receive decreasing scores from street-level to continent-level accuracy, while parse failures and predictions beyond 2500 km receive zero reward.

Format reward. The format reward r_{fmt} encourages stable and interpretable trajectories. A full score is assigned when the response contains valid `<think>`, `<useful>`, and `<answer>` tags in the required format. Partial credit is given when the reasoning and final answer are valid but evidence selection is missing. Invalid formatting receives zero reward.

Tool process reward (r_{tool}). To supervise the multi-step search process, we provide per-turn rewards that are summed and clipped to prevent reward inflation from repetitive tool usage:

$$r_{\mathrm{tool}} = \mathrm{clip}\left(\textstyle\sum_t r_t,\ -0.5,\ 1.0\right). \tag{7}$$

The turn-level reward r_t encompasses execution rewards and a discrimination reward:

- *Image search execution:* Incentivizes precise crop selection via a continuous IoU reward, $r_t^{\mathrm{img}} = \lambda_{\mathrm{iou}} \cdot \mathrm{IoU}(b_{\mathrm{pred}}, b_{\mathrm{gt}}) \cdot \mathbb{I}[\mathrm{IoU} \geq \tau]$.
- *Text search execution:* Provides a fixed base reward $r_t^{\mathrm{txt}} = \lambda_{\mathrm{base}}$ per valid query.
- *Zoom execution:* Imposes a penalty $-\delta$ for degenerate bounding boxes.

MCC discrimination reward. A core challenge in training agents to consume search results is reward hacking: if the evaluation metric for evidence selection is strictly non-negative (e.g., Accuracy or F1 score), models rapidly learn to indiscriminately tag all retrieved results as `<useful>` to guarantee a positive reward. To prevent this, we introduce a Matthews Correlation Coefficient (MCC)-based discrimination reward. After any turn receiving search results, the model's `<useful>` tag predictions are evaluated against the geo-informative ground truth labels. We compute MCC as:

$$\mathrm{MCC} = \frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}, \tag{8}$$

assigning the reward $r_t^{\mathrm{mcc}} = \lambda_{\mathrm{mcc}} \cdot \mathrm{MCC}$. Unlike F1, MCC operates in $[-1, 1]$ and strictly penalizes both false positives and false negatives. Crucially, it evaluates to zero for constant predictors (e.g., "mark all as useful"), completely eliminating the hacking vector and forcing the model to genuinely learn *what to trust*.

4 Experiments

We evaluate REVERSE on two standard geo-localization benchmarks: Im2GPS3k [17] (2,997 geotagged images sampled worldwide) and YFCC4K [18] (4,536 images sampled from YFCC100M, spanning heterogeneous scenes). We report accuracy at 1, 25, 200, 750, and 2500 km; an unparsed prediction counts as incorrect. Baselines span classification-based methods (PlaNet, CPlaNet, ISNs, GeoToken), retrieval-based methods (GeoCLIP, Img2Loc, PIGEON), VLM-based methods (G3, GeoBayes,Geo-R), plus untuned Qwen3-VL-4B/8B as zero-shot references. REVERSE is built on Qwen3-VL-4B-Instruct and trained in three stages on NVIDIA A800 GPUs; per-stage GPU counts and full hyperparameters are in Appendix D. We also conduct ablation studies on training stages, process reward components, and tool combinations.

4.1 Main results

Table 1 compares REVERSE to prior methods on Im2GPS3k and YFCC4k.

Im2GPS3k. REVERSE achieves 48.3% at 25 km and 22.5% at 1 km on Im2GPS3k, outperforming all prior methods at fine-grained thresholds. Against Geo-R [11], the strongest comparable VLM-based method, REVERSE gains 6.8 points at 25 km, from 41.5% to 48.3%. A 4B model trained with REVERSE also surpasses untuned Qwen3-VL-8B at both thresholds: 22.5% vs. 10.5% at 1 km and 48.3% vs. 37.2% at 25 km, demonstrating the value of targeted process training with a 4B backbone. Relative to direct SFT, REVERSE improves 25 km accuracy by 5.8 percentage points. The paired analysis in Appendix H connects this gain to 388 corrected SFT predictions.

YFCC4K. On YFCC4K, REVERSE reaches 27.5% accuracy at 25 km, compared with 12.4% for the untuned 4B model. Table 1 reports all methods at the five standard distance thresholds.

Table 1: **Main results on Im2GPS3k and YFCC4k.** Accuracy (%) at multiple distance thresholds. Unparsed predictions count as incorrect.

		Im2GPS3k					YFCC4k				
Method	Venue	@1km	@25km	@200km	@750km	@2500km	@1km	@25km	@200km	@750km	@2500km
Classification-based											
kNN	ICCV'17	7.2	19.4	26.9	38.9	55.9	2.3	5.7	11.0	23.5	42.0
PlaNet	ECCV'16	8.5	24.8	34.3	48.4	64.6	5.6	14.3	22.2	36.4	55.8
CPlaNet	ECCV'18	10.2	26.5	34.6	48.6	64.6	7.9	14.8	21.9	36.4	55.5
ISNs	ECCV'18	10.5	28.0	36.6	49.7	66.0	6.5	16.2	23.8	37.4	55.0
GeoToken	ICDM'25	16.8	39.6	53.8	70.8	85.0	**24.3**	35.3	46.6	64.2	**78.6**
Retrieval-based											
GeoCLIP	NeurIPS'23	14.1	34.5	50.7	69.7	83.8	9.6	19.3	32.6	55.0	74.7
Img2Loc	SIGIR'24	15.3	39.8	53.6	69.7	82.8	19.8	30.7	41.4	58.1	74.1
PIGEON	CVPR'24	11.3	36.7	53.8	72.4	85.3	10.4	23.7	40.6	62.2	77.7
G3	NeurIPS'24	16.7	40.9	55.6	71.2	84.7	24.0	**35.9**	**47.0**	**64.3**	78.2
VLM-based											
Translocator	ECCV'22	11.8	31.1	46.7	58.9	80.1	8.4	18.6	27.0	41.1	60.4
GeoDecoder	ICCV'23	12.8	33.5	45.9	61.0	76.1	10.3	24.4	33.9	50.0	68.7
GeoBayes	AAAI'26	6.3	34.7	53.6	73.7	85.9	4.9	16.1	30.9	55.8	75.4
Qwen3-VL-4B	—	8.5	32.3	47.8	62.6	71.0	3.0	12.4	23.2	38.6	51.9
Qwen3-VL-8B	—	10.5	37.2	55.3	72.8	84.9	3.3	10.8	19.6	31.1	40.4
SpotAgent	KDD'26	14.1	40.4	57.8	73.4	85.8	7.3	21.5	36.2	55.0	70.8
Geo-R	AAAI'26	18.1	41.5	58.3	**75.3**	**86.4**	10.5	22.7	40.0	60.8	75.8
REVERSE (ours)	—	**22.5**	**48.3**	**59.3**	73.5	84.8	14.1	27.5	38.1	53.8	70.6

4.2 Ablation studies

Training pipeline stages. Table 2 separates geographic pretraining from agentic process learning. REVERSE reaches 48.3% at 25 km, compared with 36.0% for the untuned model with the same three tools. Coordinate SFT establishes the geographic prior, reaching 42.5% at this threshold. Full-curriculum RL raises valid-output coverage from 90.5% after cold start to 99.1%, while increasing 25 km accuracy from 43.7% to 48.3%. Over the same transition, 750 km accuracy increases from 66.6% to 73.5%. Appendix H reports matched corrections, regressions, and persistent errors; Appendix F characterizes the coordinate-SFT scaling behavior.

Table 2: **Training pipeline ablation** on Im2GPS3k. Each row adds one stage on top of the previous. *Coverage*: fraction of samples producing a valid coordinate output (unparsed predictions count as incorrect in all accuracy columns). *AvgTool*: average number of tool calls per sample.

Stage	@1km	@25km	@200km	@750km	@2500km	Coverage	AvgTool
Qwen3-VL-4B (no tool)	8.5	32.3	47.8	62.6	71.0	71.3%	0.0
Qwen3-VL-4B (3 tools)	11.2	36.0	48.8	60.9	71.5	83.0%	3.9
Qwen3-VL-4B + SFT	18.1	42.5	57.6	74.8	85.9	100%	0.0
+ Agentic Cold Start	20.0	43.7	53.9	66.6	77.3	90.5%	2.7
+ Easy-curriculum RL	21.0	46.2	56.4	68.4	79.5	98.0%	3.1
+ Full-curriculum RL	**22.5**	**48.3**	**59.3**	**73.5**	**84.8**	**99.1%**	**3.0**

Process reward design. Table 3 ablates each component of the process reward. The full reward and the variant without IoU both reach 45.4% at 25 km. Region selection is therefore examined directly through the overlap diagnostics in Appendix I. Removing the MCC reward drops 25 km by 1.8 points to 43.6%, supporting explicit evidence discrimination as a contributor to localization accuracy. Removing the search utility reward causes the largest single-component drop, 2.4 points to 43.0%, identifying search utility as the strongest single-component contribution in this configuration. The outcome-only control in Table 3 provides the reference for the combined process reward. Relaxing the IoU threshold from 0.7 to 0.5 hurts particularly at 750 km, from 69.8% to 66.0%, supporting the stricter crop-matching gate in this configuration.

Table 3: **Process reward ablation** on Im2GPS3k. All variants share the same easy-curriculum data. γ = tool reward weight in the total reward; α = accuracy reward weight.

Variant	Acc	IoU	MCC	Search	α	γ	@25km	@750km	AvgTool
Full (ours)	✓	✓	✓	✓	0.6	0.3	**45.4**	**69.8**	3.02
w/o IoU reward	✓		✓	✓	0.6	0.3	45.4	69.4	3.04
w/o MCC reward	✓	✓		✓	0.6	0.3	43.6	69.0	3.0
w/o Search reward	✓	✓	✓		0.6	0.3	43.0	68.0	3.02
Acc only	✓				1.0	0	42.2	67.8	2.90
IoU thresh 0.5	✓	✓	✓	✓	0.6	0.3	43.4	66.0	3.15

4.3 Analysis

Tool selection. Table 7 reports localization with different tool combinations. Image search retrieves visually matched candidates, while text search supports place-name and coordinate verification. The region diagnostics in Appendix I separately measure crop overlap against useful-region references. Full tool-combination results appear in Appendix E.

Case study. Figure 4 shows a case where REVERSE succeeds while untuned Qwen3-VL-4B-Instruct fails. Both models receive identical image search results pointing clearly to Zuccotti Park, New York City. The untuned model rejects this evidence, reasoning that the architecture looks more like Chicago Loop, and follows a second search to the wrong city. REVERSE correctly reads the first result, identifies the sculpture as Joie de Vivre in Zuccotti Park, and uses text search to confirm coordinates before answering. The contrast illustrates the core failure mode of untuned VLMs as tool-users: strong visual priors override retrieved evidence even when the evidence is unambiguous.

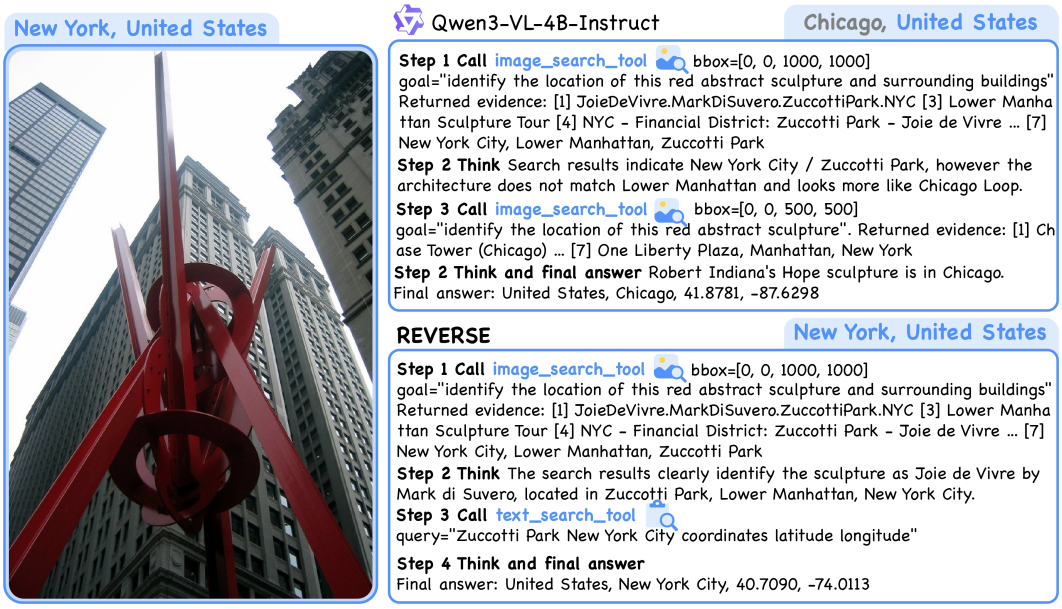


Figure 4: **Comparison of two agent trajectories on the same geo-localization case.** The Qwen3-VL-4B-Instruct trajectory first retrieves strong evidence for Joie de Vivre at Zuccotti Park, New York City, but rejects it based on a Chicago architectural prior and follows a cropped-search distractor to Chicago. **REVERSE**'s trajectory keeps the full-image evidence, identifies the sculpture as Mark di Suvero's Joie de Vivre in Zuccotti Park, and correctly predicts New York City.

Process diagnostics connect rewards to decisions. We evaluate region overlap and explicit evidence selection separately from final-coordinate accuracy. Appendix I provides image- and text-search precision, recall, F1, and MCC, with the reference-label construction and aggregation rule stated for each diagnostic. The cross-agent pilot preserves each method's native tool interface and compares search invocation and localization. These measurements make the agent's intermediate choices inspectable, linking region selection to spatial overlap and evidence filtering to result-level judgments. Together with localization accuracy, they show how the agent gathers and uses evidence along the path to a location estimate.

Offline optimization and bounded inference calls. RL serves retrieval observations from a fixed local cache. On all 2,997 Im2GPS3k images, the final agent averages 2.98 tool calls and 2.75 external search calls per image. Appendix J separates these final-agent counts from a controlled latency profile of the underlying tool service. Reusing cached observations allows repeated policy updates without issuing new external search requests during training. At inference, reporting search calls separately from local visual operations makes the external-service demand explicit and provides a concrete basis for planning deployment costs.

5 Conclusion

REVERSE trains region selection, search, and evidence verification as explicit intermediate decisions. Tool-grounded trajectories and process rewards connect these choices to the final localization objective, while cached retrieval observations support repeatable agentic optimization. With a 4B backbone, REVERSE achieves 48.3% accuracy at 25 km on Im2GPS3k. The reported experiments characterize the documented Kimi-K2.6 teacher and search configuration, with fixed retrieval observations during RL. Cross-teacher and cross-service transfer are directions for further evaluation. At inference, search results may include explicit location metadata from GPS-bearing domains such as Flickr. Evidence-conditioned routing between direct geographic priors and retrieved candidates offers an extension for inputs without selected web evidence. Because geolocation can expose individuals or sensitive locations, real-world deployment should include privacy safeguards such as metadata stripping and blurring of faces and private spaces. These results establish process supervision as an effective route to evidence-grounded, fine-grained image geo-localization.

Acknowledgements

The authors thank the National Natural Science Foundation of China for the financial support under Grant No. 42430106.

References

[1] Gengchen Mai, Weiming Huang, Jin Sun, Suhang Song, Deepak Mishra, Ninghao Liu, Song Gao, Tianming Liu, Gao Cong, Yingjie Hu, Chris Cundy, Ziyuan Li, Rui Zhu, and Ni Lao. On the opportunities and challenges of foundation models for GeoAI. *ACM Transactions on Spatial Algorithms and Systems*, 10(2):1–46, 2024.

[2] Guillaume Astruc, Nicolas Dufour, Ioannis Siglidis, Constantin Aronssohn, Nacim Bouia, Stephanie Fu, Romain Loiseau, Van Nguyen Nguyen, Charles Raude, Elliot Vincent, Lintao Xu, Hongyu Zhou, and Loic Landrieu. OpenStreetView-5M: The many roads to global visual geolocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21967–21977, 2024.

[3] Yong Li, Yingjing Huang, and Fan Zhang. Learning street view representations based on a spatiotemporal contrastive learning framework. *Computers, Environment and Urban Systems*, 125:102393, 2026.

[4] Tobias Weyand, Ilya Kostrikov, and James Philbin. PlaNet - photo geolocation with convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, 2016.

[5] Paul Hongsuck Seo, Tobias Weyand, Jack Sim, and Bohyung Han. CPlaNet: Enhancing image geolocalization by combinatorial partitioning of maps. In *European Conference on Computer Vision (ECCV)*, 2018.

[6] Eric Müller-Budack, Kader Pustu-Iren, and Ralph Ewerth. Geolocation estimation of photos using a hierarchical model and scene classification. In *European Conference on Computer Vision (ECCV)*, 2018.

[7] V. Vivanco Cepeda, Gaurav Kumar Nayak, and Mubarak Shah. GeoCLIP: Clip-inspired alignment between locations and images for effective worldwide geo-localization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[8] Lukas Haas, Michal Skreta, Silas Alberti, and Chelsea Finn. PIGEON: Predicting image geolocations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[9] Zhongliang Zhou, Jielu Zhang, Zihan Guan, Jiayu Hu, Shuwei Lao, Kaiye Mu, Yunqi Li, and Gengchen Mai. Img2Loc: Revisiting image geolocalization using multi-modality foundation models and image-based retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2024.

[10] Pengyue Jia, Yiding Liu, Xiaopeng Li, Yuhao Wang, Yantong Du, Xiao Han, Xuetao Wei, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. G3: An effective and adaptive framework for worldwide geolocalization using large multi-modality models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[11] Biao Wu, Meng Fang, Ling Chen, Ke Xu, Tao Cheng, and Jun Wang. Vision-language reasoning for geolocalization: A reinforcement learning approach. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

[12] Modi Jin, Yiming Zhang, Boyuan Sun, Dingwen Zhang, MingMing Cheng, and Qibin Hou. Geoagent: Learning to geolocate everywhere with reinforced geographic characteristics. *arXiv preprint arXiv:2602.12617*, 2026.

[13] Furong Jia, Ling Dai, Wenjin Deng, Fan Zhang, Chen Hu, Daxin Jiang, and Yu Liu. Spotagent: Grounding visual geo-localization in large vision-language models through agentic reasoning. *arXiv preprint arXiv:2602.09463*, 2026.

[14] Yuxiang Ji, Yong Wang, Ziyu Ma, Yiming Hu, Hailang Huang, Xuecai Hu, Guanhua Chen, Liaoni Wu, and Xiangxiang Chu. Thinking with map: Reinforced parallel map-augmented agent for geolocalization. *arXiv preprint arXiv:2601.05432*, 2026.

[15] Xiao Han, Chen Zhu, Xiangyu Zhao, and Hengshu Zhu. Swarm intelligence in geo-localization: A multi-agent large vision-language model collaborative framework. In *Proceedings of the 31st*

ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2, pages 814–825, 2025.

[16] Yikun Wang, Zuyan Liu, Ziyi Wang, Han Hu, Pengfei Liu, and Yongming Rao. GeoVista: Web-augmented agentic visual reasoning for geolocalization. *arXiv preprint arXiv:2511.15705*, 2025.

[17] Nam Vo, Nathan Jacobs, and James Hays. Revisiting IM2GPS in the deep learning era. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[18] Bart Thomee, David A. Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. YFCC100M: The new data in multimedia research. In *Communications of the ACM*, 2016.

[19] Yong Li, Weiyu Zhang, Ling Dai, Jian Yang, Dacheng Yin, Sirun Li, Jing Lyu, Fengyun Rao, and Fan Zhang. Unified urban tuning: Co-enhancing satellite and street view reasoning with a progressive tuning framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6302–6311, 2026.

[20] Shraman Pramanick, Ewa M. Nowara, Joshua Gleason, Carlos D. Castillo, and Rama Chellappa. Where in the world is this image? Transformer-based geo-localization in the wild. In *European Conference on Computer Vision (ECCV)*, 2022.

[21] Brandon Clark, Alec Kerrigan, Parth Parag Kulkarni, Vicente Vivanco Cepeda, and Mubarak Shah. Where we are and what we're looking at: Query based worldwide image geo-localization using hierarchies and scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23182–23190, 2023.

[22] Pengyue Jia, Seongheon Park, Song Gao, Xiangyu Zhao, and Sharon Li. GeoRanker: Distance-aware ranking for worldwide image geolocalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[23] Ling Li, Yao Zhou, Yuxuan Liang, Fugee Tsung, and Jiaheng Wei. Recognition through reasoning: Reinforcing image geo-localization with large vision-language models. *arXiv preprint arXiv:2506.14674*, 2025.

[24] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

[25] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36: 68539–68551, 2023.

[26] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11888–11898, 2023.

[27] Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023.

[28] Penghao Wu and Saining Xie. V?: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13084–13094, 2024.

[29] Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

[30] Wenxuan Huang, Yu Zeng, Qiuchen Wang, Zhen Fang, Shaosheng Cao, Zheng Chu, Qingyu Yin, Shuang Chen, Zhenfei Yin, Lin Chen, Zehui Chen, Xu Tang, Yao Hu, Shaohui Lin, Philip Torr, Feng Zhao, and Wanli Ouyang. Vision-deepresearch: Incentivizing deepresearch capability in multimodal large language models. *arXiv preprint arXiv:2601.22060*, 2026.

[31] Yu Zeng, Wenxuan Huang, Zhen Fang, Shuang Chen, Yufan Shen, Yishuo Cai, Xiaoman Wang, Zhenfei Yin, Lin Chen, Zehui Chen, Shiting Huang, Yiming Zhao, Xu Tang, Yao Hu, Philip Torr, Wanli Ouyang, and Shaosheng Cao. Vision-deepresearch benchmark: Rethinking visual and textual search for multimodal large language models. *arXiv preprint arXiv:2602.02185*, 2026.

[32] Zeyi Huang, Yuyang Ji, Anirudh Sundara Rajan, Zefan Cai, Wen Xiao, Haohan Wang, Junjie Hu, and Yong Jae Lee. Visualtoolagent (vista): A reinforcement learning framework for visual tool selection. *arXiv preprint arXiv:2505.20289*, 2025.

[33] Qihua Dong, Gozde Sahin, Pei Wang, Zhaowei Cai, Robik Shrestha, Hao Yang, and Davide Modolo. Visual reasoning through tool-supervised reinforcement learning. *arXiv preprint arXiv:2604.19945*, 2026.

[34] Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, and Qing Li. Chain-of-focus: Adaptive visual search and zooming for multimodal reasoning via rl. *arXiv e-prints*, pages arXiv–2505, 2025.

[35] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[36] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[37] Narges Ghasemi, Amir Ziashahabi, Salman Avestimehr, and Cyrus Shahabi. GeoToken: Hierarchical geolocalization of images via next token prediction. In *IEEE International Conference on Data Mining (ICDM)*, 2025.

[38] Brandon Clark, Alec Kerrigan, Parth Parag Kulkarni, Vicente Vivanco Cepeda, and Mubarak Shah. Where we are and what we're looking at: Query based worldwide image geo-localization using hierarchies and scenes. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[39] Weimin Shi, Xiang Li, Kaige Li, Junhao Fang, Qiang Zhou, Qichuan Geng, and Zhong Zhou. GeoBayes: Probabilistic image geo-localization inference via sequential bayesian updating. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

[40] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. HybridFlow: A flexible and efficient RLHF framework. *arXiv preprint arXiv:2409.19256*, 2024.

[41] Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. *arXiv preprint arXiv:2312.07104*, 2023.

A Agent Implementation Details

A.1 Prompt

To ensure consistency between training and evaluation, we adopt a unified prompt template used verbatim during both Agentic Cold Start and RL rollout. The template is designed to guide the model through systematic visual reasoning of geographic clues while enforcing a *think-then-act* pattern for tool use: the model must always reason in `<think>` before issuing any tool call or final answer. Beyond standard tool-use instructions, our prompt introduces two REVERSE-specific mechanisms: (i) explicit **decision rules** that govern which tool to call given the visual context (e.g., image search for landmarks, zoom for small text), and (ii) an **evidence discrimination** tag `<useful>` that forces the model to identify which retrieved results are geographically relevant to the specific image—this tag directly drives our MCC process reward during RL training.

A.2 Example Inference Trace

The trace below shows a complete two-turn inference from REVERSE on a real Im2GPS3k image (id: 311938754, prediction error: 3.4 km). The model identifies a distinctive astronomical instrument via image search, discriminates the relevant results with `<useful>`, and confirms the exact coordinates through a follow-up text search before issuing its final answer.

A.3 Agent Tool Infrastructure

REVERSE exposes three tools to the agent at inference time. All tool calls are formatted as structured JSON inside `<tool_call> ... </tool_call>` tags, following the same format used during Agentic Cold Start training.

Image Zoom Tool. The zoom tool implements a deterministic crop-and-resize pipeline. Given normalized bounding-box coordinates $[x_1, y_1, x_2, y_2]$ in $[0, 1000]^2$, the tool maps them to pixel-level crops via linear scaling to the image dimensions. The crop is then resized following Qwen3-VL's `smart_resize` scheme: output resolution is snapped to multiples of the patch factor (28 pixels), with a minimum of 256×256 and a maximum of 2048×1024 pixels, preserving aspect ratio. The resized crop is re-injected into the model's context as a new image token sequence, allowing fine-grained inspection of text, inscriptions, or small visual details not visible at full scale.

Image Search Tool. The reverse image search tool crops the specified region using the same pipeline as the zoom tool, uploads the crop to cloud object storage to obtain a public URL, and queries the Oxylabs Google Lens API. Results are returned as a ranked list of up to 10 web pages, each containing a title, source URL, and domain. The agent marks useful results with `<useful>[...]</useful>` tags, which serve as the discrimination signal for the MCC reward during RL training.

Web Search Tool. The text search tool submits natural-language queries to the Tavily Search API, returning up to 5 results per query with titles, snippets, and source URLs. The tool supports parallel multi-query execution: if the agent provides a list of queries, all are dispatched concurrently and results are concatenated. API keys are managed via a pool with automatic rotation on quota exhaustion. Both image search and text search results are cached in a local SQLite database (WAL mode) to ensure reproducibility across evaluation runs.

A.4 Offline Search Cache for RL Rollout

Live API calls during RL rollout are expensive and non-reproducible. We pre-annotate the training corpus with tool call–response pairs offline and cache them in a Parquet-based lookup table keyed by (image hash, bounding box) for image search and by query string for text search. During rollout, image and text searches use cached observations. Missing entries or rejected matches return the configured offline response; they do not initiate live search requests. This separates repeated policy optimization from external-service latency and changing web content.

Prompt Template

You are a geolocation expert. Given an image, identify its location.

Available Tools
The agent can invoke the following tools:

- **image_search_tool**: Reverse image search using a cropped region. Best for distinctive landmarks, buildings, or scenes. Returns matching web pages.
  - *Parameters*: `bbox_2d [x1, y1, x2, y2]` — crop region; `goal` — description of what to identify.
- **text_search_tool**: Search the web with natural language queries. Use for visible text/signs, landmark names, or any clues found from image search results. Supports a list of parallel queries.
  - *Parameters*: `query` — a string or list of search strings.
- **image_zoom_in_tool**: Zoom into a region to read text or inscriptions that are too small at full scale.
  - *Parameters*: `bbox_2d [x1, y1, x2, y2]` — region to zoom.

Decision Rules

- Distinctive landmark or scene visible (but uncertain of exact coords) → use **image_search_tool**.
- Text/signs already legible → use **text_search_tool** directly.
- Text/signs too small to read → use **image_zoom_in_tool** first, then **text_search_tool**.
- Image search returns a landmark/location name → follow up with **text_search_tool**.
- Do **not** use **image_zoom_in_tool** before **image_search_tool** — zoom does not improve image search.
- When in doubt, use a tool. Only skip tools when certain.
- **Fallback rule**: If tools have failed, returned empty results, or exceeded call limits, do **not** keep retrying. Provide your best `<answer>` based on visual priors. Never finish a response without a `<tool_call>` or `<answer>`.

Reasoning Requirement
For every response, first enclose your reasoning in `<think> ... </think>` tags, then output **exactly one** of:

```
<tool_call>{"name": "image_search_tool", "arguments": {"bbox_2d": [x1,y1,x2,y2], "goal": "..."}}</tool_call>
<tool_call>{"name": "text_search_tool", "arguments": {"query": "..."}}</tool_call>
<tool_call>{"name": "text_search_tool", "arguments": {"query": ["q1","q2"]}}</tool_call>
<tool_call>{"name": "image_zoom_in_tool", "arguments": {"bbox_2d": [x1,y1,x2,y2]}}</tool_call>
```

or your final `<answer>` when confident without needing a tool.

Evidence Discrimination
After receiving tool results, output on its own line: `<useful>[i, j, ...]</useful>` listing the 1-based indices of results that match this specific image. Results about a different place are **not** useful even if they contain geographic information. Output `<useful>[]</useful>` if none match.

Final Answer Format
When ready to answer, provide the final answer inside `<answer> ... </answer>` tags. The final answer must include Country, City, Latitude, and Longitude (decimal format).
Example: `<answer>Italy, Golfo Arnaci, 40.9606, 9.5873</answer>`
Latitude and longitude must always be valid numeric values and cannot be null.

Inference Trace — Im2GPS3k id: 311938754 GT: Jaipur, India Error: 3.4 km

User: *[Image: circular hemispherical astronomical instrument, stone construction, Jaipur, India]*

Assistant (Turn 1):
<think> The image shows a circular structure with a reflective surface, appearing to be a sundial or astronomical instrument. This is a distinctive landmark. Let me run image search to identify it. </think>
<tool_call>{"name": "image_search_tool", "arguments": {"bbox_2d": [0, 0, 1000, 1000], "goal": "identify this circular astronomical instrument"}}</tool_call>

Tool Response (image_search_tool):
[1] File:JaiPrakashYantraJaipur20080213-1.jpg — Wikimedia Commons
[2] Jai Prakash Yantra, measuring time through a hemispherical… — traveladventures.org
[3] Jantar Mantar, Jaipur — Wikipedia
[4] Detail of the Jai Prakash Yantra, a sundial which measures… — Alamy
[5] Jantar Mantar Observatory — markandchucksadventures.com
[6] File:Zodiac Circle jantar mantar, Jaipur 1.jpg — Wikimedia Commons
[7] Jantar Mantar: Jaipur's Famous Observatory — throughmylens.com
[8] Jai Prakash Yantra Jantarmantar Observatory — Tripadvisor
[9] Under Jaipur Skies: Jantar Mantar — shaopeng.blog
[10] 617 Sundial Jantar Mantar Jaipur Stock Photos — Dreamstime.com

---

Assistant (Turn 2):
<think> The results clearly identify this as the Jai Prakash Yantra at Jantar Mantar Observatory in Jaipur, Rajasthan, India. Let me confirm the exact coordinates with a text search. </think>
<useful>[1, 2, 3, 4, 5]</useful>
<tool_call>{"name": "text_search_tool", "arguments": {"query": "Jantar Mantar Observatory Jaipur coordinates latitude longitude"}}</tool_call>

Tool Response (text_search_tool):
[1] GPS coordinates of Jantar Mantar (Jaipur), India. Latitude: 26.9215, Longitude: 75.8213 — latitude.to
The Jantar Mantar monument of Jaipur is a collection of nineteen astronomical instruments built by the Rajput king Sawai Jai Singh, completed in 1738 CE. Latitude: 26°55'17.39"N, Longitude: 75°49'16.79"E. Nearest city: Jaipur.
[2] Jantar Mantar, Jaipur — Wikipedia
Kranti Vritta Yantra (measures longitude and latitude of celestial bodies)… Its face is angled at 27 degrees, the latitude of Jaipur.
[3] Jantar Mantar Jaipur India — tourhq.com
The Jantar Mantar features the world's largest stone sundial and is a UNESCO World Heritage site.
[4] Jantar Mantar — banbanjara.com
Sawai Jai Singh II constructed the astronomical observatory in Jaipur. 19 astrological and architectural instruments including the Vrihat Samrat Yantra.
[5] UNESCO Portal to the Heritage of Astronomy
The Jantar Mantar of Jaipur is an observatory built in the first half of the 18th century. One of the most complete collections of pre-telescopic masonry instruments in the world.

Assistant (Final Answer):
<think> Confirmed: Jantar Mantar Observatory, Jaipur, India. Coordinates 26.9215°N, 75.8213°E from the first result. </think>
<useful>[1]</useful>
<answer>India, Jaipur, 26.9215, 75.8213</answer>

B Training Data Construction

Our training corpora are built from teacher-generated trajectories using Kimi-K2.6 as the annotation agent. We randomly sample 100K images from MP16-Pro and prompt Kimi-K2.6 to solve each using the same agentic tool interface as REVERSE, without revealing the ground-truth location. This yields ∼30,750 valid raw trajectories after discarding failed or inaccessible cases. The remaining MP16-Pro images (approximately 4M) are used for Stage 1 SFT as coordinate-only supervision. We apply a cascade of quality filters to the 30,750 trajectories to produce training splits of varying strictness (Table 4).

Trajectory filtering. *RL base* (19,962 trajectories): retain samples where the image is accessible, the teacher's localization error $d \leq 200\,\text{km}$, at least one tool was invoked, the trajectory did not exceed the maximum turn budget, and no tool call returned an API failure response. This is the broadest usable set and forms the basis for all downstream splits.

Agentic Cold Start (4,427 trajectories, subset of RL base): additionally require at most 5 tool calls per trajectory. This stricter constraint keeps only the most concise, self-consistent reasoning paths for supervised fine-tuning.

RL full-coverage (6,996 trajectories, subset of RL base): retain samples where every image-search call in the trajectory has been annotated with geo-informative and non-informative result labels, and at least one positive result exists per search call. This annotation is required so that the MCC discrimination reward r_t^{mcc} can be computed for every tool call during RL training. Trajectories with no search calls are also included.

Easy-curriculum subset (3,104 trajectories, subset of RL full-coverage): further restrict to $d \leq 25\,\text{km}$ to form the first-stage RL curriculum, focusing training on geographically unambiguous scenes before progressing to harder long-range samples.

Offline cache construction. The image search cache groups result records by (image path, call index). Each entry stores the teacher's annotated bounding box, all geo-informative (positive) results, and all non-informative (negative) results from the same search call. During RL rollout, a model-generated bounding box is matched to a cache entry if $\text{IoU}(b_{\text{pred}}, b_{\text{gt}}) \geq \tau$; otherwise the tool returns an empty result, penalizing imprecise crops. The text search cache groups calls by image and matches the requested query to cached queries using token-level Jaccard similarity. Cache coverage is defined for the selected training and validation splits. RL accesses these frozen observations without live search calls.

Table 4: Dataset statistics for Agentic Cold Start and RL training.

Split	Samples	Filter criteria
Agentic Cold Start	4,427	$d \leq 200\,\text{km}$, $\leq$5 tool calls
RL easy-curriculum	3,104	full-coverage + $d \leq 25\,\text{km}$
RL full-curriculum	6,996	full-coverage, $\geq$1 positive result per search call

C Training Hyperparameters

Table 5 summarizes the hyperparameters for all three training stages. REVERSE follows a progressive training schedule: Stage 1 establishes broad geographic priors via large-scale supervised learning; Stage 2 fine-tunes the model on agentic tool-use trajectories at a lower learning rate; Stage 3 applies GRPO-based RL with a two-phase curriculum, starting from easy samples ($d \leq 25\,\text{km}$) before expanding to the full difficulty range.

D Experimental Setup

Benchmarks. We evaluate on two standard geolocation benchmarks. *Im2GPS3k* [17] contains 2,997 geotagged images sampled worldwide, covering diverse geographic regions and landmark types. *YFCC4k* [18] contains 4,536 images from the YFCC100M dataset and spans heterogeneous scenes. Following standard practice, accuracy at threshold r counts a prediction as correct if its Haversine distance to ground truth is within r km; unparsed predictions count as incorrect. We report accuracy at 1, 25, 200, 750, and 2500 km.

Baselines. We compare against a broad range of prior geolocation methods. *Classification-based* methods partition the Earth into geographic cells and train classifiers: PlaNet [4], CPlaNet [5], ISNs [6], kNN [17], and GeoToken [37]. *Retrieval-based* methods match query images against geo-tagged databases: GeoCLIP [7], Img2Loc [9], and PIGEON [8]. *VLM-based* methods leverage pretrained vision-language models with geographic supervision: Translocator [20], GeoDecoder [38], G3 [10], and GeoBayes [39]. We additionally evaluate Qwen3-VL-4B and Qwen3-VL-8B [35] with-

Table 5: **Training hyperparameters for all three stages of REVERSE.**

Hyperparameter	Stage 1 (SFT)	Stage 2 (Agentic Cold Start)	Stage 3 (Agentic RL)
Base model	Qwen3-VL-4B-Instruct	Stage 1 checkpoint	Stage 2 checkpoint
Training framework	FSDP	FSDP	VERL + GRPO
Rollout engine	—	—	SGLang
Nodes ($\times$ 8 GPUs)	8	1	4
Training data	MP16-Pro (4M)	Cold-start V5.2 (4,427)	Easy (3,104) $\rightarrow$ Full (6,996)
Batch size	512	128	256
Learning rate	2×10^{-5}	2×10^{-5}	1×10^{-6}
LR schedule	cosine	cosine	cosine
Warmup ratio	0.1	0.1	0.05
Min-LR ratio	—	—	0.1
Training steps	$\approx$7,800	35	80 (easy) + 135 (full)
Reward weights	—	—	α=0.6, β=0.1, γ=0.3
IoU reward coeff	—	—	0.2
MCC reward coeff	—	—	0.3
Base search reward	—	—	0.1
IoU threshold	—	—	0.7
Max turns (rollout)	—	—	10
Max response length	—	—	16,384 tokens
Hardware	8 $\times$ NVIDIA A800 80 GB per node		

Table 6: Reward hyperparameters and their default values.

Symbol	Meaning	Default
α	Geolocation reward weight	0.6
β	Format reward weight	0.1
γ	Tool reward weight	0.3
τ	IoU gate threshold	0.7
λ_{iou}	Image search IoU coeff	0.2
λ_{base}	Text search base reward	0.1
λ_{mcc}	MCC discrimination coeff	0.3
δ	Zoom invalid-box penalty	0.05

out any geographic fine-tuning as untuned VLM baselines. *RL-based* methods optimize geographic reasoning with reinforcement learning: Geo-R [11].

Implementation details. REVERSE is built on Qwen3-VL-4B-Instruct [35] and trained in three stages (full hyperparameters in Table 5). **Stage 1 (SFT):** We fine-tune on 4M geo-tagged images from MP16-Pro [10] for $\approx$7,800 steps on 8 nodes with batch size 512 and learning rate 2×10^{-5}. **Stage 2 (Agentic Cold Start):** We fine-tune on 4,427 curated teacher trajectories for 35 steps on 1 node with batch size 128 and learning rate 2×10^{-5}. **Stage 3 (Agentic RL):** We apply GRPO [36] with a two-phase curriculum: 80 steps on 3,104 easy samples ($d \leq 25$ km) followed by 135 steps on 6,996 full-coverage samples, using learning rate 1×10^{-6} and reward weights α=0.6, β=0.1, γ=0.3. RL rollout uses VERL [40] and SGLang [41]. All training runs on NVIDIA A800 80 GB GPUs (8 GPUs per node).

E Tool Combination Results

Table 7: **Tool combination ablation** on Im2GPS3k. All variants use the same REVERSE model (full-curriculum RL). Unparsed predictions count as incorrect.

Tools	@1km	@25km	@200km	@750km	@2500km
No tool	8.5	32.3	47.8	62.6	71.0
Zoom only	12.0	38.0	58.0	64.0	84.0
Image search only	20.0	48.0	64.0	72.0	76.0
Text search only	14.0	42.0	54.0	64.0	76.0
Zoom + text search	12.0	38.0	58.0	68.0	76.0
Image + text search	22.0	47.0	58.0	72.0	80.0
Image + text + zoom	**22.5**	**48.3**	**59.3**	**73.5**	**84.8**

F SFT Scaling

Figure 5 shows the accuracy of Qwen3-VL-4B and Qwen3-VL-8B on Im2GPS3k as a function of training samples seen during SFT (Stage 1). In this stage, the model is trained to directly output coordinates without any intermediate reasoning or tool use, purely consolidating geographic visual memory. Both models improve consistently across all distance thresholds with no sign of saturation at 4M samples. The 4B and 8B models reach similar accuracy under the evaluated SFT setup, supporting the 4B backbone used by REVERSE.

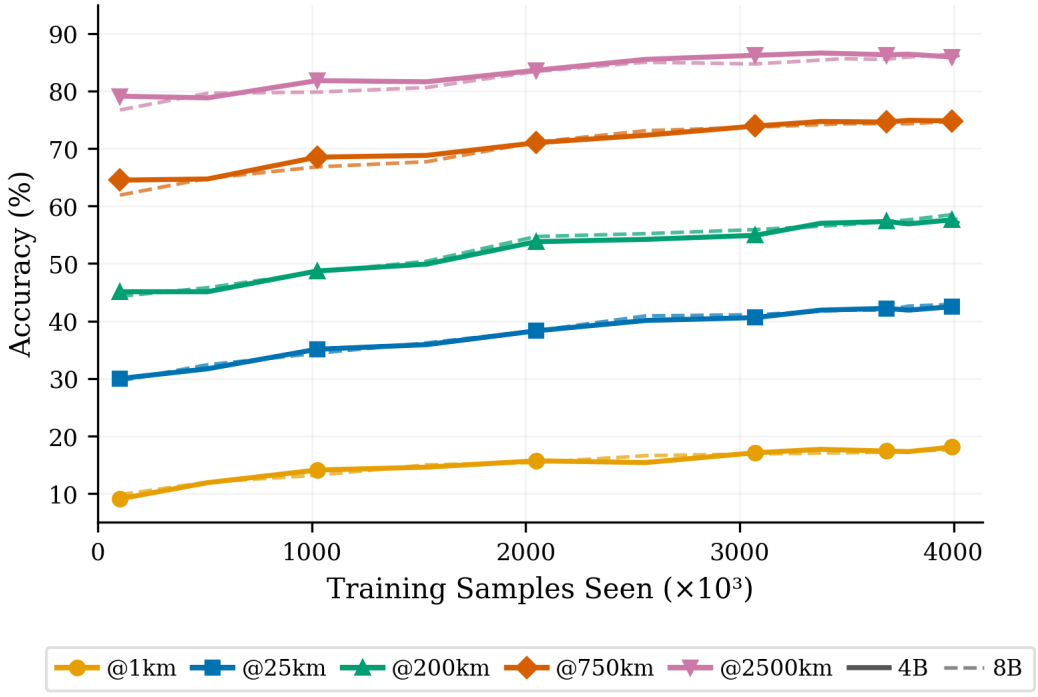


Figure 5: **SFT scaling curves** on Im2GPS3k (2,997 images). Accuracy at five distance thresholds as a function of training samples seen. Solid lines: Qwen3-VL-4B; dashed lines: Qwen3-VL-8B.

G Supervision and Diagnostic Reference Protocols

Training signals. Trajectory generation uses the query image and tool interface without exposing the reference location to the teacher. Reference coordinates enter subsequent trajectory-consistency filtering and missing-usefulness annotation. The recorded pipeline combines teacher-cited result

indices with GT-conditioned usefulness adjudication. Region re-annotation supplies useful-region targets; trajectory execution, result-label coverage, and geographic consistency are separate checks. These checks serve distinct purposes: a correctly localized trajectory does not establish the correctness of every intermediate label.

Evidence-level audit. The retained diagnostic audit contains 222 Kimi useful-selection events from the 500-image diagnostic setting. It records 108 positive, 106 negative, and eight unresolved event judgments after a targeted pass over 32 disputed events. These are internal reference judgments about the selected evidence visible in the retained records. The 214 resolved events measure audit coverage, not label accuracy. The scope of this audit is distinct from the full training corpus and from the automated diagnostic references below.

Reference definitions. Final accuracy uses benchmark GPS coordinates and the standard distance thresholds, with invalid predictions counted as errors. For the 500-image evidence-selection diagnostic, a visible result is reference-positive when its normalized title, URL, or snippet matches the GT city term. Country matching is used when the city term is unavailable, with the recorded place-name normalization rules. This lexical reference measures location-term alignment; a matching place name alone does not establish that a result depicts the photographed site. The reference boxes for region diagnostics are the union of reward-verified regions from three 235B-family annotation runs. They are useful-region proxies with their own coverage and matching protocol.

Query quality. We evaluate a query/action by the geographic evidence obtained through its tool response. For image search, the action includes selecting the crop used for reverse lookup. For text search, the action specifies a query whose returned evidence can support coordinate prediction. Search invocation, returned evidence, explicit selection, and final localization are measured separately. Each diagnostic below states its reference source and denominator, preserving the distinction between process behavior and endpoint accuracy.

H Paired SFT-to-RL Transitions

We match the direct SFT and final full-curriculum RL predictions on all 2,997 Im2GPS3k images. For each threshold, the four outcomes in Table 8 partition the evaluation set. Invalid outputs remain incorrect, so the same rule applies to fine and coarse thresholds.

Table 8: Matched outcome counts. Corrected: SFT incorrect and RL correct. Regressed: SFT correct and RL incorrect.

Threshold (km)	Both correct	Corrected	Regressed	Both incorrect
25	1059	388	215	1335
750	1957	246	285	509
2500	2380	160	195	262

At 25 km, 388 corrections and 215 regressions yield a net gain of 173 images, or 5.8 percentage points. The corrections include 77 cases moving from beyond 2,500 km to within 200 km. The cold-start-to-full-RL comparison independently gives 272 recoveries and 66 regressions at 750 km, consistent with the increase from 66.6% to 73.5% in the stage table.

Persistent errors and output states. At 750 km, SFT and RL are both incorrect on 509 images; 20 of these trajectories contain selected evidence and 489 do not. At 2,500 km, the corresponding counts are 262, eight, and 254. These counts describe the saved selections, not whether useful evidence exists elsewhere on the web. They distinguish persistent errors from the separate set of predictions that move out of a previously correct region.

For the 285 regressions at 750 km, the records contain 236 valid outputs with no selected evidence, 25 with selected evidence, and 24 invalid outputs. For the 195 regressions at 2,500 km, the counts are 149, 21, and 25. Selected evidence is therefore analyzed together with the returned content rather than treated as proof of correctness. For example, a Sandgate Town Hall query can retrieve a same-name building in Queensland for a photograph from the United States. This motivates evaluation of routing between geographic priors and retrieved evidence as a prospective extension.

I Process Diagnostics

I.1 Search invocation under native interfaces

The 50-image pilot compares the agents' native search invocation and benchmark-coordinate accuracy. An image is counted as searched if its trajectory invokes image or text/web search; geocoding alone is excluded. The REVERSE pilot is reported separately from the final full-benchmark evaluation. A reference-positive subset contains 40 images selected by the retained trace/GT proxy protocol. That protocol combines location-term matches and teacher selections conditioned on final-coordinate consistency; it describes a reference subset rather than an independently labeled necessity-to-search benchmark. Table 9 reports the count of searched images both overall and within that subset.

Table 9: Search invocation and localization on the shared 50-image pilot. Reference-positive search uses the same 40 image IDs for all methods.

Method	Searched / 50	Ref.-positive searched / 40	@25 (%)	@750 (%)
REVERSE	50	40	54.0	80.0
SpotAgent	7	5	40.0	76.0
GeoVista	47	39	30.0	62.0

The pilot separates invoking a search action from obtaining a correct final coordinate. REVERSE searches all 50 images and reaches 54.0% at 25 km, whereas GeoVista searches 47 and reaches 30.0%. This comparison motivates evaluating what the agent does with retrieved evidence, beyond counting tool invocations.

I.2 Explicit result selection

We evaluate four step-20 reward variants on a fixed 500-image subset using the lexical references defined in Appendix G. The diagnostic full-reward model is an easy-stage step-20 checkpoint, distinct from the final full-curriculum model in the main results. Each model produces its own result lists; these are behavior diagnostics rather than a controlled comparison on identical retrieved candidates. Precision, recall, F1, and MCC pool indexed result decisions separately for image and text search. An absent or unparsed `<useful>` list is treated as an empty selection by the retained parser. Empty-result events are excluded. All-negative rejection is the fraction of reference-all-negative events with an empty selection; selection rate is the fraction of nonempty-result events with at least one selected index.

Table 10: Evidence-selection diagnostics against visible-result lexical proxies. I/T denote image/text search. Rej. and Sel. are percentages.

Variant	Tool	Events	P	R	F1	MCC	Rej.	Sel.
Accuracy only	I	517	0.347	0.861	0.495	0.429	38.4	74.3
Accuracy only	T	837	0.325	0.080	0.128	0.044	88.1	11.7
w/o MCC	I	419	0.310	0.849	0.454	0.386	39.5	71.4
w/o MCC	T	400	0.383	0.044	0.079	0.068	97.2	3.0
w/o IoU	I	511	0.366	0.824	0.507	0.435	47.4	67.9
w/o IoU	T	934	0.396	0.071	0.120	0.086	93.0	7.9
Full reward	I	517	0.357	0.845	0.502	0.428	52.1	66.3
Full reward	T	893	0.222	0.257	0.239	-0.023	52.8	43.3

The explicit selection interface makes both acceptance and rejection decisions measurable. For text search, full reward reaches lexical-proxy F1 of 0.239, compared with 0.079 without MCC. The complete precision/recall and rejection columns identify how this change relates to selection behavior. These lexical diagnostics are reported alongside geographic accuracy, which uses independent benchmark GPS coordinates.

I.3 Region targeting

Of the 500 images, 103 have at least one useful-region proxy from the retained reward-verification runs. For each image with a region action, we take the maximum IoU between predicted and reference

boxes. The conditional mean averages images with both a predicted box and a reference box. A targeted crop occupies at most 50% of the normalized image plane. The opportunity-normalized hit rate uses all 103 reference-region images, with missing targeted crops counted as misses.

Table 11: Useful-region proxy diagnostics. Conditional means and opportunity-normalized hit rates have different denominators, as specified in the text.

Variant	Region use (%)	Mean max IoU	Targeted IoU	Targeted hit@0.5 (%)
Accuracy only	95.1	0.183	0.213	3.9
w/o MCC	80.6	0.166	0.000	0.0
w/o IoU	90.3	0.172	0.132	1.0
Full reward	93.2	0.178	0.194	2.9

Full reward has conditional targeted IoU of 0.194, compared with 0.132 without IoU reward. The opportunity-normalized hit rates provide the corresponding view that includes skipped region actions. Together with explicit selection metrics, this distinguishes the decisions targeted by the process rewards.

I.4 Stage-resolved pilot diagnostics

The retained 50-image reference protocol marks 40 images reference-positive and ten without a positive reference. REVERSE searches all 40 reference-positive images when image and text search are both counted. Four omit image search but invoke text search; they are not missed search opportunities. Of ten images with a useful-region reference, two have an action reaching IoU 0.5; eight do not, including missing actions. Six of the 40 reference-positive images end beyond 200 km or lack a valid prediction. These are overlapping conditional diagnostics with different denominators. The final count conditions on the reference subset, not on independent verification of the evaluated agent's successful search. They locate invocation, targeting, and coordinate decisions for further inspection without treating them as a partition of all failures.

J Training Cache and Inference Cost

Offline training. The RL tools serve image and text results from frozen caches, separating repeated rollout optimization from live API calls. For the retained cache inventory used in the cost analysis, 40,873 image-search calls contain 414,358 result rows, and 27,556 text-search calls contain 136,123 result rows. Calls, result rows, and unique images are different counting units; this inventory is reported separately from training-split sample counts. For any specified run, the counterfactual search-call budget is $N_{\text{images}}\, N_{\text{epochs}}\, N_{\text{rollouts}}\, \bar{c}_{\text{search}}$. Its interpretation depends on the run's dataset version and sampling configuration.

Final-agent tool budget. On the full 2,997-image Im2GPS3k evaluation, the final REVERSE agent averages 2.981 tool calls per image: 0.228 zoom, 1.145 image-search, and 1.608 text-search calls. The mean external-search count is 2.753, with a 90th percentile of four calls. These counts describe the final agent's saved evaluation traces.

Table 12: Service-profile latency in seconds under the specified single-request setting.

Component	Mean	P50	P90	P95	Max
End-to-end	25.60	17.07	57.26	71.10	106.83
Generation	5.23	5.58	6.53	6.67	7.55
All tools	20.36	10.75	52.54	65.23	100.25
Image search	18.76	8.31	52.26	64.02	97.67
Text search	1.56	0.00	3.24	3.92	14.80

Tool-service latency profile. A separate 50-image service profile uses Qwen3-VL-4B-Instruct on one NVIDIA A800 80 GB GPU with SGLang, a maximum of three turns, and concurrency one for generation and search. This profile uses the base instruction model to measure the shared service

path; it is not a latency benchmark of the final REVERSE checkpoint. All 50 requests contribute to the timing distribution, including 30 without parsed final coordinates. Table 12 separates generation and tool-service time. Search accounts for most of the measured wall time, locating the system component to target with caching or concurrent execution.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state that REVERSE is a training framework for agentic geolocation using GRPO-based RL with process rewards. All claimed improvements are supported by experiments in Section 4. Limitations are acknowledged in the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in the Limitations section of the paper, including reliance on external search APIs at inference time, performance gaps at coarse distance thresholds, and benchmark geographic biases.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper does not include theoretical results or proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Full hyperparameters are in Appendix C. The system prompt, tool definitions, and inference trace are in Appendix A. Data construction details are in Appendix B. The base model (Qwen3-VL-4B-Instruct) is publicly available.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will release training code, the cold-start SFT corpus, and model checkpoints upon acceptance. The training framework (VERL) and rollout engine (SGLang) are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details are provided in Appendix C and Appendix B. Evaluation follows the standard Im2GPS3k and YFCC4k protocol described in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We report point estimates on fixed benchmark splits and paired per-image transition counts. Multi-seed uncertainty and confidence intervals are not reported. Deterministic decoding does not eliminate sampling or training variability.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Per-stage hardware, node counts, optimization settings, and training steps are specified in Appendix C. Tool-service latency is profiled separately in Appendix J.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All datasets used (Im2GPS3k, YFCC4k, MP16-Pro) are publicly available research benchmarks. No private or sensitive personal data is collected or used.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Section 5 discusses privacy risks from geolocating individuals or sensitive locations and deployment safeguards, including metadata stripping and blurring of faces and private spaces.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: REVERSE is trained on publicly available benchmark datasets. We do not release a standalone inference API. Code and model release will follow standard responsible disclosure practices.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and models are properly cited. VERL and SGLang are used under their open-source licenses. External APIs (Oxylabs, Tavily) are used under their commercial terms of service.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The cold-start SFT corpus and RL training splits are documented in Appendix B with filtering criteria, statistics, and cache details. Checkpoints and code will be released with documentation upon acceptance.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subjects are involved in this research.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are central to this work. Kimi-K2.6 is used as an oracle agent to generate cold-start SFT trajectories (Appendix B). Qwen3-VL-4B-Instruct is the base model trained by REVERSE. Both are described in detail in the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.