SHOPEYE: BRIDGING AGENTIC DATA CONSTRUCTION AND ACTIVE VISUAL VERIFICATION FOR FINE-GRAINED E-COMMERCE AIGC DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid advancement of AIGC technologies provides a convenient and effective means for product display, driving an exponential growth of AI-generated content on e-commerce platforms. While substantially boosting page views and transaction conversion rates, it introduces uncontrollable generative artifacts, such as distorted product features, severely impairing the user experience. However, existing benchmarks are predominantly confined to discerning whether an image is AI-generated, rather than conducting fine-grained quality assessment on such generated content. To bridge this gap, we present ShopEye, a full-lifecycle agentic framework for fine-grained AIGC defect diagnosis in e-commerce. ShopEye integrates two core components: ShopEye-Agent for automated data production, and ShopEye-Trainer for agentic model training. Specifically, ShopEye-Agent employs an automated multi-agent pipeline to generate and curate diagnostic data. This produces ShopEye-Bench, the first benchmark tailored for low-quality e-commerce AIGC, comprising 5K training and 1K test samples. It covers prevalent defect types in e-commerce scenarios, including product inconsistency, human deformity, unrealistic scenes, and text distortion. Nonetheless, detecting these defects typically requires fine-grained visual understanding. To enhance the model's diagnostic capability, ShopEye-Trainer incorporates diverse visual tools for perception augmentation, and optimizes the diagnostic policy via reinforcement learning. Specifically, it couples Local-Global Dual-View Supervised Finetuning to model fine-grained visual details, and Evidence-Weighted Policy Optimization to incentivize active visual tool invocation for evidence gathering. By combining ShopEye-Bench with our training paradigm, we develop ShopEye-4B and ShopEye-8B, two specialized multimodal agentic models at different scales. Extensive experiments demonstrate that both models consistently outperform strong baselines, validating the effectiveness and scalability of our training framework with only a modest amount of task-specific training data.

1 INTRODUCTION

In recent years, generative models like diffusion models (Ho et al., 2020) and flow matching (Lipman et al., 2023) have shown remarkable capabilities in controllable image generation and precise visual editing. These generative capabilities significantly reduce the reliance on manual labor, photography equipment, and physical studios in traditional content creation, greatly lowering production costs and improving overall efficiency. Correspondingly, these techniques have attracted widespread attention in e-commerce content production, with diverse applications ranging from virtual try-on and product display (Mao et al., 2025; He et al., 2026) to personalized advertising creation (Xu et al., 2026) and generative recommendation (Lin et al., 2025; Wang et al.; Lin et al., 2026). As a result, such AI-generated content (AIGC) is currently experiencing exponential growth across e-commerce platforms, substantially enriching the content ecosystem and enhancing the online shopping experience for consumers.

However, while AIGC greatly accelerates the automation of content creation, it also introduces a growing number of critical challenges (Zhang et al., 2025a). Due to their probabilistic nature and the lack of explicit physical priors, current generative models still struggle to guarantee fine-grained

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

detail fidelity and geometric consistency, frequently producing subtle yet severe visual anomalies. For instance, a generated image may inadvertently alter the garment structure, length, neckline, logos, patterns, or textures of a product, leading to significant product inconsistency. Such anomalous content easily misleads consumers by creating false expectations, directly driving up complaint and return rates while severely damaging brand reputation and consumer trust.

Driven by this pressing necessity, considerable research efforts have been dedicated to AIGC evaluation and detection (Wang et al., 2020; 2023; Yan et al., 2025; Liu et al., 2026a). However, prevailing benchmarks primarily focus either on binary real-vs-fake discrimination or on generic criteria such as instruction compliance, content preservation, and perceptual quality (Basu et al., 2023; Zhang et al., 2023; Ma et al., 2024; Yu et al., 2025; Ye et al., 2026). When applied to e-commerce scenarios, these approaches face several major challenges. First, AIGC anomalies in e-commerce can arise along multiple dimensions, including product consistency, human appearance, scene plausibility, and text rendering. Yet these domain-specific anomaly categories remain largely underexplored in existing general-purpose benchmarks. Second, these anomalies are inherently fine-grained and localized, often residing in microscopic regions of the product or human subject. Therefore, e-commerce AIGC anomaly assessment should not be limited to determining *whether a defect exists*. It should further identify *what type of defect it is* and *which visual evidence supports the diagnosis*. However, equipping models with such capabilities requires supervision that goes beyond image-level labels to include localized anomaly annotations and evidence-grounded reasoning, which substantially increases the difficulty and cost of data annotation.

To fill this gap, we introduce SHOPEYE, a full-lifecycle agentic framework for fine-grained AIGC defect diagnosis in e-commerce. SHOPEYE closes the loop from data construction to model optimization, tackling the two core challenges of annotation scarcity and insufficient fine-grained perceptual reasoning. To address the lack of high-quality supervision, we first develop SHOPEYE-AGENT, a multi-agent pipeline that autonomously synthesizes and curates high-fidelity diagnostic data, thereby avoiding the prohibitive cost of manual annotation. Based on this pipeline, we construct SHOPEYE-BENCH, the first dedicated benchmark for fine-grained e-commerce AIGC defect diagnosis. SHOPEYE-BENCH contains 5K training instances and 1K test instances, systematically covering product inconsistency, human deformity, unrealistic scenes, and text distortion. We further benchmark a range of advanced existing models on SHOPEYE-BENCH and find that current approaches struggle substantially with this task, revealing a clear gap in fine-grained defect diagnosis for e-commerce AIGC. To overcome these limitations and strengthen models' ability to perceive and reason over subtle defects, we propose SHOPEYE-TRAINER. It introduces a Local-Global Dual-View Supervised Fine-Tuning (SFT) strategy to capture visual evidence at multiple scales, together with a step-level tool reward in reinforcement learning that encourages active invocation of specialized visual tools for evidence acquisition. Built upon this training paradigm, our resulting models, SHOPEYE-8B and SHOPEYE-4B, substantially outperform competitive baselines while maintaining strong sample efficiency.

In summary, the main contributions of this paper are as follows:

- We formalize the task of fine-grained AIGC defect diagnosis in e-commerce and develop SHOPEYE-AGENT, an agentic pipeline for scalable diagnostic data construction. Based on this pipeline, we build SHOPEYE-BENCH, the first dedicated benchmark covering product inconsistency, human deformity, unrealistic scenes, and text distortion.
- We benchmark advanced existing models on SHOPEYE-BENCH and show that current approaches struggle substantially with fine-grained e-commerce AIGC defect diagnosis, revealing a clear gap in existing capabilities.
- We propose SHOPEYE-TRAINER, an agentic framework that combines local-global dual-view supervision with reinforcement learning via step-level tool rewards for fine-grained evidence-based diagnosis. Built upon this training paradigm, SHOPEYE-4B and SHOPEYE-8B substantially outperform competitive baselines while demonstrating strong sample efficiency.

2 RELATED WORKS

AIGC Detection and Quality Assessment. Existing studies on synthetic content follow two complementary lines. The first line detects whether an image is AI-generated. Early detectors formulate

the task as binary classification (Wang et al., 2020; 2023; Yan et al., 2025; Liu et al., 2026a). More recently, MLLMs have brought human-readable explanations to detection (Cao et al., 2024), and benchmarks such as LOKI (Ye et al., 2025), FakeClue (Wen et al., 2026), MMFakeBench (Liu et al., 2025), and Ivy-Fake (Jiang et al., 2026) evaluate explainable detection. The second line assesses how well an image is generated or edited. It has progressed from instruction compliance and fidelity (Basu et al., 2023; Zhang et al., 2023) to comprehensive multi-dimensional evaluation (Ma et al., 2024), unified editing across diverse task types (Yu et al., 2025), and reasoning-informed modifications (Zhao et al., 2025). However, both lines remain general-domain and cannot evaluate e-commerce-specific defects such as subtle deviations in product structure, texture, or logos.

MLLM Understanding in E-commerce. The strong multimodal reasoning ability of MLLMs has driven rapid progress in e-commerce understanding. One line of work learns product-oriented multimodal representations for retrieval-centric applications. For instance, the MOON series progresses from generative MLLM-based product understanding (Zhang et al., 2026a) to modality-balanced (Nie et al., 2026) and reasoning-aware designs (Wu et al., 2026a), culminating in large-scale industrial deployment (Fu et al., 2025). Another line applies MLLMs to deeper product understanding. Representative efforts include domain adaptation to attribute-centric e-commerce data (Nulli et al., 2026), purchase-intention distillation (Xu et al., 2024), omni-modal live-streaming understanding (Hu et al., 2026), and short-video commercial reasoning (Liu et al., 2026b). Most relevant to our work, AIGuard (Zhang et al., 2025a) pioneers AIGC risk detection in real-world e-commerce. However, its annotation remains at the image level, providing neither region-level evidence nor tool-calling capabilities for fine-grained anomaly identification.

Tool-Augmented MLLMs. Augmenting MLLMs with external tools (Schick et al., 2023; Qin et al., 2024) and active perceptual actions helps overcome resolution limits, with recent paradigms exploring dynamic zooming (Shen et al., 2025b; Zhu et al., 2026a), code-driven image manipulation (Zhang et al., 2026b), and multi-agent and agentic visual inspection (Zhang et al., 2026c; Li et al., 2026; Ding et al., 2026). On the training side, RL has been extended to multimodal alignment (Shen et al., 2025a), aided by process reward models (Lightman et al., 2024; Wang et al., 2025b) and group-relative policy optimization (Shao et al., 2024; Yu et al., 2026; Zhu et al., 2026b). Recent efforts further apply RL to incentivize tool invocation and "thinking with images" (Zheng et al., 2026; Su et al., 2026; Zhang et al., 2025b; Wu et al., 2026b; Ni et al., 2026; Chng et al., 2025). Most existing approaches focus on query-driven visual question answering, where the textual query provides semantic cues for locating relevant image regions. In contrast, ShopEye addresses a more open-ended diagnostic task in which anomaly locations are unknown, requiring global assessment of source–edited image pairs to guide adaptive local inspection and determine whether and where defects occur.

3 Problem Definition

Given a source product image I_{s} and an editing instruction q, an AIGC editing model $\mathcal{G}$ produces an edited image $I_{\mathrm{e}} = \mathcal{G}(I_{\mathrm{s}}, q)$. The goal of fine-grained AIGC anomaly detection is to determine ***whether***, ***where***, and ***why*** the editing process introduces visual defects into I_{e}, with the source image I_{s} serving as a reference. Unlike conventional AI-generated image detection, visual defects in e-commerce are often local, subtle, and reference-dependent. In collaboration with e-commerce experts, we characterize such defects along four complementary dimensions:

- **Product.** Visual integrity and consistency of the product with respect to the source image I_{s}.
- **Human.** Visual plausibility of human appearance and human-product interactions.
- **Scene.** Semantic and physical plausibility of the surrounding environment.
- **Text.** Visual quality and semantic correctness of textual elements.

Based on this taxonomy, we formulate fine-grained AIGC anomaly detection over four dimensions:

$$\mathcal{D} = \{\mathrm{Product}, \mathrm{Human}, \mathrm{Scene}, \mathrm{Text}\}. \tag{1}$$

Each sample is associated with a binary label vector $\mathbf{y} \in \{0,1\}^{|\mathcal{D}|}$, where $y_d = 1$ indicates an anomaly in dimension d. Beyond the binary decision, the system performs active visual inspection through an interleaved sequence of reasoning, tool invocation, and visual feedback. For each

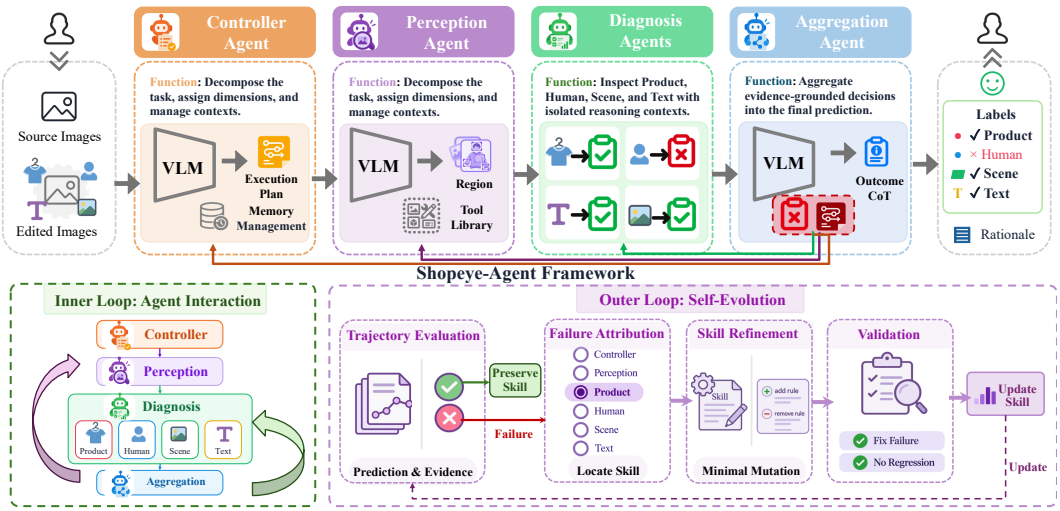


Figure 1: Overview of **ShopEye-Agent**.

dimension d, we define the diagnostic trajectory as:

$$c_d = [(t_{d,k}, a_{d,k}, o_{d,k})]_{k=1}^{K_d}, \tag{2}$$

where $t_{d,k}$ denotes the reasoning used to find a possible defect, $a_{d,k} = \text{Tool}(b_{d,k})$ applies a visual tool to the selected region $b_{d,k}$, and $o_{d,k}$ denotes the visual result returned by the tool.

This formulation naturally connects data construction and model learning. Human annotation first provides reliable dimension-level labels $\mathbf{y}$ and expert inspection knowledge, which is distilled into the detection skills of ShopEye-Agent. The agent then generates localized diagnostic traces c, which are further used to train the target model for anomaly prediction and active visual diagnosis.

4 ShopEye-Bench: Fine-Grained AIGC Anomaly Benchmark

4.1 Overview

ShopEye-Bench is built through an **AI-Human Collaborative Skill Evolution Pipeline**. Human annotators label a small set of image pairs using a four-dimensional checklist and summarize the inspection rules behind their decisions. These rules are distilled into reusable detection skills that guide ShopEye-Agent in large-scale annotation.

ShopEye-Agent identifies semantic regions in the source and edited images and routes relevant visual evidence to four parallel specialists for Product, Human, Scene, and Text. Each specialist applies dimension-specific skills within an independent context to reduce cross-dimensional interference. Their predictions and supporting evidence are aggregated into annotations describing *whether* anomalies exist, *where* they occur, and *why* they constitute defects. During annotation, a verifier examines execution trajectories to locate errors and identify their causes. This feedback guides targeted skill refinement, with updates retained after validation.

4.2 ShopEye-Agent for Collaborative Fine-Grained Diagnosis

Fine-grained AIGC anomaly detection requires heterogeneous capabilities, including task coordination, region-level perception, dimension-specific inspection, and evidence aggregation. As illustrated in Fig. 1, ShopEye-Agent decomposes this process into four functionally specialized agents:

Controller Agent. The controller defines and orchestrates the inspection workflow $\mathcal{W}$, identifying all semantic entities in x and preparing inspection instructions:

$$(\mathcal{V}, \mathcal{P}) = A_{\text{ctrl}}(x; \mathcal{W}), \qquad \mathcal{P} = \{p_d\}_{d \in \mathcal{D}}, \tag{3}$$

where $\mathcal{V} = \{v_i\}_{i=1}^{N}$ denotes the semantic entities and p_d guides inspection for dimension $d \in \mathcal{D} = \{\text{Product}, \text{Human}, \text{Scene}, \text{Text}\}$. The workflow specifies agent execution and evidence routing, maintaining separate reasoning contexts to reduce cross-dimensional interference.

Perception Agent. The perception agent localizes relevant entities in both images and acquires local visual evidence:

$$\mathcal{R}_d = A_{\text{perc}}(x, \mathcal{V}, p_d), \qquad E_d = \Phi(x, \mathcal{R}_d), \tag{4}$$

where $\mathcal{R}_d$ denotes the localized regions and Φ applies cropping or zooming. The resulting evidence E_d complements the global image pair to support fine-grained inspection.

Diagnosis Agents. Four parallel specialists $\{A_d\}_{d\in\mathcal{D}}$ inspect the global images and local evidence using dimension-specific skills:

$$o_d = A_d(x, E_d, p_d; \mathcal{S}_d, M), \qquad o_d = (\hat{y}_d, c_d), \tag{5}$$

where $\mathcal{S}_d$ denotes the detection skill and M the shared multimodal foundation model. Each output contains a binary anomaly prediction $\hat{y}_d$ and supporting visual evidence c_d.

Aggregation Agent. The aggregation agent consolidates specialist outputs into a four-dimensional prediction $\hat{\mathbf{y}}$ and diagnostic evidence c. A verifier then examines the complete execution trajectory τ to identify errors:

$$\begin{aligned} (\hat{\mathbf{y}}, c) &= A_{\text{agg}}\left(\{o_d\}_{d\in\mathcal{D}}\right), \\ \mathcal{F}_\tau &= A_{\text{ver}}(x, \tau) = \{(\ell_j, \epsilon_j)\}_{j=1}^{K_\tau}, \end{aligned} \tag{6}$$

where each pair (ℓ_j, ϵ_j) records an error location and its diagnosed cause. Trajectories with detected errors provide feedback for subsequent skill evolution.

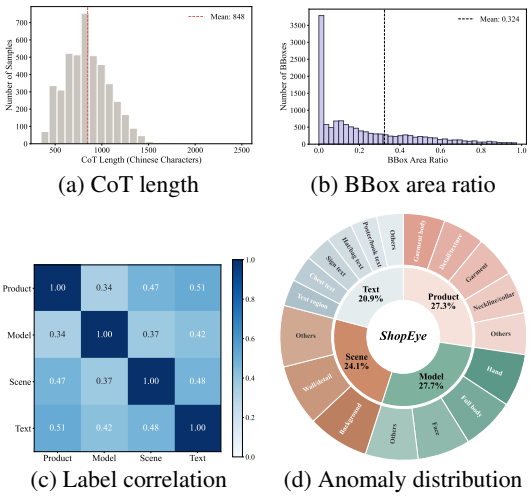
Experience-Driven Skill Evolution. At iteration t, accumulated verifier feedback $\mathcal{F}^t$ guides targeted updates to the implicated detection skills:

$$\mathcal{S}^{t+1} = \text{Validate}\left(\mathcal{S}^t, \text{Mutate}(\mathcal{S}^t, \mathcal{F}^t)\right). \tag{7}$$

Candidate updates are retained only if they pass regression checks on previous cases and held-out validation samples. Otherwise, the existing skills remain unchanged. Validated skills and lessons from successful and failed trajectories are stored for subsequent retrieval, while the workflow $\mathcal{W}$ and foundation model M remain fixed.

4.3 Benchmark Construction and Statistics

The annotation pipeline yields 5K image pairs with four-dimensional anomaly labels, diagnostic reasoning traces, and localized visual evidence. As shown in Fig. 2a, the traces average 848 Chinese characters and 2.95 tool calls per sample, documenting the evidence acquisition and diagnosis process Anomaly prevalence ranges from 44.3% to 58.6% across Product, Human, Scene, and Text in Fig. 2d. In Fig. 2c, the labels exhibit a modest mean pairwise Pearson correlation, suggesting that the dimensions capture complementary defect patterns. Product and Text show the highest label co-occurrence, reflecting overlap between these anomaly types. As shown in Figure 2b, the annotated bounding boxes typically occupy only a very small fraction of the image area. This highlights the need for targeted local inspection to capture subtle anomaly cues. ShopEye-Bench therefore provides a challenging testbed for evaluating models' ability to identify subtle visual anomalies through localized evidence.



(a) CoT length (b) BBox area ratio

(c) Label correlation (d) Anomaly distribution

Figure 2: Statistical profile of ShopEye-Bench.

5 SHOPEYE-TRAINER: LEARNING ACTIVE VISUAL VERIFICATION

ShopEye-Agent achieves strong fine-grained anomaly detection through multi-agent inspection, but incurs substantial latency and token overhead. We introduce **ShopEye-Trainer** to train a single multimodal model that adaptively inspects local regions of a source–edited image pair.

Following the diagnostic formulation in Eq. 2, the model performs active visual verification by interleaving reasoning, tool invocation, and visual feedback. This process requires two complementary capabilities. *Inspection planning* determines whether, where, and how to acquire additional visual evidence, while *evidence interpretation* connects the acquired observations to anomaly judgments. We first develop these capabilities through **Local-Global Dual-View Supervised Fine-Tuning (LGDV-SFT)**. We then refine the inspection policy through **Evidence-Weighted Policy Optimization**.

5.1 LOCAL-GLOBAL DUAL-VIEW SUPERVISED FINE-TUNING

We first use SFT to teach the base model how to call tools in multi-round interactions. However, the model still struggles with local details such as zippers, collars, and finger joints, even when the relevant crops are provided. We therefore introduce LGDV-SFT, which combines multi-round SFT with direct supervision on these local crops.

Global view. The global view applies multi-round SFT to the diagnostic trajectories generated by ShopEye-Agent. Each complete demonstration τ interleaves reasoning, tool actions, and returned observations, followed by the final anomaly prediction. At each round, the model conditions on the source–edited image pair and the available interaction history to generate the next response. For the global demonstration set $\mathcal{D}_g$, we optimize

$$\mathcal{L}_{\text{multi}} = -\mathbb{E}_{(I_{\text{s}},I_{\text{e}},\tau)\sim\mathcal{D}_g}\left[\sum_{j\in\mathcal{M}_\tau}\log\pi_\theta\left(u_j \mid I_{\text{s}}, I_{\text{e}}, \tau_{<j}\right)\right], \tag{8}$$

where $\mathcal{M}_\tau$ indexes target tokens u_j, and $\tau_{<j}$ denotes the preceding multimodal context.

Local view. The local view converts tool-returned observations into standalone visual understanding samples. Each observation $o_{d,k}$ is paired with a target $z_{d,k}$ describing its visible evidence and the corresponding local anomaly judgment. Let $\mathcal{D}_l$ denote the resulting observation–target pairs, and the local objective is:

$$\mathcal{L}_{\text{local}} = -\mathbb{E}_{(o,z)\sim\mathcal{D}_l}\left[\log\pi_\theta(z \mid o)\right]. \tag{9}$$

We jointly train the same model on both views with:

$$\mathcal{L}_{\text{LGDV}} = \mathcal{L}_{\text{multi}} + \lambda\mathcal{L}_{\text{local}}, \tag{10}$$

where λ balances global trajectory learning and local evidence interpretation.

5.2 EVIDENCE-WEIGHTED POLICY OPTIMIZATION

Outcome rewards evaluate the final prediction but provide no direct feedback on the usefulness of intermediate tool calls. We therefore introduce **Evidence-Weighted Policy Optimization**, which uses an external visual verifier to score each tool call and weight its update during DAPO (Yu et al., 2026) training.

For the t-th tool call in a sampled trajectory τ_i, the verifier assigns an evidence score:

$$f_{i,t} = \text{Judge}\left(h_{i,t}, a_{i,t}, o_{i,t}, \hat{\mathbf{y}}_i\right), \qquad f_{i,t} \geq 0, \tag{11}$$

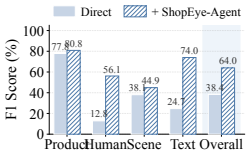
where $h_{i,t}$ is the preceding diagnostic history, $a_{i,t}$ is the tool action, and $o_{i,t}$ is the returned observation. The verifier evaluates two aspects. **Evidence quality** measures whether the observation contains clear visual details for anomaly inspection. **Task relevance** measures whether these details are relevant to the anomaly judgment given the diagnostic history and final prediction. For a group of



(a) Claude-Sonnet-5



(b) Claude-Opus-4.6



(c) Qwen3.8-Max

Figure 4: Effectiveness and generalization of **ShopEye-Agent**.

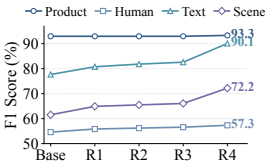


Figure 5: Self-evolution of **ShopEye-Agent**.

G trajectories sampled for the same image pair, we compute the outcome reward and group-relative advantage as:

$$R_i^{\text{out}} = R_{\text{acc}}\left(\hat{\mathbf{y}}_i, \mathbf{y}\right) + R_{\text{fmt}}(\tau_i),$$
$$A_i^{\text{out}} = \frac{R_i^{\text{out}} - \bar{R}^{\text{out}}}{\sigma_R + \epsilon}, \tag{12}$$

where $\mathbf{y}$ is the ground-truth label vector, $\bar{R}^{\text{out}}$ and σ_R are the within-group reward mean and standard deviation, and ϵ is a numerical stabilizer. Let $\mathcal{B}_{i,t}$ denote the token span specifying the t-th tool action. We reweight the advantage of its tokens using the corresponding evidence score:

$$\widetilde{A}_{i,j} = f_{i,t} A_i^{\text{out}}, \qquad j \in \mathcal{B}_{i,t}. \tag{13}$$

The resulting advantages are used in DAPO's clipped policy objective, with tool observations serving as conditioning inputs. The evidence score controls the weight of each tool-action update, while the outcome advantage determines its direction.

6 Experiments

6.1 Experimental Setup

Benchmark and data splits. We evaluate models on ShopEye-Bench. The benchmark includes approximately 5K image pairs for SFT, 12K for RL, and 1K held-out test pairs, organized according to a predefined sample-level split. The two training splits use different label distributions to support their respective training objectives. Further details are provided in Appendix A.

Evaluation protocol and metrics. We evaluate the final binary prediction for each image pair and dimension against its corresponding ground-truth label, treating anomalies as the positive class. Precision, recall, and F1 are computed separately for each dimension. The Overall score is the unweighted macro-average of the four dimension-level F1 scores.

Implementation details. ShopEye-4B and ShopEye-8B are initialized from Qwen3-VL-4B-Instruct and Qwen3-VL-8B-Instruct, respectively. Both models are first trained with Local–Global Dual-View SFT using LoRA, followed by full-parameter DAPO with a learning rate of 2×10^{-6}. Qwen3-VL-32B serves as the verifier for the Step-Level Tool Reward. Inference uses temperature 0 and at most six tool calls per trajectory. Detailed training and inference configurations are provided in the Appendix C.

6.2 Main Results

Table 1 summarizes the performance of diverse multimodal models on ShopEye-Bench across four categories. **(1) Closed-source models** include advanced proprietary systems to assess their performance on fine-grained e-commerce anomaly detection. **(2) Open-source general-purpose VLMs** cover commonly used architectures and different model scales. **(3) Agentic visual reasoning systems** assess the transfer of visual search and tool-assisted reasoning capabilities to this task.

The substantial variation across models and anomaly dimensions demonstrates the ability of ShopEye-Bench to distinguish model capabilities. Its four-dimensional evaluation reveals distinct strengths and weaknesses, supporting the effectiveness of the benchmark's fine-grained evaluation design. Among the methods with reported results, our 8B model trained with ShopEye-Trainer achieves the highest overall macro-F1 of 77.54 and the best F1 in all four dimensions. These results demonstrate its effectiveness in addressing the diverse anomaly types covered by ShopEye-Bench.

Table 1: Comparison with representative multimodal models on **ShopEye-Bench**.

Model	Params	Product			Human			Scene			Text			Overall
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	
Closed-source models														
GPT-5	–	88.82	62.73	73.53	79.31	1.84	3.60	30.95	3.78	6.74	65.69	24.36	35.54	29.85
Claude-Opus-4.8	–	92.61	44.54	60.15	53.57	2.42	4.63	43.48	1.47	2.84	72.14	35.29	47.39	28.75
Gemini-3.5-Flash	–	86.15	68.21	76.14	61.47	35.42	44.94	33.93	21.98	26.68	58.71	28.28	38.17	46.48
Gemini-3-Pro	–	79.56	94.52	86.40	64.13	76.92	69.95	37.74	61.22	46.69	56.09	63.69	59.64	65.67
Kimi-K3	–	87.91	65.56	75.10	51.52	1.36	2.66	36.00	7.87	12.92	46.69	18.18	26.17	29.21
Qwen-VL-Plus	–	79.21	75.85	77.49	62.41	7.05	12.67	36.16	19.21	25.10	61.08	27.60	38.02	38.32
Qwen-3.6-Plus	–	80.60	86.55	83.47	62.78	36.66	46.29	35.53	10.28	15.95	61.94	20.82	31.16	44.22
Qwen-3.7-Plus	–	80.13	90.59	85.04	63.47	41.94	50.51	33.89	14.70	20.51	48.22	38.43	42.77	49.71
Open-source VLMs														
Gemma-3-4B	4B	77.41	76.45	76.93	61.69	34.97	44.63	34.87	79.64	48.50	68.37	33.89	45.32	53.84
Gemma-3-27B	27B	82.82	25.15	38.58	35.00	0.07	0.14	24.26	1.35	2.55	72.96	1.96	3.82	11.27
Llama-3.2-11B-Vision	11B	76.71	95.56	85.11	62.02	94.86	75.00	33.74	93.88	49.64	44.95	66.35	53.59	65.83
Llama-3.2-90B-Vision	90B	82.14	7.81	14.27	66.67	4.36	8.18	31.43	2.63	4.86	34.29	9.21	14.52	10.46
Qwen2.5-VL-3B	3B	75.82	26.40	39.17	61.26	8.50	14.93	29.27	5.58	9.37	47.62	2.17	4.15	16.91
Qwen2.5-VL-7B	7B	82.19	4.01	7.64	76.19	0.09	0.18	27.01	2.07	3.85	84.97	1.20	2.37	3.51
Qwen2.5-VL-32B	32B	86.42	22.39	35.56	100.00	0.00	0.00	100.00	0.00	0.00	73.33	2.15	4.17	9.83
Qwen3-VL-4B	4B	85.56	54.94	66.91	62.80	1.46	2.85	35.91	8.56	13.83	54.06	23.53	32.78	29.09
Qwen3-VL-8B	8B	85.10	67.95	75.56	61.41	19.02	29.04	35.90	19.71	25.45	60.78	44.19	51.17	45.31
Qwen3-VL-32B	32B	79.32	85.86	82.41	100.00	2.09	4.10	50.00	1.57	3.05	90.91	55.94	69.26	39.71
InternVL3.5-2B	2B	80.16	78.59	79.37	61.09	32.12	42.10	67.62	34.67	45.84	36.12	60.82	45.32	53.16
InternVL3.5-8B	8B	80.05	79.55	79.80	61.06	1.34	24.06	85.70	14.00	24.06	33.22	1.79	3.39	27.47
InternVL3.5-38B	38B	83.84	70.45	76.56	42.03	0.15	0.29	75.37	12.07	20.81	46.70	0.97	1.89	24.89
Agentic Visual Reasoning Systems														
Chain-of-Focus	7B	79.14	80.84	79.98	75.00	1.10	2.17	45.63	10.68	17.31	80.65	5.32	9.98	27.36
Pixel-Reasoner	7B	83.06	51.24	63.38	71.43	3.16	6.06	55.56	11.24	18.69	93.75	15.46	26.55	28.67
DeepEyes	7B	86.76	22.68	35.96	100.00	0.09	0.18	66.67	0.32	0.64	100.00	0.15	0.30	9.27
Thyme-RL	7B	100.00	5.37	10.19	100.0	1.74	3.42	100.0	1.43	2.82	100.00	0.00	0.00	4.11
CodeV-RL	7B	81.93	26.04	39.52	59.06	13.06	21.39	62.20	21.38	31.82	39.78	21.54	27.95	30.17
InSight-o3	7B‡	82.14	53.72	64.96	62.00	2.48	4.77	78.05	4.33	8.21	51.06	6.99	12.29	22.56
GRiP	7B	82.60	40.53	54.26	100.00	0.24	0.49	42.67	7.27	12.43	71.43	6.38	11.72	19.72
SenseNova-MARS	8B	83.00	64.49	72.58	60.69	32.10	41.99	33.88	36.94	35.34	62.66	56.01	59.14	52.26
SenseNova-MARS	32B	87.52	52.85	65.90	66.67	0.73	1.45	33.33	11.36	16.95	76.92	27.66	40.69	31.25
Ours														
ShopEye-Trainer	4B	79.86	90.09	<u>84.67</u>	63.85	96.07	**76.72**	63.06	68.65	<u>65.74</u>	78.95	77.51	<u>78.23</u>	<u>76.34</u>
ShopEye-Trainer	8B	76.88	95.19	**85.06**	62.65	96.16	<u>75.87</u>	62.60	72.06	**67.00**	74.41	85.30	**79.37**	**76.83**

‡ InSight-o3 is a multi-agent system. The reported size corresponds to its Qwen2.5-VL-7B visual search agent rather than the entire system.

6.3 Effectiveness and Generalization of ShopEye-Agent.

We evaluate ShopEye-Agent from two perspectives: its generalization across different multimodal backbones and its improvement through autonomous evolution. We evaluate ShopEye-Agent on three representative frontier multimodal backbones: Claude-Sonnet-5, Claude-Opus-4.6, and Qwen3.8-Max. As shown in Fig. 4, applying the same ShopEye-Agent workflow consistently improves fine-grained anomaly detection across all evaluated backbones compared with direct inference. We further examine the autonomous evolution of ShopEye-Agent using Claude-Opus-4.6 as the backbone in Fig. 5. Performance improves progressively over successive evolution rounds, suggesting that iterative failure analysis and skill refinement effectively enhance the agent's diagnostic capability.

6.4 Ablation Studies

Local-global dual-view supervision. We first examine the design of LGDV by ablating multi-round supervision (**MR**) and local-view supervision (**LV**), with single-turn SFT serving as the baseline. An intuitive comparison of these supervision paradigms is provided in Fig. 9 of the Appendix. Adding LV consistently improves all four dimensions over MR. The improvement is particularly pronounced on Human, suggesting that direct supervision on localized visual observations helps the model better interpret fine-grained evidence that is difficult to capture from global interaction trajectories alone. These results validate the complementary roles of global trajectory learning and local evidence interpretation in LGDV.

Table 2: Ablation studies of ShopEye-Trainer on ShopEye-Bench.

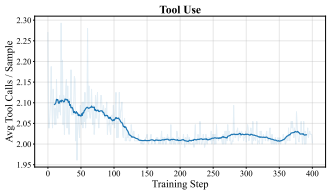
Configuration		Product			Human			Scene			Text			Overall
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	
Ablation on Local-Global Dual-View Supervision														
MR	**LV**													
–	–	87.77	48.75	62.68	62.95	15.40	24.74	49.26	19.53	27.97	76.97	55.69	**64.62**	45.00
✓	–	87.17	54.11	66.77	70.63	15.22	25.05	51.32	19.80	28.57	82.32	44.11	57.44	44.46
✓	✓	87.29	60.59	**71.53**	70.71	32.72	**44.74**	63.41	22.71	**33.44**	76.13	52.23	61.96	**52.92**
Ablation on Evidence-Guided Policy Optimization														
Reward Design														
Baseline		76.97	92.82	84.16	62.56	94.04	75.14	65.65	56.12	60.51	75.38	84.44	79.65	74.87
CG		76.84	93.32	**84.28**	62.31	94.76	75.18	61.30	70.38	65.53	76.47	82.63	79.43	76.11
EW		76.89	93.14	84.24	62.56	95.16	**75.49**	61.65	73.02	**66.85**	76.23	85.05	**80.40**	**76.75**
Ablation on the Complementarity of LGDV and EW														
LGDV	**EG**													
–	–	84.90	69.70	76.55	58.88	20.47	30.38	37.31	22.33	27.93	61.25	49.49	54.75	47.40
✓	–	87.29	60.59	71.53	70.71	32.72	44.74	63.41	22.71	33.44	76.13	52.23	61.96	52.92
–	✓	76.89	93.14	84.24	62.56	95.16	**75.49**	61.65	73.02	66.85	76.23	85.05	80.40	76.75
✓	✓	76.88	95.19	**85.06**	62.65	96.16	**75.87**	62.60	72.06	**67.00**	74.41	85.30	**79.37**	**76.83**

Evidence-guided policy optimization. We next study the optimization design of EG by comparing the reward baseline, correctness-gated tool rewards (**CG**), and the proposed evidence-weighted optimization (**EW**). A schematic illustration of **EW** is provided in Fig. 10 of the Appendix. This result indicates that final prediction correctness alone is insufficient for learning effective inspection behavior. Explicitly incorporating evidence quality and task relevance into the optimization of intermediate tool actions provides a more informative learning signal for active visual verification.

Complementarity of LGDV and EW. We evaluate LGDV and EW independently and jointly by comparing the base model, LGDV alone, EW alone, and the complete LGDV+EW training pipeline. Combining the two stages yields the strongest overall performance, demonstrating that supervised acquisition of active perception capabilities and subsequent policy refinement are complementary.



(a) Overall reward



(b) Tool usage

Figure 6: Training dynamics.

6.5 Training dynamics

Fig. 6 illustrates how EW shapes the active inspection policy during optimization. The overall reward increases steadily throughout training and eventually reaches a stable plateau, indicating effective policy improvement. Interestingly, this improvement does not rely on increasing the number of visual operations. Instead, the average number of tool calls gradually decreases in the early stage and remains stable thereafter. This behavior suggests that EW suppresses redundant or low-value visual acquisitions while preserving tool calls that provide useful evidence for the final prediction.

7 Conclusion

ShopEye highlights the value of fine-grained, evidence-grounded evaluation for assessing AIGC quality. ShopEye-Bench moves beyond coarse image-level judgments by characterizing *what* goes wrong, *where* the defect occurs, and *which visual evidence* supports the diagnosis, enabling a more informative comparison of current multimodal models. Built on this supervision, ShopEye-4B and ShopEye-8B further demonstrate that such diagnostic knowledge can be effectively distilled into compact models capable of actively acquiring and interpreting local visual evidence, providing a practical solution for fine-grained AIGC quality inspection. Together, the benchmark and trained models establish a unified framework for both evaluating and improving AIGC detection systems.

AI USE STATEMENT

Generative AI tools were used to aid and polish the writing of this paper, including language refinement, grammatical correction, and improvement of clarity and presentation. Generative AI tools were also used for retrieval and discovery of related literature, including identifying relevant research directions and candidate references. All references included in the manuscript were independently verified by the authors against the original sources. The authors remain fully responsible for the scientific content, technical accuracy, citations, and final wording of the paper.

REPRODUCIBILITY STATEMENT

We provide detailed information throughout the appendix to facilitate reproduction of our benchmark, agent pipeline, training procedure, and evaluation protocol. The construction, curation, and annotation procedures of ShopEye-Bench are described in Appendix A, with the annotation checklist in Appendix A.1.1 and human quality control in Appendix A.2. Implementation details of ShopEye-Agent, including the detection Skills and data flywheel, are provided in Appendix B. The complete training configurations of ShopEye-Trainer are reported in Appendix C, covering both Local-Global Dual-View SFT and Evidence-Weighted Policy Optimization. The diagnostic prompts and output schema are provided in Appendix D, and the visual tool definitions are specified in Appendix D.1. The implementation and inference settings of comparison baselines are described in Appendix D.1.1.

REFERENCES

Samyadeep Basu, Mehrdad Saberi, Shweta Bhardwaj, Atoosa Malemir Chegini, Daniela Massiceti, Maziar Sanjabi, Shell Xu Hu, and Soheil Feizi. Editval: Benchmarking diffusion based text-guided image editing methods. *arXiv preprint arXiv:2310.02426*, 2023.

Bin Cao, Jianhao Yuan, Yexin Liu, Jian Li, Shuyang Sun, Jing Liu, and Bo Zhao. Synartifact: Classifying and alleviating artifacts in synthetic images via vision-language model. *arXiv preprint arXiv:2402.18068*, 2024.

Yong Xien Chng, Tao Hu, Wenwen Tong, Xueheng Li, Jiandong Chen, Haojia Yu, Jiefan Lu, Hewei Guo, Hanming Deng, Chengjun Xie, et al. Sensenova-mars: Empowering multimodal agentic reasoning and search via reinforcement learning. *arXiv preprint arXiv:2512.24330*, 2025.

Shengyuan Ding, Xinyu Fang, Ziyu Liu, Yuhang Zang, Yuhang Cao, Xiangyu Zhao, Haodong Duan, Xiaoyi Dong, Jianze Liang, Bin Wang, et al. Arm-thinker: Reinforcing multimodal generative reward models with agentic tool use and visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22195–22205, 2026.

Chenghan Fu, Daoze Zhang, Yukang Lin, Zhanheng Nie, Xiang Zhang, Jianyu Liu, Yueran Liu, Wanxian Guan, Pengjie Wang, Jian Xu, et al. Moon embedding: Multimodal representation learning for e-commerce search advertising. *arXiv preprint arXiv:2511.11305*, 2025.

Yu He, Ting Zhu, Yichun Liu, Lichen Ma, Xinyuan Shan, Jingling Fu, Yu Shi, Junshi Huang, and Yan Li. Fashion130k: An e-commerce fashion dataset for outfit generation with unified multi-modal condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 4769–4779, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in neural information processing systems*, volume 33, pp. 6840–6851, 2020.

Yibo Hu, Yu Qian, Mao Gu, Yingfan Tao, Yuhao Chen, Yongdong Luo, Zhuoqun Liu, Meiguang Jin, and Junfeng Ma. Tlive-omni: An omni-modal understanding model for e-commerce live streaming. *arXiv preprint arXiv:2608.20958*, 2026.

Changjiang Jiang, Wenhui Dong, Zhonghao Zhang, Fengchang Yu, Wei Peng, Xinbin Yuan, Yifei Bi, Ming Zhao, Zian Zhou, Chenyang Si, et al. Ivy-fake: A unified explainable framework and benchmark for image and video aigc detection. In *Proceedings of the 2026 International Conference on Multimedia Retrieval*, pp. 2438–2447, 2026.

Kaican Li, Lewei Yao, Jiannan Wu, Tiezheng YU, Jierun Chen, Haoli Bai, LU HOU, Lanqing HONG, Wei Zhang, and Nevin Zhang. Insight-o3: Empowering multimodal foundation models with generalized visual search. In *International Conference on Learning Representations*, volume 2026, pp. 128950–129002, 2026.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Jianghao Lin, Yang Cao, Yong Yu, and Weinan Zhang. Diffusion models for recommender systems: From content distribution to content creation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 6074–6085, 2025.

Jianghao Lin, Peng Du, Jiaqi Liu, Weite Li, Yong Yu, Weinan Zhang, and Yang Cao. Sell it before you make it: Revolutionizing e-commerce with personalized ai-generated items. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 2323–2334, 2026.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Ruiqi Liu, Manni Cui, Ziheng Qin, Zhiyuan Yan, Ruoxin Chen, Yi Han, Zhiheng Li, Junkai Chen, ZhiJin Chen, Kaiqing Lin, et al. Mirror: Manifold ideal reference reconstructor for generalizable ai-generated image detection. *arXiv preprint arXiv:2602.02222*, 2026a.

Xianjie Liu, Yiman Hu, Liang Wu, Ping Hu, Yixiong Zou, Jian Xu, and Bo Zheng. E-vads: An e-commerce short videos understanding benchmark for mllms. *arXiv preprint arXiv:2602.08355*, 2026b.

Xuannan Liu, Zekun Li, Pei Li, Huaibo Huang, Shuhan Xia, Xing Cui, Linzhi Huang, Weihong Deng, and Zhaofeng He. Mmfakebench: A mixed-source multimodal misinformation detection benchmark for lvlms. In *International Conference on Learning Representations*, volume 2025, pp. 86327–86352, 2025.

Yiwei Ma, Jiayi Ji, Ke Ye, Weihuang Lin, Zhibin Wang, Yonghan Zheng, Qiang Zhou, Xiaoshuai Sun, and Rongrong Ji. I2ebench: A comprehensive benchmark for instruction-based image editing. In *Advances in Neural Information Processing Systems*, volume 37, pp. 41494–41516, 2024.

Chaojie Mao, Jingfeng Zhang, Yulin Pan, Zeyinzi Jiang, Zhen Han, Yu Liu, and Jingren Zhou. Ace++: Instruction-based image creation and editing via context-aware content filling. In *2025 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 1979–1987, 2025.

Minheng Ni, Zhengyuan Yang, Linjie Li, Chung-Ching Lin, Kevin Lin, Wangmeng Zuo, and Lijuan Wang. Point-rft: Improving multimodal reasoning with visually grounded reinforcement finetuning. *Advances in Neural Information Processing Systems*, 38:20538–20559, 2026.

Zhanheng Nie, Chenghan Fu, Daoze Zhang, Junxian Wu, Wanxian Guan, Pengjie Wang, Jian Xu, and Bo Zheng. Moon2. 0: Dynamic modality-balanced multimodal representation learning for e-commerce product understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22975–22985, 2026.

Matteo Nulli, Orshulevich Vladimir, Tala Bazazo, Christian Herold, Michael Kozielski, Marcin Mazur, Szymon Tuzel, Cees Snoek, Seyyed Hadi Hashemi, Omar Javed, et al. Adapting vision-language models for e-commerce understanding at scale. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 496–512, 2026.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Facilitating large language models to master 16000+ real-world apis. In *Proceedings of the International Conference on Learning Representations*, 2024.

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. In *Advances in neural information processing systems*, volume 36, pp. 68539–68551, 2023.

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

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025a.

Haozhan Shen, Kangjia Zhao, Tiancheng Zhao, Ruochen Xu, Zilun Zhang, Mingwei Zhu, and Jianwei Yin. Zoomeye: Enhancing multimodal llms with human-like zooming capabilities through tree-based image exploration. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6613–6629, 2025b.

Alex Su, Haozhe Wang, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel space reasoning via curiosity-driven reinforcement learning. *Advances in Neural Information Processing Systems*, 38:8222–8251, 2026.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8692–8701, 2020.

Weixun Wang, Shaopan Xiong, Gengru Chen, Wei Gao, Sheng Guo, Yancheng He, Ju Huang, Jiaheng Liu, Zhendong Li, Xiaoyang Li, et al. Reinforcement learning optimization for large-scale learning: An efficient and user-friendly scaling library. *arXiv preprint arXiv:2506.06122*, 2025a.

Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong Ye, Xizhou Zhu, et al. Visualprm: An effective process reward model for multimodal reasoning. *arXiv preprint arXiv:2503.10291*, 2025b.

Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Generative recommendation: Towards personalized multimodal content generation. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 2421–2425.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22388–22398, 2023.

Zhaoyang Wei, Wenchao Ding, Yanchao Hao, and Xi Chen. Guiding the inner eye: A framework for hierarchical and flexible visual grounded reasoning. *arXiv preprint arXiv:2511.22172*, 2025.

Siwei Wen, Peilin Feng, Hengrui Kang, Zichen Wen, Yize Chen, Jiang Wu, Conghui He, Weijia Li, et al. Spot the fake: Large multimodal model-based synthetic image detection with artifact explanation. *Advances in Neural Information Processing Systems*, 38:58972–59005, 2026.

Junxian Wu, Chenghan Fu, Zhanheng Nie, Daoze Zhang, Bowen Wan, Wanxian Guan, Chuan Yu, Jian Xu, and Bo Zheng. Moon3. 0: Reasoning-aware multimodal representation learning for e-commerce product understanding. *arXiv preprint arXiv:2604.00513*, 2026a.

Mingyuan Wu, Jingcheng Yang, Jize Jiang, Meitang Li, Kaizhuo Yan, Hanchao Yu, Minjia Zhang, Chengxiang Zhai, and Klara Nahrstedt. Vtool-r1: Vlms learn to think with images via reinforcement learning on multimodal tool use. In *International Conference on Learning Representations*, volume 2026, pp. 78298–78319, 2026b.

Baixuan Xu, Weiqi Wang, Haochen Shi, Wenxuan Ding, Huihao Jing, Tianqing Fang, Jiaxin Bai, Xin Liu, Changlong Yu, Zheng Li, et al. Mind: Multimodal shopping intention distillation from large vision-language models for e-commerce purchase understanding. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7800–7815, 2024.

Yexing Xu, Wei Feng, Shen Zhang, Haohan Wang, Yuxin Qin, Yaoyu Li, Ao Ma, Yuhao Luo, Lu Wang, Xudong Ren, et al. Design your ad: Personalized advertising image and text generation with unified autoregressive models. *arXiv preprint arXiv:2605.12138*, 2026.

Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable AI-generated image detection. In *Proceedings of the International Conference on Machine Learning*, volume 267, pp. 70268–70288, 2025.

Junyan Ye, Baichuan Zhou, Zilong Huang, Junan Zhang, Tianyi Bai, Hengrui Kang, Jun He, Honglin Lin, Zihao Wang, Tong Wu, et al. Loki: A comprehensive synthetic data detection benchmark using large multimodal models. In *International Conference on Learning Representations*, volume 2025, pp. 70440–70522, 2025.

Yang Ye, Xianyi He, Zongjian Li, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, Li Yuan, et al. Imgedit: A unified image editing dataset and benchmark. In *Advances in Neural Information Processing Systems*, volume 38, 2026.

Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26125–26135, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Daoze Zhang, Chenghan Fu, Zhanheng Nie, Jianyu Liu, Wanxian Guan, Yuan Gao, Jun Song, Pengjie Wang, Jian Xu, and Bo Zheng. Moon: Generative mllm-based multimodal representation learning for e-commerce product understanding. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining*, pp. 924–933, 2026a.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *Advances in neural information processing systems*, volume 36, pp. 31428–31449, 2023.

Wenhua Zhang, Weicheng Li, Xuanrong Rao, Lixin Zou, Xiangyang Luo, Chubin Zhuang, Yongjie Hong, Zhen Qin, Hengyu Chang, Chenliang Li, et al. Aiguard: A benchmark and lightweight detection for e-commerce aigc risks. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 12437–12450, 2025a.

Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, and Qing Li. Adaptive chain-of-focus reasoning via dynamic visual search and zooming for efficient vlms. *arXiv preprint arXiv:2505.15436*, 2025b.

YiFan Zhang, Xingyu Lu, Shukang Yin, Chaoyou Fu, Wei Chen, Xiao Hu, Bin Wen, Kaiyu Jiang, Changyi Liu, Tianke Zhang, Haonan fan, Kaibing Chen, Jiankang Chen, Haojie Ding, Kaiyu Tang, Zhang Zhang, Liang Wang, Fan Yang, Tingting Gao, and Guorui Zhou. Thyme: Think beyond images. In *International Conference on Learning Representations*, volume 2026, pp. 79536–79572, 2026b.

Zhehao Zhang, Ryan A Rossi, Tong Yu, Franck Dernoncourt, Ruiyi Zhang, Jiuxiang Gu, Sungchul Kim, Xiang Chen, Zichao Wang, and Nedim Lipka. Vipact: Visual-perception enhancement via specialized vlm agent collaboration and tool-use. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 36536–36546, 2026c.

Xiangyu Zhao, Peiyuan Zhang, Kexian Tang, Xiaorong Zhu, Hao Li, Wenhao Chai, Zicheng Zhang, Renqiu Xia, Guangtao Zhai, Junchi Yan, Hua Yang, Xue Yang, and Haodong Duan. Envisioning beyond the pixels: Benchmarking reasoning-informed visual editing. In *Advances in Neural Information Processing Systems*, 2025.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics*, pp. 400–410, 2024.

Ziwei Zheng, Minghao Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, and Chao Shen. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 126775–126798, 2026.

Muzhi Zhu, Hao Zhong, Canyu Zhao, Zongze Du, Zheng Huang, Mingyu Liu, Hao Chen, Cheng Zou, Jingdong Chen, Ming Yang, and Chunhua Shen. ACTIVE-O3: Empowering multimodal large language models with active perception via GRPO. In *Proceedings of the International Conference on Machine Learning*, 2026a.

Yaoyu Zhu, Di Huang, Hanqi Lyu, Xiaoyun Zhang, Chongxiao Li, Wenxuan Shi, Yutong Wu, Jianan Mu, Jinghua Wang, Pengwei Jin, et al. Qimeng-codev-r1: Reasoning-enhanced verilog generation. *Advances in Neural Information Processing Systems*, 38:154266–154300, 2026b.

A Dataset Construction and Annotation Protocol

A.1 Data Sources and Selection

Our dataset is synthesized using an internal image-based virtual try-on (VTON) pipeline. Each sample consists of a source product image and its AIGC-edited counterpart. The edited image is the target of quality assessment, while the source image provides a reference for product consistency.

To obtain effective training data, we consider sample learnability, rule consistency, and coverage across anomaly dimensions. The curation pipeline comprises three stages:

- **Learnability-oriented selection.** We probe the base model with repeated stochastic rollouts and categorize samples as *Easy*, *Learnable*, or *Hard*. Easy samples are consistently solved, Learnable samples are only partially solved with unstable performance, and Hard samples are rarely solved. We prioritize Learnable samples, which provide meaningful room for improvement within the model's current capabilities.
- **Rule-consistency filtering.** We prioritize samples governed by stable and objective inspection criteria, while filtering those dominated by subjective judgments or inconsistent rule interpretations. This process reduces annotation noise and favors visually grounded defects over ambiguous impressions, such as an image merely appearing artificial or a pose being aesthetically inappropriate.
- **Targeted augmentation.** To improve coverage of underrepresented dimensions, we use Qwen3-Image-Edit to generate additional Text and Scene anomalies. Editing instructions constrain modifications to the target region while preserving non-target content, thereby increasing training examples for dimensions that are less frequent in the original data.

Effect of data selection. To isolate the effect of data selection, all variants are evaluated under the same ShopEye-Bench setting, with only the data curation strategy varied. Table 4 reports the cumulative effects of the three stages. As the pipeline progressively incorporates learnability-oriented selection, rule-consistency filtering, and targeted augmentation, the overall macro-F1 increases from 29.27 to 77.75. The complete pipeline also produces more balanced performance across the four anomaly dimensions. Moreover, filtering noisy samples leads to a more stable reward trajectory during training.

A.1.1 Annotation Checklist

Each image pair receives four independent binary labels. A label of 1 indicates an anomaly in the corresponding dimension, while 0 indicates that no anomaly is identified. Multiple dimensions may be positive for the same sample. The checklist covers the following criteria, as shown in Table. 3.

- **Product.** Check whether the edited product preserves the key attributes of the source product, including *category, shape, color, material, structural components, logos, patterns, and textures*. A sample is marked anomalous if any product attribute is incorrectly altered, or if components are *missing, added, duplicated, misplaced, deformed, or inconsistent* with the source image.
- **Human.** Check the plausibility of the depicted person from four aspects: *body structure, facial details, hand details, and human–object interaction*. Particular attention is paid to the number and geometry of limbs and fingers, facial completeness, joint connectivity, body proportions, and contact with products or accessories. Anomalies include *extra or missing body parts, fused or distorted fingers, malformed facial features, implausible poses, and incorrect intersections or attachments* between the human body and surrounding objects.
- **Scene.** Check whether the background and surrounding environment are spatially and semantically plausible. The checklist covers *scene structure, object placement, support relations, perspective, lighting, shadows, reflections, boundaries, and background textures*. A scene is considered anomalous when it contains *broken geometry, inconsistent perspective, physically implausible support or reflection, abnormal blur or texture, duplicated structures, or semantically unreasonable background content*.
- **Text.** Check all visible textual content for *semantic correctness, spelling, character structure, legibility, completeness, and spatial attachment*. This includes both text on the product and text

Table 3: Annotation checklist and tag taxonomy used in ShopEye-Bench.

Dimension / Issue Tags	
Product	
☐ Logo or pattern inconsistency	☐ Collar, cuff, or hem defects
☐ Button, zipper, or lace defects	☐ Pocket or structural-detail errors
☐ Abnormal silhouette or fit	☐ Incorrect length or proportion
☐ Color inconsistency	☐ Material or transparency anomaly
☐ Missing or corrupted product details	☐ Cross-image product inconsistency
Other: _____	
Label: ☐ Normal (0) ☐ Anomalous (1)	
Human	
☐ Malformed hands or fingers	☐ Missing or extra fingers
☐ Abnormal joints or finger shape	☐ Extra or malformed limbs
☐ Facial feature defects	☐ Implausible body proportion
☐ Unnatural pose or posture	☐ Implausible human behavior
☐ Body–object penetration	☐ Abnormal accessory interaction
Other: _____	
Label: ☐ Normal (0) ☐ Anomalous (1)	
Text	
☐ Corrupted or garbled text	☐ Malformed characters
☐ Spelling or character errors	☐ Incomplete or distorted text
Other: _____	
Label: ☐ Normal (0) ☐ Anomalous (1)	
Scene	
☐ Excessive foreground/background blur	☐ Shadow or reflection inconsistency
☐ Implausible mirror content	☐ Deformed architecture or furniture
☐ Object–scene penetration	☐ Scene–prop incompatibility
☐ Unexpected or meaningless objects	☐ Background boundary artifacts
☐ Added or missing scene objects	☐ Inconsistent object shape or position
Other: _____	
Label: ☐ Normal (0) ☐ Anomalous (1)	

Table 4: Cumulative data curation results. LS denotes learnability-oriented selection, RCF rule-consistency filtering, and TA targeted augmentation.

LS	**RCF**	**TA**	**Product**			**Human**			**Scene**			**Text**			**Overall**
			P	R	F1	P	R	F1	P	R	F1	P	R	F1	
–	–	–	**90.45**	57.76	70.50	0.00	0.00	0.00	0.00	0.00	0.00	**85.02**	32.06	46.56	29.27
✓	–	–	84.07	71.51	77.29	**67.42**	4.87	9.08	37.50	2.65	4.95	69.38	35.26	46.76	34.52
✓	✓	–	84.81	72.55	78.20	61.05	14.81	23.84	39.86	17.58	24.40	63.94	50.37	56.35	45.70
✓	✓	✓	76.89	**99.66**	**86.81**	62.56	**99.92**	**76.94**	**61.65**	**73.02**	**66.85**	76.23	**85.05**	**80.40**	**77.75**

appearing in the surrounding scene. Anomalies include *misspelled or meaningless text, malformed characters, missing or repeated strokes, duplicated text, severe distortion, inconsistent orientation, and text that is incorrectly attached to a surface*.

A.2 Human verification

Human annotators manually correct the generated CoT rationales to mitigate hallucinated judgments, with an average correction rate of 18%. We additionally perform random quality checks with

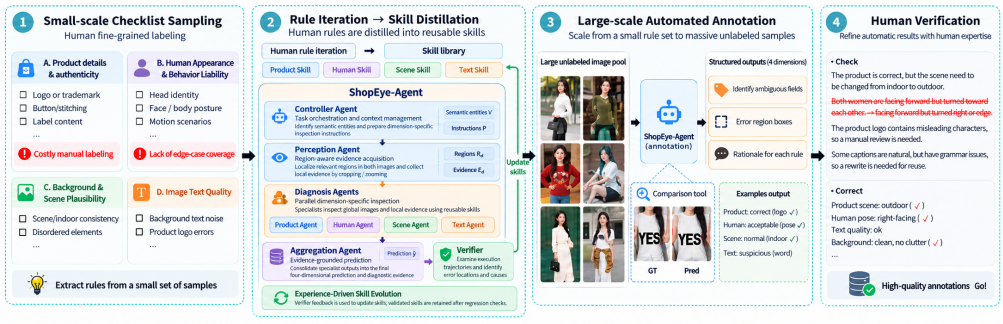


Figure 7: Overview of the ShopEye-Agent data construction pipeline.

multiple annotators and measure their agreement, providing an extra reliability check for the final annotations.

B Implementation Details and Analysis of ShopEye-Agent

System Scale. ShopEye-Agent is implemented as a collection of ten specialized detection *Skills*, each encapsulating the domain expertise required for a distinct family of quality defects in apparel e-commerce AIGC imagery. Collectively, these Skills comprise 48 pipeline detection steps, 63 execution rules, and 30 low-quality adjudication conditions, yielding a total of 109 structured review rules. This rule base provides comprehensive coverage over 40 defect types spanning the full spectrum of AIGC-generated apparel images. Decomposing the review task across multiple focused Skills, rather than relying on a single monolithic prompt, allows each Skill to apply its own targeted pipeline and decision criteria, improving both interpretability and detection precision. These specialized Skills are orchestrated within the collaborative ShopEye-Agent pipeline, whose overall workflow is illustrated in Fig. 7.

Data Flywheel. The rule base is not hand-crafted in isolation but distilled through a data-driven flywheel. Starting from 5,000 human-annotated samples, we perform rule mining and error attribution analysis to condense scattered domain knowledge into the structured Skill rule system described above. The resulting Agent system is then deployed to automatically annotate a large pool of over 30,000 unlabeled images. Human reviewers subsequently verify these automatic annotations and correct the chain-of-thought (CoT) reasoning in only 18.7% of cases, indicating that the vast majority of Agent-produced labels are directly reliable. This closed loop substantially reduces annotation cost while continuously enriching the supervision available for further refinement.

C Implementation Details and Analysis of ShopEye-Trainer

C.1 Local-Global Dual-View SFT

We fine-tune Qwen3-VL-8B-Instruct using LoRA in LLaMA-Factory (Zheng et al., 2024). LoRA adapters are applied to the query, key, value, and output projections in attention layers, as well as the gate, up, and down projections in feed-forward layers, with rank $r = 64$, $\alpha = 128$, and dropout 0.05. Training runs for 828 optimization steps with a learning rate of 1×10^{-4}, a cosine learning-rate schedule, a warmup ratio of 0.05, and weight decay of 0.01. We use eight GPUs with a per-device batch size of 1 and 16 gradient accumulation steps, yielding an effective batch size of 128. The maximum sequence length is $8{,}192$ tokens, and `image_resolution` is set to $1{,}048{,}576$. BF16 precision, gradient checkpointing, and DeepSpeed are enabled. To progressively incorporate local supervision, we combine multi-round tool-use trajectories with high-quality local-perception samples through a sampling curriculum. During the first 20% of training, we exclusively sample multi-round trajectories. Between 20% and 70% of training progress, the sampling probability of local-perception data increases linearly from 0 to 0.3. For the remaining 30%, we maintain sampling probabilities of 0.7 and 0.3 for multi-round and local-perception data, respectively.

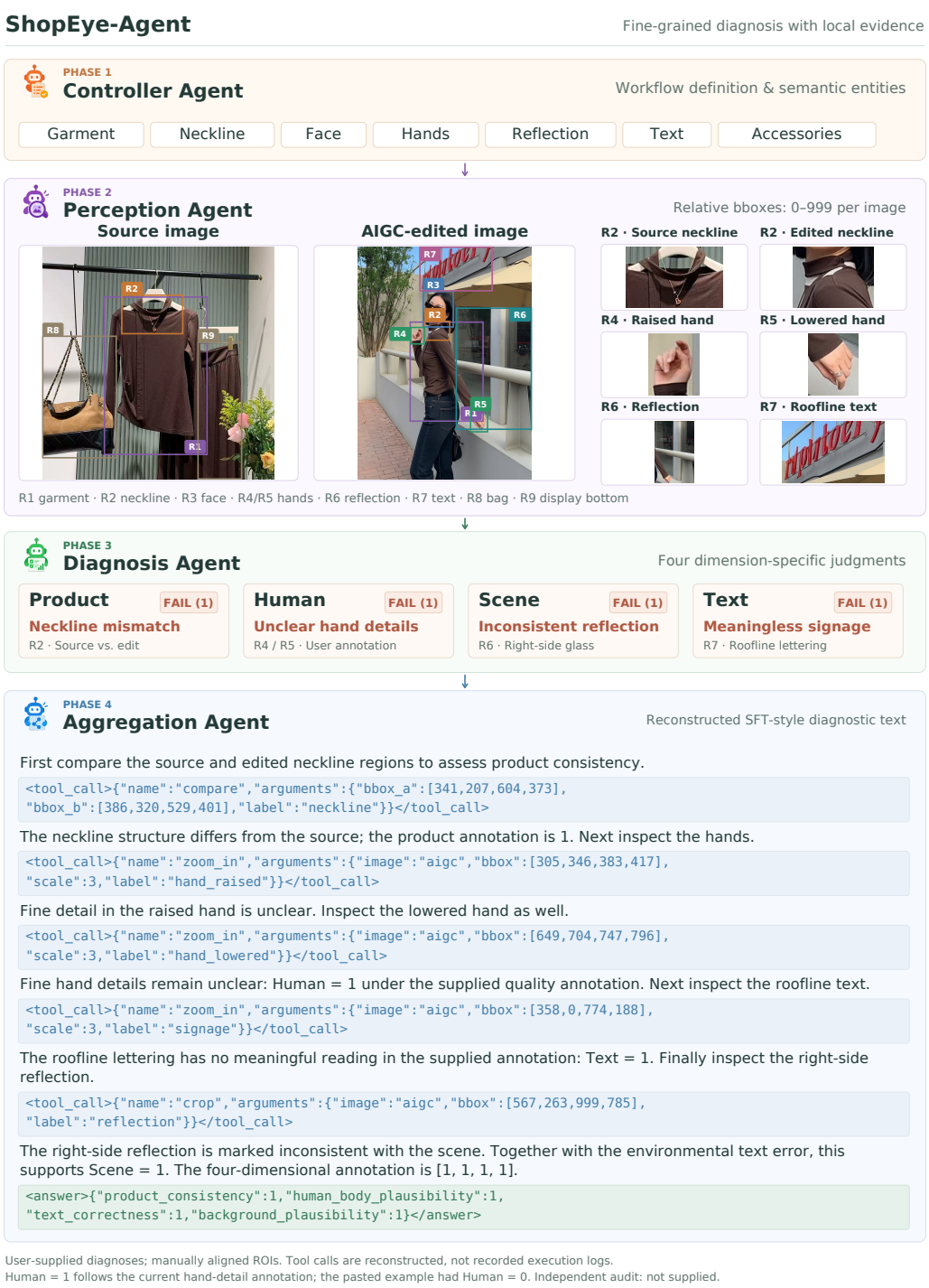


Figure 8: Overview of Shopeye-Agent.

C.2 Evidence-Weighted Policy Optimization

We implement reinforcement learning using ROLL (Wang et al., 2025a), initializing the policy from a merged Qwen3-VL-4B checkpoint obtained through multi-task supervised fine-tuning. Training runs for 748 optimization steps with a learning rate of 2×10^{-6}, cosine decay, and an effective global batch size of 128 trajectories. The vision encoder and patch-embedding layers remain frozen. Each

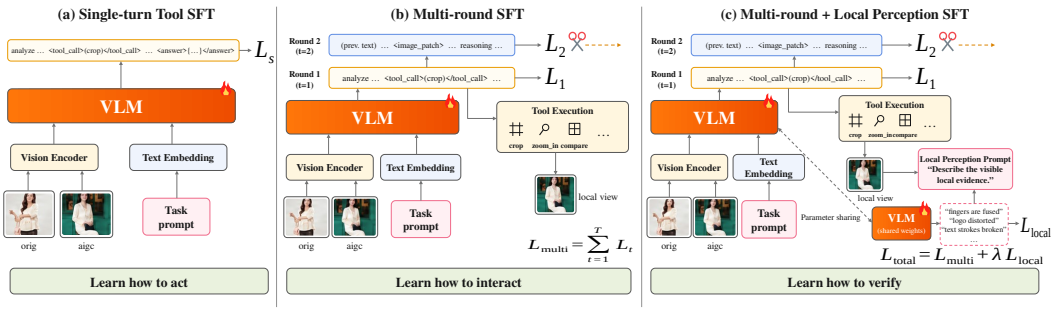


Figure 9: Comparison of single-turn tool SFT, multi-round SFT, and multi-round SFT with local perception supervision.

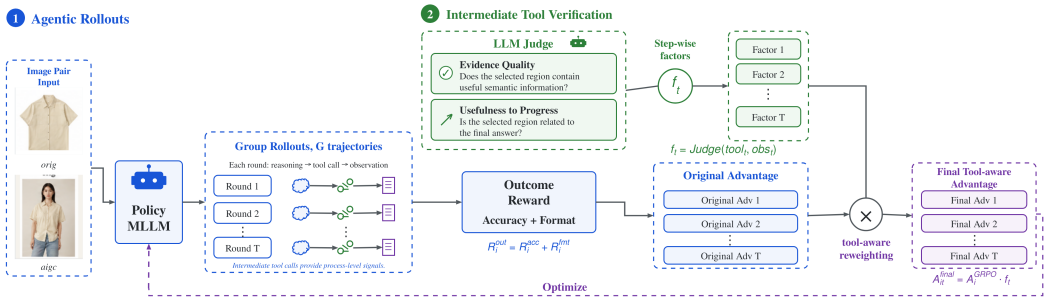


Figure 10: Overview of evidence-weighted policy optimization.

rollout batch contains 16 prompts with 8 sampled responses per prompt, generated at a temperature of 0.99. The maximum sequence and response lengths are 16,480 and 2,048 tokens, respectively. We use DAPO with a token-mean policy-gradient loss, asymmetric clipping thresholds of 0.2 and 0.28, and no KL regularization. Groups with near-zero reward variance are filtered and resampled, and overlong responses are penalized.

D Prompt Templates

Each inspection instance is formulated as a multi-turn conversation containing a system instruction and a user message with two images: the original product image I_{orig} and its AIGC-edited counterpart I_{aigc}. The system instruction specifies four binary inspection dimensions: (1) product-detail consistency, (2) human-body and pose plausibility, (3) text quality, and (4) background and scene plausibility. For each dimension, the prompt provides an operational definition together with representative failure cases, such as structural product changes, anatomical artifacts, malformed text, and inconsistent scene geometry.

The prompt follows an interleaved *analyze–act–observe–decide* protocol. At an intermediate step, the model may emit one visual action as a JSON object enclosed by `<tool_call>` tags. The environment executes the action and returns the resulting image as a tool observation. The model is explicitly instructed to describe the visible evidence before making a decision or issuing another action. Once sufficient evidence has been collected, it terminates the trajectory with a JSON object enclosed by `<answer>` tags:

```
<answer>{
  "product_consistency": 0/1,
  "human_plausibility": 0/1,
  "text_quality": 0/1,
  "background_plausibility": 0/1
```

```
}</answer>
```

where 1 denotes the presence of a defect.

For comparisons with tool-augmented VLM baselines, we retain the same task rubric, image order, label semantics, and terminal answer schema, while replacing only the tool-protocol block with the native syntax expected by each checkpoint. This avoids penalizing a model for an unfamiliar action format. All tool-capable models are assigned the same maximum number of tool interactions, whereas direct-response models receive no external visual tool.

ShopEye Diagnostic Prompt

ROLE AND INPUTS

You are an expert in assessing the quality of AIGC product images. You receive an image before editing (`orig`) and an edited image (`aigc`). Determine whether the edited image has problems in four dimensions. **Label 1 means a problem is present; label 0 means no problem.**

DIAGNOSTIC CHECKLIST

Inspect the dimensions in this order: **Product** → **Human** → **Text** → **Scene**. The following conditions are the main reasons to assign label 1.

01 PRODUCT

Goal. Determine whether the product in `aigc` remains consistent with `orig`.

- **Style or cut.** Clear changes in category, length, silhouette, neckline, or sleeve length.
- **Color.** The dominant color or main color scheme changes to a different color family.
- **Structure.** Pockets, buttons, zippers, ties, cuffs, or hems are missing, added, misplaced, or clearly deformed.
- **Material.** Clear changes in material properties, such as knit, fur, velvet, or transparency.
- **Logo, pattern, or texture.** Missing, added, mirrored, misplaced, broken, duplicated, or structurally altered elements.

02 HUMAN

Goal. Identify clear generation defects in the human body and its direct interactions in `aigc`.

- **Anatomy.** Extra, missing, repeated, incorrectly connected, or abnormally truncated body parts.
- **Hands.** Clearly extra or missing fingers, fusion, breaks, severe twisting, or implausible bending.
- **Face.** Repeated, missing, misplaced, fused, or severely distorted facial features.
- **Limbs.** Clear twisting, stretching, abnormal proportions, or incorrect connections in arms, legs, shoulders, or the neck.
- **Body–object interactions.** Clear penetration, incorrect occlusion, or implausible connections involving clothing, held objects, or accessories.
- **Accessories.** Glasses, necklaces, watches, or bag straps clearly float, penetrate, break, or deform severely.

03 TEXT

Goal. Inspect all visible text in `aigc`, including product logos, labels, printed text, and background text on signs, posters, walls, books, and buildings.

- **Garbled or incorrect text.** Meaningless characters, missing characters, incorrect order, misspellings, or unrecognizable symbols.
- **Deformation.** Strokes are severely broken, fused, duplicated, or distorted to the point of illegibility.
- **Character structure.** Character arrangements or forms clearly do not conform to normal writing.
- **Placement.** Text clearly floats, passes through surface structures, or is severely misaligned with its surface plane.

04 SCENE

Goal. Identify clear generation defects in the environment, background objects, and spatial relations in `aigc`.

- **Structure.** Furniture, buildings, or other background objects are clearly distorted, broken, incomplete, or structurally wrong.
- **Support.** People or objects clearly float, feet fail to touch the ground, or reasonable support is absent.
- **Intersections.** Clearly implausible penetration between people and objects, or between objects.
- **Space and perspective.** Clearly incorrect building lines, furniture structures, relative sizes, or perspective.
- **Lighting.** Clear contradictions in light direction, contact shadows, or lighting between the person and the scene.
- **Reflections.** Mirror or glass reflections clearly disagree with the person or scene.
- **Texture and edges.** Abnormal repeated textures, obvious jagged edges, color bleeding, halos, or stitching artifacts.
- **Background text.** An environmental text error has already been confirmed in the preceding Text inspection.

ShopEye Diagnostic Prompt (continued)

VISUAL TOOLS

Use tools as needed. Do not call them mechanically when the available evidence is sufficient.
Coordinates. All bounding boxes use relative coordinates from 0 to 999, with top-left $[0, 0]$ and bottom-right $[999, 999]$. Compute the `orig` and `aigc` boxes independently in their respective images.

`compare` Compare corresponding product regions in `orig` and `aigc`.

```
<tool_call>{"name":"compare","arguments":{
  "bbox_a":[x1,y1,x2,y2], "bbox_b":[x1,y1,x2,y2],
  "label":"region_name"}}</tool_call>
```

`crop` Inspect a local region in a single image.

```
<tool_call>{"name":"crop","arguments":{
  "image":"aigc", "bbox":[x1,y1,x2,y2],
  "label":"region_name"}}</tool_call>
```

`zoom_in` Enlarge hands, text, logos, buttons, or other small regions.

```
<tool_call>{"name":"zoom_in","arguments":{
  "image":"aigc", "bbox":[x1,y1,x2,y2], "scale":3,
  "label":"region_name"}}</tool_call>
```

INSPECTION PROCEDURE

Complete the dimensions sequentially using the **Analyze** → **Call** → **Observe** → **Judge** loop. Each round follows this format:

1. Analyze the current dimension and identify the region and tool needed for inspection.
2. Issue a `<tool_call>` using the required name and arguments.
3. The system returns the cropped or enlarged image.
4. Describe the specific visual content observed, compare it against the checklist, and determine whether it is a `FAIL`.

If further confirmation is needed, repeat the tool-call format above. **After every tool return, describe the observation before judging. Do not skip the observation and jump directly to a conclusion.**

FINAL OUTPUT TEMPLATE

After checking all dimensions, provide the final conclusion in the following format. Each value must be either `0` or `1`.

```
<answer>{
  "product_consistency": 0/1,
  "human_plausibility": 0/1,
  "text_quality": 0/1,
  "background_plausibility": 0/1
}</answer>
```

`0/1` is a placeholder; replace it with one integer in an actual response.

English translation, reorganized for paper presentation.

D.1 Visual Tool Definitions

We provide three deterministic visual operations over the original–edited image pair. A bounding box is represented as $b = (x_1, y_1, x_2, y_2)$ in a shared $[0, 999]$ coordinate space. For an image with width W and height H, it is mapped to pixel coordinates as

$$\pi(b; W, H) = \left(\frac{x_1 W}{999}, \frac{y_1 H}{999}, \frac{x_2 W}{999}, \frac{y_2 H}{999} \right). \tag{14}$$

Coordinates are validated and clipped to the corresponding image boundary. Because the original and edited images may have different layouts, their bounding boxes are always interpreted independently.

`crop`. The `crop` operation extracts a region from either I_{orig} or I_{aigc}:

$$\texttt{crop}(\texttt{image}, b, \texttt{label}).$$

It is used to inspect a localized object or artifact while preserving the native pixels within the selected region. Very small regions are expanded around their center to retain sufficient visual context.

zoom_in. The zoom_in operation first crops a region and then enlarges it by a scale factor $s \in [1, 4]$:

$$\texttt{zoom_in}(\texttt{image}, b, s, \texttt{label}).$$

We use Lanczos resampling followed by mild sharpening. This operation is intended for fine-grained evidence such as fingers, facial features, logos, buttons, seams, and small text.

compare. The compare operation accepts two independently predicted regions, b_{orig} and b_{aigc}:

$$\texttt{compare}(b_{\text{orig}}, b_{\text{aigc}}, \texttt{label}).$$

The corresponding crops are placed side by side and explicitly marked as original and AIGC-generated. Unlike a shared-coordinate crop, this design supports changes in object position, scale, or pose between the two images. It is primarily used to verify product identity, color, material, structural components, logos, and patterns.

Each successful action produces a new visual observation that is appended to the interaction history. Tool outputs are resized under a fixed pixel budget before being passed back to the VLM, controlling visual-token growth across multi-step trajectories. Invalid JSON, unsupported operations, malformed boxes, and out-of-range scale factors are rejected without terminating the rollout worker.

D.1.1 Comparison Baselines

We implement all baselines in the ROLL agentic inference framework (Wang et al., 2025a). Each model receives the original and AIGC-generated images and predicts four binary quality labels. To support different model interfaces, we convert model-specific outputs into three unified actions: visual inspection, code execution, and final answer.

Chain-of-Focus (Zhang et al., 2025b) and InSight-o3 (Li et al., 2026) use pixel-coordinate zooming on the latest visual observation. Pixel-Reasoner (Su et al., 2026) predicts normalized bounding boxes and an image-history index. Thyme-RL (Zhang et al., 2026b) generates image-processing code, whose output image is returned to the model for subsequent reasoning. DeepEyes (Zheng et al., 2026) and SenseNova-MARS-8B (Chng et al., 2025) use our shared compare, crop, and zoom_in tools with relative coordinates in $[0, 999]$. GRiP (Wei et al., 2025) and CodeV-RL (Zhu et al., 2026b) directly inspect the two input images without external tools.

All crop requests are converted to pixel coordinates and executed by the same PIL-based backend. The resulting image is appended to the conversation for the next reasoning step. We allow at most four tool calls and six model actions per sample. Inference uses greedy decoding with one trajectory per sample.

E Additional Experimental Results

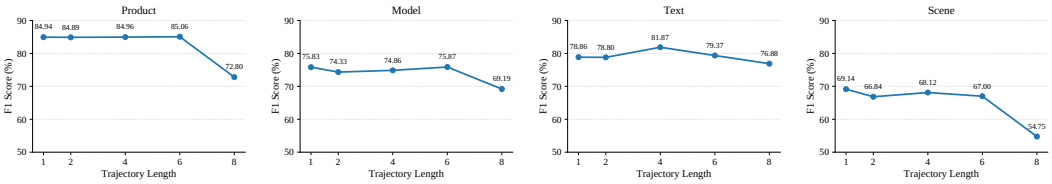
E.1 Effect of LoRA Rank

Table 5 studies the effects of the learning rate and LoRA rank. We observe that the learning rate strongly affects optimization stability. With $r = 16$, increasing the learning rate from 5×10^{-5} to 1×10^{-4} and 5×10^{-4} leads to a substantial performance drop, and the largest learning rate causes the recall of all four dimensions to nearly collapse. Meanwhile, the effect of LoRA rank is non-monotonic and depends strongly on the learning rate, indicating that the two hyperparameters should be tuned jointly.

Importantly, we do not select the final configuration solely according to the highest Overall score. ShopEye requires reliable diagnosis across all four anomaly dimensions, and several configurations with competitive aggregate performance exhibit severe dimension-specific degeneration. For example, $(1 \times 10^{-5}, r = 128)$ achieves the highest Micro-F1 but has only 3.08 F1 on Product, while $(5 \times 10^{-5}, r = 16)$ achieves the highest Macro-F1 but almost fails to detect Product anomalies (0.12 F1). In contrast, $(1 \times 10^{-4}, r = 128)$ maintains non-trivial performance on all four dimensions, achieving F1 scores of 31.08, 24.44, 14.09, and 44.98 on Product, Human, Scene, and Text, respectively. We therefore adopt 1×10^{-4} with $r = 128$ as our default configuration, favoring balanced cross-dimensional performance over a marginally higher aggregate score.

Table 5: Hyperparameter analysis of the learning rate and LoRA rank on ShopEye-Bench.

LR	Rank	Product			Human			Scene			Text			Overall
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	
1×10^{-5}	128	88.94	1.57	3.08	69.52	6.66	12.15	36.78	**56.80**	**44.65**	49.61	**68.69**	**57.61**	<u>29.37</u>
5×10^{-5}	16	**100.00**	0.06	0.12	61.76	**38.70**	**47.58**	26.21	32.82	<u>29.15</u>	51.08	36.14	42.33	**29.80**
5×10^{-5}	32	79.63	8.65	15.61	67.15	6.15	11.27	28.24	1.76	3.32	47.70	2.04	3.91	8.53
5×10^{-5}	128	78.66	2.11	4.10	65.43	4.74	8.84	35.43	8.26	13.40	64.73	11.99	20.23	11.64
1×10^{-4}	16	78.10	8.21	14.86	64.28	6.15	11.23	39.39	1.33	2.57	68.91	5.33	9.90	9.64
1×10^{-4}	32	85.28	2.82	5.47	**75.93**	2.15	4.17	35.74	1.95	3.69	65.65	2.28	4.41	4.43
1×10^{-4}	128	82.41	**19.15**	**31.08**	65.94	15.00	<u>24.44</u>	40.59	8.53	14.09	**73.24**	32.46	<u>44.98</u>	28.65
5×10^{-4}	16	82.71	0.92	1.82	60.91	0.62	1.23	**45.24**	0.36	0.71	61.28	1.75	3.40	1.79



(a) Product (b) Human (c) Text (d) Scene

Figure 11: Effect of the maximum tool-call trajectory length during training.

E.2 Effect of Maximum Tool Calls

We study the effect of the maximum tool-call trajectory length during training by limiting each rollout to at most 1, 2, 4, 6, or 8 tool invocations. As shown in Fig. 11, increasing the interaction budget does not consistently improve performance. Product and Human remain relatively stable for trajectories of up to six tool calls, while Text achieves its best performance with four calls. In contrast, allowing eight tool calls leads to a substantial degradation across all four dimensions. This suggests that excessively long trajectories may introduce redundant or noisy visual evidence and make credit assignment more difficult during policy optimization. Overall, a moderate interaction budget provides a better balance between evidence acquisition and training stability, and we therefore set the maximum trajectory length to 6 in our experiments.

E.3 Case Study

As shown in Figure 12, the proposed inspection process evaluates the generated clothing image from multiple visual dimensions, including product consistency, human plausibility, background plausibility, and text quality. Although the generated result is visually coherent in terms of overall scene layout, it still contains several typical AIGC artifacts, such as changed garment lettering, distorted hand details, and unreadable text in both the clothing region and the background posters.

F Limitations and Future Work

Efficiency of active visual inspection. Although active tool invocation improves fine-grained visual verification, executing multiple crop, zoom, and comparison operations inevitably introduces additional inference latency. A promising direction is to use the localized bounding boxes available during training as privileged supervision. For example, an OPD-style training objective could distill region-level inspection knowledge into the model, encouraging it to directly predict or internally attend to informative regions without explicitly executing every visual tool at inference time. This may retain the benefits of localized verification while substantially reducing the cost of multi-step interaction.



Product Detail and Consistency examines clothing consistency by comparing the reference image and the generated image in terms of color, style, and fine-grained details.
<tool_call>{"name": "crop", "arguments": {"image": "aigc", "bbox": [86, 138, 777, 898], "label": "garment_body"}}</tool_call>
The lettering on the chest has changed. The product check fails.

Human Appearance and Limb Plausibility examines the model's appearance. The body posture is complete, the head-to-body ratio is normal, and the facial features are well-formed and symmetrical, with no obvious abnormalities. The left hand is a high-frequency distortion region in AIGC images and requires fine-grained inspection.
<tool_call>{"name": "zoom_in", "arguments": {"image": "aigc", "bbox": [17, 691, 216, 944], "scale": 4, "label": "hand_0"}}</tool_call>
The left hand shows broken-finger deformation.



No obvious anomaly is found in the scene background. The background shows a corridor environment with cement walls and floor, as well as multiple vintage posters on the wall. The overall composition is naturally blended, so the scene check passes.

Text Quality evaluates the text quality in the image. The English print on the T-shirt is completely unreadable and exhibits severe garbling, which is a clear generation error. The posters on the background wall also contain a large amount of illegible gibberish text, with missing strokes and meaningless character combinations, which are typical AI-generated environmental text errors. Therefore, the text quality check fails.

<answer>{ "product_consistency": 1, "human_plausibility": 1, "text_quality": 1, "background_plausibility": 0}</answer>

Figure 12: Example of AIGC clothing image inspection.

Scaling the diversity of ShopEye-Bench. ShopEye-Bench currently focuses on representative fine-grained anomalies in e-commerce AIGC. While ShopEye-Agent enables scalable annotation, the benchmark can be further expanded to cover a broader range of product categories, editing scenarios, generation models, and long-tail failure patterns. In future work, we plan to leverage the evolving agent pipeline to continuously mine and annotate more diverse samples, while retaining human verification and quality control to prevent annotation noise from accumulating. Such expansion would improve both the coverage of the benchmark and its ability to evaluate model robustness under increasingly diverse AIGC distributions.