AGENTICAD: LEARNING TO ROUTE, OBSERVE, AND DETECT FOR SAMPLE-ADAPTIVE INDUSTRIAL ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-class industrial anomaly detection faces strong sample-wise variation in detector preference, defect scale, and appearance. We introduce AGENTICAD, a sample-adaptive *route–observe–detect* framework for supervised control of frozen anomaly detectors. Supervision is used only to train the lightweight routing and observation controllers (using pixel masks for pixel-level control); the anomaly detectors remain frozen. Stage I routes each image to one of four complementary experts using task-specific utilities derived from all expert outcomes. Stage II performs a bounded observe–detect loop conditioned on the initial score, heatmap, and observation history. At each step, the controller either stops or applies a heatmap-guided crop or photometric transformation; a generic local detector analyzes the resulting view and aligns its evidence to the original image. Fixed aggregation rules convert detector outputs into final scores. On IND-COM, comprising five industrial datasets and 78 object categories, Stage I improves image-level AUROC/AP from $79.82/91.36$ for category-best routing to $84.75/93.51$, and Stage II further reaches $91.34/96.34$. For pixel localization, Stage I reaches $92.08/51.22$ AUROC/AP, and Stage II reaches $92.45/51.53$, compared with $91.24/49.94$ for category-best routing. Overall, AGENTICAD offers a practical way to adapt detector selection and inspection effort to the diversity of real-world industrial products and anomalies.

1 INTRODUCTION

Industrial anomaly detection (AD) is essential to modern manufacturing for preventing defective products, reducing unplanned downtime, and lowering manual inspection costs. Because anomalies are scarce, defect taxonomies evolve, and exhaustive annotation is expensive, unsupervised anomaly detection (UAD) has become a practical solution for scalable inspection. Early UAD methods mainly address the single-class setting through feature statistics and memory banks Defard et al. (2021); Roth et al. (2022); Lee et al. (2022), reconstruction and knowledge distillation Deng & Li (2022); Salehi et al. (2021); Madan et al. (2023), synthetic anomaly generation Li et al. (2021); Zavrtanik et al. (2021); Liu et al. (2023), or diffusion-based reconstruction He et al. (2024b); Zhang et al. (2025); more recent multi-class methods instead use one unified model across many industrial categories You et al. (2022); Lu et al. (2023); Guo et al. (2023); Yin et al. (2023); He et al. (2024a). Although this progression improves scalability, practical multi-category inspection still exposes two limitations. First, detectors exhibit pronounced *anomaly-type preferences*: different modeling paradigms perform well on different defects, so one global detector wastes their complementarity, while one detector per object category remains too coarse for within-category variation. Second, anomalous regions vary by orders of magnitude in area and occur under broad imaging distributions spanning scale, location, brightness, contrast, texture, and background. A global view may reveal a large structural defect, whereas a small or low-contrast defect may require a localized crop or photometric adjustment. The appropriate observation is therefore sample dependent and may only become apparent from the score and heatmap produced by preceding detection steps. Recent agent-based AD methods automate pipeline construction Ji et al. (2025), support open-vocabulary tool-augmented reasoning Tan et al. (2026), augment inspection with adaptive memory Miao et al. (2025), or adapt anomaly-score thresholds Yang et al. (2025); However, they do not jointly learn which detector to use and how to adapt subsequent observations for each sample, as shown in Figure 1. In contrast, our method

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.

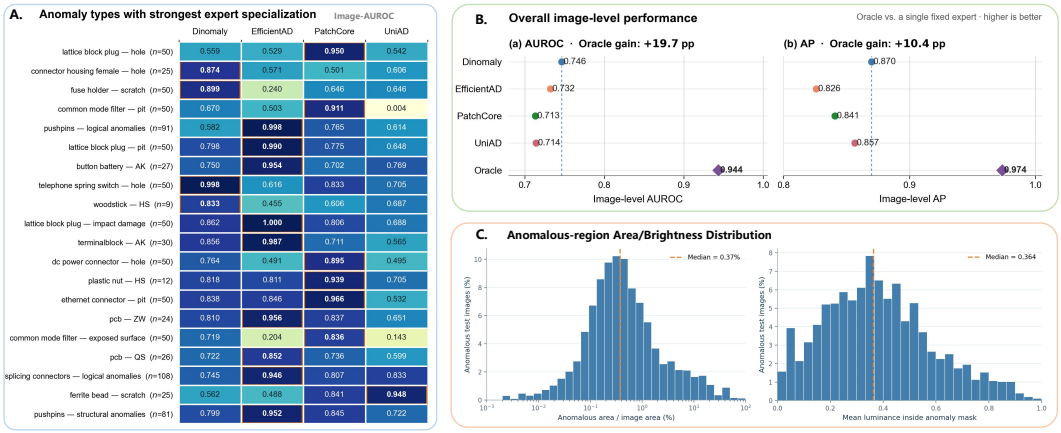


Figure 1: Motivation for industrial anomaly detection using sample-adaptive decisions. (A) The strongest expert changes across anomaly types, reflecting complementary detector biases rather than a universally optimal model. (B) The large gap between every fixed expert and the per-sample oracle shows the potential of adaptive expert selection. (C) Anomalous regions vary by orders of magnitude in area and exhibit a broad brightness distribution, making a fixed view or image transformation unreliable. AGENTICAD directly matches these challenges: Stage I **routes** each image to a suitable expert, while Stage II **observes and detects** conditioned on the current score and heatmap.

explicitly couples sample-level expert routing with a compact, auditable multi-step observe–detect policy, both learned from dense counterfactual supervision over frozen detectors as shown in Figure 2.

To this end, we propose AGENTICAD, a learnable sample-adaptive *route–observe–detect* pipeline for industrial anomaly detection. Rather than modifying the underlying detectors, AGENTICAD learns how to control complementary frozen models at the sample level. Stage I routes each input image to one of four specialized anomaly detectors using a lightweight controller over a frozen visual representation. Since all four expert outcomes are available offline, the router is trained with dense full-information supervision that preserves the relative utility of every expert instead of reducing routing to a single hard oracle label. The selected expert is executed once at its native resolution and produces the initial anomaly score and heatmap. Stage II converts further inspection into a bounded observe–detect loop. Conditioned on the original visual feature, the routed expert identity, the initial result, and the accumulated observation history, a second controller decides whether to stop or acquire a heatmap-guided crop or brightness/contrast-adjusted view. Each non-stop action is analyzed by a frozen generic local detector, and its local heatmap is mapped back to the original image coordinates before being combined with previous evidence. The updated result then determines the next action. Full-information supervision evaluates all available action outcomes during training, while inference executes only the selected trajectory. Consequently, both stages focus on sample-adaptive control, leaving the experts, the generic detector, and the detection-score fusion rule frozen.

We evaluate AGENTICAD on IND-COM, comprising five industrial AD datasets (MVTec AD Bergmann et al. (2019), MVTec AD-2 Heckler-Kram et al. (2026), MVTec LOCO Bergmann et al. (2022), Real-IAD Wang et al. (2024), and Real-IAD-D3 Zhu et al. (2025)) and 78 categories. Under our supervised-controller protocol, Stage I improves image-level AUROC/AP from $79.82/91.36$ for train-only category-best routing to $84.75/93.51$, and Stage II reaches $91.34/96.34$. The detectors remain frozen; conventional UAD results serve as reference points, not supervision-matched comparisons. In summary, our contributions are threefold:

- We introduce AGENTICAD, an agent-based pipeline that learns sample-adaptive anomaly *routing–observation–detection* instead of committing to one detector and one fixed view.
- We develop a two-stage controller with full-information supervision: Stage I routes each sample to a complementary expert, and Stage II performs a multi-step observe–detect loop.
- On IND-COM, comprising five subsets and 78 categories, AGENTICAD improves over frozen experts and fixed-control baselines under a supervised-controller protocol.

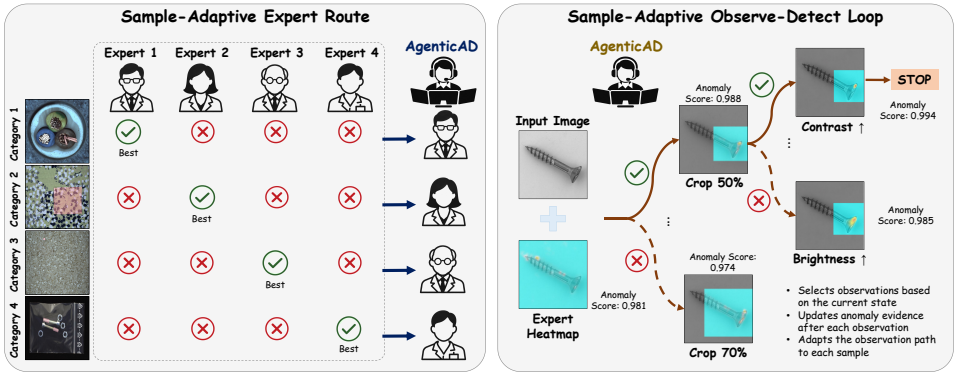


Figure 2: In multi-class industrial anomaly detection, conventional pipelines use a fixed detector and observation. AGENTICAD first **routes** each image to an expert, then uses the current score and heatmap to select the next observation or stop.

2 RELATED WORK

Unsupervised Anomaly Detection. Unsupervised anomaly detection (UAD) aims to identify abnormal patterns using only anomaly-free training data. Existing methods span feature reconstruction/distillation Deng & Li (2022); Madan et al. (2023); Hyun et al. (2024); Salehi et al. (2021), synthetic anomaly generation Li et al. (2021); Zavrtanik et al. (2021); Chen et al. (2024); Zhang et al. (2023); Liu et al. (2023), diffusion-based reconstruction He et al. (2024b); Zhang et al. (2025), and feature-statistics approaches Defard et al. (2021); Roth et al. (2022); Lee et al. (2022). Among them, PatchCore Roth et al. (2022) constructs a representative memory bank of pretrained patch features and detects anomalies via nearest-neighbor matching, while EfficientAD Batzner et al. (2024) employs a lightweight student–teacher architecture to achieve accurate and efficient anomaly detection.

Recent studies have further explored unified multi-class UAD You et al. (2022); Lu et al. (2023); Guo et al. (2023); Yin et al. (2023); He et al. (2024b;a), where a single model handles multiple object categories. UniAD You et al. (2022) pioneered this setting and introduced neighbor-masked attention and feature jittering to mitigate the "identical shortcut" problem in feature reconstruction. More recently, Dinomaly Guo et al. (2025b) leverages pretrained vision Transformer representations and a reconstruction-based framework to achieve strong generalization across diverse categories. In this paper, the term UAD describes the training regime of our frozen base detectors, not the full AgenticAD system. AgenticAD uses labeled target-category data to train its controllers and is therefore a supervised control framework built on unsupervised anomaly detectors.

Agent-based Anomaly Detection. Recent studies have introduced agents into anomaly detection at different stages. AutoIAD Ji et al. (2025) uses manager-driven multi-agent collaboration to automate IAD system development. IndusAgent Tan et al. (2026) formulates open-vocabulary IAD as tool-augmented MLLM reasoning with visual enhancement and prior retrieval, while AgentIAD Miao et al. (2025) performs iterative inspection through adaptive memory and tool use. Beyond visual inspection, Yang et al. Yang et al. (2025) employ a DQN agent for adaptive thresholding of time-series anomaly scores. In contrast, AGENTICAD learns an expert router and a structured full-information controller to adaptively orchestrate heterogeneous frozen detectors, selectively acquiring and fusing complementary visual evidence within a compact, auditable action space.

3 METHOD

3.1 PROBLEM FORMULATION AND TASK UTILITIES

Let x_i denote an image and $y_i \in \{0, 1\}$ its image-level label, where one indicates an anomaly. AgenticAD routes each image to one of four frozen experts:

$$\mathcal{E} = \{\text{Dinomaly}, \text{EfficientAD}, \text{PatchCore}, \text{UniAD}\}. \tag{1}$$

All utilities are used only to supervise decisions during training; neither labels nor counterfactual outcomes are required at inference. The overview of our proposed AgenticAD is shown in Figure 3.

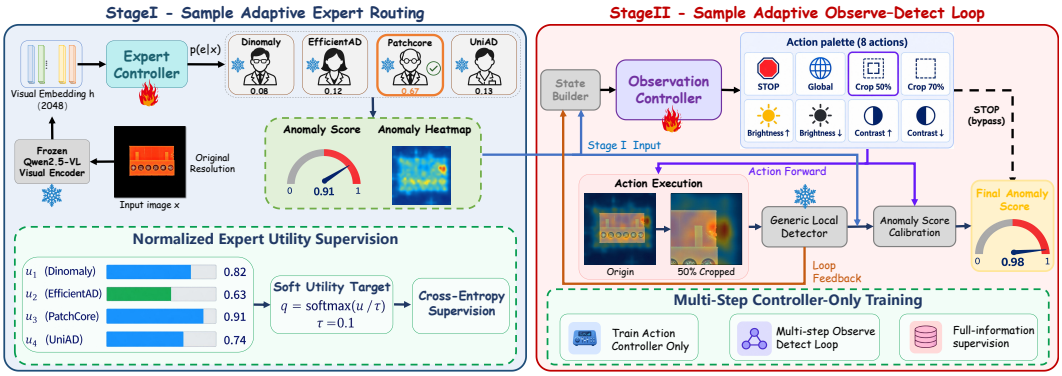


Figure 3: AGENTICAD: **Route, Observe & Detect** for sample-adaptive industrial anomaly detection. Unlike fixed detectors, category-level routing, or predefined observations, AGENTICAD adapts both *which* expert to use and *how* to inspect each image. Stage I (**Route**) selects one frozen expert to produce an initial score and heatmap. Stage II (**Observe & Detect**) iteratively chooses STOP or a heatmap-conditioned crop or photometric observation, analyzes the resulting view with a frozen generic local detector, and maps local evidence back to the original image.

Image-level utility. Each expert's raw image score is mapped to $s_{i,e} \in [0, 1]$ by an expert-specific empirical cumulative distribution function (ECDF) fitted on the training split. We define:

$$u_{i,e}^{\text{img}} = y_i s_{i,e} + (1 - y_i)(1 - s_{i,e}), \tag{2}$$

which favors high anomaly confidence on anomalous images and low confidence on normal images. Rank-percentile ties are retained rather than broken arbitrarily.

Pixel-level utility. Let $M_{i,e}$ and G_i denote the predicted heatmap and binary anomaly mask. We first calibrate each expert's pixels with an ECDF fitted only to its normal training pixels, producing $\widetilde{M}_{i,e}$. For an anomalous image with both foreground and background pixels:

$$u_{i,e}^{\text{pix}} = 0.3\ \text{AUROC}(G_i, \widetilde{M}_{i,e}) + 0.7\ \text{AP}(G_i, \widetilde{M}_{i,e}). \tag{3}$$

For a normal image, we instead use $u_{i,e}^{\text{pix}} = 1 - F_e^{\text{top1\%}}(r_{i,e})$, where $r_{i,e}$ is the mean of the highest-scoring 1% of pixels and $F_e^{\text{top1\%}}$ is its train-normal ECDF. This penalizes localized false positives without assigning an undefined pixel AUROC or AP. An anomalous image without a valid positive mask receives the neutral utility 0.5.

Controller-only learning. AgenticAD improves detection solely by learning lightweight decision controllers; the visual encoder, specialist experts, generic local detector, and evidence-aggregation rules remain frozen. Both stages use the same generic controller block with a 256-dimensional hidden layer and dropout rate 0.1. They differ only in their decision context and output space: Stage I maps the frozen image representation to four expert values, whereas Stage II maps the current observation state to seven action logits. Figure 4 summarizes this shared controller design. The controllers adapt how the fixed detector and observation tools are used; they do not update detector parameters or learn the score-aggregation rules.

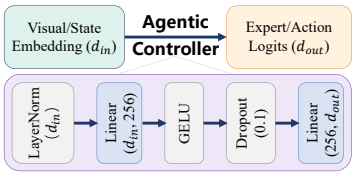


Figure 4: The shared lightweight controller for both expert and observation routing in AgenticAD.

3.2 STAGE I: SAMPLE-ADAPTIVE EXPERT ROUTING

A frozen visual encoder ϕ represents the image as $h_i = \phi(x_i)$, and a lightweight router f_θ produces one value for each expert. Training has full information because all four frozen outcomes are available for every training image.

For image-level detection, the utility vector is converted into a soft target:

$$q_{i,e}^{\text{img}} = \frac{\exp(u_{i,e}^{\text{img}}/\tau)}{\sum_{e'} \exp(u_{i,e'}^{\text{img}}/\tau)}, \qquad \mathcal{L}_{\text{img-route}} = -\sum_{e} q_{i,e}^{\text{img}} \log p_\theta(e \mid x_i), \tag{4}$$

where $p_\theta = \text{softmax}(f_\theta(h_i))$. This target preserves near-ties and the relative quality of all experts. For pixel localization, the router regresses the utility advantage over a fixed fallback expert e_0:

$$\Delta u_{i,e}^{\text{pix}} = u_{i,e}^{\text{pix}} - u_{i,e_0}^{\text{pix}}, \qquad \mathcal{L}_{\text{pix-route}} = \text{SmoothL1}\Big(f_\theta(h_i), \Delta \mathbf{u}_i^{\text{pix}}\Big). \tag{5}$$

At inference, Stage I selects one expert. The pixel router falls back to e_0 if no predicted advantage is positive; otherwise both tasks take the highest-valued expert. Only this expert is executed at its native preprocessing and resolution. Its normalized score $s_i^{(0)}$, raw score, identity, and heatmap $M_i^{(0)}$ initialize Stage II, while the other experts are never called for that sample.

3.3 Stage II: Sample-Adaptive Observe–Detect Loop

Stage II learns only an action controller. The four experts and frozen generic local detector remain frozen, and the image- and pixel-level aggregation rules are fixed. Thus, additional labeled data cannot fit a second anomaly detector or a learned fusion model.

State and actions. The controller chooses from the action set:

$$\mathcal{A} = \{\text{Stop}, \text{Crop50}, \text{Crop70}, \text{BrightUp}, \text{BrightDown}, \text{ContrastUp}, \text{ContrastDown}\}.$$

The observation controller receives state:

$$z_i^t = [b_i, \psi(\mathcal{H}_i^t)], \tag{6}$$

and assigns probabilities to valid actions as follows. Let $\mathcal{A}_i^t \subseteq \mathcal{A}$ be the actions valid in state z_i^t:

$$\pi_\omega(a \mid z_i^t) = \frac{\mathbb{1}[a \in \mathcal{A}_i^t] \exp(g_\omega(z_i^t)_a)}{\sum_{a' \in \mathcal{A}_i^t} \exp(g_\omega(z_i^t)_{a'})}. \tag{7}$$

Here g_ω outputs seven action logits. The fixed initial descriptor b_i concatenates the frozen image representation, the selected-expert one-hot vector, its normalized and raw anomaly scores, and statistics of the initial heatmap $M_i^{(0)}$. The history descriptor $\psi(\mathcal{H}_i^t)$ encodes, for each completed observation, the frozen generic local detector score, aligned-heatmap statistics, action identity, and decision depth; unused history slots are padded up to the three-call budget. At step t, the controller also maintains the current view:

$$\mathcal{V}_i^t = (x_i^t, M_i^t, T_i^t), \tag{8}$$

where T_i^t maps the current coordinates back to the original image. Crop actions center on the peak of the latest heatmap, and all photometric actions operate on the latest view. Hence, later decisions depend on earlier observations rather than restarting from the original image. Stop terminates the trajectory; every other action invokes the frozen generic local detector once.

Observe, align, and update. For a non-stop action a_i^t, the observation operator and frozen generic local detector produce:

$$(\widetilde{x}_i^{t+1}, T_i^{t+1}) = \mathcal{O}_{a_i^t}(\mathcal{V}_i^t), \qquad (d_i^{t+1}, \widetilde{M}_i^{t+1}) = \mathcal{G}(\widetilde{x}_i^{t+1}). \tag{9}$$

Here $d_i^{t+1} \in [0,1]$ is the generic local detector's abnormal-class probability. Composing the transformation traces projects the normalized local map to the original coordinates as $\bar{M}_i^{t+1}$ with binary validity mask W_i^{t+1}. The Stage I map $M_i^{(0)}$ and projected maps $\bar{M}_i^j$ are normalized to $[0,1]$ before fusion. The resulting evidence is appended to the history to form the next controller input z_i^{t+1}.

Fixed evidence aggregation. Stage II uses no learned outcome-fusion network. After t observations, the image score is the equal-weight mean:

$$\widehat{s}_i^t = \frac{s_i^{(0)} + \sum_{j=1}^{t} d_i^j}{t+1}. \tag{10}$$

For localization, aligned maps are averaged only where they are valid:

$$\widehat{M}_i^t(p) = \frac{M_i^{(0)}(p) + \sum_{j=1}^{t} W_i^j(p)\bar{M}_i^j(p)}{1 + \sum_{j=1}^{t} W_i^j(p)}. \tag{11}$$

Thus, stopping at the root exactly reproduces Stage I, while later stopping retains all acquired evidence with no trainable fusion parameters.

Full-information controller learning. Because the detector and observation tools are frozen, every legal next action can be executed offline at each training state. This yields a full counterfactual utility vector rather than a reward only for the sampled action. For each valid action a, let $\widehat{s}_{i,a}^{t+1}$ and $\widehat{M}_{i,a}^{t+1}$ be the aggregated score and map that would result from taking that action; for STOP, they equal the current outputs $\widehat{s}_i^t$ and $\widehat{M}_i^t$. Image- and pixel-level action utilities are then:

$$U_{i,a}^{t,\mathrm{img}} = y_i \widehat{s}_{i,a}^{t+1} + (1-y_i)(1-\widehat{s}_{i,a}^{t+1}) - \lambda \mathbb{1}[a \neq \textsc{Stop}]. \tag{12}$$

$$\ell_i(M) = \begin{cases} 0.3\ \mathrm{AUROC}(G_i, M) + 0.7\ \mathrm{AP}(G_i, M), & y_i = 1 \text{ and } G_i \text{ has pixels}, \\ 1 - r_i^{\mathrm{top1\%}}(M), & y_i = 0, \\ 0.5, & y_i = 1 \text{ and } G_i \text{ has no valid mask}. \end{cases} \tag{13}$$

Here $r_i^{\mathrm{top1\%}}(M)$ is the mean of the highest-scoring 1% of pixels in map M. The pixel-level action utility is:

$$U_{i,a}^{t,\mathrm{pix}} = \ell_i(\widehat{M}_{i,a}^{t+1}) - \lambda \mathbb{1}[a \neq \textsc{Stop}]. \tag{14}$$

Here λ discourages unnecessary calls. STOP evaluates the current output and incurs no call cost.

For either task $T \in \{\mathrm{img}, \mathrm{pix}\}$, the complete utility vector defines a soft action target and a cross-entropy objective:

$$q_i^{t,T}(a) = \frac{\exp(U_{i,a}^{t,T}/\tau_a)}{\sum_{a' \in \mathcal{A}_i^t} \exp(U_{i,a'}^{t,T}/\tau_a)}, \qquad \mathcal{L}_{\mathrm{ctrl}} = -\mathbb{E}_{(i,t)}\left[\sum_{a \in \mathcal{A}_i^t} q_i^{t,T}(a) \log \pi_\omega(a \mid z_i^t)\right]. \tag{15}$$

Here τ_a is the action-target temperature, and invalid actions are excluded from both distributions. The soft target preserves near-tied actions, while full-information supervision (FIS) provides dense, lower-variance supervision for estimating relative action quality than GRPO Shao et al. (2024).

Sequential inference. At deployment, the controller selects the highest-probability action from its seven outputs. STOP returns the current score or heatmap; otherwise, the selected observation is executed and the state is updated as described above. This repeats until STOP or the three-call budget, producing one auditable, sample-dependent trajectory without evaluating counterfactual branches.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmarks. We evaluate on IND-COM (short for Industrial Comprehension), a unified multi-domain benchmark comprising MVTec AD Bergmann et al. (2019), MVTec-AD-2 Heckler-Kram et al. (2026), MVTec LOCO Bergmann et al. (2022), Real-IAD Wang et al. (2024), and Real-IAD-D3 Zhu et al. (2025). Together, these five subsets contain 78 object categories and cover diverse defect scales, structures, acquisition conditions, and anomaly semantics. As summarized in Table 1, the anomaly experts are trained on 14,769 official `train/good` images. For controller training and evaluation, we partition the official test pool once by image into 7,316 labeled Router-training images (2,017 normal and 5,299 anomalous) and a disjoint 4,876-image held-out test set (1,341

Table 1: Complete IND-COM data composition. "expert train" is the official `train/good` partition used to train the anomaly-detection experts and other comparison methods, therefore contains only normal images. The official test pool is partitioned into the Router training set and a shared test set used by all methods. N and A denote normal and anomalous images, respectively.

		Expert train			Router train			Unified test		
Subset	Categories	N	A	Total	N	A	Total	N	A	Total
MVTec AD	15	3,629	0	3,629	281	755	1,036	186	503	689
MVTec-AD-2	8	2,528	0	2,528	409	423	832	272	282	554
MVTec LOCO	5	2,083	0	2,083	344	596	940	231	397	628
Real-IAD	30	2,529	0	2,529	383	1,335	1,718	252	893	1,145
Real-IAD-D3	20	4,000	0	4,000	600	2,190	2,790	400	1,460	1,860
Total	78	14,769	0	14,769	2,017	5,299	7,316	1,341	3,535	4,876

normal and 3,535 anomalous), used consistently by all methods. Image-level labels supervise the image controllers; available pixel masks additionally supervise pixel controllers. This is a custom supervised-controller split of the official test pool, rather than the standard normal-only UAD training protocol. We report results both by constituent subset and over the complete IND-COM collection.

Baselines. We compare three groups of baselines. *Individual detectors* include the four routing experts together with UniMMAD, Dinomaly2, and DeCo-Diff. These conventional UAD methods are trained on anomaly-free data and serve as unsupervised reference points; their results are not supervision-matched to AgenticAD. *Routing baselines* use random selection or the best expert for each category estimated from labeled Router-training data; a ground-truth per-sample oracle provides a non-deployable upper bound. *Observation baselines* apply one fixed global, crop, brightness, or contrast operation to the Stage I output. These policies share the same frozen detectors and equal-weight aggregation as AgenticAD. We additionally compare full-information supervision with GRPO using the same controller state and action space, providing within-protocol comparisons of adaptive control strategies under the labeled controller-training data.

Protocol and metrics. IND-COM is split once into 60% Router training data and a shared 40% test set, whose composition is reported in Table 1. Unlike conventional unsupervised AD, AgenticAD uses image labels to train its routing and observation controllers, and pixel masks for pixel-level control. The detectors remain frozen: the controllers learn to select experts and observation actions, while fixed rules compute and aggregate anomaly scores. Thus, AgenticAD does not fine-tune detectors or learn a new anomaly-scoring or fusion function. We report image-level AUROC and AP. Pixel AUROC/AP are computed independently on every test image with a valid anomalous mask and then averaged over images. Category-averaged image-level results are provided separately in the appendix. For Stage I we additionally report average utility, oracle regret, and oracle-set hit rate, which counts a prediction as correct when it belongs to the complete tied oracle set.

Implementation details. The four anomaly experts and frozen generic local detector (AnomalyCLIP Zhou et al. (2024)) remain frozen throughout controller training. Stage I uses the mean-pooled, ℓ_2-normalized 2,048-dimensional visual feature from Qwen2.5-VL-3B-Instruct. Both controllers use a LAYERNORM–LINEAR$(d, 256)$–GELU–DROPOUT(0.1)–LINEAR$(256, K)$ architecture, with $K = 4$ experts in Stage I and $K = 7$ actions in Stage II. Stage II allows at most three observations and records the selected action, local view evidence, heatmap statistics, and call ratio in the padded history. Further architectural, optimization, and preprocessing details are provided in Appendix A.2.

4.2 MAIN RESULTS

Image-Level Anomaly Detection. Table 2 compares our router with the four candidate experts, recent detector baselines, random routing, and train-only category-best routing. Stage I reaches $84.75/93.51$ AUROC/AP, improving category-best by $4.93/2.15$ points. It also outperforms every individual routing expert by a clear margin. The improvement is seen on four of the five constituent datasets, with MVTec LOCO being the only exception. This pattern is consistent with our motivation: the best detector is not fixed for an object category, and the visual appearance of each sample provides useful evidence for choosing among complementary experts. The routing statistics in Table 4 tell a similar story. The learned policy raises average utility to 74.53%, reduces oracle regret to 9.15%,

Table 2: Sample-averaged image-level AUROC/AP on IND-COM unified test subsets. "Category-best" selects the best expert on the training split. Bold marks the best result and oracle is not deployable.

Method	MVTec AD	MVTec-AD-2	MVTec LOCO	Real-IAD	Real-IAD-D3	Overall
Dinomaly Guo et al. (2025b)	93.66/97.19	74.86/76.19	77.87/87.98	78.90/92.60	72.53/91.01	76.20/89.77
EfficientAD Batzner et al. (2024)	91.53/94.32	72.35/70.95	87.41/93.43	86.05/94.67	59.11/87.56	70.74/87.40
PatchCore Roth et al. (2022)	83.45/93.22	73.75/75.08	83.33/90.65	82.02/93.75	75.01/91.97	76.32/89.65
UniAD You et al. (2022)	87.76/95.31	59.37/62.32	76.38/86.18	75.45/91.31	67.79/89.24	71.21/87.18
UniMMAD Zhao et al. (2026)	91.19/96.95	68.23/63.88	71.02/80.15	82.34/93.83	64.72/83.72	71.41/84.38
Dinomaly2 Guo et al. (2025a)	**97.20/99.06**	68.62/73.39	76.42/87.46	86.64/95.96	63.42/83.85	77.03/88.25
DeCo-Diff Beizaee et al. (2025)	44.71/69.82	53.82/55.15	48.09/62.91	51.31/78.00	53.32/80.93	43.95/68.50
Random Expert	88.75/95.76	69.22/70.39	80.18/89.25	79.31/92.43	67.65/89.74	72.82/88.28
Train-Set Category Best	89.49/95.29	76.50/78.07	**88.39/93.58**	81.42/92.61	79.74/93.41	79.82/91.36
AGENTICAD Stage I	94.56/97.72	80.34/81.48	86.59/92.79	87.55/95.28	84.12/95.06	84.75/93.51
AGENTICAD Stage II	96.35/98.55	**85.11/85.97**	87.14/92.99	**90.08/96.49**	**93.63/98.03**	**91.34/96.34**
Per-sample Oracle (upper bound)	99.69/99.88	94.98/95.76	99.16/99.53	99.45/99.84	93.87/98.34	96.28/98.63

Table 3: Sample-averaged pixel-level AUROC/AP on IND-COM unified test subsets. Metrics are computed independently for each anomalous test image and then macro-averaged over all valid anomalous images within each subset. Bold marks the best result and oracle is not deployable.

Method	MVTec AD	MVTec-AD-2	MVTec LOCO	Real-IAD	Real-IAD-D3	Overall
Dinomaly Guo et al. (2025b)	95.95/58.27	**89.69**/12.54	72.46/30.06	97.71/29.49	91.83/10.26	82.75/18.70
EfficientAD Batzner et al. (2024)	97.12/66.03	62.19/8.430	65.67/36.65	95.17/12.33	76.34/2.620	70.38/3.890
PatchCore Roth et al. (2022)	95.04/56.16	80.19/4.030	75.48/29.66	96.37/31.29	91.37/6.410	86.14/10.23
UniAD You et al. (2022)	94.11/33.87	82.36/4.79	75.03/19.02	96.24/16.34	90.99/10.16	84.04/10.11
UniMMAD Zhao et al. (2026)	96.46/53.90	80.09/6.030	77.65/25.97	98.07/29.59	92.58/11.33	85.55/14.54
Dinomaly2 Guo et al. (2025a)	95.69/54.88	89.27/23.81	74.40/28.08	**98.42**/40.78	**93.80**/12.55	83.98/17.54
DeCo-Diff Beizaee et al. (2025)	51.85/3.950	55.97/0.720	50.13/6.380	78.67/0.910	77.64/2.500	66.69/2.930
Random Expert	97.18/73.01	76.28/21.19	88.19/54.45	94.53/45.35	83.15/25.09	88.04/40.01
Train-Set Category Best	97.81/**80.67**	75.23/20.38	87.65/59.54	95.46/**57.47**	90.45/37.84	91.24/49.94
AGENTICAD Stage I	**97.95**/80.45	81.00/29.52	88.18/62.26	95.07/56.04	91.43/39.39	92.08/51.22
AGENTICAD Stage II	**98.21/80.76**	81.45/**29.84**	**88.72/62.63**	95.46/56.34	91.75/**39.70**	**92.45/51.53**
Per-sample Oracle (upper bound)	98.90/87.26	89.69/34.16	94.26/71.63	98.55/68.69	94.34/45.59	95.67/59.37

and selects a member of the tied oracle set for 56.95% of the test images. Importantly, it continues to use all four experts instead of collapsing to a single dominant choice. The gap to the per-sample oracle ($96.28/98.63$) also makes clear that sample-adaptive routing is useful, but far from solved.

Stage II starts from the frozen Stage I result and decides whether the sample needs another observation and, if so, how it should be inspected. This observe–detect loop raises AUROC/AP from $84.75/93.51$ to $91.34/96.34$. All five subsets benefit, with the largest AUROC gain appearing on Real-IAD-D3 ($84.12 \rightarrow 93.63$), whose diverse defects make a single fixed observation particularly limiting. All Stage II controls use the same frozen detectors and fixed equal-weight score aggregation; they differ only in how the observation action is chosen. The gain therefore cannot be attributed to a separately trained fusion model. It reflects the benefit of deciding both whether another observation is needed and which transformation should be applied over a trajectory of up to three calls.

Pixel-Level Anomaly Localization. Table 3 evaluates the same routing decisions using the selected experts' heatmaps. Stage I obtains $92.08/51.22$ pixel AUROC/AP, compared with $91.24/49.94$ for category-best routing. The gain is smaller than in image-level detection, but it is achieved without modifying or retraining any expert. This distinction matters: selecting an expert with a strong image score does not automatically guarantee the best spatial localization. The sizeable gap to the pixel oracle ($95.67/59.37$) confirms that predicting which expert will produce the most accurate heatmap remains a challenging problem in its own right.

Stage II brings a modest localization gain, from $92.08/51.22$ to $92.45/51.53$ pixel AUROC/AP (+0.37/+0.31 points). This is substantially smaller than its image-level gain. A complementary observation can improve image-level confidence without greatly improving the ordering of thousands of pixels. After a crop or appearance transformation, the new frozen generic local detector heatmap must also be projected back accurately and remain spatially compatible with the Stage I heatmap. In the current equal-mean fusion scheme, these observations are often helpful for image-level judgment but partly redundant or noisy for localization. Accordingly, the current evidence for Stage II is

Table 4: Stage I utility, oracle agreement, and expert selection rates (%). Lower regret is better.

Method	Utility↑	Regret↓	Hit↑	Dinomaly	EfficientAD	Patchcore	UniAD
Random	65.99	17.69	35.44	24.38	25.02	25.10	25.49
Category-Best	71.43	12.25	48.52	29.98	32.18	30.87	6.97
Image+Category	73.44	10.24	55.39	32.12	26.56	29.98	11.34
Stage I (Image Only)	**74.53**	**9.15**	**56.95**	31.46	22.97	28.40	17.17

strongest for image-level detection; its localization benefit is positive but limited, leaving room for localization-aware control.

4.3 Ablation Study

Table 5 isolates observation control while keeping the detectors and score aggregation fixed. The Stage I + AnomalyCLIP global-view ensemble obtains $82.79/92.40$, below the Stage I baseline ($84.75/93.51$). Most fixed one-step operations also underperform Stage I; only the 50% crop gives a small gain, reaching $85.12/93.89$. Since every fixed policy makes one additional call, simply adding AnomalyCLIP is insufficient—the observation must be selected for the current sample. Adaptive control is considerably stronger. GRPO reaches $88.04/95.02$ with 0.704 average calls, whereas full-information supervision reaches $91.34/96.34$ with 0.969 calls, outperforming GRPO by $3.30/1.32$ points. Although FIS uses slightly more observations, the weak fixed one-call results show that call count alone cannot explain the gain. When counterfactual outcomes are available, directly supervising their relative utilities is better aligned with this compact action-selection problem than learning from sampled GRPO trajectories.

Table 5: Stage II ablations. We report sample-pooled image-level metrics (%).

Configuration	Max. obs.	AUROC	AP	Mean calls
Baseline (Stage I)	0	84.75	93.51	0.000
Random Obs.	1	83.25	92.81	1.000
Stage I + Global	1	82.79	92.40	1.000
Fixed 50% Crop	1	85.12	93.89	1.000
Fixed 70% Crop	1	84.20	93.30	1.000
Fixed Bright. ↑	1	82.90	92.48	1.000
Fixed Bright. ↓	1	83.25	92.81	1.000
Fixed Contr. ↑	1	82.69	92.36	1.000
Fixed Contr. ↓	1	82.76	92.56	1.000
GRPO Training	3	<u>88.04</u>	<u>95.02</u>	0.704
AGENTICAD	3	**91.34**	**96.34**	0.969

Table 6: First-decision distribution of the final Stage II-FIS controller (%). "Normal" and "Anomalous" are post-hoc slices. *Mean calls* counts all non-stop generic local observations over the trajectory.

Slice	Stop	Crop50	Crop70	Bright+	Bright−	Contrast+	Contrast−	Mean calls
Overall	62.76	2.24	14.52	2.15	1.58	5.37	11.38	0.969
Normal	55.11	2.24	11.33	3.36	1.94	8.35	17.67	1.128
Anomalous	65.66	2.23	15.73	1.70	1.44	4.24	9.00	0.756

Observation Selection. Table 6 shows that the Stage-II-FIS controller makes sample-adaptive decisions, stopping immediately for 62.76% of test images. Among non-stop actions, 70% cropping is most frequent (14.52%), followed by contrast reduction (11.38%) and enhancement (5.37%), highlighting the value of focused spatial inspection. Anomalous images receive 70% crops more often than normal images (15.73% versus 11.33%), suggesting that localized anomalies benefit from focused inspection. In contrast, normal images receive more frequent appearance adjustments, particularly contrast reduction (8.35% versus 4.24%), and require more calls on average (1.128 versus 0.756). This suggests that normal samples often require repeated appearance checks to resolve ambiguous evidence and reject low-confidence false positives.

5 Conclusion

We introduced AGENTICAD, a two-stage framework for sample-adaptive industrial anomaly detection that jointly learns expert routing and sequential observe–detect decisions. The lightweight controllers are the only learnable components in AGENTICAD and are trained with supervised signals, while all anomaly detectors remain frozen throughout training. By combining full-information supervision with adaptive observation and evidence fusion, AGENTICAD dynamically selects both the detector and inspection strategy for each sample. Across five IND-COM subsets, AGENTICAD achieves $91.34/96.34$ image-level AUROC/AP and $92.45/51.53$ pixel-level AUROC/AP, demonstrating gains over fixed and category-level selection. Future work will explore richer observation strategies, cost-aware control, and generalization to unseen categories and acquisition shifts.

AI USE STATEMENT

We used AI-assisted tools to improve the clarity, grammar, and overall presentation of the manuscript. These tools were used solely for language editing and polishing purposes and were not used to generate research ideas, conduct experiments, analyze results, or draw scientific conclusions. All scientific content and claims were developed, reviewed, and verified by the authors, who take full responsibility for the content of this manuscript.

REPRODUCIBILITY STATEMENT

We are committed to ensuring the reproducibility of our work. The manuscript provides detailed descriptions of the proposed method, experimental settings, and implementation details necessary to reproduce the reported results. Upon publication, we will publicly release all relevant resources associated with this work, including the datasets, source code, and model weights, to facilitate independent reproduction and further research.

REFERENCES

Kilian Batzner, Lars Heckler, and Rebecca König. Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 128–138, 2024.

Farzad Beizaee, Gregory A Lodygensky, Christian Desrosiers, and Jose Dolz. Correcting deviations from normality: A reformulated diffusion model for multi-class unsupervised anomaly detection. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19088–19097. IEEE, 2025.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten 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.

Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *International Journal of Computer Vision*, 130(4):947–969, 2022.

Qiyu Chen, Huiyuan Luo, Chengkan Lv, and Zhengtao Zhang. A unified anomaly synthesis strategy with gradient ascent for industrial anomaly detection and localization. In *European Conference on Computer Vision*, pp. 37–54. Springer, 2024.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. PaDiM: A patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pp. 475–489. Springer, 2021.

Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9737–9746, 2022.

Jia Guo, Shuai Lu, Lize Jia, Weihang Zhang, and Huiqi Li. Recontrast: Domain-specific anomaly detection via contrastive reconstruction. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 10721–10740, 2023.

Jia Guo, Shuai Lu, Lei Fan, Zelin Li, Donglin Di, Yang Song, Weihang Zhang, Wenbing Zhu, Hong Yan, Fang Chen, et al. One dinomaly2 detect them all: A unified framework for full-spectrum unsupervised anomaly detection. *arXiv preprint arXiv:2510.17611*, 2025a.

Jia Guo, Shuai Lu, Weihang Zhang, Fang Chen, Huiqi Li, and Hongen Liao. Dinomaly: The less is more philosophy in multi-class unsupervised anomaly detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 20405–20415, 2025b.

Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. MambaAD: Exploring state space models for multi-class unsupervised anomaly detection. *Advances in Neural Information Processing Systems*, 37:71162–71187, 2024a.

Haoyang He, Jiangning Zhang, Hongxu Chen, Xuhai Chen, Zhishan Li, Xu Chen, Yabiao Wang, Chengjie Wang, and Lei Xie. A diffusion-based framework for multi-class anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8472–8480, 2024b.

Lars Heckler-Kram, Jan-Hendrik Neudeck, Ulla Scheler, Rebecca König, and Carsten Steger. The mvtec ad 2 dataset: Advanced scenarios for unsupervised anomaly detection. *International Journal of Computer Vision*, 134:175, 2026.

Jeeho Hyun, Sangyun Kim, Giyoung Jeon, Seung Hwan Kim, Kyunghoon Bae, and Byung Jun Kang. Reconpatch: Contrastive patch representation learning for industrial anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2052–2061, 2024.

Dongwei Ji, Bingzhang Hu, and Yi Zhou. Autoiad: Manager-driven multi-agent collaboration for automated industrial anomaly detection. *arXiv preprint arXiv:2508.05503*, 2025.

Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. CFA: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10:78446–78454, 2022.

Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9664–9674, 2021.

Zhikang Liu, Yiming Zhou, Yuansheng Xu, and Zilei Wang. SimpleNet: A simple network for image anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20402–20411, 2023.

Ruiying Lu, YuJie Wu, Long Tian, Dongsheng Wang, Bo Chen, Xiyang Liu, and Ruimin Hu. Hierarchical vector quantized transformer for multi-class unsupervised anomaly detection. *Advances in Neural Information Processing Systems*, 36:8487–8500, 2023.

Neelu Madan, Nicolae-Cătălin Ristea, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised masked convolutional transformer block for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(1): 525–542, 2023.

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*, 2025.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter 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.

Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14902–14912, 2021.

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

Rongbin Tan, Fangfang Lin, Zhenlong Yuan, Min Qiu, Kejin Cui, Mengmeng Wang, Yi Wang, Zijian Song, Zhiyuan Wang, Jiyuan Wang, et al. Indusagent: Reinforcing open-vocabulary industrial anomaly detection with agentic tools. *arXiv preprint arXiv:2605.20682*, 2026.

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.

Xue Yang, Enda Howley, and Michael Schukat. Agent-based dynamic thresholding for adaptive anomaly detection using reinforcement learning. *Neural Computing and Applications*, 37(23): 18775–18791, 2025.

Haonan Yin, Guanlong Jiao, Qianhui Wu, Borje F Karlsson, Biqing Huang, and Chin Yew Lin. LafitE: Latent diffusion model with feature editing for unsupervised multi-class anomaly detection. *arXiv preprint arXiv:2307.08059*, 2023.

Zhiyuan You, Lei Cui, Yujun Shen, Kai Yang, Xin Lu, Yu Zheng, and Xinyi Le. A unified model for multi-class anomaly detection. *Advances in Neural Information Processing Systems*, 35: 4571–4584, 2022.

Vitjan Zavrtanik, Matej Kristan, and Danijel Skoč aj. 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.

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*, 2025.

Xuan Zhang, Shiyu Li, Xi Li, Ping Huang, Jiulong Shan, and Ting Chen. DeSTSeg: Segmentation guided denoising student-teacher for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3914–3923, 2023.

Yuan Zhao, Youwei Pang, Lihe Zhang, Hanqi Liu, Jiaming Zuo, Huchuan Lu, and Xiaoqi Zhao. Unimmad: Unified multi-modal and multi-class anomaly detection via moe-driven feature decompression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28502–28511, 2026.

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*, volume 2024, pp. 49705–49737, 2024.

Wenbing Zhu, Lidong Wang, Ziqing Zhou, Chengjie Wang, Yurui Pan, Ruoyi Zhang, Zhuhao Chen, Linjie Cheng, Bin-Bin Gao, Jiangning Zhang, Zhenye Gan, Yuxie Wang, Yulong Chen, Shuguang Qian, Mingmin Chi, Bo Peng, and Lizhuang Ma. Real-iad d3: A real-world 2d/pseudo-3d/3d dataset for industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15214–15223, 2025.

Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, pp. 392–408. Springer, 2022.

A Appendix

A.1 Dataset Composition and Splits

IND-COM combines MVTec AD Bergmann et al. (2019), MVTec AD-2 Heckler-Kram et al. (2026), MVTec LOCO Bergmann et al. (2022), Real-IAD Wang et al. (2024), and Real-IAD-D3 Zhu et al. (2025). The benchmark contains 78 object categories and covers structural, logical, surface, and localized defects under diverse imaging conditions. We use the official `train/good` images to train the frozen anomaly experts. For supervised controller training and evaluation, the official test pool is partitioned once by image into 7,316 labeled Router-training images (2,017 normal and 5,299 anomalous) and a disjoint 4,876-image held-out test set (1,341 normal and 3,535 anomalous). Image labels supervise image-level controllers, and pixel masks additionally supervise pixel-level controllers. The same split is used for all routing, observation, and baseline comparisons; this is a custom supervised-controller split, not the standard normal-only UAD protocol. Within the Router-training set, 20% of the samples are reserved for image-router checkpoint selection; the router is then refitted on all 7,316 samples.

A.2 Implementation Details

Stage I image routing uses soft utility targets with temperature 0.1. Its training epoch is selected on an internal 20% split, after which the router is refitted on all 7,316 Router-training samples. Pixel

routing uses 100 FULL_TRAIN epochs and SmoothL1 regression of each expert's utility advantage over the fallback expert. At inference, the pixel router selects the predicted expert only when its maximum predicted advantage exceeds δ; otherwise it selects e_0. Stage I controllers use AdamW with learning rate 10^{-3}, weight decay 10^{-4}, and batch size 256.

The Stage II input combines the 2,048-dimensional visual feature, expert one-hot vector, and ten heatmap statistics; image control additionally uses normalized and raw Stage I scores. Three padded history slots store frozen generic local detector scores, heatmap statistics, and six-dimensional action one-hot vectors, along with a scalar call ratio, yielding 2,116-dimensional image and 2,114-dimensional pixel inputs. Brightness factors are $1.15^{\pm 1}$ and contrast factors are $1.35^{\pm 1}$. Local heatmaps are normalized and mapped to a 64×64 canvas in original-image coordinates. Image scores average the Stage I ECDF score and the generic local detector's abnormal-class probability. Pixel utility uses $0.3\,\mathrm{AUROC} + 0.7\,\mathrm{AP}$ on anomalous images, one minus the hottest-1% mean on normal images, and 0.5 for anomalous images without valid masks; non-stop actions incur a 0.01 penalty. The image controller receives a 15-epoch full-information initialization followed by 100 controller-training epochs; the pixel controller trains for 100 epochs. Stage II controllers use AdamW with learning rate 10^{-4}, weight decay 10^{-4}, batch size 256, and temperature 0.1. AnomalyCLIP uses ViT-L/14-336 with a prompt trained on VisA Zou et al. (2022).

A.3 Category-Averaged Image-Level Evaluation

In addition to sample-pooled results, we report category-averaged image-level metrics in Table 7. For each object category, AUROC and AP are computed from all images belonging to that category. We then average the category-level values with equal weight, so categories with more test images do not dominate the result. The Overall column is the macro-average over all 78 categories. Under this protocol, Stage I obtains $87.83/93.92$ AUROC/AP, improving over category-best routing at $86.78/93.05$, while Stage II reaches $91.58/95.69$. These results confirm that the improvement is not caused solely by the image-count imbalance among the five constituent datasets.

A.4 Computational Resources

All experiments use CUDA-enabled NVIDIA GPUs on a single machine. Frozen visual-feature and anomaly-map caches are constructed in parallel across four or eight GPUs, while the lightweight Stage I and Stage II controllers are trained on one GPU after cache construction. Multi-GPU jobs use the provided distributed configuration with BF16 mixed precision. During controller training, the four experts, Qwen2.5-VL-3B-Instruct, and AnomalyCLIP are not fine-tuned; the additional trainable components are only the small routing and observation controllers. This separation substantially reduces the memory and compute requirements of controller optimization, while inference cost is dominated by the selected expert and the number of AnomalyCLIP calls.

A.5 Why Per-Image Localization Utility Does Not Optimize Pooled Pixel Metrics

We analyze a pixel-level extension of Stage I in which each frozen expert produces an anomaly map $\mathbf{H}_{i,e} \in \mathbb{R}^{H \times W}$ for image i. To make the outputs of heterogeneous experts comparable, each pixel score is transformed by an expert-specific empirical cumulative distribution function fitted exclusively to normal pixels in the training set. The resulting map $\widetilde{\mathbf{H}}_{i,e} \in [0,1]^{H \times W}$ therefore measures how extreme each response is relative to the corresponding expert's normal-score distribution. This train-normal ECDF is frozen before it is applied to validation or test images.

A natural full-information target is a weighted per-image localization utility. For an anomalous image with pixel mask $\mathbf{M}_i$, we may define:

$$u_{i,e}^{\mathrm{loc}} = \alpha\ \mathrm{AP}(\widetilde{\mathbf{H}}_{i,e}, \mathbf{M}_i) + (1-\alpha)\ \mathrm{AUROC}(\widetilde{\mathbf{H}}_{i,e}, \mathbf{M}_i), \tag{16}$$

where a larger α emphasizes small anomalous regions and penalizes false positive pixels more strongly. A normal image has no positive pixels and hence admits neither a meaningful per-image AP nor a per-image AUROC. Its localization utility can instead penalize the mean of the highest-scoring 1% of pixels, which measures spurious localized responses on an all-background mask. This use of normal images remains pixel-level false-positive supervision and does not introduce an image-classification objective.

Table 7: Stage I and Stage II image-level AUROC/AP (%) on the five IND-COM subsets using category-averaged aggregation. Metrics are first computed independently for each object category and then averaged with equal category weight; Overall averages all 78 categories. Bold marks the best deployable result in each column.

Method	MVTec AD	MVTec-AD-2	MVTec LOCO	Real-IAD	Real-IAD-D3	Overall
Dinomaly Guo et al. (2025b)	98.30/99.00	74.67/74.32	78.24/86.84	87.25/95.45	72.53/89.17	83.73/91.80
EfficientAD Batzner et al. (2024)	97.68/98.01	71.79/71.03	87.30/92.47	88.61/95.88	58.91/86.66	80.93/91.16
PatchCore Roth et al. (2022)	96.50/98.53	72.84/73.86	82.30/88.64	86.48/95.30	72.93/89.94	83.27/91.92
UniAD You et al. (2022)	93.75/97.12	60.23/63.40	75.06/83.23	78.77/92.32	67.84/88.11	76.71/88.61
UniMMAD Zhao et al. (2026)	98.48/99.53	70.62/70.13	76.34/86.34	86.13/94.91	70.04/87.82	82.16/90.89
Dinomaly2 Guo et al. (2025a)	**98.66/99.60**	74.31/75.83	78.71/88.04	**90.93/97.07**	71.96/89.67	85.07/92.90
DeCo-Diff Beizaee et al. (2025)	44.21/73.97	50.67/56.31	51.24/67.16	52.58/80.24	56.51/82.13	51.70/76.22
Uniform random	94.77/98.09	68.83/70.52	80.21/88.50	80.18/92.81	67.12/88.54	78.48/90.17
Category-best	97.81/97.78	76.41/77.22	**87.68**/92.01	88.78/95.72	79.43/92.10	86.78/93.05
AGENTICAD Stage I	97.76/98.74	80.70/79.59	86.87/92.29	88.41/95.74	82.59/93.73	87.83/93.92
AGENTICAD Stage II	97.98/98.96	**85.20/84.37**	87.28/**92.73**	90.21/96.64	**92.45/97.07**	**91.58/95.69**
Per-sample Oracle (upper bound)	100.0/100.0	94.53/94.93	99.04/99.45	99.51/99.85	93.50/98.25	97.52/98.94

Despite being intuitive, Eq. equation 16 is not equivalent to the final pooled pixel metrics. Per-image AUROC contains only comparisons between positive and negative pixels within the same image. Sample-pooled AUROC additionally contains all cross-image positive–negative pairs. For example, suppose one image assigns scores 0.6 and 0.5 to its anomalous and background pixels, while another assigns 0.4 and 0.3. Both images have perfect within-image ordering, but the first image's background score exceeds the second image's anomaly score. Their individual AUROCs are one, whereas the pooled AUROC is lower. Per-image supervision cannot observe or penalize this cross-image ordering error.

Pixel AP has the same non-additivity and is additionally sensitive to anomaly size. Averaging per-image utilities gives equal weight to an image with a few anomalous pixels and one with thousands of anomalous pixels, whereas sample-pooled AP weights their positive pixels according to their actual counts. This distinction is substantial in IND-COM because anomalous regions vary by orders of magnitude in area. Similarly, a Top-1% penalty on a normal image summarizes its strongest false responses, while the pooled metrics rank all of its background pixels. Consequently,

$$\frac{1}{N}\sum_{i}\mathrm{Metric}(\widetilde{\mathbf{H}}_{i,a_i},\mathbf{M}_i)\neq\mathrm{Metric}\Big(\mathrm{concat}_i\,\widetilde{\mathbf{H}}_{i,a_i},\mathrm{concat}_i\,\mathbf{M}_i\Big)\,, \tag{17}$$

where a_i is the expert selected for image i. Category-averaged evaluation introduces a third aggregation level: pixels are first pooled within each object category and the resulting category metrics are then macro-averaged. Thus, per-image, sample-pooled, and category-averaged objectives are three distinct optimization criteria.

Train-normal ECDF calibration addresses score-scale mismatch but not this objective mismatch. It is a monotone marginal transformation and therefore does not make equal percentiles from different experts equally likely to represent anomalous pixels. Moreover, routing combines maps drawn from several independently calibrated experts, changing the score mixture used by the pooled ranking. A router can therefore improve its per-image utility regression while leaving pooled AUROC/AP unchanged or even degrading them.

A metric-aligned alternative is to form multi-image training batches, concatenate the selected maps and masks, and optimize differentiable pooled objectives directly. A SoftAP loss approximates global precision–recall ranking, while a cross-image pairwise ranking loss approximates AUROC by requiring anomalous pixels to outrank hard background pixels from both the same and other images. If inference must execute exactly one expert, a straight-through categorical estimator can retain a hard expert choice in the forward pass while propagating gradients to the routing controller. The validation checkpoint should then be selected with the same pooled aggregation and AUROC/AP weighting used for the reported evaluation.

B LIMITATIONS

Despite these advantages, AgenticAD has two limitations. First, image-level detection and pixel-level localization are optimized under different utility functions. Image-level detection emphasizes ranking images by anomaly confidence, whereas localization requires accurate pixel-wise ranking and suppression of background responses. Consequently, the two tasks currently require separate controllers, which limits direct transfer between them. Second, sequential observation increases inference cost. Each additional step may require image transformation, an AnomalyCLIP forward pass, and heatmap alignment with the original image. The current stopping policy does not explicitly account for computational budget or latency, motivating future work on cost-aware control.