INDUSAGENT: AGENTIC REINFORCEMENT LEARNING FOR CROSS-CATEGORY INDUSTRIAL ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Small defects in unfamiliar product categories can be difficult to distinguish from benign appearance variations in a global view. Additional reasoning over the same view may leave this ambiguity unresolved. We introduce **IndusAgent**, a Qwen3-VL-8B inspection policy trained through supervised fine-tuning and tool-augmented reinforcement learning for cross-category industrial anomaly detection. The policy selects among six tools for cropping, enhancement, measurement, within-image comparison, texture analysis, and external visual opinions from Qwen3-VL-Max, then integrates returned observations into a structured diagnosis. We construct Indus-CoT with 3,000 trajectories from 26 Real-IAD categories, generated with actual tool feedback and screened against annotations after teacher generation. GRPO optimizes an accuracy-gated objective that conditions localization, type, and tool rewards, including call costs, on final diagnostic correctness while retaining an independent format reward. Under the common protocol, IndusAgent exceeds direct Max inference in mean balanced accuracy across five benchmarks (83.5% versus 79.4%). Across three training seeds, the gated policy achieves higher five-benchmark mean balanced accuracy than the weight-matched ungated control (83.5% versus 80.5%). Evaluation uses disjoint product categories and a fixed defect vocabulary. Inference requires no paired normal exemplar but can access the external prior API.

1 INTRODUCTION

Industrial anomaly detection aims to identify defective products and localize abnormal regions, supporting quality inspection in manufacturing (Bergmann et al., 2019; Zou et al., 2022). Reliable inspection requires detecting both surface defects and structural irregularities across diverse products. Conventional reconstruction and feature-based methods learn normal appearance from product-specific examples and identify deviations from this learned distribution (Zavrtanik et al., 2021; Roth et al., 2022). Advances in vision-language modeling have introduced semantic knowledge and visual reasoning into inspection, creating opportunities for transferring detection capabilities across product categories (Jeong et al., 2023; Gu et al., 2024; Xu et al., 2025). However, industrial defects often occupy small image regions and resemble benign texture variations, reflections, or structural details. Distinguishing these subtle defects becomes especially challenging on unfamiliar products, where differences in geometry, material, and texture change what should be considered normal.

Multimodal large language models provide a foundation for addressing these challenges by combining image understanding with language-based reasoning (OpenAI, 2023; Bai et al., 2025b;a). Beyond describing visible objects, they can interpret inspection instructions, discuss suspicious regions, and produce structured diagnostic outputs. Industrial post-training further adapts these capabilities through supervised fine-tuning and reinforcement learning (Li et al., 2026; Chao et al., 2025). Tool-assisted approaches extend the available evidence by allowing models to inspect selected regions and incorporate external observations. AgentIAD, for example, combines tool-integrated trajectories with supervised fine-tuning and reinforcement learning, supporting both local inspection and normal-exemplar retrieval (Miao et al., 2026). These developments support a transition from predicting anomalies directly to learning how to inspect an image before reaching a decision.

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.

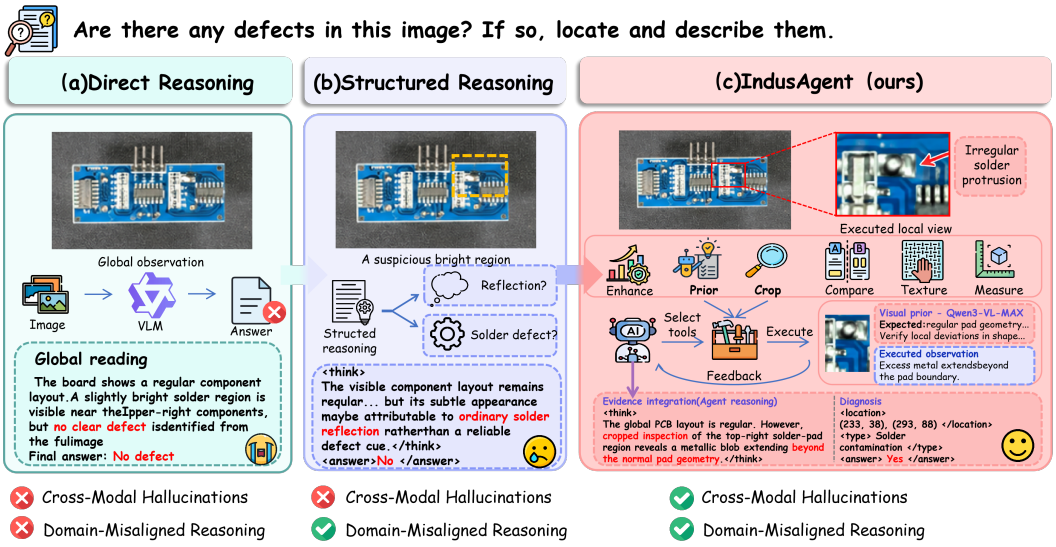


Figure 1: **Insight of IndusAgent.** On the illustrated PCB query, direct reasoning and structured reasoning produce a no-defect decision, while IndusAgent inspects a returned local crop and incorporates an external visual opinion before predicting an anomaly. The three columns illustrate alternative inspection paths for this example. The visual-prior tool queries **Qwen3-VL-Max**.

Nevertheless, reliable inspection of unfamiliar products requires addressing two related limitations. ❶ **Global visual understanding may overlook the local structural deviations that define industrial defects.** Recognizing a regular component layout does not establish whether an individual solder joint or surface region is normal. As shown in Figure 1(a), direct reasoning describes the overall PCB layout and notices a bright solder region, yet concludes that no clear defect is present. ❷ **Structured reasoning can leave competing visual explanations unverified.** Organizing an assessment into reasoning steps does not provide the additional observations needed to distinguish a defect from a benign appearance variation. In Figure 1(b), the model considers the suspicious region but interprets its brightness as a possible ordinary solder reflection and retains a normal decision. The uncertainty is therefore carried into the conclusion without inspecting the local geometry that could resolve it.

To address these limitations, we argue that effective industrial inspection requires two complementary capabilities. ❶ **Adaptive acquisition of local evidence** enables the policy to select informative regions and inspection operations when the global view is insufficient. ❷ **Verification of domain-specific hypotheses** connects returned observations with expectations about normal geometry, texture, and structure, while treating external opinions as hypotheses to verify. Figure 1(c) illustrates how these capabilities work together: IndusAgent requests a local view and an external visual prior, examines the metal extending beyond the solder-pad boundary, and integrates this evidence into an anomalous diagnosis. This example motivates learning an inspection policy that can decide what evidence to acquire and how that evidence should affect its conclusion.

We introduce **IndusAgent**, a tool-augmented reinforcement learning framework for cross-category industrial anomaly detection. Its Qwen3-VL-8B policy selects among six tools for cropping, enhancement, geometric measurement, within-image comparison, texture analysis, and an external visual prior from Qwen3-VL-Max. Figure 2 presents the three-stage training pipeline. ❶ As shown in Figure 2(a), we construct *Indus-CoT*, a corpus of 3,000 trajectories from 26 Real-IAD categories, using teacher-selected tool calls and actual execution feedback. Ground-truth annotations screen completed trajectories rather than guide the teacher's inspection actions. ❷ In Figure 2(b), response-only supervised fine-tuning initializes the policy to select tools and integrate their observations. ❸ In Figure 2(c), online two-round GRPO refines this behavior: the first response provides a preliminary assessment and optional tool requests, and the second produces the final diagnosis using the available evidence. An accuracy-gated objective makes final diagnostic correctness a prerequisite for localization, defect-type, and tool-related rewards, including call costs, while supervising output format

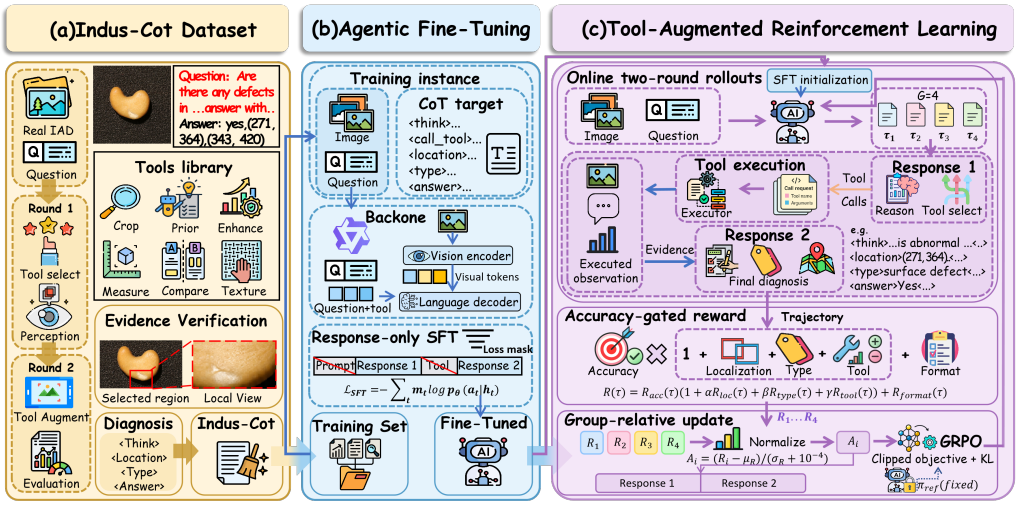


Figure 2: **Overview of the IndusAgent pipeline.** (a) Indus-CoT contains 3,000 screened trajectories from 26 Real-IAD categories, pairing teacher-selected tool calls with executed feedback. (b) Response-only SFT initializes the inspection policy. (c) Online two-round GRPO refines the policy with an accuracy-gated reward, conditioning auxiliary inspection rewards and call costs on final diagnostic correctness while supervising format independently.

independently. We evaluate this framework on target product categories excluded from industrial post-training, without providing paired normal exemplars at inference. Our contributions are:

- **Tool-Augmented Industrial Inspection.** We propose **IndusAgent**, a framework that enables a Qwen3-VL-8B policy to select among six tools and integrate their feedback for cross-category industrial anomaly detection. It combines local inspection with an external visual prior and supports inference without a paired normal exemplar.
- **Indus-CoT Dataset and Agentic Training.** We construct **Indus-CoT**, a dataset of 3,000 inspection trajectories from 26 Real-IAD categories, generated with teacher-selected tool calls, executed feedback, and post-generation annotation screening. Our two-stage SFT–GRPO training pipeline builds on this dataset and uses an accuracy-gated objective to condition auxiliary inspection rewards and call costs on final diagnostic correctness while supervising format independently.
- **Empirical Effectiveness.** IndusAgent achieves 83.5% mean balanced accuracy across five industrial anomaly detection benchmarks, exceeding direct Qwen3-VL-Max inference by 4.1 percentage points under the common evaluation protocol. Across three training seeds, the accuracy-gated objective further yields a 3.0-point mean improvement over a weight-matched ungated objective.

2 METHODOLOGY

Overview. IndusAgent consists of teacher-trajectory construction, supervised policy initialization, and online tool-augmented GRPO (Fig. 2). We first screen generated trajectories against source annotations, then train the policy to produce structured responses, and finally optimize the policy with the accuracy-gated reward.

2.1 SYSTEMATIC DEFINITION AND AGENTIC TOOLKIT

Problem Formulation. Given a query image $I \in \mathbb{R}^{H\times W\times 3}$ and instruction Q, the policy π_θ produces a binary diagnosis and an inspection trace. When predicting an anomaly, it also returns a final defect box and a label from a fixed canonical defect vocabulary. We represent the final defect box as $B = [x_1, y_1, x_2, y_2]$ in original-image pixel coordinates, with (x_1, y_1) and (x_2, y_2) denoting the top-left and bottom-right corners, respectively. The backbone is Qwen3-VL-8B-Instruct (Bai et al., 2025a). Source training categories are disjoint from the evaluation categories, and no paired normal exemplar is supplied at inference.

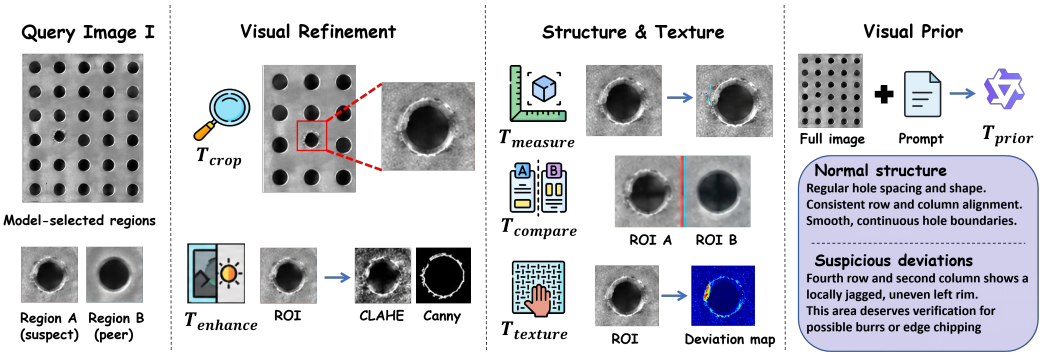


Figure 3: **Six-tool inspection library.** Crop, enhancement, measurement, within-image comparison, and texture analysis provide observations derived from the query image. The comparison regions shown here come from the same query, without a paired normal exemplar. The visual-prior tool queries the external Qwen3-VL-Max API for descriptions of structure and visible deviations.

Fixed Two-round Inference with Optional Tool Feedback. Training and inference use the same two-round protocol. The first response produces a preliminary diagnosis and selects zero or more calls from $\mathcal{T} = \{T_{\text{crop}}, T_{\text{prior}}, T_{\text{enhance}}, T_{\text{measure}}, T_{\text{compare}}, T_{\text{texture}}\}$. The environment executes the selected calls and returns their observations. If no tool is selected, the call and observation lists are both empty. A second response is generated in every case:

$$a^{(1)} \sim \pi_\theta(\cdot \mid I, Q), \qquad a^{(2)} \sim \pi_\theta(\cdot \mid I, Q, a^{(1)}, o), \tag{1}$$

where o is the observation list. Only the second response provides the final diagnosis and, for an anomalous prediction, the defect box in original-image pixel coordinates and canonical type. Appendix E specifies the shared output schema.

Agentic Toolkit. As shown in Fig. 3, we equip IndusAgent with six complementary tools that provide visual, quantitative, and textual observations for category-disjoint industrial anomaly detection. All six are available during training and target inference.

(1) Crop (T_{crop}) extracts model-selected regions; **(2) Enhance** (T_{enhance}) applies CLAHE and Canny operators to reveal local details; **(3) Measure** (T_{measure}) computes pixel distances and angles; **(4) Compare** (T_{compare}) contrasts similar regions within the same query image; **(5) Texture** (T_{texture}) characterizes local texture variations; **(6) Visual Prior** (T_{prior}) queries Qwen3-VL-Max with the full query image to describe expected normal structure and suspicious deviations, without returning a final Yes/No diagnosis.

2.2 INDUS-COT DATASET

We construct **Indus-CoT** for supervised initialization using 3,000 Real-IAD trajectories, balanced between 1,500 normal and 1,500 anomalous images (Wang et al., 2024). Four product categories overlapping with the evaluation benchmarks are excluded before construction, leaving 26 categories.

- **Inspection and tool selection.** Given the query image and task instruction, the teacher forms a preliminary diagnosis and selects tools and arguments without ground-truth labels, boxes, types, or a paired normal reference.
- **Tool execution and diagnosis.** The environment executes the selected calls and returns the resulting images or text. The teacher integrates this feedback into a final diagnosis and, when predicting an anomaly, a defect box and type.
- **Post-generation screening.** Completed trajectories are checked against source annotations for output format, binary correctness, and required auxiliary fields. Anomalous samples must match the canonical defect type and achieve box IoU ≥ 0.5. Normal samples must end with `No` and no final box or type.

Initial screening retains 2,747 trajectories. We obtain 253 accepted replacements from additional images in the same source categories using the same procedure. Failed candidates are discarded without annotation-guided repair. Appendix B.1 provides the category audit and screening details.

2.3 SUPERVISED FINE-TUNING

We initialize Qwen3-VL-8B-Instruct with supervised fine-tuning on Indus-CoT before online reinforcement learning. This stage provides supervision for the diagnostic output schema, tool routing, and interpretation of returned evidence. The no-SFT ablation evaluates the effect of this initialization within our training setup; it does not establish that SFT is universally necessary for inspection.

Formally, let $A = (a_1, \ldots, a_T)$ contain all policy-generated response tokens, including reasoning, tool routing and arguments, the final binary answer, and the optional location and type fields. Prompt tokens and externally returned tool observations are provided as context but are masked from the loss. With $m_t \in \{0, 1\}$ denoting this response mask, SFT minimizes

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{\tau \sim \mathcal{D}} \left[\sum_{t=1}^{T} m_t \log \pi_\theta(a_t \mid h_t) \right]. \tag{2}$$

Here h_t includes the query, instruction, preceding responses, and any executed observations available before token t. The objective supervises the policy output, including the text inside `<think>`; environment-returned observations remain non-differentiable context for the next response.

2.4 AGENTIC REINFORCEMENT LEARNING

Online Two-round Rollouts. For each source query, the rollout policy $\pi_{\theta_{\text{old}}}$ samples G first responses. Each response selects zero or more calls; the environment executes them and returns the observations. A second response is generated for every trajectory, including no-tool trajectories. Ground-truth labels, boxes, and types are supplied to reward computation, not to the policy input. Both policy-generated response segments contribute to the loss; prompts and environment-returned observations do not. Inference follows the same two-round protocol described in Section 2.1.

Group Relative Policy Optimization (GRPO). We adopt GRPO (Shao et al., 2024) for policy optimization. For each query q, we sample G rollout trajectories $\{o_1, \ldots, o_G\}$ from the previous policy $\pi_{\theta_{\text{old}}}$ and compute their advantages from within-query reward comparisons.The objective below uses compact trajectory-indexed notation. Within each trajectory contribution, probability ratios are evaluated separately at each policy-generated token, and clipping precedes token averaging. We first average the token losses within each trajectory and then average over sampled trajectories. Token indices and conditioning on the preceding history are suppressed: the ratios in the clipped objective refer to token-level conditional probabilities rather than joint trajectory probabilities.

$$\begin{aligned}\mathcal{L}_{GRPO}(\theta) = -\mathbb{E}_{q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)} \Bigg[\frac{1}{G} \sum_{i=1}^{G} \\ \left(\min \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)} A_i, \text{clip} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\epsilon, 1+\epsilon \right) A_i \right) - \beta \mathbb{D}_{KL}(\pi_\theta \| \pi_{ref}) \right) \Bigg],\end{aligned} \tag{3}$$

The KL expression below likewise suppresses token indices and history conditioning and denotes a sampled penalty relative to the fixed reference policy. KL computation is separate from groupwise reward normalization.

$$\mathbb{D}_{KL}(\pi_\theta || \pi_{ref}) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \tag{4}$$

Here, ϵ controls clipping, and β denotes the KL regularization coefficient within this subsection; it is distinct from the type-reward coefficient denoted by β in the reward design. Each trajectory shares one advantage across its policy tokens. For a group with nonzero reward standard deviation, the advantage is

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \ldots, r_G\})}{\text{std}(\{r_1, r_2, \ldots, r_G\})}. \tag{5}$$

When all rewards in a group are equal, we set $A_i = 0$ for every trajectory without evaluating the division above. Such a group contributes no reward-driven policy update. Each trajectory receives the composite reward defined below.

Accuracy-gated Reward. For a completed trajectory τ, final diagnostic correctness controls the contribution of the auxiliary inspection terms:

$$R(\tau) = R_{\text{acc}}(\tau)\big(1 + \alpha R_{\text{loc}}(\tau) + \beta R_{\text{type}}(\tau) + \gamma R_{\text{tool}}(\tau)\big) + R_{\text{format}}(\tau), \tag{6}$$

with $\alpha = 0.8$, $\beta = 0.6$, and $\gamma = 0.5$. Binary correctness gives $R_{\text{acc}} \in \{0, 1\}$. For truly anomalous images, R_{loc} is continuous IoU between predicted and ground-truth boxes expressed in the same original-image pixel coordinate system, and R_{type} is exact agreement after deterministic canonicalization. Missing or invalid predictions receive zero, and both terms are zero for truly normal images. The teacher-screening threshold IoU≥ 0.5 is separate from the continuous RL localization reward.

The tool term combines a confidence-gain bonus with a cost per parsed call:

$$R_{\text{tool}} = 0.3\,\mathbb{I}[N_{\text{ok}} > 0 \ \wedge \ \Delta m > 0] - 0.1 N_{\text{call}}. \tag{7}$$

Here N_{call} counts up to four parsed calls admitted to execution, including repeated calls; excess requests are ignored. A counted call incurs cost even if execution fails. N_{ok} counts recognized observations with status `ok`. The margin change Δm uses Yes/No logits oriented toward the final predicted label. It measures confidence change across the two response contexts, which also differ in reasoning history (Appendix A). A compliant two-round trajectory receives $R_{\text{format}} = 0.05$; otherwise this term is zero. With $N_{\text{call}} \leq 4$, correct diagnoses yield $R \geq 0.8$, whereas incorrect diagnoses yield $R \leq 0.05$.

3 Experiments

3.1 Experimental Setup

Datasets and Benchmarks. We evaluate **IndusAgent** on five industrial anomaly detection benchmarks: MVTec-AD (Bergmann et al., 2019), VisA (Zou et al., 2022), MPDD (Jezek et al., 2021), DTD (Aota et al., 2023), and SDD (Tabernik et al., 2020). These datasets cover two representative scenarios: (1) *industrial objects*, which are characterized by complex structures, poses, and geometries; (2) *surface textures*, where defects are often subtle and embedded within repetitive patterns.

Common-protocol Comparisons. We evaluate the direct Qwen3-VL-8B backbone, direct Qwen3-VL-Max, IAD-R1, the full system, and the full system with the prior disabled at inference. These rows use the same images, preprocessing, binary task semantics, and metric definitions, with model-appropriate answer extraction described in Appendix B.3.

Implementation Details. We train Qwen3-VL-8B-Instruct on a single node with four 80GB A100 GPUs. Indus-CoT SFT and accuracy-gated, fixed-two-round GRPO each run for one epoch, taking approximately 21.6 minutes and 43.5 hours, respectively. GRPO uses four generations per prompt, a per-device batch size of 1, two gradient accumulation steps, and prompt/completion limits of 4096/512 tokens, with bfloat16 precision, gradient checkpointing, and DeepSpeed ZeRO-3.

3.2 Main Results

Table 1 provides literature-reported context across five industrial-object and surface-texture benchmarks for IndusAgent's hybrid 8B policy and external visual-prior API. Table 2 compares systems under a common protocol.

IndusAgent obtains 83.5% mean balanced accuracy, exceeding direct Max inference (79.4%) by 4.1 points. The IAD-R1 rerun uses Qwen2.5-VL-7B. Prior disabling removes its prompt description and Max calls while retaining the checkpoint and training-time prior exposure. The resulting 79.7% (a 3.8-point decrease) thus reflects both inference-time prior availability and the prompt change. Across three training seeds, the full policy achieves higher mean balanced accuracy than the weight-matched ungated control (83.5 ± 0.4 versus 80.5 ± 0.5 five-benchmark mean balanced accuracy; Table 9), supporting the complete accuracy-gated objective under the reported training setup.

Qualitative comparison. Figure 4 contrasts the backbone's normal prediction with IndusAgent's suspected insulation break, supported by crop-based comparison with adjacent intact insulation. The second response retains the first-round anomaly judgment and reports defect location and type.

Table 1: Image-level balanced accuracy (%) on five IAD benchmarks. Baseline results other than our Qwen3-VL-8B rerun are transcribed from IAD-R1 (Li et al., 2026). IndusAgent combines an 8B policy with the Qwen3-VL-Max API.

Type	Model	Param.	Industrial Workpieces			Surface Texture		Avg.
			MVTec	MPDD	VisA	DTD	SDD	
Commercial	GPT-4o (OpenAI, 2024)	/	69.6	60.3	63.5	69.9	65.7	65.8
	GPT-4.1-nano (OpenAI, 2025)	/	74.7	61.7	60.5	78.3	50.0	65.0
	GPT-4.1 (OpenAI, 2025)	/	81.9	66.7	69.1	90.1	79.9	77.5
	Claude-Sonnet-4 (Anthropic, 2025)	/	67.6	65.9	63.5	88.4	81.7	73.4
Open Source	LLaVA-OneVision-SI (Li et al., 2024)	0.5B	50.0	50.0	50.0	54.3	50.0	50.9
	Qwen2.5-VL-Instruct (Bai et al., 2025b)	3B	62.6	52.9	58.4	64.4	50.3	57.7
	AnomalyR1 (Chao et al., 2025)	3B	69.4	56.0	59.8	61.0	57.6	60.8
	InternVL-2.5 (Chen et al., 2024)	4B	56.6	59.1	53.7	81.3	64.1	63.0
	Qwen2.5-VL-Instruct (Bai et al., 2025b)	7B	66.0	56.0	58.4	59.2	67.4	61.4
	Qwen3-VL-Instruct (rerun) (Bai et al., 2025a)	8B	67.0	50.0	46.8	70.2	50.0	56.8
	AnomalyGPT (Gu et al., 2024)	7B	46.6	54.2	57.3	64.1	49.5	54.3
	Anomaly-OV (Xu et al., 2025)	7B	74.3	70.3	74.3	90.7	88.7	79.7
	LLaVA-1.5 (Liu et al., 2023a)	13B	61.4	61.4	67.2	75.9	50.0	63.2
	LLaVA-1.6 (Liu et al., 2024)	34B	53.7	50.0	53.9	52.7	50.0	52.1
	Qwen2.5-VL-Instruct (Bai et al., 2025b)	72B	77.4	64.7	68.2	81.9	69.0	72.2
	IAD-R1 (Li et al., 2026)	7B	81.9	65.8	75.4	90.8	83.4	79.5
Hybrid	IndusAgent (8B policy + prior API)	8B	83.6	72.7	76.8	95.6	88.9	83.5

Table 2: **Common-protocol system-level balanced accuracy (%).** Max access indicates use of the external Qwen3-VL-Max component during inference. Prior disabling uses the full trained checkpoint with that tool removed from the prompt; it does not remove prior access from training.

Method	Policy	Max access	MVTec	MPDD	VisA	DTD	SDD	Avg.
Qwen3-VL-8B direct	8B	No	67.0	50.0	46.8	70.2	50.0	56.8
IAD-R1, rerun	7B	No	78.4	68.1	70.2	90.9	82.0	77.9
Qwen3-VL-Max direct	API	Yes	80.2	68.9	73.5	90.8	83.4	79.4
Prior disabled at inference	8B	No	81.1	70.4	69.3	92.1	85.8	79.7
IndusAgent	8B	Yes	83.6	72.7	76.8	95.6	88.9	83.5

3.3 Classification and Grounding

Table 3 separates anomaly recall, normal-class specificity, and grounding quality, distinguishing binary detection performance from the accuracy of defect localization and type prediction. On MPDD, 95.4% recall is accompanied by 50.0% specificity: the system detects most anomalous images but incorrectly flags half of the normal images, yielding 72.7% balanced accuracy. This operating point would impose a substantial reinspection burden. Grounding metrics provide complementary evidence, since a correct anomaly decision does not guarantee an accurate defect box or type. These metrics are evaluated over all ground-truth anomalous images, with missing or invalid fields receiving zero, so incomplete outputs remain reflected in the reported grounding performance.

Table 3: **Classification and grounding of the full IndusAgent system (%).** Grounding metrics use all ground-truth anomalous images; missing or invalid fields receive zero. Loc.@0.5 denotes the fraction with box IoU ≥ 0.5.

Dataset	BA	AUROC	Recall	Specificity	Box mIoU	Loc.@0.5	Type Acc.
MVTec-AD	83.6	92.1	85.5	81.7	48.7	46.2	72.8
MPDD	72.7	88.5	95.4	50.0	41.5	38.6	68.9
VisA	76.8	85.8	73.4	80.2	45.1	42.7	71.4
DTD	95.6	95.8	94.1	97.1	59.8	61.5	80.7
SDD	88.9	88.7	83.3	94.5	52.3	54.1	77.2
Macro Avg.	83.5	90.2	86.3	80.7	49.5	48.6	74.2

3.4 Ablation Studies

We compare training-stage variants and test-time restrictions on three benchmarks. These interventions answer different questions and are specified below; additional results appear in Appendix B.

Training Stages and Inference Restrictions.

Table 4 compares SFT only with all tools, RL from the base model under the same reward and tool environment, and the full checkpoint with all six tool descriptions removed at inference. The last variant is evaluated without retraining, retaining two responses with empty call and observation lists. On VisA, the full system achieves 76.8% BA, versus 57.6% for SFT only, 55.5% without SFT, and 67.5% with tools disabled. Even without tools, the policy outperforms SFT only; however, prompt changes prevent isolating the contribution of executed observations.

Table 4: **Training-stage comparisons and test-time tool restrictions.** Image-level balanced accuracy (%) is reported on MVTec-AD, VisA, and DTD. The last row disables all six tools at inference; the prior-disabled variant in Table 2 removes only the visual-prior tool.

Method	MVTec	VisA	DTD
(a) Qwen3-VL-8B	67.0	46.8	70.2
(b) IndusAgent	83.6	76.8	95.6
(c) SFT only (w/o RL)	72.3	57.6	74.1
(d) RL from base (w/o SFT)	69.5	55.5	72.8
(e) Tools disabled at inference	78.1	67.5	87.9

Table 5: **Accuracy-gate and reward ablations (%).** Avg., grounding, and format metrics are macro-averages over MVTec-AD, VisA, and DTD. Grounding follows Table 3; format failure measures two-round schema violations over all queries. Base reward is $R_{\text{acc}} + R_{\text{format}}$. Ungated removes only the accuracy gate; other variants remove the named term, including both bonus and call cost for tool.

RL reward	Balanced accuracy ↑				Grounding ↑			Format fail. ↓
	MVTec	VisA	DTD	Avg.	Box mIoU	Loc.@0.5	Type Acc.	
Base reward	76.0	64.9	79.1	73.3	40.8	37.3	62.9	3.4
Ungated additive	80.2	70.1	91.3	80.5	47.9	46.8	72.2	1.6
Full accuracy-gated	83.6	76.8	95.6	85.3	51.2	50.1	75.0	1.2
Gated w/o tool utility	81.9	71.5	92.8	82.1	48.6	48.3	73.9	1.1
Gated w/o localization	79.5	68.8	89.6	79.3	38.7	34.9	72.6	1.4
Gated w/o type	78.5	72.8	90.6	80.6	49.6	47.5	60.9	1.0
Gated w/o format	76.6	65.7	82.5	74.9	45.8	42.9	68.8	12.8

On MVTec-AD, VisA, and DTD (Table 5), the full accuracy-gated objective improves mean BA from 80.5% to 85.3% over the weight-matched ungated control, with gains of 3.3, 3.3, and 2.8 points in Box mIoU, Loc.@0.5, and type accuracy, respectively. Removing localization or type rewards primarily degrades the corresponding grounding metrics, while removing tool utility reduces BA to 82.1%. Removing the format reward increases format failures from 1.2% to 12.8% and lowers BA to 74.9%. Because malformed binary answers count as incorrect, this BA decline may partly reflect reduced output compliance.

4 Related Work

Industrial Anomaly Detection and Category Transfer. Reconstruction methods such as DRAEM and DiffusionAD model defect-free appearance or reconstruct normal structure (Zavrtanik et al., 2021; Zhang et al., 2025). Feature-based approaches such as PaDiM, PatchCore, and flow models provide strong inspection baselines when their required training data are available (Defard et al., 2021; Roth et al., 2022; Yu et al., 2021; Rudolph et al., 2022). Vision-language methods including WinCLIP, AnomalyGPT, and Anomaly-OV extend inspection through text-image alignment and multimodal reasoning (Jeong et al., 2023; Gu et al., 2024; Xu et al., 2025). Our evaluation focuses on transfer from the stated source-category pool without paired normal images at inference; zero-shot inspection itself is not a new task introduced here.

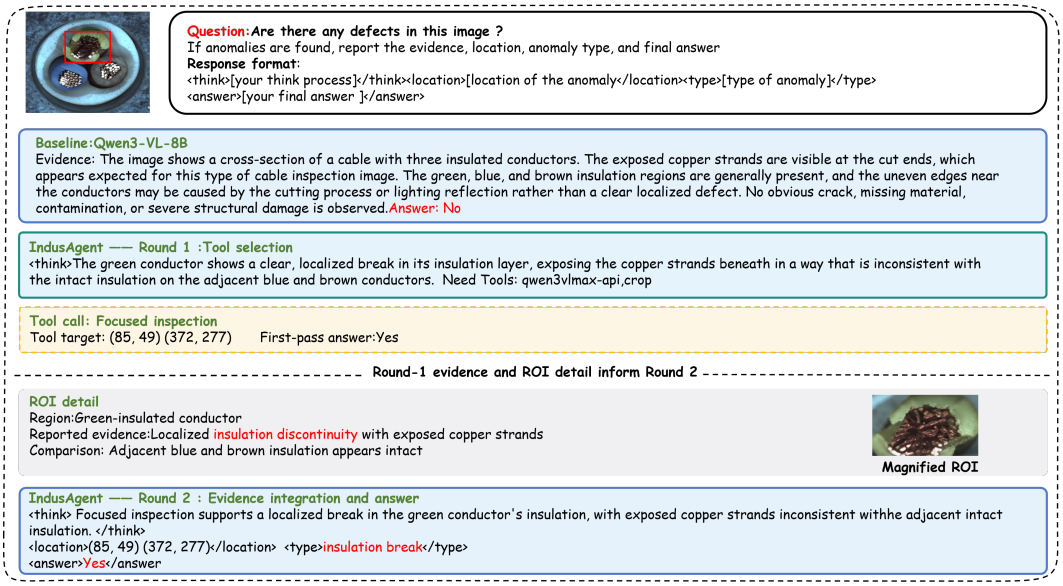


Figure 4: **Qualitative comparison on a cable image.** Direct Qwen3-VL-8B inference predicts no anomaly. IndusAgent flags a suspected insulation break, selects cropping and the external visual-prior tool, and integrates the available evidence into a final diagnosis with a defect box and type.

Post-training for Industrial Reasoning. SFT and reinforcement learning have been used for language and multimodal reasoning (DeepSeek-AI, 2025; Peng et al., 2025). IAD-R1 and AnomalyR1 apply post-training to industrial anomaly reasoning (Li et al., 2026; Chao et al., 2025). IndusAgent builds on these training methods using actual tool-feedback trajectories and a final-correctness gate over auxiliary reward. The classification ablations assess this combination within the reported setup.

Tool-augmented Visual Agents. Toolformer, AgentTuning, and multimodal tool-use frameworks study learning or orchestrating external operations (Schick et al., 2023; Zeng et al., 2024; Liu et al., 2023b; Ma et al., 2025; Zhao et al., 2025). AgentIAD combines tool-integrated trajectories with SFT and reinforcement learning, including a PZ-only mode without normal-exemplar retrieval (Miao et al., 2026). Its anomalous trajectory construction obtains inspection regions from ground-truth masks and supplies annotation information during reasoning generation. Indus-CoT instead uses teacher-selected calls and executed observations, with annotations consulted only to screen completed trajectories. IndusAgent couples this data construction with source–target product-category exclusion and a final-correctness gate over auxiliary rewards. Its external visual prior provides a query-conditioned opinion without retrieving a paired normal image. These distinctions concern data construction, evaluation scope, and optimization.

5 Conclusion

We presented IndusAgent for cross-category industrial anomaly detection without paired normal exemplars. Indus-CoT initializes inspection behavior with teacher-selected actions, executed feedback, and post-generation annotation screening. The accuracy-gated objective then couples auxiliary inspection credit to final diagnostic correctness. The hybrid system achieves 83.5% mean balanced accuracy across five benchmarks, with the weight-matched three-seed comparison supporting a classification benefit from the gated objective. Test-time restrictions quantify sensitivity to tool availability, and full-system metrics report grounding performance. Improving normal-image specificity and resolving the contributions of individual reward terms remain priorities for more reliable inspection.

AI use statement

Generative AI was used to synthesize the teacher inspection trajectories in Indus-CoT and to provide external visual-prior responses during tool-augmented inspection.AI tools also assisted with language

editing, figure preparation, and manuscript assessment and revision.The authors are responsible for verifying the scientific claims, experimental results, citations, and final manuscript.

Reproducibility statement

The appendix specifies the source categories, post-generation screening, two-round reward protocol, pixel-coordinate box convention, metric aggregation, test-time tool restrictions, and training-set hyperparameter selection. Hyperparameters were fixed before target evaluation and were not adjusted using target results. Three-seed results are provided in Table 9. Exact launch configurations, canonical type mappings, and sample manifests are required to reproduce the experiments.

References

Anthropic. Introducing Claude 4. Anthropic, May 22, 2025. URL https://www.anthropic.com/news/claude-4.

Toshimichi Aota, Lloyd Teh Tzer Tong, and Takayuki Okatani. Zero-shot versus many-shot: Unsupervised texture anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5564–5572, 2023. URL https://openaccess.thecvf.com/content/WACV2023/html/Aota_Zero-Shot_Versus_Many-Shot_Unsupervised_Texture_Anomaly_Detection_WACV_2023_paper.html.

S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a. URL https://arxiv.org/abs/2511.21631.

S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b. URL https://arxiv.org/abs/2502.13923.

P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger. MVTec AD–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9592–9600, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Bergmann_MVTec_AD_--_A_Comprehensive_Real-World_Dataset_for_Unsupervised_Anomaly_CVPR_2019_paper.html.

Y. Chao, J. Liu, J. Tang, and G. Wu. AnomalyR1: A GRPO-based end-to-end MLLM for industrial anomaly detection. *arXiv preprint arXiv:2504.11914*, 2025. URL https://arxiv.org/abs/2504.11914.

Xuhai Chen, Yue Han, and Jiangning Zhang. APRIL-GAN: A zero-/few-shot anomaly classification and segmentation method for CVPR 2023 VAND workshop challenge tracks 1&2: 1st place on zero-shot AD and 4th place on few-shot AD. *arXiv preprint arXiv:2305.17382*, 2023. URL https://arxiv.org/abs/2305.17382.

Zhe Chen, Weiyun Wang, Yue Cao, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024. URL https://arxiv.org/abs/2412.05271.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. URL https://arxiv.org/abs/2501.12948.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: A patch distribution modeling framework for anomaly detection and localization. In *Pattern Recognition. ICPR International Workshops and Challenges*, pp. 475–489. Springer, 2021. doi: 10.1007/978-3-030-68799-1_35. URL https://link.springer.com/chapter/10.1007/978-3-030-68799-1_35.

Z. Gu, B. Zhu, G. Zhu, Y. Chen, M. Tang, and J. Wang. AnomalyGPT: Detecting industrial anomalies using large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1932–1940, 2024. doi: 10.1609/aaai.v38i3.27963. URL https://ojs.aaai.org/index.php/AAAI/article/view/27963.

J. Jeong, Y. Zou, T. Kim, D. Zhang, A. Ravichandran, and O. Dabeer. WinCLIP: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19606–19616, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Jeong_WinCLIP_Zero-Few-Shot_Anomaly_Classification_and_Segmentation_CVPR_2023_paper.html.

Stepan Jezek, Martin Jonak, Radim Burget, Pavel Dvorak, and Milos Skotak. Deep learning-based defect detection of metal parts: evaluating current methods in complex conditions. In *2021 13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 66–71, 2021. doi: 10.1109/ICUMT54235.2021.9631567. URL https://doi.org/10.1109/ICUMT54235.2021.9631567.

Bo Li, Yuanhan Zhang, Dong Guo, et al. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. URL https://arxiv.org/abs/2408.03326.

Yanhui Li, Yunkang Cao, Chengliang Liu, Yuan Xiong, Xinghui Dong, and Chao Huang. IAD-R1: Reinforcing consistent reasoning in industrial anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 6583–6591, 2026. doi: 10.1609/aaai.v40i8.37588. URL https://ojs.aaai.org/index.php/AAAI/article/view/37588.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a. URL https://arxiv.org/abs/2310.03744.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. LLaVA Blog, January 30, 2024. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

S. Liu, H. Cheng, H. Liu, H. Zhang, F. Li, T. Ren, X. Zou, J. Yang, H. Su, J. Zhu, L. Zhang, J. Gao, and C. Li. LLaVA-Plus: Learning to use tools for creating multimodal agents. *arXiv preprint arXiv:2311.05437*, 2023b. URL https://arxiv.org/abs/2311.05437.

Z. Ma, J. Zhang, Z. Liu, J. Zhang, J. Tan, M. Shu, J. C. Niebles, S. Heinecke, H. Wang, C. Xiong, R. Krishna, and S. Savarese. TACO: Learning multi-modal models to reason and act with synthetic chains-of-thought-and-action. In *ICLR 2025 Workshop on Reasoning and Planning for Large Language Models*, 2025. URL https://iclr.cc/virtual/2025/32802.

Junwen Miao, Penghui Du, Yingying Fan, Yi Liu, Yu Wang, Runze He, Lida Huang, and Yan Wang. AgentIAD: Agentic industrial anomaly detection via adaptive memory augmentation. *arXiv preprint arXiv:2512.13671*, 2026. URL https://arxiv.org/abs/2512.13671.

OpenAI. GPT-4V(ision) system card. OpenAI, September 25, 2023. URL https://openai.com/index/gpt-4v-system-card/.

OpenAI. Hello GPT-4o. OpenAI Blog, 2024. URL https://openai.com/index/hello-gpt-4o/.

OpenAI. Introducing GPT-4.1 in the API. OpenAI, April 14, 2025. URL https://openai.com/index/gpt-4-1/.

Y. Peng, G. Zhang, M. Zhang, Z. You, J. Liu, Q. Zhu, K. Yang, X. Xu, X. Geng, and X. Yang. LMM-R1: Empowering 3B LMMs with strong reasoning abilities through two-stage rule-based RL. *arXiv preprint arXiv:2503.07536*, 2025. URL https://arxiv.org/abs/2503.07536.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14318–14328, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Roth_Towards_Total_Recall_in_Industrial_Anomaly_Detection_CVPR_2022_paper.html.

M. Rudolph, T. Wehrbein, B. Rosenhahn, and B. Wandt. Fully convolutional cross-scale-flows for image-based defect detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1088–1097, 2022. URL https://openaccess.thecvf.com/content/WACV2022/html/Rudolph_Fully_Convolutional_Cross-Scale-Flows_for_Image-Based_Defect_Detection_WACV_2022_paper.html.

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, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

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

Domen Tabernik, Samo Šela, Jure Skvarč, and Danijel Skočaj. Segmentation-based deep-learning approach for surface-defect detection. *Journal of Intelligent Manufacturing*, 31(3):759–776, 2020. doi: 10.1007/s10845-019-01476-x. URL https://link.springer.com/article/10.1007/s10845-019-01476-x.

Chengjie Wang, Wenbing Zhu, Bin-Bin Gao, Zhenye Gan, Jiangning Zhang, Zhihao Gu, Shuguang Qian, Mingang Chen, and Lizhuang Ma. Real-IAD: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22883–22892, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wang_Real-IAD_A_Real-World_Multi-View_Dataset_for_Benchmarking_Versatile_Industrial_Anomaly_CVPR_2024_paper.html.

Jiacong Xu, Shao-Yuan Lo, Bardia Safaei, Vishal M. Patel, and Isht Dwivedi. Towards zero-shot anomaly detection and reasoning with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20370–20382, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Xu_Towards_Zero-Shot_Anomaly_Detection_and_Reasoning_with_Multimodal_Large_Language_CVPR_2025_paper.html.

Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. FastFlow: Unsupervised anomaly detection and localization via 2D normalizing flows. *arXiv preprint arXiv:2111.07677*, 2021. URL https://arxiv.org/abs/2111.07677.

V. Zavrtanik, M. Kristan, and D. Skočaj. DRAEM: A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8330–8339, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Zavrtanik_DRAEM_-_A_Discriminatively_Trained_Reconstruction_Embedding_for_Surface_Anomaly_ICCV_2021_paper.html.

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. Agent-Tuning: Enabling generalized agent abilities for LLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3053–3077. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.181. URL https://aclanthology.org/2024.findings-acl.181/.

Hui Zhang, Zheng Wang, Dan Zeng, Zuxuan Wu, and Yu-Gang Jiang. DiffusionAD: Norm-guided one-step denoising diffusion for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(8):7140–7152, 2025. doi: 10.1109/TPAMI.2025.3570494. URL https://ieeexplore.ieee.org/document/11005495.

S. Zhao, H. Zhang, S. Lin, M. Li, Q. Wu, K. Zhang, and C. Wei. PyVision: Agentic vision with dynamic tooling. *arXiv preprint arXiv:2507.07998*, 2025. URL https://arxiv.org/abs/2507.07998.

Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. AnomalyCLIP: Object-agnostic prompt learning for zero-shot anomaly detection. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.18961.

Y. Zou, J. Jeong, L. Pemula, D. Zhang, and O. Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pp. 392–408. Springer, 2022. URL https://arxiv.org/abs/2207.14315.

A Detailed Reward Design

Equation 6 is implemented with $\alpha = 0.8$, $\beta = 0.6$, $\gamma = 0.5$, a tool bonus of 0.3, a per-call coefficient of 0.1, and an independent format weight of 0.05.

Ungated control. The weight-matched additive objective used in the reward ablation is

$$R_{\text{add}} = R_{\text{acc}} + \alpha R_{\text{loc}} + \beta R_{\text{type}} + \gamma R_{\text{tool}} + R_{\text{format}}. \tag{8}$$

All component definitions and coefficients are shared with the gated objective, including the format reward and the tool-call cost.

Binary correctness and auxiliary fields. $R_{\text{acc}} = 1$ only when a parsed final answer equals the source label. For ground-truth anomalous images, the box reward is continuous IoU after validating both boxes as finite and non-degenerate in the original-image pixel coordinate system. Type reward is exact agreement after deterministic canonicalization within the 29-label vocabulary in Appendix B.2, with unknown or missing types receiving zero. Both auxiliary terms are zero for ground-truth normal images. Parse failures receive zero binary and spatial/type credit. This runtime reward is distinct from the IoU≥ 0.5 criterion used to screen teacher trajectories.

Tool counts and success condition. N_{call} counts parsed calls admitted to the tool-execution round, including repeated calls, up to a maximum of four. Excess requests are ignored and do not enter this count. An admitted call incurs cost even if execution fails. N_{ok} counts successful observation records from these calls whose tool is one of the six tools in $\mathcal{T}$. The bonus in Eq. 7 requires $N_{\text{ok}} > 0$ and a positive margin change. A successful status establishes execution validity, not correctness or diagnostic usefulness of the returned evidence.

Decision-margin change. For the final predicted label $\hat{y} \in \{\texttt{Yes}, \texttt{No}\}$ and opposite label $\bar{y}$, define

$$m(\hat{y} \mid h) = \log p(\hat{y} \mid h) - \log p(\bar{y} \mid h), \qquad \Delta m = m(\hat{y} \mid h_{\text{post}}) - m(\hat{y} \mid h_{\text{pre}}). \tag{9}$$

The trainer records raw Yes/No logit differences at the first response's <pre_answer> position and the second response's <answer> position, and orients both toward the final label. The raw-logit difference equals the log-probability difference at that position. The scoring context includes the generated history available at each decision, and the second context includes the tool feedback and additional reasoning. A margin increase can therefore reflect extra deliberation or changed context as well as returned observations. Missing tagged decision positions yield a zero margin; a missing final binary answer yields zero oriented margins. The reward helper treats non-finite or unparseable margin inputs as zero gain. No verbalized confidence is used. Because the contexts and generated histories differ between rounds, Δm is a confidence proxy rather than an isolated estimate of the feedback's causal effect.

Format reward.

$$R_{\text{format}} = 0.05\,\mathbb{I}[\text{trajectory satisfies the two-round schema}]. \tag{10}$$

A compliant RL trajectory has exactly two non-empty `<think>` blocks, a valid preliminary Yes/No field, and a final answer. Tool-using trajectories have parsed calls and a JSON observation list with a matching record count. No-tool trajectories have empty call and observation lists and still contain both response segments. For an anomalous final answer, a valid defect box in original-image pixel coordinates and a non-empty type in the second response precede the final answer; for a normal final answer, these final auxiliary fields are absent. Invalid structures receive zero format reward, not a negative format penalty. Inference uses the same two-round schema specified in Appendix E.

Reward ordering. Under the four-call execution limit, the smallest possible tool contribution is $\gamma R_{\text{tool}} = -0.2$. Since localization, type, and format rewards are nonnegative, a correctly diagnosed trajectory receives at least 0.8, whereas an incorrectly diagnosed trajectory receives at most the independent format reward of 0.05. Thus, before groupwise normalization, every correctly diagnosed trajectory has a higher return than every incorrectly diagnosed trajectory. This ordering prioritizes diagnostic correctness while allowing auxiliary terms to distinguish trajectories with correct diagnoses.

Scope of the gate. For trajectories that satisfy the output schema, localization and type rewards are already zero on incorrect diagnoses: a false negative has no final box or type, and a truly normal image is ineligible for either reward. On this subset, $R_{\text{acc}}R_{\text{loc}} = R_{\text{loc}}$ and $R_{\text{acc}}R_{\text{type}} = R_{\text{type}}$. An additional effect on auxiliary-field credit is possible when an inconsistent trajectory predicts No on an anomalous image but still provides annotation-matching auxiliary fields. If the auxiliary parser accepts those fields, the additive objective can assign them credit, whereas the gated objective suppresses it.

The tool term also changes on incorrect trajectories: both its bonus and its call cost are removed, leaving at most the format reward. The gated–ungated comparison therefore evaluates the complete coupling of diagnostic correctness, auxiliary credit, and tool costs. The classification and grounding results in Table 5, together with the three-seed classification results in Table 9, support the benefit of this combined objective. They do not isolate the contribution of each gated component or measure the frequency of reward exploitation during training.

Class-dependent reward range. Correct anomalous predictions can receive localization and type rewards that are absent on normal images. This difference in reward range is a design asymmetry, but does not by itself establish a preference for anomalous predictions: GRPO normalizes rewards among rollouts of the same image, whose ground-truth label is fixed. In particular, a false-positive decision on a normal image receives no gated auxiliary credit. Class-stratified reward and error diagnostics are needed to test whether the training dynamics contribute to the observed normal-image errors. An alternative with call cost outside the correctness gate would change the training objective and requires a separately trained comparison; it is not evaluated here.

B Additional Experimental Results

B.1 More Experimental Details

Category-Disjoint Training Protocol. The category audit compares Real-IAD object names with the five evaluation benchmarks, including semantic matches such as `pcb` with VisA's PCB categories and `transistor1` with MVTec-AD's transistor category. The four overlapping categories, `pcb`, `toothbrush`, `transistor1`, and `zipper`, were excluded before constructing the final training corpus. All 3,000 training trajectories come from the following 26 categories:

```
audiojack, bottle_cap, button_battery, end_cap, eraser,
fire_hood, mint, mounts, phone_battery, plastic_nut,
plastic_plug, porcelain_doll, regulator, rolled_strip_base,
sim_card_set, switch, tape, terminalblock, toy, toy_brick,
u_block, usb, usb_adaptor, vcpill, wooden_beads, woodstick.
```

Table 6: **Reported corpus-size comparison.** Balanced accuracy (%) for the 3,000-trajectory corpus and the 2,747-trajectory accepted subset. The completed corpus contains accepted replacements, not rejected or annotation-repaired trajectories.

SFT corpus	MVTec	MPDD	VisA	DTD	SDD	Avg.
All 3,000 trajectories	83.6	72.7	76.8	95.6	88.9	83.5
2,747 accepted trajectories	83.9	72.4	77.1	95.4	89.2	83.6

This is a product-category exclusion protocol; it does not demonstrate that all visual attributes or canonical defect types are unseen during training.

Trajectory Generation and Curation. The teacher generates an inspection trajectory without access to ground-truth labels, locations, or defect types. Model-selected tool calls are executed, and actual feedback is supplied before the final diagnosis. Annotations are consulted only after generation to screen the completed trajectory. We retain candidates that satisfy the output format and agree with the binary annotation; anomalous candidates must additionally satisfy box IoU≥ 0.5 and canonical defect-type agreement. Tool observations must correspond to executed calls. Normal candidates must not fabricate a final defect box or type.

Initially, 2,747 trajectories passed screening. The 253 unsuccessful samples were discarded. We generated candidates on other images from the same allowed source pool and selected 253 additional trajectories that passed the same criteria, completing a corpus of 3,000 accepted trajectories with 1,500 normal and 1,500 anomalous images. Failed trajectories were not repaired using ground-truth-conditioned text or included as placeholders.

Evaluation Metrics and Inference Details. For each dataset, balanced accuracy is computed from the pooled image-level confusion matrix, and F1 treats anomalous images as positive. Categories are not averaged separately within a dataset. The five-dataset average assigns equal weight to each dataset. Full-model localization and type metrics use all ground-truth anomalous images, including those with incorrect binary predictions; missing or malformed required fields receive zero credit. IoU≥ 0.5 is the localization hit threshold. Training and test inference always generate two responses. Without tools, the intervening call and observation lists are empty. Evaluation uses only the second response's final diagnosis and auxiliary fields.

B.2 Canonical Defect Vocabulary

We use a fixed vocabulary of 29 defect labels for post-generation trajectory screening, reinforcement-learning type rewards, and canonical type accuracy. Table 7 lists the complete vocabulary. Predicted types are evaluated by exact agreement after deterministic normalization and dictionary matching. Missing, unrecognized, or non-matching types receive zero type credit. Normal predictions omit the final type field. Cross-category transfer is therefore evaluated over unseen product categories within a fixed defect-label space.

B.3 Baseline Prompting and Answer Parsing

The common binary task and metric definitions apply to the rerun models in Table 2, including direct Qwen3-VL-8B and Qwen3-VL-Max. IndusAgent additionally uses the tool-routing protocol. It does not imply that literature-only rows used the same prompt or parser. For each query image, the model is asked to determine whether the image contains an anomaly and to provide a final answer in a normalized binary form, where `Yes` denotes anomalous and `No` denotes normal. No paired normal reference image, category-specific exemplar, or dataset-specific prompt is provided during these reruns.

For IndusAgent, the final prediction is always extracted from the second response's structured `<answer>...</answer>` field, including when no tool is used. For IAD-R1, we apply a rule-based parser fixed before examining target labels. The parser first searches the final-answer span for explicit decisions such as "yes", "anomalous", "defective", "abnormal", "no", "normal", and "defect-free". When both positive and negative expressions occur, only the final stated conclusion is

Table 7: **Fixed canonical defect vocabulary.** The 29 labels used for defect-type supervision and evaluation. Each cell contains one complete label.

Canonical label	Canonical label
dark stain contamination	surface abrasion
foreign object	frayed area
scratch	frayed bristles
crack	frayed wire
surface pit	frayed fiber
local dent	frayed traces
deformation	chipped damage
missing structural part	broken tip
missing material	broken area
missing tooth	misaligned pins
missing pin	short circuit
missing solder pad	solder protrusion
missing bristle cluster	surface defect
missing corner	insulation break
missing trace	

used. Responses that remain ambiguous or malformed are counted as incorrect rather than manually relabeled.

B.4 More Hyperparameter Configurations

Table 8: **Reward-weight selection by separate post-training evaluation on the source training set.** These are training-set metrics, not held-out development results; target benchmarks were not used for selection.

α (Loc)	β (Type)	γ (Tool)	Train-set BA	Train-set mIoU	Train-set Type Acc.	Calls/query
0.0	0.0	0.0	78.9	37.8	62.4	0.66
0.8	0.0	0.0	80.7	49.1	63.0	0.70
0.8	0.6	0.0	81.5	50.3	70.8	0.73
1.0	1.0	1.0	81.8	51.0	72.5	1.28
0.8	0.6	0.5	82.6	50.8	72.1	0.91

Reward weights were selected by a separate evaluation pass on the same 3,000-image source training set after training each configuration. Balanced accuracy was the primary selection metric; localization, type accuracy, and call frequency were secondary diagnostics. There was no independent development set. The selected weights were frozen before evaluation on the five target benchmarks and were not adjusted using target results. Reusing training images for selection can give optimistic source-domain estimates.

All experiments were conducted on a single compute node equipped with four NVIDIA A100 GPUs, each with 80GB memory. We used Qwen3-VL-8B-Instruct as the backbone model. The supervised fine-tuning stage was trained for one epoch on the Indus-CoT corpus and took approximately 21.6 minutes. The subsequent reinforcement learning stage was implemented with our accuracy-gated, fixed-two-round GRPO training pipeline and optimized for one epoch using four generations per prompt. This stage took approximately 43.5 hours. During RL training, we used a per-device batch size of 1, gradient accumulation steps of 2, a maximum prompt length of 4096, a maximum completion length of 512, bfloat16 precision, gradient checkpointing, and DeepSpeed ZeRO-3.

Table 9: **Three-seed results for the central comparisons.** Values are balanced accuracy averaged equally across the five target benchmarks. The full and ungated rows correspond to their respective trained policies. For each seed, prior disabling uses the corresponding full checkpoint with the prior tool description removed at inference; it is not a separately trained policy.

Variant	Seed 1	Seed 2	Seed 3	Mean $\pm$ std.
Ungated additive	79.9	80.7	80.8	80.5 ± 0.5
Prior disabled at inference	79.4	79.8	79.9	79.7 ± 0.3
Full accuracy-gated IndusAgent	83.1	83.5	83.9	83.5 ± 0.4

Table 10: **Image-level F1 (%).** For rerun IAD-R1 and IndusAgent results, anomalous images are positive and counts are pooled over all images within each dataset.

Model	DTD	MPDD	MVTec	VisA
Qwen3-VL-8B	84.8	66.2	77.8	58.3
Claude-4-Sonnet	88.5	69.8	81.2	64.5
IAD-R1 (Qwen2.5-VL-7B) (Li et al., 2026)	91.1	75.2	85.7	70.8
IndusAgent (Ours)	96.1	77.3	88.9	77.6

B.5 More Experimental results

Table 10 reports F1 scores on four shared datasets. IndusAgent has the highest displayed F1 on each dataset under the reported protocols. The separate test-time disabling experiments characterize the trained policy's dependence on tool availability.

B.6 Tool Usage Analysis

Table 11: **Tool-call composition and frequency during target inference.** Each tool percentage is its share of all counted tool calls on the corresponding dataset, with repeated calls counted separately. Shares sum to 100% up to rounding. Average calls is the total number of counted calls divided by all query images, including those requiring no tool. Execution success measures valid-status returns rather than diagnostic usefulness.

Metric	MVTec	VisA	DTD
Crop (%)	47.2	42.9	29.8
Prior (%)	14.1	19.2	6.5
Enhance (%)	16.1	14.8	36.3
Measure (%)	7.4	5.9	2.2
Compare (%)	9.1	11.7	4.2
Texture (%)	6.1	5.5	20.9
Avg. calls	1.32	1.30	0.94
Success rate (%)	99.1	98.7	99.4

Table 11 reports tool-call composition and calls per query on three benchmarks. For dataset d, let $C_{d,t}$ be the number of counted calls to tool t and N_d the total number of query images. The tool share is $100C_{d,t}/\sum_{u\in\mathcal{T}} C_{d,u}$, while average calls is $\sum_{u\in\mathcal{T}} C_{d,u}/N_d$. Repeated calls contribute separately; images with no calls remain in N_d. Crop accounts for the largest call share on MVTec and VisA, and enhancement on DTD. Execution success is the fraction of counted calls returning a valid success status. Prior calls query the external Qwen3-VL-Max API.

Table 12 reports the measured end-to-end cost of the resulting inference paths. IndusAgent has a mean latency of 1.46 seconds per image, lower than the 2.31 seconds for direct Qwen3-VL-Max but higher than the 0.84 seconds for direct Qwen3-VL-8B. The 0.62-second increase over direct 8B inference is an end-to-end system difference encompassing additional policy generation and selected

Table 12: **Measured end-to-end inference efficiency.** For IndusAgent, latency includes generation of both policy responses, plus tool execution and API waiting when invoked, under the measured hardware and network conditions. Latency is computed by averaging dataset-level mean per-image latencies with equal weights. API costs are reported in Chinese yuan (CNY) per query.

Method	Tool calls/query	Max-API calls/query	Latency (s)	Output tokens	API cost/query (CNY)
Qwen3-VL-8B direct	0	0	0.84	108	0
Qwen3-VL-Max direct	1	1	2.31	142	0.0038
Prior disabled at inference	0.79	0	1.13	184	0
IndusAgent	0.96	0.17	1.46	196	0.0006

tool execution. The full system makes 0.17 Max-API calls per query and incurs a reported API cost of CNY 0.0006 per query under the measured service conditions. Prior disabling reduces latency to 1.13 seconds while retaining a policy trained with prior access. The 0.33-second difference includes changes in policy generation and tool routing and cannot be attributed solely to API waiting. The available timing measurements do not separate policy generation, local tool execution, and remote waiting. Deployment therefore requires assessing the end-to-end inspection cost against the required cycle time and throughput under the intended hardware and service conditions.

B.7 Image-Level AUROC Analysis

Table 13: **Image-level AUROC (%).** Published ZSAD baselines and IndusAgent on five IAD benchmarks. DTD denotes DTD-Synthetic.

Method	MVTec	MPDD	VisA	DTD	SDD	Avg.
CLIP (Radford et al., 2021)	74.1	54.3	66.4	71.6	65.7	66.4
WinCLIP (Jeong et al., 2023)	91.8	63.6	78.1	93.2	84.3	82.2
APRIL-GAN (Chen et al., 2023)	86.1	73.0	78.0	86.4	79.8	80.7
AnomalyCLIP (Zhou et al., 2024)	91.5	77.0	82.1	93.5	84.7	85.8
IndusAgent	**92.1**	**88.5**	**85.8**	**95.8**	**88.7**	**90.2**

Table 13 reports image-level AUROC using the normalized log-probability margin between the "Yes" and "No" decision tokens at the second response's final-answer position as the anomaly score. The five IndusAgent values yield a macro-average of 90.2%. The baseline values are contextual because they were quoted rather than rerun under the same implementation. Different scalar scoring functions can be compared by AUROC under a matched evaluation protocol; differences in score construction alone do not invalidate that metric.

B.8 More Ablation Studies

Table 14: **Individual tools disabled at inference.** Every row uses the same full trained checkpoint, removing only the indicated tool description from the inference prompt. These are availability interventions, without retraining.

Method	MVTec	VisA	DTD
IndusAgent	83.6	76.8	95.6
Crop disabled at inference	79.4	68.6	91.2
Prior disabled at inference	81.1	69.3	92.1
Enhance disabled at inference	80.5	70.8	88.8
Measure disabled at inference	81.5	70.2	91.5
Compare disabled	80.9	74.1	91.8
Texture disabled	81.4	72.2	91.7

Individual Tool Ablations. Table 14 disables one tool at a time through the inference prompt at the same full checkpoint. Crop removal gives the largest VisA decrease, while enhancement removal gives the largest DTD decrease among the reported variants. Removing measurement or the Qwen3-VL-Max prior also reduces balanced accuracy on all three datasets. These observations are specific to the evaluated policy and benchmarks.

Table 15: **GRPO group-size selection.** Source metrics are from separate post-training evaluation on the 3,000-image training set; step time is measured during training.

Group size	Train-set BA	Train-set mIoU	Step time
2	80.9	48.6	8.7
4	82.6	50.8	17.4
6	82.4	51.2	25.8
8	81.9	51.0	34.1

Impact of GRPO Group Size. Table 15 compares group sizes using training-set evaluation and measured training cost. Group size 4 was selected and fixed before target-benchmark evaluation, without adjustment using target results. The source metrics are not independent generalization estimates.

C Discussion on Novelty and Problem Setting

Problem scope. The study evaluates transfer from 26 Real-IAD source product categories to the stated target benchmarks without paired normal exemplars. This study is defined by category-disjoint and reference-free evaluation; it does not claim to introduce zero-shot IAD. The fixed canonical type vocabulary does not test unconstrained naming or disjoint defect-type generalization. Unknown pretraining overlap is outside the category audit.

Training-data contribution. Indus-CoT links teacher-selected actions to actual executed tool feedback and a final diagnosis. The teacher is not given annotations during generation; annotations screen completed candidates for binary agreement, box IoU≥ 0.5, and canonical type agreement. The 3,000 accepted trajectories initialize the policy for online optimization. Selection favors teacher-solvable examples and can limit coverage of difficult defects.

Reward contribution and evidence. The correctness gate conditions auxiliary rewards and call costs on the final binary diagnosis while retaining a separate format reward. The weight-matched additive comparison evaluates this complete coupling. Three-seed results support a classification benefit within the reported training setup. As detailed in Appendix A, the gate changes auxiliary-field credit on inconsistent outputs and tool-related returns on incorrect diagnoses;

Relation to prior agentic inspection. AgentIAD already combines tool-integrated trajectories, SFT, and reinforcement learning, including inspection without exemplar retrieval (Miao et al., 2026). Our study combines teacher-selected, executed-feedback trajectories with product-category exclusion and a correctness-gated objective. The visual-prior tool supplies an external opinion about the query and may identify suspicious deviations, although it is instructed not to return the final binary decision. Test-time disabling measures the trained policy's sensitivity to tool availability.

D Tool Library Specifications

The tool library provides observations selected by the policy. Each tool-execution round admits at most four parsed calls across the six tool types, including repeated calls; requests beyond this limit are ignored. Successful execution does not guarantee that the returned evidence is correct or interpreted correctly.

D.1 DYNAMIC REGION CROPPING TOOL (T_{CROP})

The teacher or policy selects image coordinates, and the environment executes the requested crop and returns the local image. The same model-selected-coordinate mechanism is used in teacher generation, online RL, and target inference. Ground-truth defect coordinates are not supplied to the crop tool. A crop window specifies what to inspect; it is distinct from the final anomaly box evaluated against the annotation. Cropping exposes a local view without creating new optical information.

D.2 NORMALCY PRIOR EXPLANATION TOOL (T_{PRIOR})

The external Qwen3-VL-Max API receives the query image, the inspection question, and a fixed instruction template with a policy-specified focus supplied through the tool's `query` argument. The instruction requests expected defect-free geometry, texture, repeated structure, and plausible manufacturing variation, while prohibiting a final anomaly decision. Thus, the returned text is a query-conditioned visual opinion. Ground-truth annotation metadata and paired normal reference images are not supplied to this API call.

Normalcy-Prior Instruction Template

```
# TOOL -- Industrial Normalcy Prior

You are an expert in the normal appearance of industrial
products. Use the query image and general industrial knowledge
to describe expected defect-free structure and visible
deviations that require further inspection.

Important:
- Address the inspection question and the requested focus.
- Consider geometry, texture, repeated structures, and
  plausible manufacturing variation.
- Distinguish visible observations from expectations.
- Do not assume a paired normal reference image.
- Do not make the final Yes/No anomaly decision.
- No ground-truth labels, boxes, masks, or defect types
  are provided.

## Input
Query image: attached.

Inspection question:
{inspection_question}

Policy-requested focus:
{query}

Reply format:
Describe the expected normal structure, relevant visible
deviations, and uncertainties that require verification.
Return a concise textual explanation without a final
binary anomaly decision.
```

D.3 LOW-LEVEL VISUAL ENHANCER (T_{ENHANCE})

The tool returns deterministic transformations such as Canny edges and contrast-limited adaptive histogram equalization (CLAHE). These transformations may make contrast or boundaries easier to inspect, but can also amplify artifacts. The agent must interpret them alongside the query image.

D.4 GEOMETRIC VERIFIER (T_{MEASURE})

The tool receives model-selected image coordinates and returns pixel distances and angular relations. It does not use camera calibration or report physical dimensions.

D.5 WITHIN-IMAGE COMPARISON TOOL (T_{COMPARE})

The tool compares a target region with other regions in the same query image. It takes a normalized `target_box` and exactly one of `search_box` or `reference_boxes`. Automatic search uses

normalized correlation, excludes candidates with excessive target overlap, and applies spatial non-maximum suppression. The tool supports edge and grayscale comparison, with up to five references and three by default. Alignment is limited to three pixels in each direction at the working resolution, whose longest side is at most 384 pixels. It returns the selected regions, matching and difference measurements, and visual comparison artifacts. Comparison regions are not assumed to be normal; the measurements describe differences within the query image.

D.6 Texture Analysis Tool (T_{TEXTURE})

The tool analyzes periodic structure within a normalized region of interest. It removes a fitted illumination plane, applies a Hann window, and computes non-DC Fourier power globally and in local windows. Periodicity quality is the strongest eligible spectral-bin power divided by the total eligible power. Local deviation combines differences in period, orientation, and periodicity quality with weights 0.45, 0.45, and 0.10, respectively. Unreliable local windows are assigned a deviation of 1, and overlapping window scores are averaged to form the heatmap. Periods are reported in original-image pixels and orientations in degrees modulo 180. The heatmap describes variation within the selected region and is not a calibrated defect probability. Regions with insufficient stable periodicity return an error.

E Prompt Templates and Output Protocol

Shared two-round protocol. Teacher generation, online RL, and target inference use the same two-round structure. The first response provides a preliminary diagnosis and optional tool calls. The environment executes the admitted calls and returns actual observations. The second response provides the final diagnosis, including when the call and observation lists are empty. The full setting provides six tools and admits at most four calls per trajectory, including repeated calls. During training, generated responses are supervised or optimized, while prompts and environment observations are masked from the loss.

Round 1 – Visual Inspection and Tool Selection

```
You are an expert industrial quality control inspector.
Inspect the query image for localized anomalies. Form a
preliminary diagnosis and select tools when additional
evidence would help resolve uncertainty.

Every reply MUST contain <think>...</think>,
<pre_answer>Yes or No</pre_answer>,
<location>...</location>, and
<call_tool>...</call_tool>, in that order.

Important:
- Begin with visible evidence in the query image.
- Consider whether suspicious regions reflect defects
  or plausible benign appearance variations.
- Yes means anomalous; No means normal.
- Use only the supplied tools and follow their exact
  argument schemas and coordinate conventions.
- Select tool arguments from the query image.
- Request at most four calls, including repeated calls.
- If no tool is needed or available, return an empty
  call list: <call_tool>[]</call_tool>.
- No paired normal reference image is provided.
  Regions from the same image are not guaranteed
  to be defect-free.
- Do not invent observations. The environment executes
  the calls and supplies the results.

## Input
Query image: attached.

Inspection question:
{inspection_question}

Available tools and exact call schemas:
{tool_definitions}

Reply format:
```

```
Replace {tool_calls} with a call list following the
supplied schemas, or [] when no tool is requested.

<think>
Describe the relevant visual evidence, preliminary
assessment, and uncertainty motivating any tool calls.
</think>
<pre_answer>Yes or No</pre_answer>
<call_tool>{tool_calls}</call_tool>

Output only these fields.
```

Round 2 – Evidence Integration and Final Diagnosis

```
You are an expert industrial quality control inspector.
Use the query image and any returned tool evidence to
make the final anomaly decision. Reassess the preliminary
diagnosis using the available observations.

Every reply MUST contain a non-empty <think>...</think>
block and a final <answer>...</answer> field.

Important:
- Yes means anomalous; No means normal.
- Ground the decision in the query image and actual
  returned observations. Revise the preliminary
  diagnosis when the evidence warrants it.
- Verify external visual-prior opinions against the
  image. Within-image comparison regions may contain
  defects, and enhanced views may amplify artifacts.
- Failed or empty tool results provide no additional
  evidence. If the observation list is [], complete
  the diagnosis using the query image.
- For Yes, provide one defect box [x1,y1,x2,y2] in
  original-image pixel coordinates: top-left followed
  by bottom-right. Convert crop-relative or normalized
  coordinates back to the original image.
- Choose one defect label from the canonical vocabulary.
- For No, omit <location> and <type> entirely.
- Do not request more tools or generate observations.

## Input
Query image: attached in the inspection context.

Inspection question:
{inspection_question}

Round-1 response:
{round1_response}

Tool evidence, inserted by the environment:
<observation>{tool_observations}</observation>
Returned tool images are attached when available.
For no-tool trajectories, {tool_observations} is [].

Canonical defect vocabulary:
{canonical_defect_vocabulary}

Reply format:
Use exactly one of the following formats.

If anomalous:
<think>
Explain the visible defect evidence and how the available
feedback supports or changes the preliminary assessment.
</think>
<location>[x1,y1,x2,y2]</location>
<type>canonical defect label</type>
<answer>Yes</answer>

If normal:
<think>
Explain the evidence supporting a normal decision,
addressing suspicious regions and relevant feedback.
</think>
<answer>No</answer>
```

```
Output only the chosen fields.
```

Teacher screening and reward computation. Ground-truth labels, locations, and defect types are withheld during teacher generation and policy inference. Completed teacher trajectories are screened for format and binary agreement; anomalous trajectories must additionally achieve box IoU ≥ 0.5 and canonical-type agreement. Accepted trajectories provide SFT targets. During online RL, annotations are used for reward computation outside the policy interaction. The fixed canonical vocabulary is listed in Table 7.

Tool availability interventions. Individual-tool and all-tool disabling remove the corresponding definitions from the inference prompt at the same full trained checkpoint. With all tools disabled, the call and observation lists are empty, while both response rounds are retained. Other evaluation settings remain fixed. The no-RL row uses the SFT checkpoint with the full tool library. The no-SFT row applies the same RL reward and executed tool environment directly to the base model.

Direct-model evaluation. Direct baselines inspect the query without tools or paired normal references. The template below preserves the supplied direct-output convention, including `none` for the location and type of normal predictions. Exact model-specific wrappers and decoding configurations remain necessary to reproduce the reported reruns. This template does not specify the prompts used for literature-transcribed results.

Direct Inspection – Anomaly Diagnosis

```
You are an expert industrial quality control inspector.
Inspect the query image for localized anomalies and
determine whether it is anomalous or normal.

Every reply MUST contain <think>, <location>, <type>,
and <answer> fields, in that order.

Important:
- Base the diagnosis on visible evidence in the image.
- No tools or paired normal references are available.
- Yes means anomalous; No means normal.
- For Yes, provide one defect box [x1,y1,x2,y2] in
  original-image pixels, with top-left coordinates
  followed by bottom-right coordinates.
- Choose the defect label from the canonical vocabulary.
- For No, write none inside both <location> and <type>.

## Input
Query image: attached.

Inspection question:
{inspection_question}

Canonical defect vocabulary:
{canonical_defect_vocabulary}

Reply format:
<think>Brief visual evidence supporting the diagnosis.</think>
<location>[x1,y1,x2,y2] or none</location>
<type>canonical defect label or none</type>
<answer>Yes or No</answer>

Output only these fields.
```

F LIMITATIONS

The evaluation covers public benchmarks; production-line deployment remains unevaluated. Bounding boxes provide coarser localization than masks, and a fixed canonical vocabulary does not evaluate unconstrained defect naming. The category audit concerns industrial post-training; unknown backbone and API pretraining overlap cannot be excluded. Teacher-success filtering can underrepresent difficult images, and the corpus may inherit teacher and prompt biases.

Reward weights and group size were selected on the source training set without an independent development set and were frozen before target evaluation; the source selection metrics may therefore be optimistic. The correctness gate removes call costs on incorrect trajectories and does not guarantee minimal tool use. Reward ablations in Table 5 report classification, grounding, and format-compliance metrics on three benchmarks, but do not separate the effects of gating localization and type rewards from those of gating tool bonuses and call costs. Tool-usage and individual-tool availability analyses cover all six tools on three of the five benchmarks. Prompt-based disabling jointly changes available feedback and input context. The MPDD specificity of 50.0% remains an unresolved normal-image error limitation.

The full system depends on an external Qwen3-VL-Max visual-prior API. The reported inference-time prior restriction retains prior exposure during training and does not establish the performance of a policy trained entirely without external visual opinions. Latency and cost describe the measured service and network conditions. Misleading tool outputs can affect diagnosis. Literature-transcribed baseline results provide context, and there is no matched AgentIAD rerun.