ANOMALYPROMPT: LLM-GUIDED ADAPTIVE PROMPTING VIA VISUAL PERCEPTION FOR ZERO-SHOT ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

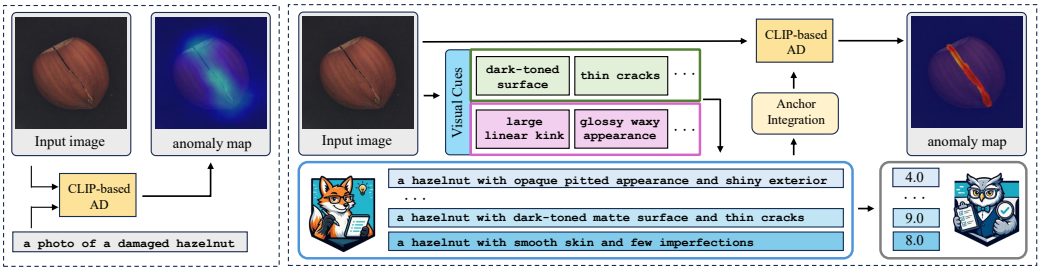
Industrial visual anomaly detection has long been constrained by the scarcity of anomalous samples and the high diversity of defect appearances. Recent CLIP/VLM-based methods have demonstrated promising cross-category generalization for ZSAD through prompt optimization. Most existing methods rely on static or manually crafted prompts or perform only passive adjustments, lacking mechanisms for active prompt exploration and quality assessment. Consequently, they struggle to capture fine-grained distinctions between normal and abnormal states and ultimately fail to consistently generate reliable prompts for anomaly detection. To address this limitation, we propose AnomalyPrompt, a training-efficient vision-language anomaly detection framework that unifies prompt generation, prompt evaluation, and visual-semantic alignment through a multi-stage prompting process. AnomalyPrompt operates through a structured "perception-reasoning-decision-execution" pipeline: it first uses the image-level embedding and patch embeddings of the input image to retrieve global surface features and fine-grained local features from hierarchical concept banks; then, an LLM-based Prompt Generator produces diverse, CLIP-friendly natural-language candidate prompts under structured constraints; next, an LLM Prompt Evaluator scores each candidate in terms of cue faithfulness, CLIP suitability, and clarity/readability, selecting the optimized prompt; finally, the selected prompt is encoded together with learnable normal/abnormal state vectors for anomaly scoring and localization. In this way, AnomalyPrompt enables image-specific prompt adaptation while preserving the semantic transparency of natural-language prompts. Extensive experiments on seven industrial anomaly detection datasets and four medical benchmarks demonstrate that AnomalyPrompt achieves SOTA performance in zero-shot anomaly detection.

1 INTRODUCTION

Automated inspection is essential to modern industrial production, yet reliable surface anomaly detection remains a long-standing challenge. In real-world manufacturing scenarios, defective samples are inherently scarce and often exhibit substantial appearance variability, making it difficult for supervised models to generalize without extensive fine-grained annotations. Zero-shot anomaly detection (ZSAD) offers a promising alternative by aiming to detect and localize anomalies in categories unseen during training. Recent vision-language models (VLMs), such as CLIP Radford et al. (2021), have further increased the appeal of this setting by enabling zero-shot recognition through natural language supervision. However, their anomaly detection performance remains highly dependent on the quality of textual prompts.

Early CLIP-based anomaly detection methods, such as WinCLIP Jeong et al. (2023), often rely on handcrafted descriptions or fixed prompt templates to represent defective states (e.g., "a photo of a cracked object"). Such prompts are inherently rigid and struggle to capture the subtle visual differences between normal and abnormal patterns, especially in industrial inspection, where fine-grained features such as scratches, dents, and discolorations are crucial for reliable detection. More recent approaches improve prompt flexibility through learnable or dynamically optimized prompts. Nevertheless, these methods still passively rely on training data for prompt optimization, lacking active prompt exploration and explicit prompt quality assessment. As a result, it remains challenging to

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.



(a) Conventional CLIP-based method (b) Our AnomalyPrompt

Figure 1: Comparison of conventional CLIP-based method and AnomalyPrompt. By replacing static prompting with feature-guided prompt generation and evaluator-based selection, AnomalyPrompt produces a more accurate anomaly map. Details of anchor integration are provided in Section 3.5.

construct prompts that generalize well to unseen categories while being both semantically meaningful and aligned with image-specific visual characteristics. In this regard, the emergence of instruction-following large language models (LLMs) opens up a new opportunity. With their strong semantic reasoning and generative capabilities, LLMs offer the potential to automatically produce more expressive and visually grounded prompts, thereby enhancing the generalizability of the prompts, as illustrated by the comparison in Figure 1.

Motivated by this observation, we introduce **AnomalyPrompt**, a multi-stage vision-language anomaly detection framework that organizes prompt construction into a structured "perception–reasoning–decision–execution" pipeline. To provide image-conditioned semantic guidance for prompt construction, the framework uses the image-level embedding and patch embeddings from the input image to retrieve global surface features and fine-grained local features from a hierarchical Concept Bank in the perception stage. Given that global information often overlooks fine-grained local variations while local information lacks category semantics, AnomalyPrompt jointly retrieves global surface and fine-grained local concepts. This allows text descriptions to better align with both image-level semantics and fine-grained local visual details, thereby providing richer context for the subsequent LLM-based reasoning.

During the reasoning stage, an LLM-based Prompt Generator produces a diverse set of CLIP-compatible candidate prompts under four structured constraints: (1) caption-style phrasing, (2) appearance-focused attributes, (3) grounding in retrieved visual cues, and (4) diversity across candidates. Therefore, our method leverages fine-grained visual information and the generalization capability of LLMs to generate expressive, image-aligned prompts.

To select the most suitable prompt, AnomalyPrompt introduces a multi-dimensional Prompt Evaluator that scores each candidate based on cue faithfulness, CLIP suitability, and clarity/readability. Finally, in the execution stage, the selected prompt is encoded together with learnable normal/abnormal state vectors as textual features and fed into CLIP, enabling the model to more accurately distinguish normal from abnormal patterns in the semantic space.

Our main contributions are summarized as follows:

- We present **AnomalyPrompt**, a multi-stage LLM-driven vision-language framework that reformulates prompt construction for zero-shot anomaly detection as a structured perception–reasoning process, using a hierarchical *Concept Bank* to retrieve global surface features and fine-grained local features for generating image-aligned prompts.
- We introduce a multi-dimensional Prompt Evaluator for quality-aware prompt decision, enabling prompt assessment, selection, and downstream execution within a unified framework to produce prompts that generalize better to unseen categories.
- Extensive experiments on seven industrial anomaly detection datasets and four medical anomaly detection benchmarks show that **AnomalyPrompt** achieves state-of-the-art performance in zero-shot anomaly detection, establishing a new paradigm for LLM-assisted vision-language anomaly detection.

2 Related Work

Zero-shot Anomaly Detection and Prompt Design. WinCLIP Jeong et al. (2023) adapts CLIP to ZSAD with handcrafted prompt ensembles and window-based visual aggregation. APRIL-GAN Chen et al. (2023a) and CLIP-AD Chen et al. (2023b) improve patch-level alignment for fine-grained localization, while SAA+ Cao et al. (2023) explores external foundation-model assistance for zero-shot anomaly segmentation. Prompt design remains a key factor in CLIP-based ZSAD, as anomaly scores depend on how image-level and patch-level features align with textual anchors for normal and abnormal states. Inspired by CoOp Zhou et al. (2022b) and CoCoOp Zhou et al. (2022a), AnomalyCLIP Zhou et al. (2024) learns object-agnostic prompts to capture generic normality and abnormality; AdaCLIP Cao et al. (2024) combines static and dynamic prompts for image-wise adaptation. FiLo Gu et al. (2024a) introduces fine-grained textual descriptions for anomaly localization, while Bayes-PFL Qu et al. (2025) improves prompt-space coverage through probabilistic prompt modeling. MRAD Xu et al. (2026) instead performs retrieval-based anomaly scoring with image-level and pixel-level memory banks. In contrast, AnomalyPrompt uses retrieved global and local visual features to guide LLM-based candidate prompt generation and selects the optimized text anchor through a multi-dimensional prompt decision module.

MLLMs for Anomaly Reasoning. MLLMs have recently been explored for anomaly detection and reasoning. AnomalyGPT Gu et al. (2024b) enables interactive anomaly localization, while Anomaly-OV Xu et al. (2025) introduces anomaly instruction tuning. In contrast, AnomalyPrompt uses an LLM for feature-grounded prompt generation and multi-dimensional prompt decision, while anomaly scoring is performed by the CLIP-based detection framework.

3 Method

3.1 Framework Overview

AnomalyPrompt addresses cross-dataset zero-shot anomaly detection, where the model is trained on auxiliary categories and evaluated on unseen target categories. Rather than treating prompt construction as a passive single-step procedure, it formulates prompt construction as a feature-guided multi-stage process consisting of perception, reasoning, decision, and execution. Figure 2 provides a framework overview of AnomalyPrompt.

Given an input image, the framework first encodes it into an image-level embedding and patch embeddings, which are then used to retrieve global and local features from hierarchical concept banks. The retrieved features are organized into a structured feature text, providing image-conditioned semantic guidance for prompt construction. Based on the class anchor and the structured feature text, an LLM-based prompt reasoning module generates a diverse set of candidate prompts under structured constraints, so that the resulting prompts remain suitable for vision-language anomaly detection. A prompt evaluator further assesses the generated candidates and selects the most suitable one for the current image. Together, prompt reasoning and prompt decision enable active prompt construction through candidate generation, quality-aware evaluation, and prompt selection. The selected prompt is finally integrated into the retained vision-language anomaly detection framework for image-level anomaly scoring and pixel-level localization.

The overall pipeline can be summarized as

$$x \xrightarrow{\text{Enc}} (z^I, Z^P) \xrightarrow{\mathcal{H}} (\mathcal{F}_g, \mathcal{F}_l, t_f) \xrightarrow{\mathcal{G}} \mathcal{P} \xrightarrow{\mathcal{E}} p^* \xrightarrow{\mathcal{I}} (s, M),$$

where x denotes the input image, and z^I and Z^P denote the image-level embedding and patch embeddings extracted by the image encoder, respectively. $\mathcal{H}$ denotes the hierarchical feature retrieval module, which retrieves global features $\mathcal{F}_g$ and local features $\mathcal{F}_l$ from the concept banks and organizes them into the structured feature text t_f. $\mathcal{G}$ denotes the structure-constrained prompt reasoning module, which generates the candidate prompt set $\mathcal{P} = \{p_i\}_{i=1}^K$. $\mathcal{E}$ denotes the multi-dimensional prompt decision module that selects the optimized text anchor p^* from $\mathcal{P}$. $\mathcal{I}$ denotes the integration and execution module that produces image-level anomaly score s and pixel-level anomaly map M.

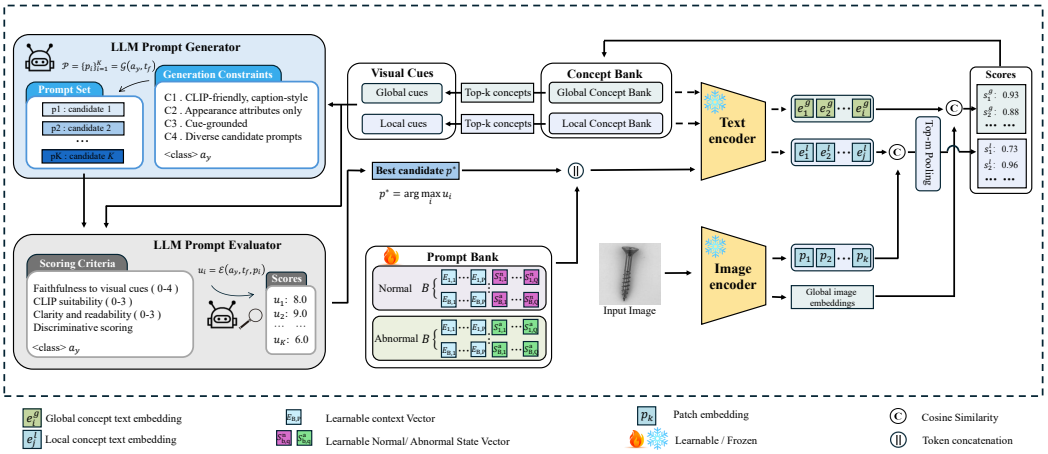


Figure 2: Framework overview of AnomalyPrompt. The framework retrieves global and local features from hierarchical concept banks, constructs structured feature text for prompt generation and evaluation, and integrates the selected prompt for image-level anomaly scoring and pixel-level localization.

3.2 Visual Perception via Hierarchical Concept Bank

AnomalyPrompt performs visual perception by using image-level embedding and patch embeddings to retrieve relevant global and local features from hierarchical concept banks, producing LLM-readable feature descriptions for prompt construction. Relying only on global information tends to produce overly generic descriptions that overlook fine-grained local variations, whereas using only local information often yields fragmented features without stable image-level semantics. To address this issue, AnomalyPrompt uses the current test image as the retrieval query and selects global and local features from two concept banks, thereby providing image-conditioned semantic guidance for subsequent prompt construction.

Specifically, we maintain two concept banks: a Global Concept Bank B_g and a Local Concept Bank B_l. The Global Concept Bank stores semantic concepts describing overall appearance attributes, such as material, texture, color, and surface condition, while the Local Concept Bank stores fine-grained concepts describing local visual patterns and appearance states. Both banks are instantiated as an LLM-generated semantic resource, and each concept phrase is encoded into the CLIP text embedding space for similarity-based retrieval. Formally, the two concept banks are defined as

$$B_g = \{(c_i^g, e_i^g)\}_{i=1}^{N_g}, \quad B_l = \{(c_j^l, e_j^l)\}_{j=1}^{N_l},$$

where c_i^g and c_j^l denote the corresponding concept phrases, and $e_i^g, e_j^l \in \mathbb{R}^D$ denote their CLIP text embeddings.

Conditioned on the image-level embedding z^I of the current image, we compute its similarity with each global concept embedding e_i^g:

$$s_i^g = \mathrm{sim}(z^I, e_i^g).$$

The retrieved global features are obtained by selecting the top-K_g concept phrases according to their similarity scores:

$$\mathcal{F}_g = \mathrm{TopK}_{K_g}\left(\{(c_i^g, s_i^g)\}_{i=1}^{N_g}\right).$$

For local feature retrieval, we use the patch embeddings $Z^P = \{z_n^P\}_{n=1}^{N_p}$ of the current image as queries to search the Local Concept Bank B_l. In the main implementation, we use the last-layer patch embeddings for local retrieval. For the j-th local concept, we compute its similarity to all patch embeddings and aggregate the strongest patch responses by top-m pooling:

$$s_j^l = \mathrm{TopMMean}_m\left(\{\mathrm{sim}(z_n^P, e_j^l)\}_{n=1}^{N_p}\right).$$

The retrieved local features are then obtained by selecting the top-K_l concept phrases according to their aggregated scores:

$$\mathcal{F}_l = \mathrm{TopK}_{K_l}\left(\{(c_j^l, s_j^l)\}_{j=1}^{N_l}\right).$$

Finally, the retrieved global and local features are organized into a structured feature text:

$$t_f = \mathrm{Format}(\mathcal{F}_g, \mathcal{F}_l),$$

where $\mathrm{Format}(\cdot)$ denotes a text formatting operation that arranges the retrieved features into two fields, `GLOBAL_FEATURES` and `LOCAL_FEATURES`. The resulting t_f serves as the semantic condition for subsequent prompt reasoning and decision.

3.3 Structure-Constrained Prompt Reasoning

With the original class anchor text a_y and the structured feature text t_f, AnomalyPrompt performs prompt reasoning to construct a set of candidate prompts for the current image. This step aims to transform the category-level anchor and the retrieved global and local features into candidate text anchors that are more suitable for downstream vision-language anomaly detection. Formally, the LLM-based reasoning module generates a set of candidate prompts.

$$\mathcal{P} = \{p_i\}_{i=1}^{K} = \mathcal{G}(a_y, t_f),$$

where $\mathcal{G}(\cdot)$ denotes the prompt reasoning function and K is the number of candidates generated.

Rather than allowing unconstrained free-form generation, AnomalyPrompt imposes four structured constraints so that the generated prompts remain suitable for vision-language anomaly detection: (1) CLIP-friendly, caption-style phrasing, which encourages concise visual descriptions instead of long instructions or diagnostic statements; (2) appearance-focused attributes, which restrict the prompts to visible properties such as material, texture, color, surface condition, shape, and defect patterns; (3) cue grounding, which requires the prompts to remain consistent with the retrieved global and local features summarized in t_f; and (4) candidate diversity, which encourages different prompts to cover complementary appearance attributes and anomaly-related patterns.

The resulting set of candidates $\mathcal{P}$ is then passed to the multi-dimensional decision module for quality assessment and prompt selection.

3.4 Multi-dimensional Prompt Decision

After obtaining the set of candidate prompts $\mathcal{P} = \{p_i\}_{i=1}^{K}$, AnomalyPrompt evaluates their quality before selecting the final text anchor. For each candidate prompt p_i, the evaluator considers the class anchor a_y, the structured feature text t_f, and the candidate itself, and assigns a scalar score:

$$u_i = \mathcal{E}(a_y, t_f, p_i), \quad u_i \in [0, 10],$$

where $\mathcal{E}(\cdot)$ denotes the prompt evaluator.

The score reflects three aspects of prompt quality: (1) faithfulness to the retrieved cues, which measures whether the prompt is consistent with the global and local features summarized in t_f; (2) CLIP suitability, which measures whether the prompt is concise, visual, and appropriate for vision-language matching; and (3) clarity and readability, which measures whether the prompt is linguistically clear and unambiguous. To avoid indistinguishable candidates, the evaluator is further instructed to assign discriminative scores, allowing prompts with clearer semantic advantages to stand out. The final optimized text anchor is selected as the highest-scoring candidate:

$$p^* = \arg\max_{p_i \in \mathcal{P}} u_i.$$

The selected prompt p^* is then used for subsequent integration and execution.

3.5 Integration and Execution

The optimized prompt p^* is integrated into the retained vision-language anomaly detection framework as the final text anchor. Specifically, we preserve the learnable context vectors E and the normal and abnormal state vectors S^n and S^a in the retained prompt parameterization, while replacing the

original class anchor with the selected prompt p^*. The downstream matching pipeline for image-level anomaly scoring and pixel-level localization is then executed with the optimized anchor.

Formally, the normal and abnormal prompt representations are written as

$$T_b^n = [E_b \mid S_b^n] \oplus \tau(p^*), \quad T_b^a = [E_b \mid S_b^a] \oplus \tau(p^*),$$

where $\tau(\cdot)$ denotes the token embeddings of the selected prompt, and $\oplus$ denotes token concatenation. The resulting text representations are encoded by the text encoder and aligned with the image-level embedding and patch embeddings for anomaly inference.

Following the retained base framework, the image-level anomaly score s is computed from the matching between the image-level embedding and the normal/abnormal text representations, while the pixel-level anomaly map M is obtained by aligning patch embeddings with the corresponding text representations. In this way, the optimized anchor produced by the multi-stage prompting process is used for both image-level anomaly detection and pixel-level anomaly localization.

For optimization, we adopt a two-stage training strategy. In the first stage, the model is jointly optimized for image-level anomaly recognition, pixel-level anomaly localization, and prompt diversity:

$$\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{seg}} + \mathcal{L}_{\text{Orth}},$$

where $\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{Focal}} + \mathcal{L}_{\text{Dice}}$. In the second stage, we further refine image-level anomaly discrimination using only $\mathcal{L}_{\text{CE}}$. Detailed formulations of the loss terms are provided in Appendix A.2.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate AnomalyPrompt on seven industrial anomaly detection datasets, including MVTec-AD Bergmann et al. (2019), VisA Zou et al. (2022), BTAD Mishra et al. (2021), KSDD2 Božič et al. (2021), RSDD Gan et al. (2017), DAGM (Wieler & Hahn, 2007), and DTD-Synthetic Aota et al. (2023). We further evaluate cross-domain generalization on four medical anomaly detection benchmarks, including BrainAD Baid et al. (2021); Bakas et al. (2017); Menze et al. (2015), HIS Ehteshami Bejnordi et al. (2017), Endo Hicks et al. (2021), and Kvasir Jha et al. (2020). Following a cross-dataset zero-shot evaluation protocol adopted in prior ZSAD works Zhou et al. (2024); Qu et al. (2025), we use MVTec-AD as the source training dataset for all target datasets except MVTec-AD itself. For MVTec-AD, we instead train on VisA to avoid training and testing on the same dataset.

Evaluation Metrics. For image-level anomaly detection, we report the Area Under the Receiver Operating Characteristic curve (AUROC), Average Precision (AP), and F_1-max. For pixel-level anomaly localization, we report pixel-level AUROC, Per-Region Overlap (PRO) Bergmann et al. (2020), and Average Precision (AP). We additionally report pixel-level F_1-max for the medical benchmarks.

Implementation Details. We instantiate the vision-language backbone with OpenAI CLIP ViT-L/14@336 Radford et al. (2021) and use patch-level features from the 6th, 12th, 18th, and 24th layers for anomaly localization. For hierarchical retrieval, we select the top-$K_g = 8$ global concepts from the image-level embedding and top-$K_l = 4$ local concepts from the last-layer patch embeddings, with top-m pooling ($m = 15$) for local relevance. The generator produces $K = 5$ candidate prompts per image, and the evaluator selects the highest-scoring one as the optimized text anchor. For the LLM, Qwen2.5-1.5B-Instruct is used for all datasets. All experiments are conducted on a single NVIDIA GeForce RTX 4090 GPU (24 GB). Additional implementation details are provided in Appendix A.3.

4.2 Comparison with State-of-the-art Methods

We compare AnomalyPrompt with representative CLIP-based zero-shot anomaly detection methods, including WinCLIP Jeong et al. (2023), AnomalyCLIP Zhou et al. (2024), AdaCLIP Cao et al. (2024), Bayes-PFL Qu et al. (2025), and MRAD Xu et al. (2026).

Table 1: Comparison with state-of-the-art methods on industrial anomaly detection datasets. Best and second-best results are shown in bold and underline, respectively.

(a) Pixel-level ZSAD (AUROC% / PRO%).

Dataset	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
MVTec-AD	(85.1, 64.6)	(91.1, 81.4)	(86.8, 33.8)	(91.8, 87.4)	(**93.0**, 86.8)	(91.9, **88.2**)
VisA	(79.6, 56.8)	(95.5, 87.0)	(95.1, 71.3)	(95.6, 88.9)	(**95.9**, 88.0)	(**95.9**, **90.3**)
BTAD	(71.4, 32.8)	(93.3, 69.3)	(87.7, 17.1)	(93.9, 76.6)	(**95.4**, 72.8)	(94.9, **82.5**)
KSDD2	(97.9, 91.2)	(99.4, 92.7)	(96.1, 70.8)	(**99.6**, 97.6)	(98.9, 95.6)	(**99.6**, **98.8**)
RSDD	(95.1, 75.4)	(99.1, 92.0)	(99.5, 50.5)	(99.6, 98.0)	(99.1, 95.0)	(**99.8**, **98.7**)
DAGM	(83.2, 55.4)	(99.1, 93.6)	(97.0, 40.9)	(99.3, 98.0)	(97.4, 90.3)	(**99.4**, **98.2**)
DTD-Synthetic	(82.5, 55.4)	(97.6, 88.3)	(94.1, 24.9)	(97.8, 94.3)	(98.1, 89.8)	(**98.5**, **95.7**)
Mean	(85.0, 61.7)	(96.4, 86.3)	(93.8, 44.2)	(96.8, 91.5)	(96.8, 88.3)	(**97.1**, **93.2**)

(b) Image-level ZSAD (AUROC% / AP%).

Dataset	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
MVTec-AD	(91.8, 95.1)	(91.5, 96.2)	(92.0, 96.4)	(92.3, 96.7)	(**94.0**, **97.4**)	(93.2, 96.9)
VisA	(78.1, 77.5)	(82.1, 85.4)	(83.0, 84.9)	(87.0, 89.2)	(85.7, 88.3)	(**87.1**, **89.3**)
BTAD	(83.3, 84.1)	(89.1, 91.1)	(91.6, 92.4)	(93.2, **96.5**)	(92.8, 94.2)	(**94.9**, 95.4)
KSDD2	(93.5, 77.9)	(92.1, 77.8)	(95.9, 95.9)	(97.3, **97.9**)	(95.1, 88.9)	(**97.6**, **97.9**)
RSDD	(85.3, 65.3)	(73.5, 55.0)	(89.1, 70.8)	(94.1, 92.3)	(96.2, 97.3)	(**98.2**, **98.5**)
DTD-Synthetic	(95.0, 97.9)	(94.5, 97.7)	(92.8, 97.0)	(95.1, 98.4)	(96.0, 98.4)	(**97.5**, **99.1**)
Mean	(87.8, 83.0)	(87.1, 83.9)	(90.7, 89.6)	(93.2, 95.2)	(93.3, 94.1)	(**94.8**, **96.2**)

Table 2: Cross-domain evaluation on medical anomaly detection benchmarks. Image-level results are reported as (AUROC, AP), while pixel-level results are reported as (AUROC, F1-max, PRO). Best and second-best results are shown in bold and underline, respectively.

Level	Dataset	WinCLIP (CVPR'23)	CLIP-AD (IJCAI'24 W)	AdaCLIP (ECCV'24)	VisualAD(CLIP) (CVPR'26)	AnomalyPrompt (Ours)
Image	BrainAD	(72.5, 91.6)	(72.1, 91.5)	(80.0, 94.1)	(80.8, **94.7**)	(**81.2**, 94.6)
	HIS	(47.1, 40.9)	(44.7, 46.0)	(59.9, 55.8)	(60.1, 55.9)	(**62.0**, **56.6**)
	Mean	(59.8, 66.3)	(58.4, 68.8)	(70.0, 75.0)	(70.5, 75.3)	(**71.6**, **75.6**)
Pixel	BrainAD	(87.6, 21.7, 59.7)	(94.1, 42.9, 74.2)	(95.2, 40.6, 36.4)	(95.2, 46.7, 79.5)	(**96.8**, **51.8**, **83.9**)
	Endo	(68.2, 32.9, 28.3)	(78.1, 41.4, 46.4)	(82.9, 47.9, 55.8)	(84.9, **52.0**, **66.3**)	(**86.1**, **52.0**, 64.7)
	Kvasir	(69.7, 35.9, 24.5)	(73.6, 38.9, 30.6)	(80.3, 47.7, 27.5)	(80.3, 47.8, 50.8)	(**81.4**, **47.9**, **53.8**)
	Mean	(75.2, 30.2, 37.5)	(81.9, 41.1, 50.4)	(86.1, 45.4, 39.9)	(86.8, 48.8, 65.5)	(**88.1**, **50.5**, **67.5**)

Quantitative and Qualitative Comparison. Table 1 reports pixel-level and image-level results across industrial anomaly detection datasets. AnomalyPrompt achieves SOTA or highly competitive performance across diverse industrial scenarios. Compared with static or passively optimized prompts, the proposed multi-stage prompt construction process produces image-conditioned text anchors that better capture both global appearance properties and fine-grained local patterns, leading to stronger zero-shot generalization. Figure 3 compares anomaly maps produced by different methods on representative industrial samples. AnomalyPrompt produces anomaly maps that are more concentrated on true defect regions and less affected by irrelevant background responses. The qualitative results further show that feature-guided prompt generation and evaluator-based prompt selection help construct text anchors that better match image-specific defect characteristics, leading to more robust anomaly localization.

Cross-Domain Generalization. We further evaluate AnomalyPrompt on four medical anomaly detection benchmarks and additionally compare with VisualAD Hou et al. (2026). As shown in Table 2, AnomalyPrompt achieves strong performance in both image-level anomaly recognition and pixel-level localization, demonstrating effective transfer beyond industrial anomaly detection.

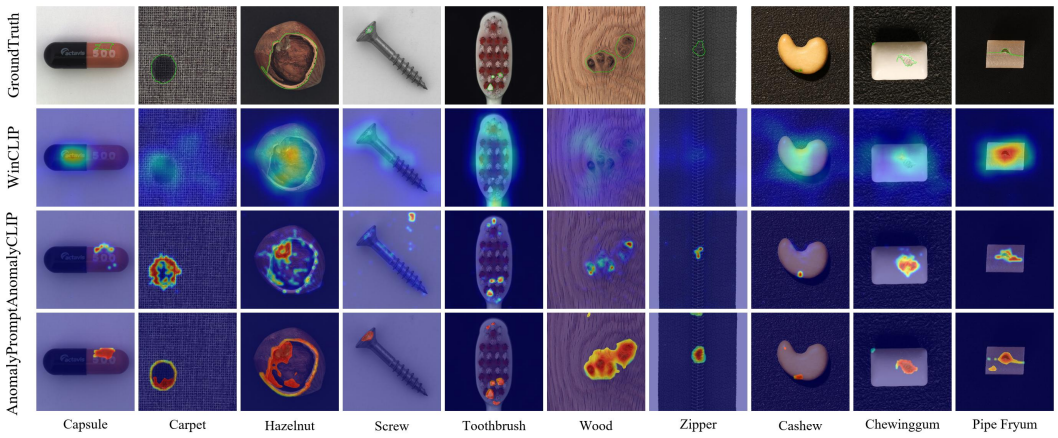


Figure 3: Comparison of anomaly segmentation results between AnomalyPrompt and competitors.

4.3 Ablation Study

Ablation on Main Components. Table 3 compares AnomalyPrompt with degraded variants to assess the contribution of its main components. We introduce a fixed-template variant that retains the same visual retrieval while removing the Prompt Generator and Evaluator (Appendix A.4). The results show that LLM-based prompt generation and evaluation further improve the utilization of retrieved visual cues. Removing LLM-based prompt optimization together with visual features leads to performance degradation, indicating that the proposed multi-stage prompt construction process provides important benefits for constructing effective text anchors. The variant without the prompt bank also shows a substantial decline. Overall, these results verify that LLM-based prompt optimization, visual features, and the downstream prompt structure jointly contribute to the final performance.

Table 3: Main component ablation on VisA.

Setting	Pixel-level		Image-level	
	AUROC	PRO	AUROC	AP
AnomalyPrompt	**95.9**	**90.3**	**87.1**	**89.3**
w/o LLM optimization (fixed template)	95.7	89.6	84.9	87.9
w/o LLM optimization & visual retrieval	95.6	88.9	84.8	87.3
w/o prompt bank	92.1	78.0	71.1	74.7

Ablation on Global and Local Features. Table 4 analyzes the roles of global and local features in the hierarchical visual feature design. Adding local features improves the base model, while further incorporating global features yields additional gains, showing that the two types of features provide complementary guidance for prompt construction. The improvement brought by global features is more evident at the image level, indicating that global features contribute important surface-level and category-related semantics for anomaly recognition. Local features, on the other hand, introduce fine-grained local visual information and help improve localization quality when combined with global features.

Table 4: Ablation study of visual features on MVTec-AD.

Setting	Image-level (AUROC, AP, F1-max)	Pixel-level (AUROC, PRO, AP)
Base (w/o visual features)	(89.8, 95.4, 92.4)	(91.4, 86.3, 45.0)
+ Local features	(90.6, 95.8, 92.8)	(91.8, 88.1, 46.5)
+ Local & Global features	(**93.2**, **96.9**, **93.4**)	(**91.9**, **88.2**, **46.8**)

Ablation on Prompt Evaluator. To assess the effect of the Prompt Evaluator, we compare evaluator-based selection with random selection from the same $K = 5$ candidate set. Figure 4 reports the absolute image-level performance improvement introduced by the prompt evaluator across five industrial datasets. The evaluator consistently improves both AUROC and AP on all evaluated datasets, showing that explicit prompt assessment helps select more reliable optimized text anchors after candidate generation.

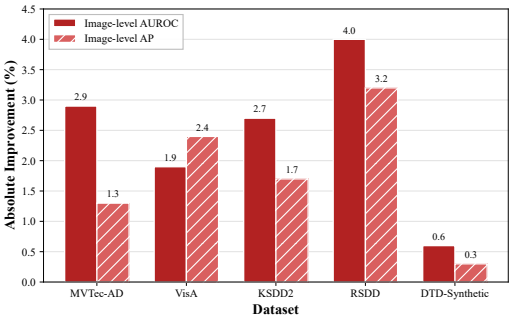


Figure 4: Performance gains of evaluator-based selection over random selection.

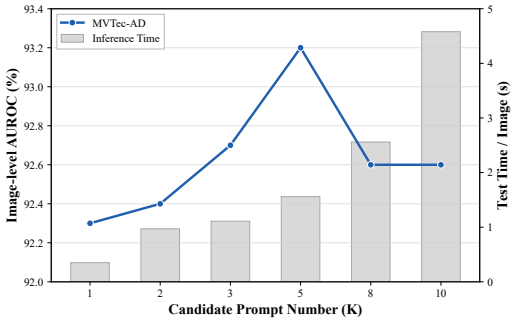


Figure 5: Effect of candidate prompt number K on performance and inference cost.

4.4 Further Analysis

Influence of Candidate Prompt Number. Figure 5 analyzes the effect of the candidate prompt number K on both performance and inference cost. Image-level AUROC improves from $K = 1$ to $K = 5$ on MVTec-AD, but decreases when further increasing K to 8 or 10, while inference time continues to grow substantially. At $K = 5$, AnomalyPrompt achieves 93.2% AUROC on MVTec-AD. We therefore set $K = 5$ as the default, providing a favorable balance between prompt exploration and inference efficiency.

Table 5: Robustness to different LLM scales and families on MVTec-AD and VisA.

	LLM Scale (Qwen2.5-Instruct)				**LLM Family**		
Metric	0.5B	1.5B	3B	7B	Qwen2.5-3B	Llama-3.2-3B	Gemma-3-4B
MVTec AUROC	92.5	93.2	92.7	**93.3**	92.7	92.3	**93.1**
MVTec AP	96.4	**96.9**	96.5	96.7	96.5	96.3	**96.7**
VisA AUROC	84.1	**87.1**	**87.1**	86.4	**87.1**	86.5	87.0
VisA AP	86.4	89.3	**89.5**	89.2	**89.5**	88.7	89.3

Robustness to LLM Scale and Family. Table 5 evaluates AnomalyPrompt with different LLM backbones, where the Prompt Generator and Prompt Evaluator use the same model. Across Qwen2.5-Instruct Yang et al. (2024) models from 0.5B to 7B, the 0.5B model performs relatively worse, while the 1.5B, 3B, and 7B models achieve comparable results, with variations within 0.7 points across all reported metrics. This suggests that once the LLM can effectively use the retrieved visual cues and follow the prompting instructions, increasing the model scale provides only limited benefits. We further evaluate Llama-3.2-3B-Instruct Meta (2024) and Gemma-3-4B-IT Gemma Team (2025) at comparable model scales. Performance across the three LLM families varies by at most 0.8 points, suggesting that the effectiveness of AnomalyPrompt is not specific to a particular LLM family. Overall, these results demonstrate that AnomalyPrompt remains stable across different LLM scales and families, while maintaining strong performance with relatively lightweight language models.

5 Conclusion

In this paper, we presented AnomalyPrompt, a framework for zero-shot anomaly detection that formulates prompt construction as a multi-stage prompting process. By retrieving global and local visual cues to ground LLM-based candidate generation and evaluator-based selection, AnomalyPrompt constructs image-conditioned text anchors for anomaly recognition and localization while preserving the readability and semantic transparency of natural-language prompts. Extensive experiments on seven industrial anomaly detection datasets and four medical benchmarks demonstrate overall state-of-the-art performance in zero-shot anomaly detection, while ablation studies validate the contributions of visual grounding, LLM-based prompt optimization, prompt evaluation, and complementary global and local cues.

AI USE STATEMENT

Generative AI tools were used during manuscript preparation to assist with language polishing, grammatical correction, and LaTeX formatting. All technical content, methodological design, experimental implementation, result analysis, and conclusions were determined and verified by the authors. The use of LLMs within AnomalyPrompt is part of the proposed methodology and is described separately in the main paper and appendix.

REFERENCES

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.

Ujjwal Baid, Satyam Ghodasara, Suyash Mohan, Michel Bilello, Evan Calabrese, Errol Colak, Keyvan Farahani, Jayashree Kalpathy-Cramer, Felipe C. Kitamura, Sarthak Pati, et al. The RSNA-ASNR-MICCAI BraTS 2021 benchmark on brain tumor segmentation and radiogenomic classification. *arXiv preprint arXiv:2107.02314*, 2021.

Spyridon Bakas, Hamed Akbari, Aristeidis Sotiras, Michel Bilello, Martin Rozycki, Justin S. Kirby, John B. Freymann, Keyvan Farahani, and Christos Davatzikos. Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. *Scientific Data*, 4: 170117, 2017.

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, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4183–4192, 2020.

Jakob Božič, Domen Tabernik, and Danijel Skočaj. Mixed supervision for surface-defect detection: From weakly to fully supervised learning. *Computers in Industry*, 129:103459, 2021.

Yunkang Cao, Xiaohao Xu, Chen Sun, Yuqi Cheng, Zongwei Du, Liang Gao, and Weiming Shen. Segment any anomaly without training via hybrid prompt regularization. *arXiv preprint arXiv:2305.10724*, 2023.

Yunkang Cao, Jiangning Zhang, Luca Frittoli, Yuqi Cheng, Weiming Shen, and Giacomo Boracchi. AdaCLIP: Adapting CLIP with hybrid learnable prompts for zero-shot anomaly detection. In *European Conference on Computer Vision*, pp. 55–72. Springer, 2024.

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*, 2023a.

Xuhai Chen, Jiangning Zhang, Guanzhong Tian, Haoyang He, Wuhao Zhang, Yabiao Wang, Chengjie Wang, Yunsheng Wu, and Yong Liu. CLIP-AD: A language-guided staged dual-path model for zero-shot anomaly detection. *CoRR*, abs/2311.00453, 2023b.

Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318(22): 2199–2210, 2017.

Jinrui Gan, Qingyong Li, Jianzhu Wang, and Haomin Yu. A hierarchical extractor-based visual rail surface inspection system. *IEEE Sensors Journal*, 17(23):7935–7944, 2017.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Hao Li, Ming Tang, and Jinqiao Wang. FiLo: Zero-shot anomaly detection by fine-grained description and high-quality localization. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 2041–2049, 2024a.

Zhaopeng Gu, Bingke Zhu, Guibo Zhu, Yingying Chen, Ming Tang, and Jinqiao Wang. AnomalyGPT: Detecting industrial anomalies using large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1932–1940, 2024b.

Steven A. Hicks, Debesh Jha, Vajira Thambawita, Pål Halvorsen, Hugo L. Hammer, and Michael A. Riegler. The EndoTect 2020 challenge: Evaluation and comparison of classification, segmentation and inference time for endoscopy. In *Pattern Recognition. ICPR International Workshops and Challenges*, pp. 263–274. Springer, 2021.

Yanning Hou, Peiyuan Li, Zirui Liu, Yitong Wang, Yanran Ruan, Jianfeng Qiu, and Ke Xu. VisualAD: Language-free zero-shot anomaly detection via vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21346–21356, 2026.

Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar 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.

Debesh Jha, Pia H. Smedsrud, Michael A. Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D. Johansen. Kvasir-SEG: A segmented polyp dataset. In *MultiMedia Modeling*, pp. 451–462. Springer, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Bjoern H. Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Transactions on Medical Imaging*, 34(10): 1993–2024, 2015.

Meta. Llama 3.2: 3B instruct model card. `https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct`, 2024. Accessed: 2026.

Pankaj Mishra, Riccardo Verk, Daniele Fornasier, Claudio Piciarelli, and Gian Luca Foresti. VT-ADL: A vision transformer network for image anomaly detection and localization. In *2021 IEEE 30th International Symposium on Industrial Electronics (ISIE)*, pp. 1–6. IEEE, 2021.

Zhen Qu, Xian Tao, Xinyi Gong, Shichen Qu, Qiyu Chen, Zhengtao Zhang, Xingang Wang, and Guiguang Ding. Bayesian prompt flow learning for zero-shot anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30398–30408, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Matthias Wieler and Tobias Hahn. Weakly supervised learning for industrial optical inspection, 2007. DAGM 2007 Challenge Dataset.

Chaoran Xu, Chengkan Lv, Qiyu Chen, Feng Zhang, and Zhengtao Zhang. MRAD: Zero-shot anomaly detection with memory-driven retrieval. In *International Conference on Learning Representations*, 2026.

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*, pp. 20370–20382, 2025.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16816–16825, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022b.

Qihang Zhou, Guansong Pang, Yu Tian, Shibo He, and Jiming Chen. AnomalyCLIP: Object-agnostic prompt learning for zero-shot anomaly detection. In *The Twelfth International Conference on Learning Representations*, 2024.

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 Implementation Details

A.1 Concept Bank Construction

The hierarchical concept banks are constructed once offline before model training and evaluation. The Global Concept Bank contains 320 concepts describing object-level appearance properties, including material, texture, color tone, reflectance, and finish, while the Local Concept Bank contains 700 concepts describing fine-grained local visual patterns and appearance states, including both abnormality-related and neutral local descriptions. The construction process consists of the following three steps.

Candidate Generation. We use an instruction-following LLM under anomaly-detection-oriented role instructions to generate 400 global candidates and 850 local candidates. The Global Concept Bank targets object-level appearance properties, while the Local Concept Bank focuses on fine-grained local visual patterns and appearance states.

Automatic Cleaning and Finalization. The generated candidates are processed automatically through text normalization, format validation, prohibited-term filtering, exact duplicate removal, and reordered-token duplicate removal. After filtering, the first 320 global and 700 local valid phrases are retained in their original order. No manual rewriting or random sampling is involved in this process.

Encoding and Freezing. The finalized concept phrases are combined with fixed text templates, encoded once by the CLIP text encoder, ℓ_2-normalized, and saved as fixed bank files. The same concept banks are then reused unchanged across all datasets, random seeds, LLM configurations, and experimental runs.

No target-domain annotations or test masks are used during concept bank construction. To facilitate reproducibility, the finalized text banks, encoded bank files, raw candidate outputs, rejection report, construction manifest, and file hashes will be released with the code. Representative examples from both concept banks are shown in Table 6.

Table 6: Representative examples from the hierarchical concept banks.

Concept Bank	Semantic Scope	Representative Concepts
Global	Material / finish	brushed metallic finish; polished smooth finish; coated grainy texture
	Reflectance	glossy waxy appearance; low reflectance finish; highly reflective finish
	Surface texture	matte rough appearance; opaque pitted appearance; sanded smooth surface
	Color tone	dark-toned glossy surface; warm-toned matte surface; cool-toned opaque texture
Local	Structural pattern	crack; hairline crack; broken edge; intact edge; no visible crack
	Surface pattern	linear scratch; stain patch; smudge area; smooth local patch; clean surface area
	Local appearance state	pitting cluster; peeling; dent; bulge; wrinkle; flat local region

A.2 Training Objectives and Loss Functions

As described in the main paper, AnomalyPrompt adopts a two-stage optimization strategy. In the first stage, the model is jointly optimized for image-level anomaly recognition, pixel-level anomaly localization, and prompt diversity:

$$\mathcal{L}_{\text{stage1}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{seg}} + \mathcal{L}_{\text{Orth}}, \tag{1}$$

where

$$\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{Focal}} + \mathcal{L}_{\text{Dice}}. \tag{2}$$

In the second stage, we further refine image-level anomaly discrimination using

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{CE}}. \tag{3}$$

Image-level Classification Loss. For each training iteration, one normal–abnormal prompt pair is randomly selected from the prompt bank and used for image-level supervision. Let $\mathbf{o}_b = [o_b^n, o_b^a]$ denote the corresponding normal and abnormal logits for the b-th image, and let $y_b \in \{0, 1\}$ denote its image-level label. The classification loss is the standard cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{B}\sum_{b=1}^{B} \log \frac{\exp(o_b^{y_b})}{\exp(o_b^n) + \exp(o_b^a)}, \tag{4}$$

where B is the batch size.

Pixel-level Localization Loss. Pixel-level supervision is applied to the anomaly maps extracted from the selected visual layers. For each layer, the same normal–abnormal prompt pair used for image-level classification is selected, and the corresponding two-channel anomaly map is normalized using softmax. The segmentation objective combines Focal loss and Dice loss.

To reduce the influence of the strong imbalance between normal background pixels and anomalous regions, we compute the Focal loss separately over foreground and background pixels:

$$\mathcal{L}_{\text{Focal}} = -\frac{1}{N_+}\sum_{i\in\Omega_+} (1-p_i)^\gamma \log(p_i) - \frac{1}{N_-}\sum_{i\in\Omega_-} (1-p_i)^\gamma \log(p_i), \tag{5}$$

where p_i denotes the predicted probability of the correct class at pixel i, Ω_+ and Ω_- denote the anomalous and normal pixel sets, respectively, and N_+ and N_- are their numbers of pixels. We set $\gamma = 2$.

The Dice loss is applied to the anomalous foreground region:

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2\sum_i y_i \hat{y}_i + \epsilon}{\sum_i y_i + \sum_i \hat{y}_i + \epsilon}, \tag{6}$$

where y_i and $\hat{y}_i$ denote the ground-truth anomaly label and predicted anomaly probability at pixel i, respectively, and $\epsilon = 10^{-5}$. The Focal and Dice losses are accumulated over the selected visual layers.

Orthogonal Regularization. To prevent the learnable prompts from collapsing to highly similar representations, we apply orthogonal regularization separately to the normal and abnormal prompt sets. Let $\bar{\mathbf{t}}_{b,i}^n$ and $\bar{\mathbf{t}}_{b,i}^a$ denote the ℓ_2-normalized normal and abnormal text representations of the i-th prompt for the b-th sample, respectively. The orthogonal loss is defined as

$$\mathcal{L}_{\text{Orth}} = \frac{1}{BN_p^2}\sum_{b=1}^{B}\sum_{i=1}^{N_p}\sum_{\substack{j=1\\ j\neq i}}^{N_p}\left[\left(\left(\bar{\mathbf{t}}_{b,i}^n\right)^\top \bar{\mathbf{t}}_{b,j}^n\right)^2 + \left(\left(\bar{\mathbf{t}}_{b,i}^a\right)^\top \bar{\mathbf{t}}_{b,j}^a\right)^2\right], \tag{7}$$

where B is the batch size and N_p denotes the number of learnable prompts for each normal or abnormal state. The regularization penalizes pairwise similarity within the normal and abnormal prompt sets separately, thereby encouraging prompt diversity without directly enforcing orthogonality between the two states.

A.3 Additional Implementation Details

All input images are resized to 518×518. We train the model for 20 epochs using Adam Kingma & Ba (2014) with an initial learning rate of 1×10^{-4} and a batch size of 32. Patch-level features from the 6th, 12th, 18th, and 24th layers of the CLIP image encoder are used for anomaly localization.

For hierarchical concept retrieval, the numbers of retrieved global and local concepts are set to $K_g = 8$ and $K_l = 4$, respectively. Local concept relevance is aggregated using top-m pooling with $m = 15$. For prompt reasoning, the Prompt Generator produces $K = 5$ candidate prompts for each input image, from which the Prompt Evaluator selects the highest-scoring candidate.

We use Qwen2.5-1.5B-Instruct (Yang et al., 2024) for both the Prompt Generator and Prompt Evaluator across all datasets. LLM inference is performed in bfloat16 precision with a temperature of 0.7, top-p sampling with $p = 0.95$, a maximum generation length of 128 tokens, and a random seed of 333. All experiments are conducted on a single NVIDIA GeForce RTX 4090 GPU with 24 GB of memory.

A.4 Fixed-Template Variant

For the fixed-template variant, we keep the global and local concept retrieval identical to the full model while removing the Prompt Generator and Prompt Evaluator. Up to 8 global cues and 4 local cues are retrieved and inserted into the fixed template "a photo of a {class} with {global cues}, showing {local cues}". Duplicate local cues that already appear among the global cues are removed, and the remaining cues are inserted deterministically according to their retrieval ranks. At least one global and one local cue are retained, while lower-ranked cues are omitted when the text-length limit is reached. No similarity scores, random selection, manual filtering, or LLM processing are involved.

B LLM Instructions for Prompt Optimization

AnomalyPrompt uses an instruction-following LLM for candidate prompt generation and prompt evaluation. The LLM does not take raw images as input. Instead, it receives the normalized class anchor and the retrieved visual cues represented as structured text. In the following templates, `VISUAL_EVIDENCE` denotes the visual cues from the global and local concept banks.

Table 7: LLM instructions used in AnomalyPrompt.

Instruction	Input	Output	Purpose
System prompt	Role and global constraints	Shared behavior specification	Enforces CLIP-friendly, appearance-focused, and cue-grounded prompt design.
Generator prompt	Class anchor and retrieved visual cues	K candidate prompts	Produces diverse caption-style text anchors for the current image.
Evaluator prompt	Class anchor, retrieved visual cues, and candidate prompts	JSON score list	Selects the optimized text anchor through quality-aware scoring.

System Prompt

```
You are an expert in industrial visual anomaly detection and CLIP prompt design.
Your job is to write short, CLIP-friendly English caption-style prompts that serve as
     class anchor phrases.
Describe only visible appearance attributes such as material, texture, color, shape, parts,
      layout, viewpoint, or background when they are supported.
Do not write inspection instructions or quality control procedures.
Avoid anomaly or defect words unless explicitly allowed and strictly supported by evidence.

Avoid extreme normality claims such as perfect or flawless.
If VISUAL_EVIDENCE is provided, only use attributes grounded in that evidence.
If VISUAL_EVIDENCE is missing, stay conservative and class-relevant.
The CLASS_ANCHOR has already been normalized to a natural object name. Do not rename or
     reinterpret it.
```

Generator Prompt

```
Generate EXACTLY {k} English candidate prompts for CLIP-based visual class anchoring in
     industrial anomaly detection.

CLASS_ANCHOR: "{raw_phrase}"
```

```
VISUAL_EVIDENCE (Top-K concepts from the image encoder): {ve}

Task:
- Write short, CLIP-friendly, caption-style prompts that can serve as semantic anchor
    phrases.
- Describe only visible appearance attributes such as material, texture, color, shape,
    layout, parts, or surface finish.
- If VISUAL_EVIDENCE is available, stay grounded to it and do not add unsupported details.
- If VISUAL_EVIDENCE is None or empty, stay conservative and class-relevant.
- Do not write inspection instructions, quality-control procedures, or unsupported anomaly
     claims.
- Use defect-related words only when they are explicitly supported by VISUAL_EVIDENCE.
- Do not use the token X.

Output requirements:
- Return exactly {k} lines
- One prompt per line
- Do not number the lines
- Do not add quotes or any extra text
```

Evaluator Prompt

```
Score the following {len(candidates)} English candidate prompts (0-10) for CLIP-based
    visual class anchoring in industrial anomaly detection.

CLASS_ANCHOR: "{original_phrase}"
VISUAL_EVIDENCE (Top-K concepts from the image encoder): {ve}

Candidate prompts:
{candidates_text}

Important:
- Scores must be discriminative. Do not assign nearly identical scores to all candidates.
- High-scoring prompts should be concise, caption-style, category-consistent, and visually
     grounded.

Scoring criteria (0-10):
1) Faithfulness and correctness (0-4)
- If VISUAL_EVIDENCE is available, the prompt must be consistent with the evidence and
    must not add unsupported details.
- If VISUAL_EVIDENCE is None or empty, the prompt must stay conservative and avoid
    guessing specific colors, backgrounds, lighting, viewpoints, or fine-grained part
    details.

2) CLIP suitability (0-3)
- Short, caption-style phrasing is preferred.
- Avoid long or multi-clause sentences.

3) Clarity and readability (0-3)
- Single sentence
- Easy to read
- No redundant wording

Output requirements:
- Return JSON only: {"scores":[...]}
- The length of scores must be exactly {len(candidates)}
- Do not provide explanations or extra fields.
```

C Additional Quantitative and Qualitative Results

C.1 Category-Level Results

We provide category-level results on MVTec-AD and VisA to complement the aggregate comparisons reported in the main paper. Tables 8–15 report detailed pixel-level and image-level performance across individual object and texture categories. These results provide a fine-grained view of the generalization performance of AnomalyPrompt under the cross-dataset zero-shot setting.

Table 8: Category-level comparison on MVTec-AD in terms of pixel-level AUROC. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
bottle	89.5	90.4	83.8	94.0	92.2	**94.1**
cable	77.0	78.9	**85.6**	78.4	78.4	78.4
capsule	86.9	95.8	86.2	96.4	**97.8**	96.5
carpet	95.4	98.8	94.8	**99.6**	99.4	**99.6**
grid	82.2	97.3	90.6	98.2	**98.4**	98.3
hazelnut	94.3	97.2	**98.7**	97.3	97.5	97.9
leather	96.7	98.6	97.8	**99.6**	99.1	**99.6**
metal_nut	61.0	74.6	55.4	77.5	**79.4**	78.1
pill	80.0	**91.8**	77.5	89.1	91.3	86.7
screw	89.6	97.5	**99.2**	98.4	97.6	98.7
tile	77.6	94.7	83.9	92.8	**97.0**	92.6
toothbrush	86.9	91.9	93.4	93.7	**96.1**	93.9
transistor	74.7	70.8	71.4	67.5	**75.0**	69.8
wood	93.4	96.4	91.2	97.1	**98.0**	96.8
zipper	91.6	91.2	91.8	97.4	96.7	**97.7**
mean	85.1	91.1	86.8	91.8	**93.0**	91.9

Table 9: Category-level comparison on MVTec-AD in terms of pixel-level PRO. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
bottle	76.4	80.8	26.9	87.6	84.7	**87.9**
cable	42.9	64.0	15.2	69.1	69.8	**71.0**
capsule	62.1	87.6	65.7	93.4	**94.7**	**94.7**
carpet	84.1	90.0	19.6	**98.6**	96.5	98.5
grid	57.0	75.4	46.2	93.5	84.8	**93.8**
hazelnut	81.6	92.5	42.3	87.1	92.9	**94.0**
leather	91.1	92.2	55.9	**99.1**	97.8	99.0
metal_nut	31.8	71.1	20.7	74.9	**75.7**	73.8
pill	65.0	88.1	37.0	93.8	92.6	**94.6**
screw	68.5	88.0	75.3	93.0	90.4	**94.4**
tile	51.2	87.4	7.7	90.3	**91.3**	89.5
toothbrush	67.7	88.5	25.6	87.6	89.5	**89.8**
transistor	43.4	58.2	6.7	55.5	**58.7**	55.3
wood	74.1	91.5	58.3	95.1	94.3	**95.2**
zipper	71.7	65.4	3.4	90.1	88.2	**91.3**
mean	64.6	81.4	33.8	87.4	86.8	**88.2**

Table 10: Category-level comparison on MVTec-AD in terms of image-level AUROC. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
bottle	**99.2**	88.7	95.6	95.6	92.7	93.7
cable	86.5	70.3	79.0	81.5	**92.0**	81.3
capsule	72.9	89.5	89.3	92.4	**96.4**	94.6
carpet	**100.0**	99.9	**100.0**	**100.0**	**100.0**	**100.0**
grid	98.8	97.8	99.2	**99.7**	98.9	99.4
hazelnut	93.9	**97.2**	95.5	95.9	93.0	96.8
leather	**100.0**	99.8	**100.0**	**100.0**	**100.0**	**100.0**
metal_nut	**97.1**	92.4	79.9	75.9	78.8	81.0
pill	79.1	81.1	**92.6**	82.2	85.8	90.5
screw	83.3	82.1	83.9	**89.4**	86.7	88.8
tile	**100.0**	**100.0**	99.7	99.3	99.7	98.5
toothbrush	87.5	85.3	95.2	90.5	**97.2**	92.5
transistor	88.0	**93.9**	82.0	84.5	89.1	83.5
wood	**99.4**	96.9	98.5	98.1	99.0	**99.4**
zipper	91.5	98.4	89.4	99.7	**99.9**	97.8
mean	91.8	91.5	92.0	92.3	**94.0**	93.2

Table 11: Category-level comparison on MVTec-AD in terms of image-level AP. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
bottle	98.3	96.8	98.6	**98.7**	97.9	98.0
cable	86.2	81.7	87.3	90.1	**95.4**	89.1
capsule	93.4	97.8	97.8	98.4	**99.2**	98.9
carpet	99.9	99.9	**100.0**	**100.0**	**100.0**	**100.0**
grid	99.8	99.3	99.7	**99.9**	99.6	99.8
hazelnut	96.3	**98.5**	97.5	98.0	96.6	98.4
leather	**100.0**	99.9	**100.0**	**100.0**	**100.0**	**100.0**
metal_nut	97.9	**98.1**	95.6	94.6	95.2	95.7
pill	96.5	95.3	**98.6**	95.9	97.0	98.0
screw	88.4	92.9	93.0	**96.0**	94.5	95.7
tile	99.9	**100.0**	99.9	99.7	99.9	99.4
toothbrush	96.7	93.9	97.9	96.8	**98.9**	97.2
transistor	74.9	**92.1**	83.8	82.3	86.4	83.3
wood	98.8	99.2	99.5	99.7	99.7	**99.8**
zipper	98.9	99.5	97.1	99.9	**100.0**	99.4
mean	95.1	96.2	96.4	96.7	**97.4**	96.9

Table 12: Category-level comparison on VisA in terms of pixel-level AUROC. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
candle	88.9	98.8	98.6	**99.2**	98.9	98.8
capsules	81.6	94.9	96.1	**98.7**	95.4	98.5
cashew	84.7	93.7	97.2	91.6	**97.9**	94.1
chewinggum	93.3	99.2	99.2	99.6	99.5	**99.7**
fryum	88.5	94.6	93.6	**94.7**	94.3	94.5
macaroni1	70.9	98.3	98.8	99.3	98.8	**99.6**
macaroni2	59.3	97.6	98.2	98.3	97.6	**98.6**
pcb1	61.2	**94.0**	90.7	93.8	92.3	92.5
pcb2	71.6	**92.4**	91.3	**92.4**	91.4	92.1
pcb3	85.3	88.3	87.7	88.4	88.7	**90.6**
pcb4	94.4	95.7	94.6	95.3	**96.8**	95.6
pipe_fryum	75.4	**98.2**	95.7	95.4	**98.2**	96.5
mean	79.6	95.5	95.1	95.6	**95.9**	**95.9**

Table 13: Category-level comparison on VisA in terms of pixel-level PRO. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
candle	83.5	**96.5**	71.6	95.6	94.6	95.2
capsules	35.3	78.9	80.3	87.4	79.4	**88.8**
cashew	76.4	91.9	45.6	90.5	93.8	**94.8**
chewinggum	70.4	90.9	53.9	87.4	91.8	**92.8**
fryum	77.4	86.8	55.6	90.4	88.5	**90.9**
macaroni1	34.3	89.7	86.6	96.5	93.3	**97.5**
macaroni2	21.4	83.9	84.8	89.3	85.1	**91.1**
pcb1	26.3	80.7	52.3	**89.3**	84.3	88.1
pcb2	37.2	78.2	77.5	78.3	76.0	**79.6**
pcb3	56.1	76.8	76.2	79.3	**82.1**	80.1
pcb4	80.4	89.4	84.3	88.6	**90.3**	88.0
pipe_fryum	82.3	96.1	86.3	95.3	96.4	**96.8**
mean	56.8	87.0	71.3	88.9	88.0	**90.3**

Table 14: Category-level comparison on VisA in terms of image-level AUROC. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
candle	95.4	80.9	**95.9**	92.3	82.8	93.8
capsules	85.0	82.7	81.1	92.1	91.5	**92.8**
cashew	**92.1**	76.0	89.6	91.3	90.6	91.1
chewinggum	96.5	97.2	**98.5**	97.5	97.6	97.2
fryum	80.3	92.7	89.5	**95.8**	93.5	91.5
macaroni1	76.2	86.7	86.3	90.8	85.6	**92.0**
macaroni2	63.7	72.2	56.7	67.6	**78.4**	69.3
pcb1	73.6	**85.2**	74.0	66.1	83.4	74.2
pcb2	51.2	62.0	71.1	**76.5**	64.8	74.6
pcb3	73.4	61.7	75.2	**79.0**	68.9	74.6
pcb4	79.6	93.9	89.6	96.3	**97.1**	96.4
pipe_fryum	69.7	92.3	88.8	**98.7**	94.3	97.3
mean	78.1	82.1	83.0	87.0	85.7	**87.1**

Table 15: Category-level comparison on VisA in terms of image-level AP. The best results are shown in **bold**.

Method → Category ↓	WinCLIP (CVPR'23)	AnomalyCLIP (ICLR'24)	AdaCLIP (ECCV'24)	Bayes-PFL (CVPR'25)	MRAD (ICLR'26)	AnomalyPrompt (Ours)
candle	95.6	82.6	**96.4**	93.8	86.2	95.0
capsules	80.9	89.4	86.7	95.9	95.0	**96.4**
cashew	95.2	89.3	95.4	**96.3**	96.1	95.9
chewinggum	98.8	98.8	99.4	**99.5**	99.1	98.9
fryum	92.5	96.6	95.1	**98.4**	97.0	96.3
macaroni1	64.5	85.5	85.0	92.2	85.7	**93.1**
macaroni2	65.2	70.8	54.3	70.4	**78.0**	68.1
pcb1	74.6	**86.7**	73.5	70.0	84.9	78.8
pcb2	44.2	64.4	71.6	77.3	67.9	**78.8**
pcb3	66.2	69.4	77.9	**81.1**	74.9	77.5
pcb4	70.1	94.3	89.8	96.1	**96.8**	96.2
pipe_fryum	82.1	96.3	93.9	**99.4**	97.1	98.8
mean	77.5	85.4	84.9	89.2	88.3	**89.3**

C.2 Prompt Behavior on Normal Images

We further analyze the behavior of image-conditioned prompt optimization on normal images using all normal test samples from MVTec-AD and VisA. For each image, we inspect the generated candidate pool and the final prompt selected by the Prompt Evaluator, and measure the frequency of affirmative explicit-defect claims.

On MVTec-AD, explicit-defect claims appear in 6.3% of the generated candidates, while this rate decreases to 4.3% after evaluator-based selection. On VisA, the corresponding rate decreases from 2.9% to 2.1%. Consequently, the final selected prompts contain no affirmative explicit-defect claim for 95.7% of the 467 normal MVTec-AD images and 97.9% of the 962 normal VisA images. These results show that the Prompt Evaluator not only selects prompts with stronger semantic alignment, but also tends to suppress defect-oriented candidates when they are less consistent with the retrieved visual evidence.

C.3 Additional Segmentation Visualizations

We provide additional qualitative segmentation results across representative categories from MVTec-AD, VisA, BTAD, DAGM, DTD-Synthetic, and KSDD2. For each example, the ground-truth anomaly contour is overlaid on the input image, and the corresponding anomaly map predicted by AnomalyPrompt is shown below. These visualizations complement the qualitative comparison in the main paper and illustrate the localization behavior of AnomalyPrompt across diverse object, texture, and defect patterns.

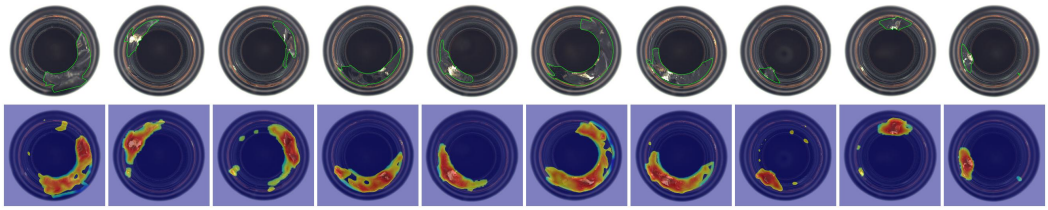


Figure 6: Visualization of segmentation results for the *bottle* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.



Figure 7: Visualization of segmentation results for the *capsule* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

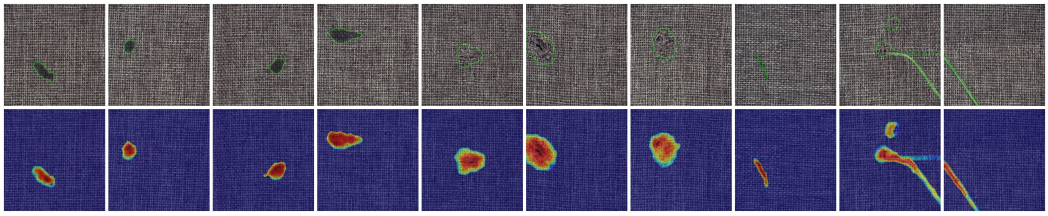


Figure 8: Visualization of segmentation results for the *carpet* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

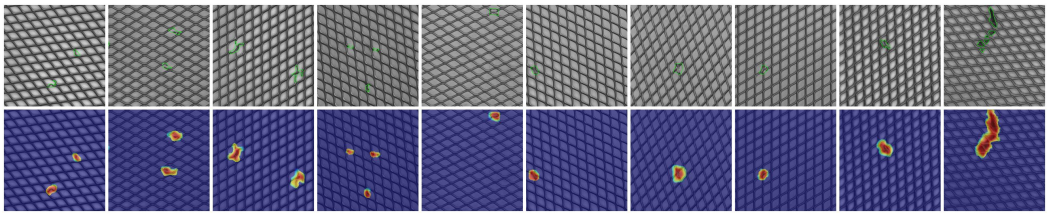


Figure 9: Visualization of segmentation results for the *grid* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

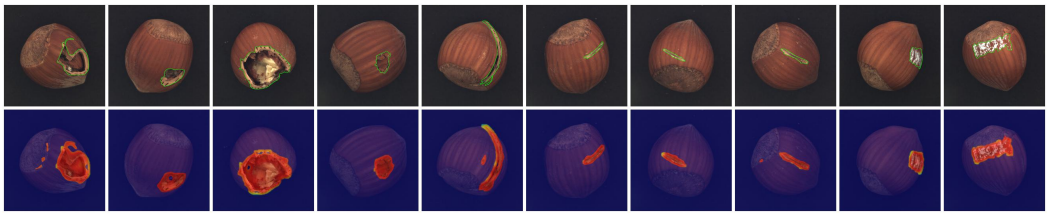


Figure 10: Visualization of segmentation results for the *hazelnut* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

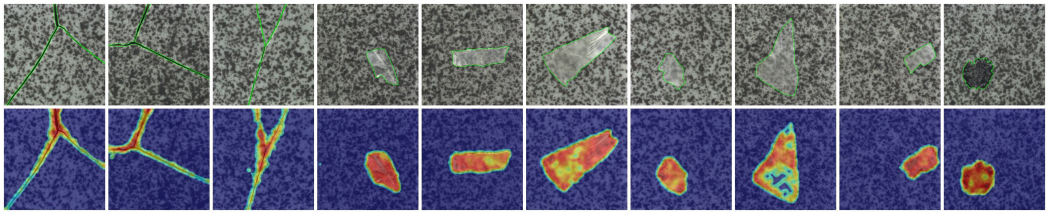


Figure 11: Visualization of segmentation results for the *tile* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

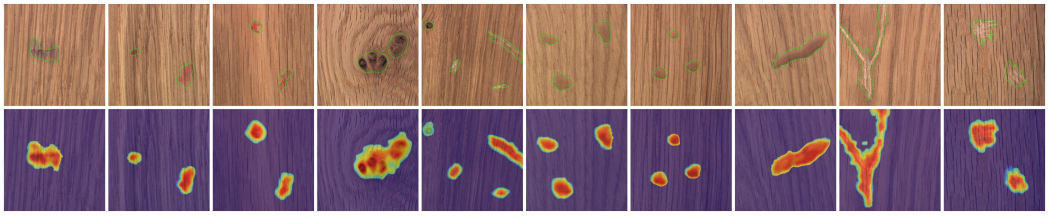


Figure 12: Visualization of segmentation results for the *wood* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

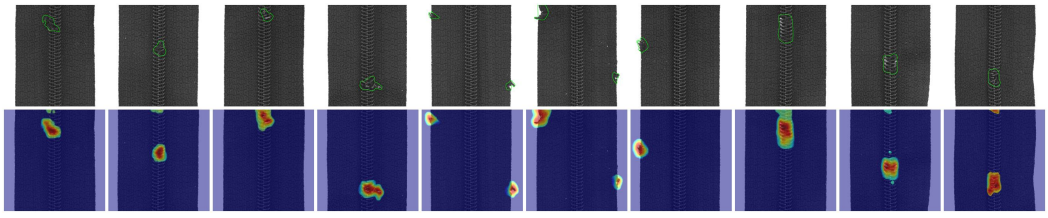


Figure 13: Visualization of segmentation results for the *zipper* category on MVTec-AD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

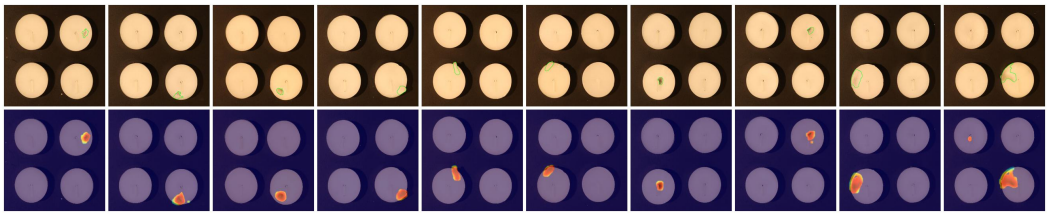


Figure 14: Visualization of segmentation results for the *candle* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

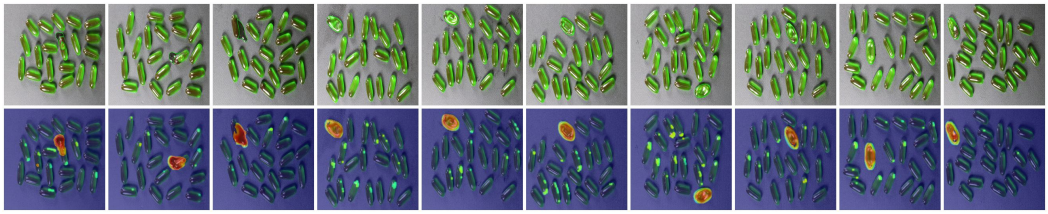


Figure 15: Visualization of segmentation results for the *capsules* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

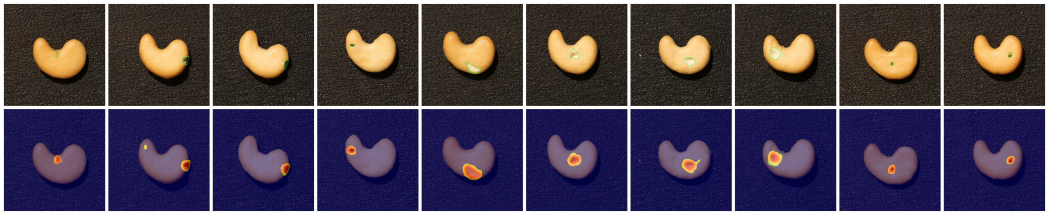


Figure 16: Visualization of segmentation results for the *cashew* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

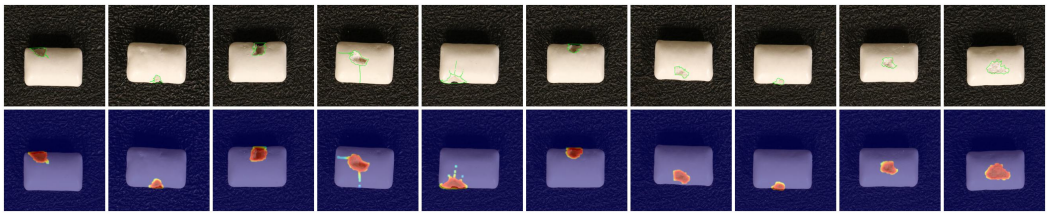


Figure 17: Visualization of segmentation results for the *chewing gum* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

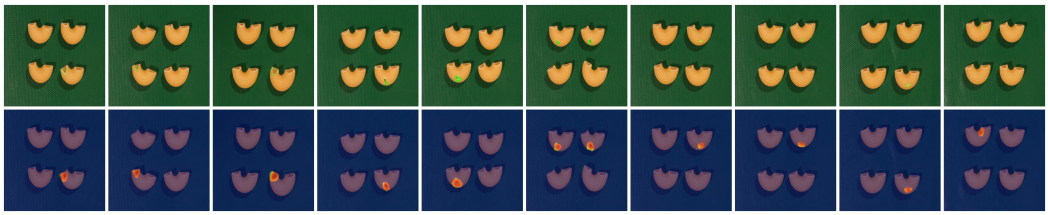


Figure 18: Visualization of segmentation results for the *macaroni1* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

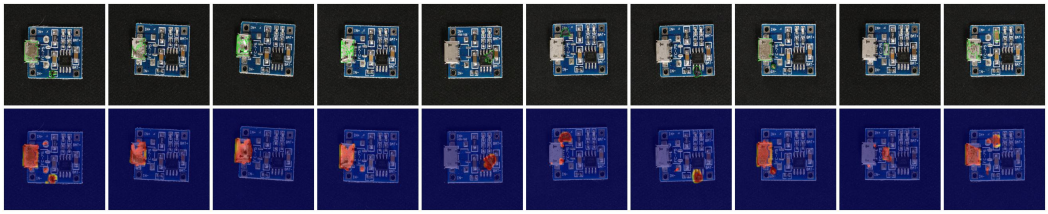


Figure 19: Visualization of segmentation results for the *pcb* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

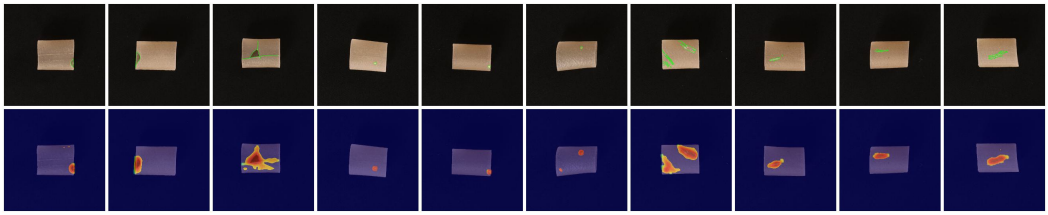


Figure 20: Visualization of segmentation results for the *pipe fryum* category on VisA. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

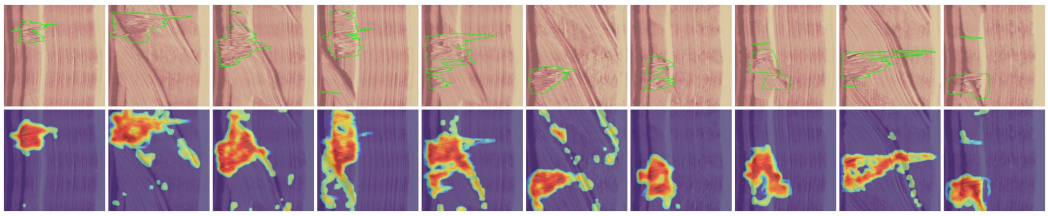


Figure 21: Visualization of segmentation results for the *wood* category on BTAD. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

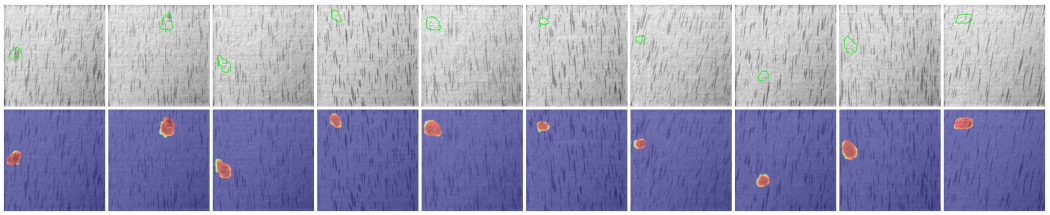


Figure 22: Visualization of segmentation results for the *fabric* category on DAGM. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

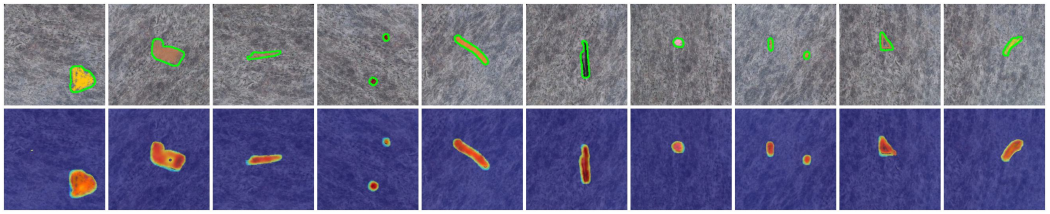


Figure 23: Visualization of segmentation results for the *marbled* category on DTD-Synthetic. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

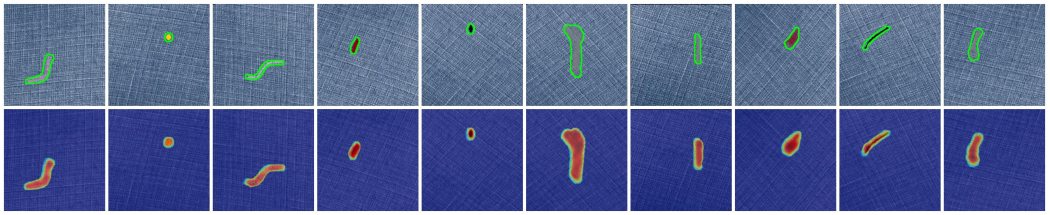


Figure 24: Visualization of segmentation results for the *woven* category on DTD-Synthetic. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.

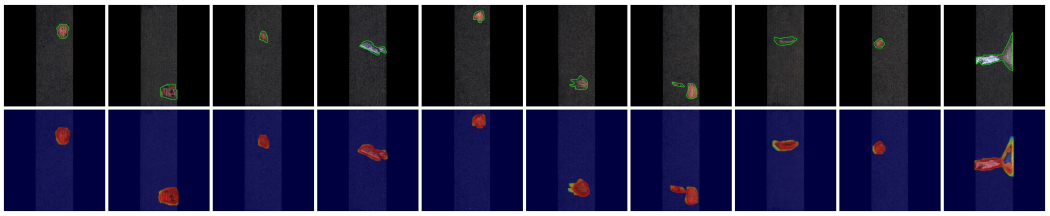


Figure 25: Visualization of segmentation results for the *metal* category on KSDD2. Ground-truth contours are overlaid on the input images in the upper row, and predicted anomaly maps are shown in the lower row.