LocalAgent: Collaborative Agentic Verification for Fine-Grained Instance-Level Consistency

Jiachen Guo[1,*], **Xinshan Zhu**[1,*], **Boyun Wang**[3,5], **Yanyan Liang**[3], **Jun Wan**[2,3], **Feng Lin**[4], **Ajian Liu**[2,3,4,†]

* Equal Contributors, † Corresponding Author

[1] School of Electrical and Information Engineering, Tianjin University
[2] MAIS, Institute of Automation, Chinese Academy of Sciences (CASIA)
[3] SCSE, Macau University of Science and Technology (MUST)
[4] State Key Laboratory of Blockchain and Data Security, Zhejiang University
[5] AIX, University of Macau

{gjiachen2025,xszhu}@tju.edu.cn, ajian.liu@ia.ac.cn

Abstract

Multimodal large language models (MLLMs) have made rapid progress in general visual understanding, yet they still struggle with fine-grained instance-level consistency verification, where the goal is to determine whether two images depict the exact same physical object under varying viewpoints, illumination, and backgrounds. This task requires both global semantic perception and precise examination of subtle local evidence, since semantically similar objects may differ only in small suspicious regions. To address this challenge, we propose **LocalAgent**, a collaborative agentic framework that integrates the complementary strengths of a generalist MLLM and lightweight specialist verifiers. The MLLM first reasons globally to identify suspicious local regions, which are then delegated to a Local Evidence Verifier (LEV) toolkit for fine-grained appearance and geometric matching. LEV comprises **Local Appearance Consistency (LAC)**, which estimates appearance similarity under illumination and color changes, and **Local Geometric Correspondence (LGC)**, which estimates spatial correspondence under viewpoint changes. LEV returns structured scalar evidence that the MLLM integrates for the final decision. We further introduce an auxiliary-capped reinforcement fine-tuning (RFT) strategy, in which answer correctness dominates auxiliary tool-use and format signals, enabling the policy to learn when verification is necessary. Experiments show that LocalAgent improves mean accuracy by **4.9%** over the strongest CoT-RFT baseline while invoking a verifier on only **12.13%** of samples and preserving the general multimodal capabilities of the underlying MLLM. The open-source repository is available at `https://github.com/Firefly-zoo/LocalAgent-main`.

1 Introduction

Multimodal Large Language Models (MLLMs) have demonstrated remarkable proficiency in general visual understanding, yet their reliability falters in fine-grained instance-level consistency verification. This task requires the model to determine whether two images depict the exact same physical object despite drastic variations in viewpoint, illumination, and background, while a different object may look nearly identical [1]. Unlike category-level recognition, the decision therefore often hinges on subtle local differences in texture, geometry, or brand-specific details. As illustrated in Fig. 1, current MLLMs can err by over-relying on coarse-grained global semantic perception while overlooking decisive local cues. Related perceptual weaknesses of MLLMs have been documented by several recent benchmarks [2–7].

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

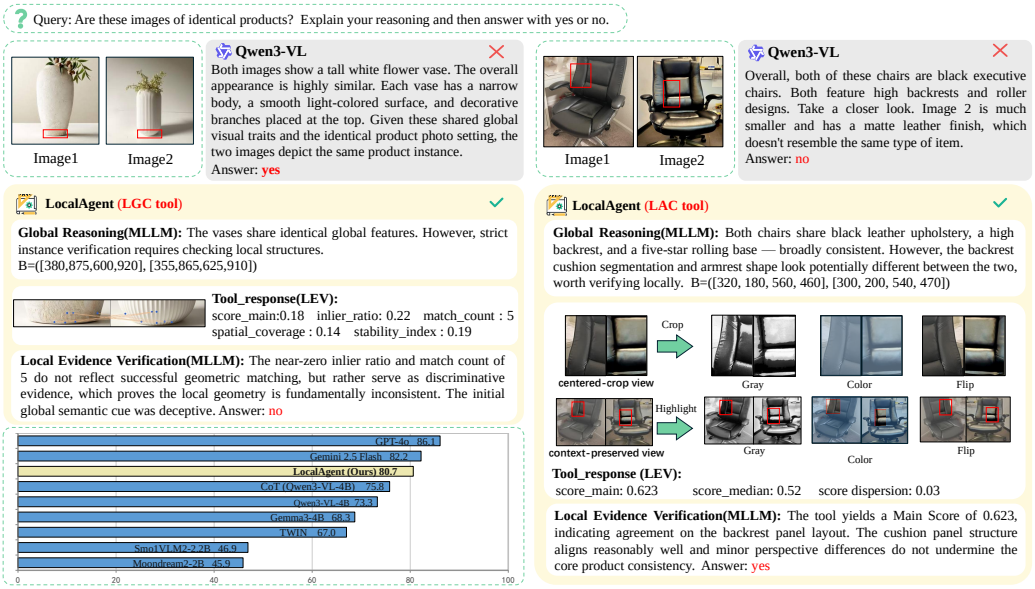


Figure 1: The motivation of LocalAgent. Standard MLLMs over-rely on global similarity and fail on instance-level verification in both directions. LocalAgent uses the MLLM as a spatial controller to localize suspicious regions, delegates them to the LEV toolkit (LAC, LGC) for invariant matching, and integrates the returned structured scalar evidence into the final decision.

To address these limitations, prior research has explored chain-of-thought (CoT) reasoning, visual backtracking, and tool-augmented inference [8–12]. While these methods enhance general reasoning robustness, they are not designed specifically for instance-level consistency verification. Moreover, providing a localized crop does not by itself guarantee reliable instance-level reasoning, because a localized crop does not isolate the discriminative cues from the rest of the image. We argue that an effective verification system should not merely increase reasoning complexity, but must strategically localize suspicious regions and delegate them to specialized modules capable of matching the localized regions directly in the visual domain.

The fundamental issue is not that MLLMs lack reasoning capacity, but that their vision encoders are optimized for cross-modal alignment. This is effective for global semantic perception, yet such encoders tend to miss fine-grained visual details [4, 13], such as the subtle texture, geometric, and brand-specific cues that determine instance-level identity. Motivated by this, we propose LocalAgent, which keeps the MLLM in charge of what it does well, namely global semantic perception and the localization of suspicious regions [14–16], while delegating the fine-grained discriminative judgment that lies beyond its perceptual reach to a Local Evidence Verifier (LEV) toolkit built on visual-domain features. Unlike tools that return raw visual crops, LEV provides structured, interpretable evidence that is fed back into the MLLM, enabling an iterative decision-making process.

To train this selective verification behavior, we introduce an auxiliary-capped Reinforcement Fine-Tuning (RFT) strategy. The training objective prioritizes final answer correctness while incorporating tool invocation consistency and output format as auxiliary signals, preventing degenerate strategies such as unconditional tool invocation. We further construct weak behavioral prior labels to guide selective tool invocation without additional human annotation. Experimental results demonstrate that LocalAgent improves mean accuracy by 4.9% over the strongest CoT-RFT baseline while invoking a verifier on only 12.13% of samples and preserving the general multimodal capabilities of the underlying MLLM.

Our contributions are as follows.

- We propose LocalAgent, a collaborative framework that decomposes fine-grained instance-level consistency verification into global perception, spatial localization, and local evidence verification. The MLLM provides global semantic reasoning and identifies potentially

discriminative regions, while the LEV toolkit applies appearance and geometric matching to those regions and returns structured evidence for the final decision.

- We design structured scalar evidence through which LEV's matching results are surfaced to the MLLM as interpretable indicators rather than as visual content, allowing verification evidence to be integrated through textual reasoning without re-feeding by the vision encoder.
- We introduce an auxiliary-capped RFT strategy with weak behavioral priors that allows the policy to learn when verification is necessary, rather than relying on hand-crafted invocation rules, improving mean accuracy by 4.9% across six diverse data sources while invoking a verifier on only 12.13% of samples.

2 Related Work

2.1 Fine-Grained Recognition and MLLMs

Fine-grained visual modeling aims to distinguish instances with highly similar global appearances, where the key challenge lies in capturing subtle yet discriminative local details. Despite strong general visual understanding, MLLMs remain unreliable in fine-grained instance-level consistency verification. POPE [5] reveals object hallucination in MLLMs, and HallusionBench [2] exposes entangled language hallucination and visual illusion. MMVP [4] and BLINK [3] further show that MLLMs miss fine-grained visual patterns and correspondences that humans perceive easily. These weaknesses have been linked to vision encoders trained for cross-modal alignment, which can attenuate instance-specific detail [13, 17]. TWIN [1] addresses this through targeted task formulation, yet still relies on global semantic perception without explicit mechanisms for invariant matching under complex physical transformations. LocalAgent reframes this as a collaborative verification process, leveraging the MLLM as a spatial controller to localize suspicious regions while delegating precise invariant matching to specialized verifiers trained on visual signals alone, with results surfaced as structured scalar evidence.

2.2 Multimodal Reasoning and Tool-Augmented Inference

MLLMs have been augmented with chain-of-thought reasoning [8], visual backtracking [9], and tool invocation [10] to enhance decision-making. Frameworks such as MM-REACT [8] and ViperGPT [10] interleave reasoning with external tool execution, while recent methods including DeepEyesV2 [18] and VisTA [19] employ reinforcement learning to optimize tool invocation strategies. However, these approaches target general-purpose reasoning and rely on visual feedback such as cropping or region enlargement. LocalAgent instead delegates suspicious regions to the LEV toolkit, comprising LAC and LGC, which perform invariant matching directly on raw visual features and return structured, interpretable evidence for iterative decision-making. A complementary auxiliary-capped RFT strategy further enables selective tool invocation without hand-crafted triggering rules.

3 Methodology

3.1 Overview

LocalAgent decomposes instance-level verification into three roles (Fig. 2). The MLLM acts as a spatial controller responsible for global semantic perception and the localization of suspicious regions. The Local Evidence Verifier (LEV) toolkit operates on vision-only features and returns structured scalar indicators. An RFT-trained policy governs whether verification is invoked at all. This division allocates each subtask to the component best suited to it.

3.2 The MLLM as Spatial Controller

Within LocalAgent, the MLLM is not bound to a fixed output role. Given an input image pair, the MLLM first performs global semantic perception to form an initial reasoning state. When this assessment is sufficient, the MLLM commits directly to a final answer, consistent with its strength on holistic understanding. When the global assessment is inconclusive, typically when two instances

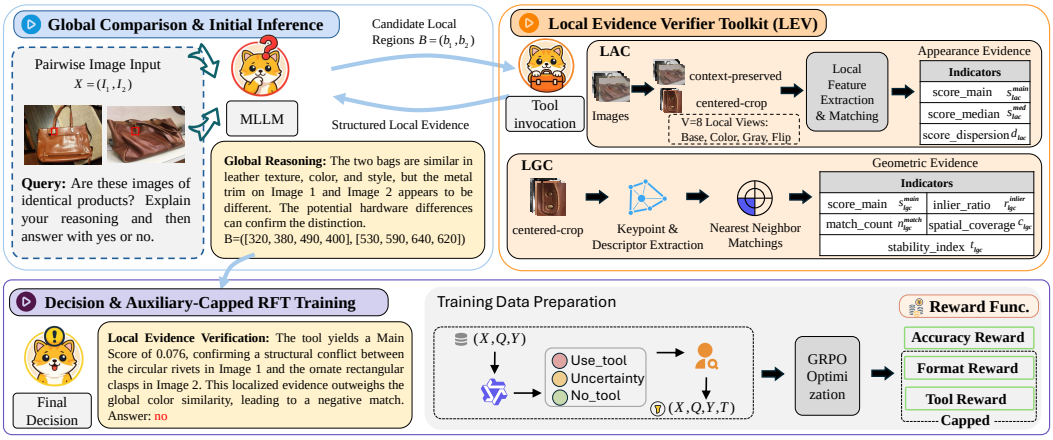


Figure 2: Overview of the LocalAgent framework. Given an image pair, the MLLM performs initial global semantic perception and either commits to a direct answer or, when the assessment is inconclusive, acts as a spatial controller to localize suspicious regions. These regions are conditionally delegated to the LEV toolkit, which returns structured scalar evidence that the MLLM integrates with its initial reasoning state to produce the final prediction. The strategic verification policy is optimized via auxiliary-capped RFT.

share strong semantic similarity yet may differ in localized details, the MLLM acts as a spatial controller and outputs a pair of region coordinates $B = (b_1, b_2)$, specifying where verification should be performed. In practice, the MLLM is prompted under a fixed instruction template to emit B as normalized $[x_1, y_1, x_2, y_2]$ coordinates within `<tool_call>` tags.

This division aligns with two distinct visual abilities. Global semantic perception and region localization rely on semantic encoding, whereas deciding whether two localized regions correspond to the same physical instance requires fine-grained visual detail, and is therefore delegated to LEV (Sec. 3.3). After LEV returns structured evidence, the MLLM integrates it with the initial reasoning state to produce the final prediction, closing the loop without re-feeding raw visual content.

3.3 Local Evidence Verifier Toolkit

The LEV toolkit supplies the discriminative evidence required when global semantic perception alone is insufficient. Matching is performed on features from encoders trained purely on visual signals, rather than on representations aligned with language. Self-supervised objectives at the patch level (DINO [20]) and keypoint-descriptor learning (XFeat [21]) preserve spatial and textural detail that can be attenuated by cross-modal alignment. We instantiate LEV with two sub-modules targeting complementary failure modes. DINO and XFeat are chosen as representative, well-established backbones for their respective categories. LAC is built upon DINO and performs patch-level appearance matching that remains stable under photometric variations where local structure is preserved despite changes in surface coloring. LGC is built upon XFeat and performs keypoint-based geometric matching that remains effective under viewpoint changes where appearance breaks but spatial correspondence is recoverable. The contribution of LEV lies not in the choice of backbones but in the unified scalar evidence interface that surfaces their outputs to the MLLM, enabling integration through evidential reasoning.

3.3.1 Local Appearance Consistency (LAC)

Given the localized regions B, LAC builds $V = 8$ view variants per region so that its estimate is less sensitive to background and photometric changes. For each region, it generates a context view that highlights the region within the full image and a crop view that isolates it from the background. Each view is kept in its original form and is also transformed by color jittering, grayscale conversion, and horizontal flipping, which gives four variants per view. For each view v, we extract patch-level DINO

features and establish patch correspondences by cosine similarity, which yields the consistency ratio

$$s_v = \frac{|M^{(v)}|}{N^{(v)}}, \tag{1}$$

where $|M^{(v)}|$ denotes the number of stable patch-level matches and $N^{(v)}$ represents the total number of patches. To provide LocalAgent with interpretable feedback, LAC aggregates these scores into a structured evidence triplet $z_{\text{lac}} = (s_{\text{lac}}^{\text{main}}, s_{\text{lac}}^{\text{med}}, d_{\text{lac}})$. Specifically, the main appearance score ($s_{\text{lac}}^{\text{main}}$) is the mean of the top-k scores, reflecting similarity under the most favorable views. The median appearance score ($s_{\text{lac}}^{\text{med}}$) captures the overall central tendency, and the view dispersion (d_{lac}) is the standard deviation across all views, reflecting the stability of the evidence. This compact representation allows the MLLM to perform evidential reasoning over explicit similarity indicators rather than high-dimensional raw features.

3.3.2 Local Geometric Correspondence (LGC)

The Local Geometric Correspondence (LGC) module characterizes spatial structural consistency between localized regions, providing robustness against drastic viewpoint and geometric deformations. Unlike LAC, the geometric estimation in LGC is less sensitive to color distribution. Therefore, the construction of view variants prioritizes perturbations in contrast and orientation by applying three fixed transformations, namely base, contrast equalization, and horizontal flip.

For each view v, XFeat [21] as a lightweight local feature extractor is utilized to identify keypoints and descriptors. A set of candidate matches $C^{(v)}$ is established based on descriptor similarity, which is subsequently filtered into an inlier set $I^{(v)} \subseteq C^{(v)}$ via robust geometric estimation. This process suppresses unstable correspondences induced by background clutter or repetitive textures. The single-view geometric score is defined as

$$g_v = \frac{|I^{(v)}|}{|C^{(v)}|}, \tag{2}$$

representing the proportion of matches supported by a consistent spatial relationship. To provide LocalAgent with comprehensive structural feedback, LGC aggregates these scores into a five-dimensional evidence vector $z_{\text{lgc}} = (s_{\text{lgc}}^{\text{main}}, r_{\text{lgc}}^{\text{inlier}}, n_{\text{lgc}}^{\text{match}}, c_{\text{lgc}}, t_{\text{lgc}})$. Specifically, the main geometric score ($s_{\text{lgc}}^{\text{main}}$) reflects the overall correspondence strength across all views, and the inlier ratio ($r_{\text{lgc}}^{\text{inlier}}$) measures matching precision. The effective match count ($n_{\text{lgc}}^{\text{match}}$) prevents reliance on sparse points. The spatial coverage (c_{lgc}) quantifies the distribution of matches to distinguish point-wise versus structural alignment, and the geometric stability (t_{lgc}) reflects evidence consistency under varying conditions.

3.4 Learning When to Verify

Many samples are resolvable through global semantic perception alone, whereas unconditional LEV invocation incurs unnecessary overhead. The decision of when to invoke verification thus becomes a learned behavior in itself, one that hand-crafted rules cannot easily capture across heterogeneous instance distributions. We formulate this as a strategic policy optimized via Group Relative Policy Optimization (GRPO)-based RFT [22, 23], with the reward signal dominated by final answer correctness and shaped by auxiliary signals on tool-use behavior and output format.

Weak behavioral priors. To stabilize early exploration and prevent data leakage, we construct weak behavioral priors on the training split using only its ground-truth labels. For each sample, the base model is evaluated under two zero-shot conditions, without tool access and with the actual LEV outputs supplied as reference feedback. A sample is labeled `No_tool` when the model answers correctly without tools, and `Use_tool` when it answers correctly only with tools. A held-out fraction of both categories is relabeled as `Uncertainty`, which breaks the deterministic `Use_tool`/`No_tool` boundary and serves as label-smoothing regularization for tool-use behavior. The labels shape only the auxiliary reward R_{tool} and never enter R_{acc}, which remains supervised exclusively by the original ground-truth answers.

Table 1: Main quantitative results (Accuracy, %). P, M, and T denote Pair, Mul, and Total accuracy. The leading Mean, Pair, and Mul columns are the unweighted averages over the six data sources. † indicates closed-source models. Bold marks the best open-source result in each column.

Method	Mean	Pair	Mul	TWIN-Eval			ILIAS			Landmarks			MET			CUB			Inquire		
				P	M	T	P	M	T	P	M	T	P	M	T	P	M	T	P	M	T
Closed-source Models																					
GPT-4o†	86.1	91.1	81.1	92.2	71.2	81.7	95.0	95.5	95.3	87.3	74.7	81.0	86.3	72.9	79.6	91.7	86.0	88.9	94.3	86.3	90.3
Gemini2.5 Flash†	82.2	87.0	77.5	89.1	68.1	78.6	92.2	90.5	91.4	85.4	77.0	81.2	82.6	71.9	77.3	84.1	77.9	81.0	88.3	79.4	83.9
Open-source Baselines																					
PerceptionLM-3B	37.8	50.0	25.5	46.9	25.2	36.1	49.6	25.0	37.3	51.5	25.7	38.6	50.4	26.1	38.3	49.6	24.9	37.3	52.1	26.1	39.1
Moondream2-2B	45.9	61.6	30.2	64.5	30.0	47.3	58.7	30.3	44.5	59.8	25.2	42.5	84.3	27.0	55.7	50.6	43.3	47.0	51.9	25.1	38.5
Gemma3-4B	68.3	86.1	50.4	68.1	39.5	53.8	88.4	46.5	67.5	**91.2**	55.2	73.2	**87.9**	57.5	72.7	91.2	55.2	73.2	90.0	48.7	69.4
SmolVLM2-2.2B	46.9	64.9	28.8	54.7	25.1	39.9	52.2	29.7	41.0	54.1	29.6	41.9	73.9	27.9	50.9	81.5	35.0	58.3	73.0	25.7	49.4
Qwen3-VL-4B	73.3	76.8	69.7	**89.1**	**76.7**	**82.9**	**89.2**	**79.3**	**84.3**	50.9	60.1	55.5	54.0	50.1	52.1	89.6	77.5	83.6	87.9	74.6	81.3
RFT Baselines																					
CoT (Qwen3-VL-4B)	75.8	83.4	68.1	86.6	66.4	76.5	84.4	75.8	80.1	75.7	52.9	64.3	76.9	58.9	67.9	84.4	79.7	82.1	92.3	75.0	83.7
TWIN	67.0	86.7	47.2	88.9	45.7	67.3	79.9	43.7	61.8	80.8	34.9	57.9	85.4	46.5	66.0	92.5	57.7	75.1	92.7	54.6	73.7
Ours																					
LocalAgent	**80.7**	**87.6**	**73.9**	80.5	68.5	74.5	82.7	76.4	79.6	87.3	**65.0**	**76.2**	86.7	**71.9**	**79.3**	**93.2**	**82.5**	**87.9**	**94.9**	**78.8**	**86.9**

Auxiliary-capped reward. The reward combines an answer-correctness term R_{acc}, a tool-use term R_{tool} aligned with the prior label $c \in \{\texttt{Use_tool}, \texttt{No_tool}, \texttt{Uncertainty}\}$, and a format term R_{fmt} enforcing well-formed `<tool_call>` and `<answer>` tags. R_{tool} asymmetrically encourages verification on `Use_tool` samples and restraint on `No_tool` samples without dictating fixed rules. A valid tool call receives a positive reward on `Use_tool` samples, a small penalty on `No_tool` samples, and an intermediate positive reward on `Uncertainty` samples. Not invoking the verifier reverses the sign on the first two groups and is neutral on the third. Failed, repeated, or malformed calls receive the lowest tool reward regardless of the label. A naive linear combination of the three terms, however, would allow the policy to accumulate auxiliary rewards even when the final answer is wrong, yielding degenerate strategies that prioritize tool-use behavior over discriminative accuracy. The auxiliary contribution is therefore bounded whenever the prediction $\hat{y}$ is incorrect

$$r = \begin{cases} \alpha R_{\text{acc}} + \beta R_{\text{tool}} + \gamma R_{\text{fmt}}, & \hat{y} = y, \\ \alpha R_{\text{acc}} + \max(\min(\beta R_{\text{tool}} + \gamma R_{\text{fmt}},\, \delta)\,,\, -\epsilon)\,, & \hat{y} \neq y, \end{cases} \tag{3}$$

where δ and ϵ are small positive constants. The cap δ prevents the policy from accumulating positive auxiliary reward on wrong answers, while the floor $-\epsilon$ bounds the auxiliary penalty so that it does not outweigh the answer term. Within these bounds, negative auxiliary penalties still discourage degenerate tool-use behavior.

4 Experiments

4.1 Experimental Setup

Train Data and Protocols. We conduct experiments following the evaluation sets and protocols established by the TWIN benchmark [1]. As a large-scale VQA dataset designed for fine-grained visual perception, TWIN includes positive, negative, and hard negative pairs. TWIN is constructed from product images sourced from Amazon Reviews, with visually similar hard negatives selected and manually screened during dataset construction. For the RFT stage, we select training samples with an emphasis on hard negative pairs with very high visual similarity and on positive pairs with large changes in viewpoint, illumination, or background.

Benchmarks and Evaluation Metrics. The evaluation suite spans six distinct data sources to ensure robust generalization, namely TWIN-Eval for in-distribution testing, ILIAS [24] for retail products, Landmarks [25] for geographical sites, MET [26] for artwork, CUB [27] for birds, and Inquire [28] for flora and fauna. In total, 12k queries across these fine-grained domains are evaluated. We utilize two task formats for assessment. The pairwise task requires a binary judgment on whether two images correspond to the same instance. The multiple-choice task presents a query image alongside three candidate images and asks the model to determine how many candidates depict

Table 2: Ablation study on the LEV toolkit. Pair/Mul/All Call report the tool-invocation rates for each variant.

Method	Performance (%)			Invocation Rate (%)		
	Mean	Pair	Mul	Pair	Mul	All
Only LAC	78.82	85.45	72.18	31.80	10.77	21.29
Only LGC	79.94	**88.57**	71.32	30.47	8.53	19.50
LAC+LGC (Ours)	**80.70**	87.55	**73.85**	7.85	16.40	12.13

Table 3: Ablation results of the reward function components and gating mechanisms on instance level verification tasks.

Method	Performance (%)			Invocation Rate (%)		
	Mean	Pair	Mul	Pair	Mul	All
w/o weak prior	79.39	86.55	72.23	38.45	22.90	30.68
Hard gating	80.23	**87.72**	72.73	10.58	17.83	14.21
LocalAgent (Ours)	**80.70**	87.55	**73.85**	7.85	16.40	12.13

the same physical instance as the query. Performance is quantified using mean accuracy alongside specific metrics for each task format across the six sources.

General Capability Benchmarks. To ensure that the RFT process does not compromise general multimodal and linguistic capabilities, evaluations are performed on two sets of general benchmarks. Vision-language benchmarks include MME-RealWorld-Lite [29], AI2D [30], SEED [31], ChartQA [32], TextVQA [33], MMMU [34], and BLINK [3]. Text-only benchmarks include MMLU [35], HellaSwag [36], and GSM8K [37].

Baselines. The evaluation includes a broad spectrum of comparison methods. We select closed-source models including GPT-4o [38] and Gemini2.5 Flash [39] as references for frontier performance. Open-source baselines comprise models of similar parameter scales such as Qwen3-VL-4B [40], PerceptionLM-3B [41], Moondream2-2B [42], Gemma3-4B [43], and SmolVLM2-2.2B [44]. We also compare against the TWIN model [1] and a CoT-RFT baseline, denoted CoT (Qwen3-VL-4B) in the tables, which is trained on the same dataset as LocalAgent using RFT with accuracy and format rewards and without any external tools.

4.2 Main Results

Table 1 presents the comparative results across the six data sources. LocalAgent improves mean accuracy by 4.9% over the strongest RFT baseline CoT (Qwen3-VL-4B) while invoking a verifier on only 12.13% of samples. Paired-bootstrap confidence intervals and McNemar tests confirm that this improvement is statistically significant (Appendix F). These results further narrow the gap with closed-source frontiers like Gemini2.5 Flash. LocalAgent improves both the mean pairwise and the mean multiple-choice accuracy. On TWIN-Eval, TWIN reaches 88.9% accuracy on pairwise queries but 45.7% on multiple-choice queries. In contrast, LocalAgent maintains balanced performance across the two formats. On TWIN-Eval, however, LocalAgent remains below the untrained base model, indicating that the current RFT policy and selective verification strategy do not fully match this benchmark's evaluation distribution. We therefore treat this benchmark-specific regression as a limitation of the current training setup rather than attributing it to a single mechanism. LocalAgent improves over CoT-RFT on four of the five out-of-distribution sources, while its mixed performance on ILIAS and TWIN-Eval indicates that the current training and selective-verification setup does not transfer uniformly across evaluation distributions.

Comparison with crop-based verification. To test whether re-feeding localized crops is sufficient, we compare LocalAgent with Qwen3-VL-4B equipped with a crop tool that returns the image regions inside the predicted boxes, trained and evaluated under the same protocol. As shown in Table 4, LocalAgent improves pairwise accuracy from 76.81% to 87.55% and mean accuracy from 75.64% to 80.70%, while its multiple-choice accuracy is slightly lower. Pairwise verification maps localized evidence to one binary decision. Multiple-choice verification also requires comparing evidence across three candidates, which the current policy does not jointly aggregate.

4.3 Ablation Study

Tool Ablation. To evaluate the distinct contributions of individual verification modules, we analyze the performance and invocation patterns of the LAC and LGC components in Table 2. It shows that LAC and LGC each surpass the CoT-RFT baseline individually, while their integration reaches 80.70% mean accuracy, suggesting the complementarity of appearance and geometric evidence. The coordinated policy also invokes LEV less often overall, on 12.13% of samples compared with 21.29%

Table 4: Accuracy (%) of crop-based tool augmentation and LocalAgent on the pairwise, multiple-choice, and mean settings.

Method	Pair	Mul	Mean
Qwen3-VL-4B + crop	76.81	**74.47**	75.64
LocalAgent (Ours)	**87.55**	73.85	**80.70**

Table 5: Tool invocation behavior of LocalAgent. Easy and Hard denote invocation rates (%) on baseline-correct and baseline-incorrect samples.

Setting	Easy	Hard	Call Rate
Pair	6.81	11.27	7.85
Mul	13.71	22.55	16.40



Figure 3: Qualitative results of LocalAgent on pairwise verification cases.

and 19.50% for the single-tool variants. This decrease comes from the pairwise task, whereas the multiple-choice call rate rises from 10.77% and 8.53% to 16.40%.

Reward Ablation. Table 3 examines two design choices. Without the weak prior, the pairwise invocation rate rises to 38.45% and mean accuracy drops by 1.31%, which suggests that the prior helps the policy identify samples that benefit from verification. Replacing the auxiliary cap with hard gating removes all positive auxiliary reward on incorrect predictions, leading to a 0.47% accuracy drop. Both components therefore contribute to the final performance. A sensitivity analysis of the auxiliary bounds is given in Appendix E.

4.4 Diagnostic Analysis

Tool Invocation Behavior. To examine whether LocalAgent learns a genuinely selective verification policy rather than invoking tools indiscriminately, we analyze invocation behavior across task formats and sample difficulty. As shown in Table 5, the overall call rates remain low at 7.85% and 16.40% for pairwise and multiple-choice respectively, confirming that accuracy gains are not simply a product of increased tool usage. Across both settings, the call rate (hard) on baseline-incorrect samples is consistently higher than the call rate (easy) on baseline-correct. This difficulty-sensitive pattern indicates that the learned policy concentrates LEV invocations on harder cases across both task formats.

Table 6: Policy-level end-to-end inference latency under different invocation settings, reported in ms per sample on the pairwise and multiple-choice tasks.

Setting	Pair	Mul	Mean
Direct inference	**184.72**	**331.54**	**258.13**
Only LAC	290.50	350.48	320.49
Only LGC	294.87	379.18	337.03
LAC+LGC (Ours)	220.54	453.17	336.86
Always-call policy	400.53	820.29	610.41

Table 7: Accuracy (%) of pure visual matching baselines under the optimal threshold, computed on the full images and on the regions localized by Qwen3-VL-4B.

Method	Pair	Mul	Mean
DINO (full)	73.18	75.28	74.23
XFeat (full)	68.96	69.50	69.23
DINO (localized)	74.75	**75.36**	75.06
XFeat (localized)	70.88	71.33	71.11
LocalAgent (Ours)	**87.55**	73.85	**80.70**

Table 8: General capability evaluation (%). LocalAgent matches Qwen3-VL-4B across vision-language and text-only benchmarks.

	Vision-Language							Text-only		
Method	MME-RW-L	AI2D	SEED	ChartQA	TextVQA	MMMU	BLINK	MMLU	HellaSwag	GSM8K
Qwen3-VL-4B	29.03	84.20	73.05	75.00	70.35	77.20	65.8	68.80	86.90	89.54
LocalAgent	29.39	84.44	73.28	75.08	70.39	77.22	65.9	68.84	86.96	89.54

Inference Latency. Table 6 reports policy-level end-to-end latency. The selective LAC+LGC policy invokes tools on 7.85% of pairwise samples, 16.40% of multiple-choice samples, and 12.13% overall, keeping mean latency at 336.86 ms compared with 258.13 ms for direct inference. Selective invocation reduces mean latency from 610.41 ms/sample for the always-call policy to 336.86 ms/sample by avoiding unnecessary verifier executions, while the table makes the remaining per-sample overhead explicit.

Comparison with pure visual matching. To examine whether visual matching alone is sufficient for instance-level consistency verification, we evaluate two non-MLLM baselines built on the LEV backbones, DINO and XFeat. These baselines compute similarity scores between images and use a threshold t to obtain matching decisions. To avoid relying on an arbitrary threshold, we sweep t over a dense grid on each evaluation subset and report the best observed accuracy. In full-image matching, the whole images are compared without localization. For multiple-choice inputs, similarity scores are computed between the query image and the three candidate images. In MLLM-localized matching, matching is performed on regions predicted by Qwen3-VL-4B, using the localization procedure described in Appendix A. As shown in Table 7, MLLM-localized matching improves the mean accuracy of DINO from 74.23% to 75.06% and that of XFeat from 69.23% to 71.11%, suggesting that localization helps expose discriminative visual cues. Both baselines nevertheless achieve lower mean accuracy than LocalAgent, which reaches 80.70%. On the multiple-choice task, localized DINO achieves 75.36%, compared with 73.85% for LocalAgent.

4.5 General Capability Evaluation

A primary concern in RFT for domain-specific tasks is the potential for catastrophic forgetting of the model's general capabilities [45, 46]. We evaluate LocalAgent on ten general-purpose benchmarks spanning vision-language and text-only tasks. As shown in Table 8, LocalAgent matches or slightly exceeds the Qwen3-VL-4B baseline on every benchmark, indicating that our RFT improves fine-grained verification without degrading general capabilities.

4.6 Qualitative Analysis

Fig. 3 illustrates two complementary verification patterns. Low LAC main scores overturn deceptive global similarity, while a high main score confirms identity despite packaging-induced appearance shifts. These cases demonstrate that structured scalar evidence enables LocalAgent to resolve ambiguous cases beyond the reach of global semantic perception.

5 Conclusion

We present LocalAgent, a collaborative agentic framework for fine-grained instance-level consistency verification. The MLLM contributes global semantic perception and suspicious-region localization, while the LEV toolkit supplies appearance and geometric evidence from vision-only encoders, returned as structured scalar indicators that are integrated through textual reasoning rather than re-fed by the vision encoder. An auxiliary-capped RFT strategy further enables the policy to learn when verification is necessary. Experiments across six diverse data sources show that LocalAgent improves mean accuracy while preserving general multimodal capabilities, narrowing the gap to closed-source counterparts. Beyond instance-level verification, this paradigm of coupling MLLM-level semantic control with specialized vision-only verifiers suggests a general recipe for tasks where vision-language alignment alone can attenuate the fine-grained signals required by the task.

Limitations. LEV examines only the regions proposed by the MLLM, so the quality of its evidence depends on the localization step. The current policy does not aggregate evidence jointly across the candidates of a multiple-choice query. In addition, LocalAgent remains below the base model on TWIN-Eval (Appendix B) and is 30.5% slower than direct inference (Table 6).

Acknowledgments and Disclosure of Funding

This work was supported by the General Program of the National Natural Science Foundation of China (No. 62676278), the New Generation Artificial Intelligence—National Science and Technology Major Project (No. 2025ZD0123501), the National Natural Science Foundation of China (No. 62406320), the State Key Laboratory of Multimodal Artificial Intelligence Systems (No. MAIS20260206), the Open Research Fund of the State Key Laboratory of Blockchain and Data Security, Zhejiang University (No. A2542), and the AI Super Computing Platform of Macau University of Science and Technology. The authors declare no competing interests.

References

[1] Damiano Marsili, Aditya Mehta, Ryan Y Lin, and Georgia Gkioxari. Same or not? Enhancing visual perception in vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2026.

[2] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. HallusionBench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14375–14385, 2024.

[3] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pages 148–166, 2024.

[4] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? Exploring the visual shortcomings of multimodal LLMs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9568–9578, 2024.

[5] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 292–305, 2023.

[6] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. MME: A comprehensive evaluation benchmark for multimodal large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[7] Zhecan Wang, Junzhang Liu, Chia-Wei Tang, Hani Alomari, Anushka Sivakumar, Rui Sun, Wenhao Li, Hammad Ayyubi, Haoxuan You, Alvi Ishmam, et al. JourneyBench: a challenging one-stop vision-language understanding benchmark of generated images. *Advances in Neural Information Processing Systems*, 37:63110–63123, 2024.

[8] 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.

[9] Shuo Yang, Yuwei Niu, Yuyang Liu, Yang Ye, Bin Lin, and Li Yuan. Look-back: Implicit visual re-focusing in MLLM reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11694–11702, 2026.

[10] 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.

[11] Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-V: Exploring long-chain visual reasoning with multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9062–9072, 2025.

[12] Tianhong Gao, Yannian Fu, Weiqun Wu, Haixiao Yue, Shanshan Liu, and Gang Zhang. MMAT-1M: A large reasoning dataset for multimodal agent tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1484–1494, 2025.

[13] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. *Advances in Neural Information Processing Systems*, 37:87310–87356, 2024.

[14] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. GLaMM: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024.

[15] Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. Groma: Localized visual tokenization for grounding multimodal large language models. In *European Conference on Computer Vision*, pages 417–435. Springer, 2024.

[16] Hongyu Li, Jinyu Chen, Ziyu Wei, Shaofei Huang, Tianrui Hui, Jialin Gao, Xiaoming Wei, and Si Liu. LLaVA-ST: A multimodal large language model for fine-grained spatial-temporal understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8592–8603, 2025.

[17] Chaoya Jiang, Hongrui Jia, Haiyang Xu, Wei Ye, Mengfan Dong, Ming Yan, Ji Zhang, Fei Huang, and Shikun Zhang. MaVEn: An effective multi-granularity hybrid visual encoding framework for multimodal large language model. *Advances in Neural Information Processing Systems*, 37:101992–102010, 2024.

[18] Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. DeepEyesV2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

[19] 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.

[20] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[21] Guilherme Potje, Felipe Cadar, André Araujo, Renato Martins, and Erickson R Nascimento. XFeat: Accelerated features for lightweight image matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2682–2691, 2024.

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

[23] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[24] Giorgos Kordopatis-Zilos, Vladan Stojnić, Anna Manko, Pavel Suma, Nikolaos-Antonios Ypsilantis, Nikos Efthymiadis, Zakaria Laskar, Jiri Matas, Ondrej Chum, and Giorgos Tolias. ILIAS: Instance-level image retrieval at scale. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14777–14787, 2025.

[25] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2575–2584, 2020.

[26] Nikolaos-Antonios Ypsilantis, Noa Garcia, Guangxing Han, Sarah Ibrahimi, Nanne Van Noord, and Giorgos Tolias. The MET dataset: Instance-level recognition for artworks. In *Thirty-fifth conference on neural information processing systems datasets and benchmarks track (Round 2)*, 2021.

[27] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD birds-200-2011 dataset. 2011.

[28] Edward Vendrow, Omiros Pantazis, Alexander Shepard, Gabriel Brostow, Kate E Jones, Oisin Mac Aodha, Sara Beery, and Grant Van Horn. INQUIRE: A natural world text-to-image retrieval benchmark. *Advances in Neural Information Processing Systems*, 37:126500–126514, 2024.

[29] YiFan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, et al. MME-RealWorld: Could your multimodal LLM challenge high-resolution real-world scenarios that are difficult for humans? In *International Conference on Learning Representations*, volume 2025, pages 89655–89701, 2025.

[30] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pages 235–251, 2016.

[31] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. SEED-Bench: Benchmarking multimodal LLMs with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023.

[32] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pages 2263–2279, 2022.

[33] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019.

[34] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9556–9567, 2024.

[35] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

[36] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800, 2019.

[37] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[38] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[39] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[40] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[41] Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, et al. PerceptionLM: Open-access data and models for detailed visual understanding. *arXiv preprint arXiv:2504.13180*, 2025.

[42] Moondream. moondream2, 2025. URL `https://github.com/m87-labs/moondream`.

[43] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[44] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. SmolVLM: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

[45] Size Wu, Sheng Jin, Wenwei Zhang, Lumin Xu, Wentao Liu, Wei Li, and Chen Change Loy. F-LMM: Grounding frozen large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24710–24721, 2025.

[46] Ziang Yan, Zhilin Li, Yinan He, Chenting Wang, Kunchang Li, Xinhao Li, Xiangyu Zeng, Zilei Wang, Yali Wang, Yu Qiao, et al. Task preference optimization: Improving multimodal large language models with vision task alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29880–29892, 2025.

Appendix

A Localization Quality Analysis

A natural concern is whether the MLLM, when acting as a spatial controller, produces region coordinates that actually cover the discriminative cues required for downstream verification. To assess localization quality, we let the base Qwen3-VL-4B model output region coordinates on all 12k samples in our evaluation set under both task formats. Under the pairwise task, the model produces a pair of bounding boxes, one per image. Under the multiple-choice task, the model first selects the most suspicious image among three references and then produces bounding boxes on both the selected image and the query image.

We then employ Qwen3-VL-30B-A3B-Instruct as an independent judge to evaluate whether each predicted bounding box covers the discriminative components relevant to instance identity and remains usable for subsequent local verification. The judge model is given the original image pair, the predicted bounding boxes, and the verification query, and is asked to determine whether the localization is acceptable. The judge accepts 94.22% of the boxes in the pairwise task and 87.32% in the multiple-choice task. The lower rate in the multiple-choice task reflects its additional step of selecting a candidate before localization.

B TWIN-Eval Regression Analysis

We further analyze why LocalAgent falls below the base model on TWIN-Eval. Most of this gap already appears without any verifier. The no-tool CoT-RFT control, trained on the same data with accuracy and format rewards only, drops from 82.9% to 76.5% on this source, whereas LocalAgent reaches 74.5%. To check whether the remaining change is uniform, we sample 1,000 TWIN-Eval examples, 500 pairwise and 500 multiple-choice, and use a larger MLLM as a blind stratification judge. Given the images, the question, and the ground-truth relation, the judge labels a sample as training-emphasis-aligned if it is a highly similar hard negative or a positive pair with substantial viewpoint, illumination, or background change, and assigns all remaining samples to the other group. The judge never sees predictions or correctness information from any evaluated model.

As shown in Table 9, LocalAgent leads the base model by 8.8% and CoT-RFT by 4.2% on the training-emphasis-aligned cases, with a 43.8% invocation rate. The aggregate regression concentrates in the other cases, where LocalAgent invokes LEV on 17.9% of samples and reaches 69.2% compared with 88.8% for the base model and 78.2% for CoT-RFT. LEV thus recovers performance on some difficult cases, but this descriptive analysis does not identify a single cause for the overall decrease. We therefore treat the benchmark-specific regression as a limitation of the current training and routing setup.

Table 9: Stratified TWIN-Eval accuracy (%) by visual characteristics associated with the training emphasis, with LocalAgent's LEV invocation rate per stratum.

TWIN-Eval subset	**Base**	**CoT-RFT**	**LocalAgent**	**LEV invocation**
Training-emphasis-aligned hard cases	72.1	76.7	**80.9**	43.8
Other cases	**88.8**	78.2	69.2	17.9

C Failure Mode Analysis

A qualitative audit of incorrect predictions identifies four recurring categories, which delimit where the framework does not yet work.

- Localization and visibility. Predicted boxes miss the identifying region, or occlusion, blur, and extremely small details leave insufficient evidence for any verifier to act on.
- Verifier ambiguity. Repetitive textures, large viewpoint changes, or sparse keypoints yield weak or mutually conflicting LAC and LGC evidence.
- Incomplete coverage. A single invocation does not capture all inconsistent regions when several exist.
- Policy and evidence integration. The MLLM omits a useful call, invokes an unnecessary verifier, or weighs the returned evidence incorrectly against its global impression.

These failure modes follow from the controller-bounded design and from the lack of joint evidence aggregation across several candidates in the multiple-choice task.

Table 10: Sensitivity of LocalAgent to the auxiliary bounds on incorrect predictions.

Setting	$(\delta,\ \epsilon)$	**Mean**
Default	(0.06, 0.12)	**80.7**
No auxiliary reward on incorrect answers	(0, 0)	79.5
Looser bounds	(0.10, 0.20)	77.1

D Implementation Details

We train the policy with GRPO based on Qwen3-VL-4B-Instruct using LoRA with rank 16 and $\alpha_{\text{LoRA}} = 32$ to the language backbone, while the visual encoder and the multimodal aligner are



Figure 4: **Illustrative failure modes.**

frozen. The main configuration uses 5 NVIDIA H20 GPUs, 1 for rollout and 4 for DDP training. Each prompt samples 8 rollout responses. We use AdamW with a learning rate of 1×10^{-5}, a weight decay of 0.1, and $\beta_1 = 0.9$, $\beta_2 = 0.95$. The learning rate follows cosine decay with a warmup ratio of 0.05, and gradient clipping is applied with a maximum norm of 0.5.

E Reward Sensitivity Analysis

We vary the auxiliary bounds (δ, ϵ) in Eq. 3, which apply to incorrect predictions, while keeping the reward coefficients fixed. Results are reported in Table 10. Removing the auxiliary reward on incorrect predictions lowers mean accuracy by 1.2%, and loosening the bounds lowers it by 3.6%. Bounding the auxiliary contribution on incorrect predictions therefore matters for the final performance.

F Statistical Reliability

To assess the sample-level statistical reliability of the reported improvement, we align the LocalAgent and CoT-RFT predictions at the question level and compute paired-bootstrap 95% confidence intervals and exact McNemar tests for the accuracy gain of LocalAgent over CoT-RFT. Results are reported in Table 11. The paired mean difference is positive and its confidence interval excludes zero in all three splits, and every McNemar test is significant, so the reported gain is not an artifact of a single evaluation run.

Table 11: Paired-bootstrap 95% confidence intervals and exact McNemar tests for the accuracy gain of LocalAgent over CoT-RFT.

Split	**95% bootstrap CI**	**McNemar p**
Overall	[4.27, 5.70]	4.034×10^{-41}
Pairwise	[3.28, 5.07]	7.849×10^{-20}
Multiple	[4.67, 6.95]	5.181×10^{-23}

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 state our three contributions (the collaborative agentic framework, the LEV toolkit with structured scalar evidence, and the auxiliary-capped RFT strategy), and the experimental results in Section 4 support the reported 4.9% improvement in mean accuracy over the CoT-RFT baseline.

   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: We discuss limitations in Section 4, the Limitations paragraph of Section 5, and Appendices B and C, including the regression on TWIN-Eval, the dependence on localization quality, and the lack of joint evidence aggregation in the multiple-choice task.

   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: The paper presents an empirical framework and does not contain formal theoretical results requiring 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: We specify the LocalAgent framework, the LEV toolkit (including the DINO and XFeat backbones), and the auxiliary-capped RFT objective with its reward components and bounds. Section 4 and Appendix D detail the training data, evaluation benchmarks, baselines, and hyperparameters.

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: The codes are publicly available at `https://github.com/Firefly-zoo/LocalAgent-main`. All evaluation data come from public benchmarks.

   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: Section 4.1 and Appendix D describe the data, protocols, and training hyperparameters, and Appendix E reports the sensitivity to the auxiliary bounds.

   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: [Yes]

   Justification: Appendix F reports paired-bootstrap 95% confidence intervals and exact McNemar tests for LocalAgent's gain over CoT-RFT, which are significant on the overall, pairwise, and multiple-choice splits.

   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: Experiment compute resources are reported in Appendix D.

   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: The research uses publicly available models and benchmarks and conforms with the NeurIPS Code of Ethics.

   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: LocalAgent narrows the performance gap between open-source and closed-source MLLMs on fine-grained instance-level verification, contributing to the accessibility of capable visual reasoning systems. The work poses no foreseeable negative societal impact beyond standard considerations applicable to general-purpose MLLMs.

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: The paper does not release models or datasets with a high risk of misuse. The framework is built on publicly available models and standard benchmarks.

    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 external assets used in this work are properly cited in the references with their original publications.

    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: We release the LocalAgent implementation at `https://github.com/Firefly-zoo/LocalAgent-main`.

    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: The work does not involve crowdsourcing or research with 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: The work does not involve human subjects and therefore does not require IRB review.

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: LocalAgent uses Qwen3-VL-4B-Instruct as its multimodal controller. Sections 3 and 4 and Appendix D describe its role, interaction with the LEV toolkit, and reinforcement fine-tuning.

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.