RECAL-CLIP: PATCH-TOKEN RECALIBRATION FOR CROSS-DOMAIN ZERO-SHOT ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-domain zero-shot anomaly detection aims to identify and localize abnormal regions in unseen target domains without using target-domain anomalous samples, labels, or masks for optimization. Existing CLIP-based methods benefit from vision-language priors, but their patch-level representations are often not calibrated for context-dependent defects: a patch may look ambiguous in isolation, while its abnormality becomes clear only when compared with surrounding normal patterns. This limitation is particularly problematic for industrial inspection, where defects can be small, low-contrast, spatially fragmented, or visually defined by deviations from normal texture. We propose **ReCal-CLIP**, a patch-token recalibration framework that addresses this issue before anomaly scoring. The core idea is to recalibrate image-side CLIP tokens by routing anomaly-suspect evidence, contrasting it with normal-context cues, and injecting the resulting update through confidence-gated residual fusion. With the recalibrated tokens, ReCal-CLIP uses a lightweight token-space consistency step and a training-only Sampled Discriminative Patch Alignment (SDPA) objective that applies a mask-guided similarity-margin constraint during source-domain training. Experiments on industrial benchmarks, with supplementary medical-domain evaluation, show competitive image-level detection and pixel-level localization performance.

1 INTRODUCTION

Industrial anomaly detection aims to identify defective samples and localize abnormal regions in visual inspection scenarios Tao et al. (2022); Bergmann et al. (2019). In many production lines, models must detect image-level abnormality and pixel-level defect regions in unseen target domains, where target-domain anomalous samples, labels, and masks are unavailable for optimization. This requirement motivates cross-domain zero-shot methods that generalize across categories, domains, and defect types Zhou et al. (2022); Wang et al. (2022).

As illustrated in Figure 1, vision-language models provide a promising foundation for this problem. CLIP Radford et al. (2021) enables anomaly scoring through normal and anomalous language descriptions, reducing dependence on category-specific classifiers or dense defect annotations. However, directly applying pretrained CLIP patch tokens remains insufficient because CLIP is mainly optimized for image-level semantic alignment rather than dense industrial localization Rao et al. (2022). Small, low-contrast, or texture-like defects can therefore lead to fragmented patch-wise responses.

The central problem is that patch-level anomaly evidence is context-dependent. A suspicious patch should not be judged only by isolated similarity to anomalous text; it should also be compared with the normal patterns that define local visual regularity. Thus, the key requirement is not merely to add more prompts or smooth the final heatmap, but to recalibrate image-side patch tokens before scoring while preserving the transferable CLIP feature space.

In this paper, we propose **ReCal-CLIP**, a patch-token recalibration framework for cross-domain zero-shot anomaly detection. Anomaly-Suspect Routing estimates which patch tokens require stronger context-aware refinement. Then, *Hybrid Recalibration* preserves raw instance information, constructs normal-context calibration cues, and models local defect structure. These cues are fused as a confidence-gated residual update to the original patch tokens, keeping the pretrained representation as an anchor within the same scoring pathway rather than forming separate prediction

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.

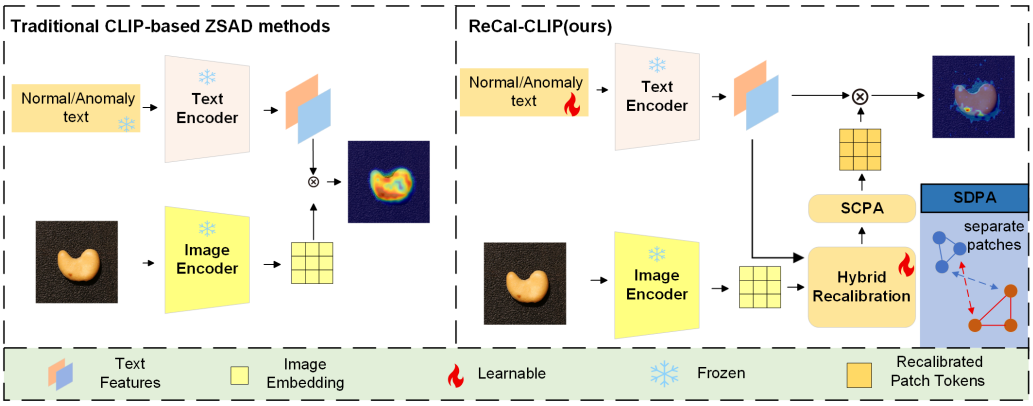


Figure 1: Motivation of ReCal-CLIP. Left: conventional CLIP-based zero-shot anomaly detection uses direct image–text similarity for patch responses. Right: ReCal-CLIP recalibrates patch tokens with normal-context evidence before token-space aggregation, while training-only SDPA adds a source-mask-guided similarity-margin constraint.

branches. A lightweight token-space aggregation step reduces isolated score fluctuations, while SDPA regularizes sampled source-domain patch similarities using mask-derived normal/anomalous sets. Together, these steps complement learnable normal/anomalous text prototypes and remain applicable to unseen target domains.

Our contributions are summarized as follows:

- We formulate ReCal-CLIP as an image-side patch-token recalibration framework for cross-domain zero-shot anomaly detection, targeting the context-dependent ambiguity of raw CLIP patch responses.
- We introduce Hybrid Recalibration, which refines patch tokens through anomaly-suspect routing, three context-calibration branches—Identity (IDE), Suspect-Evidence (SUS), and Structure (STR)—and confidence-gated residual fusion, while preserving the pretrained feature space.
- We integrate lightweight token-space aggregation and training-only SDPA into the recalibration pipeline; SDPA imposes a sampled similarity-margin constraint defined by source-domain anomaly masks.

2 Related Work

Cross-Domain Zero-Shot Industrial Anomaly Detection. Industrial anomaly detection has been studied through reconstruction, self-supervised transformation, feature embedding, density estimation, and memory-based comparison Tao et al. (2022); Zuo et al. (2024); Fang et al. (2023); Im et al. (2025); Li et al. (2021); Defard et al. (2021); Roth et al. (2022); Yu et al. (2021). These methods work well with target-category data but require category-specific normal images and generalize poorly to unseen objects, domains, or defects, motivating cross-domain zero-shot detection with vision-language models such as CLIP Radford et al. (2021).

CLIP-based approaches improve normal/anomalous descriptions, prompt learning, patch-text matching, and multi-level features Jeong et al. (2023); Chen et al. (2023); Zhou et al. (2024); Cao et al. (2024); Chen et al. (2024); Liu et al. (2025); Zhu et al. (2025); Hou et al. (2026); Guo et al. (2026). ReCal-CLIP instead targets the image-side patch tokens responsible for dense localization errors under a target-domain zero-shot protocol.

Patch-Level Feature Refinement. Dense localization requires visual tokens that are semantically aligned with normal/anomalous textual states, locally discriminative, and spatially stable. Prior work improves dense prediction through feature refinement, context modeling, multi-scale representation, and local aggregation Yu & Koltun (2015); Huang et al. (2021); Xu et al. (2022); Yang et al. (2025);

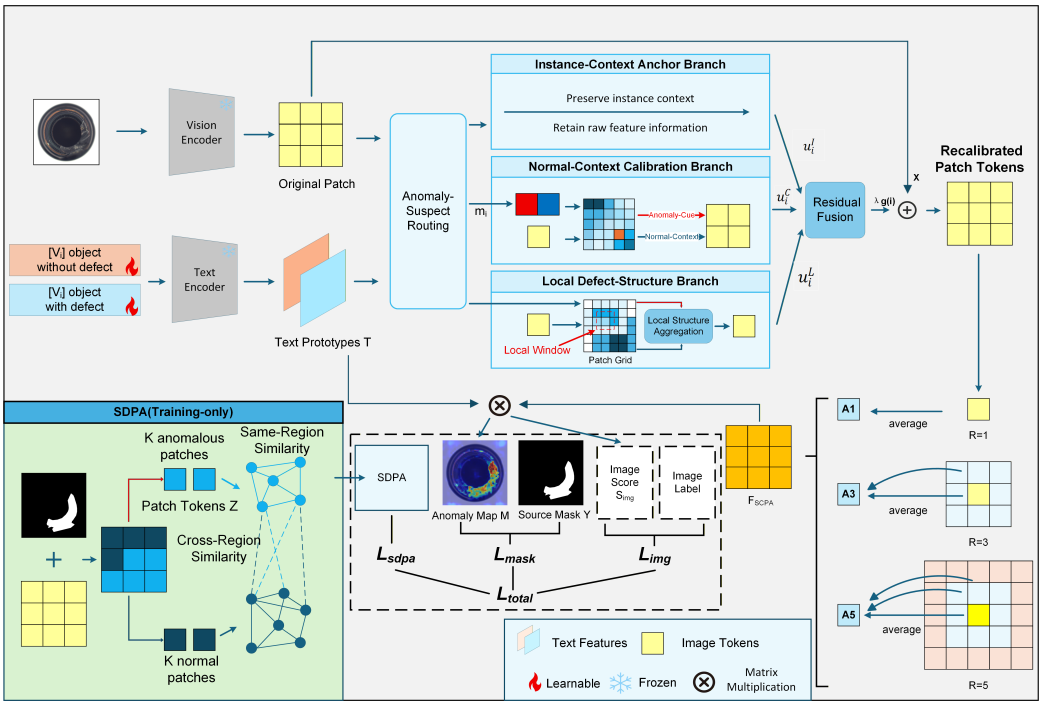


Figure 2: Overview of ReCal-CLIP. Frozen CLIP encoders produce image tokens and normal/anomalous text prototypes. Anomaly-Suspect Routing guides three-branch Hybrid Recalibration (IDE, SUS, and STR), followed by residual fusion, SCPA, and text-prototype scoring for image-level detection and pixel-level localization. SDPA is training-only.

Sultana et al. (2019), which enhances object boundaries, suppresses irrelevant responses, and aggregates neighboring regions. For industrial defects this matters because local abnormality is defined by inconsistency with nearby normal context rather than an isolated semantic class.

Neighborhoods, region cues, multi-scale features, and discriminative constraints improve dense responses Yu & Koltun (2015); Liu et al. (2021); Zhang et al. (2020); Huang et al. (2021); Sun et al. (2025). For CLIP, excessive visual adaptation risks transfer, while heatmap smoothing does not correct patch features. ReCal-CLIP instead recalibrates suspect tokens against normal context, aggregates tokens locally, and uses a training-only sampled similarity-margin constraint (SDPA).

Unlike prompt-centric methods that mainly improve text-side state descriptions, ReCal-CLIP targets the image-side patch representation where localization errors emerge, complementing CLIP's language prior: learnable normal/anomalous text prototypes define the scoring anchors, while recalibrated tokens provide cleaner local evidence. Unlike generic visual adapters, the recalibration follows the specific ambiguity of localization: which patches are suspect, what normal context to compare against, and how much residual correction to inject.

3 METHOD

3.1 OVERVIEW

For image x, frozen CLIP extracts image token $\mathbf{x}_0$ and $N = H_p W_p$ patch tokens $\mathbf{x}_{1:N}$, forming $\mathbf{X} \in \mathbb{R}^{(N+1)\times D}$; its frozen text encoder maps learnable prompts to $\mathbf{T} = [\mathbf{t}_n; \mathbf{t}_a] \in \mathbb{R}^{2\times D}$. As in Figure 2, routing weights $\{m_i\}$ guide Hybrid Recalibration to produce $\mathbf{X}' = \{\mathbf{x}'_i\}_{i=1}^{N}$; SCPA aggregates neighboring tokens before text matching yields anomaly map $\mathbf{M}$ and image score s_{img}. Source training uses image-level, pixel-level, and SDPA objectives; inference uses neither SDPA nor source labels or masks.

3.2 LEARNABLE TEXT PROMPTS

Following the learnable text-prompt setting in CLIP-based zero-shot anomaly detection Zhou et al. (2024); Liu et al. (2025), ReCal-CLIP uses normal and anomalous prompts to construct text prototypes:

$$\begin{aligned} P_n &= [V_1][V_2]\dots[V_E] \text{ object without defect,} \\ P_a &= [W_1][W_2]\dots[W_E] \text{ object with defect.} \end{aligned} \tag{1}$$

The frozen text transformer fuses each prompt's token features by mean, learned-attention, and EOT pooling before normalization (Appendix A). Denoting this encoder by $E_t(\cdot)$, the prototypes are

$$\begin{aligned} \mathbf{t}_n &= \frac{E_t(P_n)}{\|E_t(P_n)\|_2}, \qquad & \mathbf{t}_a &= \frac{E_t(P_a)}{\|E_t(P_a)\|_2}, \\ \mathbf{T} &= [\mathbf{t}_n; \mathbf{t}_a]. \end{aligned} \tag{2}$$

Source-domain prompt learning yields text prototypes for image- and patch-level visual similarities.

3.3 HYBRID RECALIBRATION

Hybrid Recalibration produces context-aware patch tokens by preserving CLIP representations, comparing suspect evidence with normal context, and incorporating local structure before residual correction. Its D-dimensional branches are IDE (Identity), SUS (Suspect-Evidence), and STR (Structure).

3.3.1 ANOMALY-SUSPECT ROUTING

For each patch token $\mathbf{x}_i$, complementary token-context discrepancies, rather than a segmentation target, define routing coefficient m_i. Let $\bar{\mathbf{x}}_i^{\mathrm{loc}}$ and $\bar{\mathbf{x}}$ be the one-dimensional local and image-level patch means. The four cues are

$$\begin{aligned} \bar{\mathbf{x}}_i^{\mathrm{loc}} &= \mathrm{AvgPool}_{1\mathrm{D},3}(\mathbf{x})_i, \qquad & \bar{\mathbf{x}} &= \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}_j, \\ s_i^{loc} &= \|\mathbf{x}_i - \bar{\mathbf{x}}_i^{\mathrm{loc}}\|_2, \qquad & s_i^{ctx} &= \|\mathbf{x}_i - \bar{\mathbf{x}}\|_2, \\ s_i^{img} &= 1 - \frac{\mathbf{x}_i^\top \mathbf{x}_0}{\|\mathbf{x}_i\|_2\|\mathbf{x}_0\|_2}, \qquad & s_i^{lay} &= \|\mathbf{x}_i^{(\ell)} - \mathbf{x}_i^{(\ell-1)}\|_2. \end{aligned} \tag{3}$$

where $\mathrm{AvgPool}_{1\mathrm{D},3}$ is stride-one, width-three average pooling; the layer cue is zero if the previous-layer token is unavailable. After per-image min–max normalization $\widetilde{(\cdot)}$, fixed weights combine the cues:

$$\begin{aligned} r_i &= 0.30\widetilde{s}_i^{loc} + 0.30\widetilde{s}_i^{ctx} + 0.30\widetilde{s}_i^{img} + 0.10\widetilde{s}_i^{lay}, \\ \bar{r}_i &= \mathrm{AvgPool}_{1\mathrm{D},5}(r)_i, \\ m_i &= \begin{cases} 1.0, & \bar{r}_i > 0.5, \\ 0.5, & \bar{r}_i \le 0.5. \end{cases} \end{aligned} \tag{4}$$

Width-five pooling slides with reflected padding. Thus, $m_i \in \{0.5, 1.0\}$ is input-dependent but fixed, not a learned sigmoid output. The cues are relative discrepancies within the input's CLIP sequence; their weights and threshold are dataset-invariant, with no optimized routing parameters. Hence m_i guides context gathering rather than acting as a mask, segmentation prediction, or dataset-specific rule.

3.3.2 THREE-BRANCH CONTEXT CALIBRATION

The *Identity* branch (IDE) sets $\mathbf{u}_i^I = \mathbf{x}_i$, anchoring the instance to the pretrained representation. The *Suspect-Evidence* branch (SUS, output $\mathbf{u}_i^C$) constructs suspect evidence and a normal-context reference from routing coefficients and each query's image-token similarity. Let

$$c_i = \frac{\mathbf{x}_i^\top \mathbf{x}_0}{\|\mathbf{x}_i\|_2\|\mathbf{x}_0\|_2}. \tag{5}$$

The suspect and normal attention logits are

$$\begin{aligned} h^a_{ij} &= (1 - c_i)m_i m_j, \\ h^n_{ij} &= c_i(1 - m_i)(1 - m_j), \end{aligned} \tag{6}$$

and are row-normalized over key index j:

$$w^a_{ij} = \frac{\exp(h^a_{ij})}{\sum_{k=1}^{N} \exp(h^a_{ik})}, \qquad w^n_{ij} = \frac{\exp(h^n_{ij})}{\sum_{k=1}^{N} \exp(h^n_{ik})}. \tag{7}$$

The routed summaries and SUS output are

$$\begin{aligned} \mathbf{z}^a_i &= \sum_{j=1}^{N} w^a_{ij}\mathbf{x}_j, & \mathbf{z}^n_i &= \sum_{j=1}^{N} w^n_{ij}\mathbf{x}_j, \\ \mathbf{u}^C_i &= 0.6\,[m_i\mathbf{z}^a_i + (1 - m_i)\mathbf{z}^n_i] + 0.4\mathbf{x}_i. \end{aligned} \tag{8}$$

Thus, image-inconsistent tokens contribute stronger suspect evidence, while image-consistent tokens support the normal path. The *Structure* branch (STR, output $\mathbf{u}^L_i$) applies the same routing on a fixed 5×5 patch neighborhood, separately aggregates suspect and stable-context features, and adds the summaries; its stability, richness, normalization, and padding rules are given in Appendix B. IDE preserves instance information, SUS compares suspect and normal context, and STR captures local defect structure. The outputs remain separate recalibration views rather than being concatenated.

3.3.3 Branch-wise Confidence-Gated Residual Fusion

Residual fusion is performed independently for each branch and selected visual layer. Let $v \in \{I, C, L\}$ index IDE, SUS, and STR, and let $\mathbf{y}_{i,v}$ be the branch output before fusion. Each branch uses the same type of context adapter and is compared with the original layer token $\mathbf{x}_i$:

$$\begin{aligned} \mathbf{e}_{i,v} &= A_{\text{adapt}}(\mathbf{y}_{i,v}), \\ \Delta\mathbf{x}_{i,v} &= \mathbf{e}_{i,v} - \mathbf{x}_i, \\ \mathbf{x}'_{i,v} &= \mathbf{x}_i + \lambda_v g_{i,v}\Delta\mathbf{x}_{i,v}. \end{aligned} \tag{9}$$

A shared confidence-gate module is evaluated separately for each branch and controls how much residual information is injected:

$$g_{i,v} = G(\mathbf{x}_i), \quad 0 \le g_{i,v} \le 1. \tag{10}$$

Adapter and gate parameters are shared across branches, with separate scales λ_v; the reported IDE/SUS/STR scales are $0.25/0.25/0.35$. The gate acts on the original layer token, not a concatenation: no operation of the form $A_\Delta([\mathbf{u}^I_i; \mathbf{u}^C_i; \mathbf{u}^L_i])$ is used. Hybrid Recalibration produces separate views $\mathbf{x}'_{i,v}$ that remain in the CLIP-space scoring pathway and are later scored and aggregated separately; m_i guides context gathering and $g_{i,v}$ controls branch-wise residual injection.

3.4 Spatially Coherent Patch Aggregation

After Hybrid Recalibration, the main visual correction has already been applied to the patch tokens. We use SCPA as a lightweight parameter-free token-space aggregation step to reduce isolated patch fluctuations before text-prototype scoring. It does not introduce a separate trainable prediction head or target-domain adaptation; it only aggregates nearby recalibrated tokens on the patch grid. Let $\mathbf{P}' = \{\mathbf{p}'_{u,v}\} \in \mathbb{R}^{H_p \times W_p \times D}$ be the two-dimensional grid reshaped from $\mathbf{X}'$. For a neighborhood size r, SCPA computes

$$\mathbf{f}^r_{u,v} = \frac{1}{|\Omega_r(u, v)|} \sum_{(a,b)\in\Omega_r(u,v)} \mathbf{p}'_{a,b}, \tag{11}$$

where $\Omega_r(u, v)$ is the valid $r \times r$ neighborhood centered at (u, v). The aggregated grid at scale r is denoted as $\mathbf{F}^r = \{\mathbf{f}^r_{u,v}\}_{u,v}$. ReCal-CLIP uses $r \in \{1, 3, 5\}$ and fuses the three scales as

$$\mathbf{F}_{\text{SCPA}} = 0.5\mathbf{F}^1 + 0.3\mathbf{F}^3 + 0.2\mathbf{F}^5. \tag{12}$$

The fixed weights follow a simple fine-to-coarse prior: the original recalibrated token receives the largest weight to preserve local defect evidence, while the 3×3 and 5×5 neighborhoods provide progressively weaker context support. We keep this step parameter-free to avoid introducing another source-trained spatial fusion module. The same weights are used for all datasets and ablation variants.

3.5 SAMPLED DISCRIMINATIVE PATCH ALIGNMENT

SDPA is a training-only regularizer defined on source-domain mask-guided patch groups. It adjusts the relative cosine-similarity margin between sampled cross-region and same-region pairs; this operational interpretation does not assume semantic purification, causal deconfounding, or complete removal of domain bias. Let $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^{N}$ denote projected recalibrated patch tokens before SCPA, and let $\bar{\mathbf{z}}_i = \mathbf{z}_i / \|\mathbf{z}_i\|_2$. The resized source mask defines normal and anomalous patch sets

$$\mathcal{I}_n = \{i \mid y_i = 0\}, \quad \mathcal{I}_a = \{i \mid y_i = 1\}. \tag{13}$$

If either set is empty, this image contributes zero to SDPA while image-level and pixel-level losses are still computed. Otherwise, SDPA samples $K = \min(K_0, |\mathcal{I}_n|, |\mathcal{I}_a|)$ normal and anomalous patches. The cross-region and same-region similarities are

$$s_{na} = \frac{1}{K^2} \sum_{p=1}^{K} \sum_{q=1}^{K} \langle \bar{\mathbf{z}}_p^n, \bar{\mathbf{z}}_q^a \rangle, \tag{14}$$

$$s_{same} = \frac{1}{2K^2} \sum_{p=1}^{K} \sum_{q=1}^{K} \left(\langle \bar{\mathbf{z}}_p^n, \bar{\mathbf{z}}_q^n \rangle + \langle \bar{\mathbf{z}}_p^a, \bar{\mathbf{z}}_q^a \rangle \right). \tag{15}$$

The SDPA loss is

$$\mathcal{L}_{\text{SDPA}} = \max(0, s_{na} - s_{same} + \mu), \tag{16}$$

where μ is the margin. Thus, SDPA only penalizes cases in which the sampled cross-region similarity is not sufficiently below the sampled same-region similarity. The loss is discarded during inference, and we interpret its empirical effect through the ablation and sensitivity results rather than as evidence of a stronger feature-space mechanism.

3.6 ANOMALY PREDICTION AND TRAINING OBJECTIVE

Let $\widetilde{\mathbf{F}}$ denote the sequence containing the image-level token and the patch tokens produced by Hybrid Recalibration. Following the implementation used for all reported results, each visual token is first row-wise ℓ_2 normalized and SCPA is then applied to the normalized sequence. The normalized text prototypes are compared with this SCPA output by a scaled dot product:

$$\widehat{\mathbf{F}} = \text{Norm}(\widetilde{\mathbf{F}}), \qquad \mathbf{F}^{score} = \text{SCPA}(\widehat{\mathbf{F}}), \tag{17}$$

$$\mathbf{S} = \mathbf{F}^{score} \text{Norm}(\mathbf{T})^{\top} / \tau_s, \quad \mathbf{P} = \text{softmax}(\mathbf{S}). \tag{18}$$

No second visual normalization is applied after SCPA. Consequently, although the inputs to SCPA and the text prototypes are normalized, $\mathbf{S}$ is a dot-product score rather than a strict cosine similarity of the aggregated visual tokens. The anomaly map and image-level anomaly score are

$$\mathbf{M} = \text{Reshape}(\{\mathbf{P}_{i,a}\}_{i=1}^{N}), \quad s_{img} = \mathbf{P}_{0,a}. \tag{19}$$

During source-domain training, the objective is

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{focal}(s_{img}, y) + \mathcal{L}_{reg}^{it} + \mathcal{L}_{focal}(\mathbf{M}, \mathbf{Y}) \\ + \beta \mathcal{L}_{dice}(\mathbf{M}, \mathbf{Y}) + \lambda_{sdpa} \mathcal{L}_{\text{SDPA}}. \end{aligned} \tag{20}$$

At inference time, all training losses, source-domain masks, and SDPA sampling are removed; ReCal-CLIP only uses Hybrid Recalibration, parameter-free SCPA, and text-prototype scoring.

4 EXPERIMENTS

4.1 EXPERIMENT SETTINGS

Datasets and protocol. We evaluate ReCal-CLIP under a target-domain zero-shot protocol. Industrial benchmarks include MVTec AD Bergmann et al. (2019), VisA Zou et al. (2022), MPDD Jezek et al. (2021), BTAD Mishra et al. (2021), and DTD-Synthetic Cimpoi et al. (2014); medical benchmarks include HeadCT Chilamkurthy et al. (2018), Br35H Hamada (2020), ISIC Codella et al. (2019), CVC-ClinicDB Bernal et al. (2015), CVC-ColonDB Bernal et al. (2012), and Kvasir Jha

Table 1: Unified comparison on industrial and medical anomaly detection benchmarks. Baseline entries follow their cited sources. Results are reported as (AUROC, AP) for image-level detection and (AUROC, AUPRO) for localization; best and second-best results are **bold** and underlined.

Metric	Dataset	AnomalyCLIP ICLR'24	AdaCLIP ECCV'24	LECLIP TIM'25	FAPrompt ICCV'25	VisualAD CVPR'26	CHSA-CLIP ESWA'26	ReCal-CLIP Ours
Industrial benchmarks								
Image-level (AUROC, AP)	MVTec AD	(91.5, 96.2)	(89.8, 95.6)	(89.4, 95.0)	(91.9, 95.7)	(92.2, 96.7)	(**92.8**, **96.8**)	(92.7, **96.8**)
	VisA	(82.1, 85.3)	(81.2, 85.4)	(81.6, 84.5)	(83.2, 85.8)	(84.7, 87.6)	(83.5, 86.3)	(**87.5**, **89.2**)
	MPDD	(75.9, 79.9)	(69.5, 75.0)	(72.0, 79.5)	(76.1, 81.3)	(71.9, 76.6)	(76.6, 81.5)	(**82.9**, **87.5**)
	BTAD	(86.4, 85.0)	(87.7, 86.7)	(91.6, 92.8)	(92.2, 89.1)	(**94.9**, 97.0)	(92.9, 93.2)	(94.4, **97.2**)
	DTD-Synthetic	(94.0, 97.1)	(91.8, 95.6)	(94.4, 97.1)	(94.4, 96.8)	(97.5, 99.1)	(95.0, 97.6)	(**97.7**, **99.2**)
	Average	(86.0, 88.7)	(84.0, 87.7)	(85.8, 89.8)	(87.6, 89.7)	(88.2, 91.4)	(88.2, 91.1)	(**91.0**, **94.0**)
Pixel-level (AUROC, AUPRO)	MVTec AD	(91.1, 81.4)	(89.9, —)	(88.7, 73.3)	(90.6, 82.3)	(90.8, **87.5**)	(91.3, 82.9)	(**92.1**, 85.6)
	VisA	(95.5, 86.7)	(92.3, —)	(92.8, 82.1)	(95.5, 86.5)	(95.8, 91.0)	(95.7, 87.4)	(**96.0**, **91.1**)
	MPDD	(95.5, 87.7)	(96.1, —)	(96.6, 89.9)	(95.1, 86.6)	(**96.8**, **91.6**)	(95.7, 87.9)	(**96.8**, 88.5)
	BTAD	(93.8, 73.7)	(94.4, —)	(95.4, 77.3)	(95.4, 74.8)	(91.1, **80.4**)	(95.5, 77.6)	(**95.8**, 78.1)
	DTD-Synthetic	(97.8, 92.4)	(96.9, —)	(98.1, 92.5)	(97.7, 92.3)	(98.1, **94.8**)	(98.1, 92.9)	(**98.4**, 93.2)
	Average	(94.7, 84.4)	(93.9, —)	(94.3, 83.0)	(94.9, 84.5)	(94.5, **89.1**)	(95.3, 85.7)	(**95.8**, 87.3)
Medical benchmarks								
Image-level (AUROC, AP)	HeadCT	(93.4, 91.6)	(93.5, 92.0)	(91.9, 89.5)	(**93.9**, 92.0)	—	(**93.9**, 92.1)	(**93.9**, **93.9**)
	Br35H	(94.6, 94.7)	(95.9, 95.3)	(93.8, 93.6)	(97.0, 95.4)	—	(96.1, 95.5)	(**97.3**, **97.5**)
	Average	(94.0, 93.2)	(94.7, 93.7)	(92.8, 91.5)	(95.5, 93.7)	—	(95.0, 93.8)	(**95.6**, **95.7**)
Pixel-level (AUROC, AUPRO)	ISIC	(88.9, 78.0)	(90.3, —)	(90.5, 80.9)	(88.9, 81.2)	—	(89.7, 78.4)	(**94.9**, **89.0**)
	CVC-ClinicDB	(82.9, 67.8)	(85.3, 14.6)	(84.2, 69.8)	(84.7, 70.1)	(85.9, 67.3)	—	(**87.2**, **74.5**)
	CVC-ColonDB	(81.9, 71.3)	(82.9, —)	(**83.9**, 73.7)	(82.5, 70.7)	—	(83.2, 71.9)	(83.4, **74.8**)
	Kvasir	(78.9, 45.6)	(79.4, —)	(78.7, 45.6)	(79.0, 43.9)	(82.6, 40.5)	(79.4, 45.9)	(**84.8**, **48.8**)
	Average	(83.2, 65.7)	(84.5, —)	(84.3, 67.5)	(83.8, 66.5)	—	—	(**87.6**, **71.8**)

et al. (2020). The main MVTec-source experiments use the complete labeled MVTec AD test split: 1,725 source images (467 normal and 1,258 anomalous) across 15 categories. For the reverse direction, the MVTec AD target result is produced by training on the complete labeled VisA evaluation split: 2,162 source images (962 normal and 1,200 anomalous) across 12 categories. Anomalous source images include pixel-level masks; these masks supervise the pixel loss and define the normal/anomalous patch sets used by SDPA, while normal source images use all-zero masks. The trained source model is evaluated directly on the designated unseen targets. No target-domain image, label, mask, or statistic is used for optimization, checkpoint selection, or test-time adaptation.

Metrics and implementation. We report image AUROC/AP and pixel AUROC/AUPRO using frozen CLIP ViT-L/14@336px and multi-level tokens. Appendix D details training and evaluation. Baselines follow their reported settings; unavailable entries are "—".

4.2 Main Results

Table 1 compares ReCal-CLIP with recent CLIP-based anomaly detection methods. On industrial benchmarks, ReCal-CLIP achieves the best average image-level AUROC/AP (91.0/94.0), with clear gains on VisA (87.5/89.2) and MPDD (82.9/87.5), indicating that cleaner local responses also benefit image-level decisions. For localization, it obtains the best average pixel AUROC (95.8) and the second-best AUPRO (87.3), behind VisualAD's 89.1. Results across the industrial datasets support robust transfer across categories and defect patterns.

The medical-domain results provide an additional cross-domain generalization check. Although the primary focus is industrial inspection, ReCal-CLIP achieves the best average pixel-level performance on these benchmarks, with 87.6 AUROC and 71.8 AUPRO, and the best average image-level AUROC/AP. This suggests that patch-level recalibration can also help in domains where abnormal regions are local and structurally irregular. Importantly, these medical results are obtained under

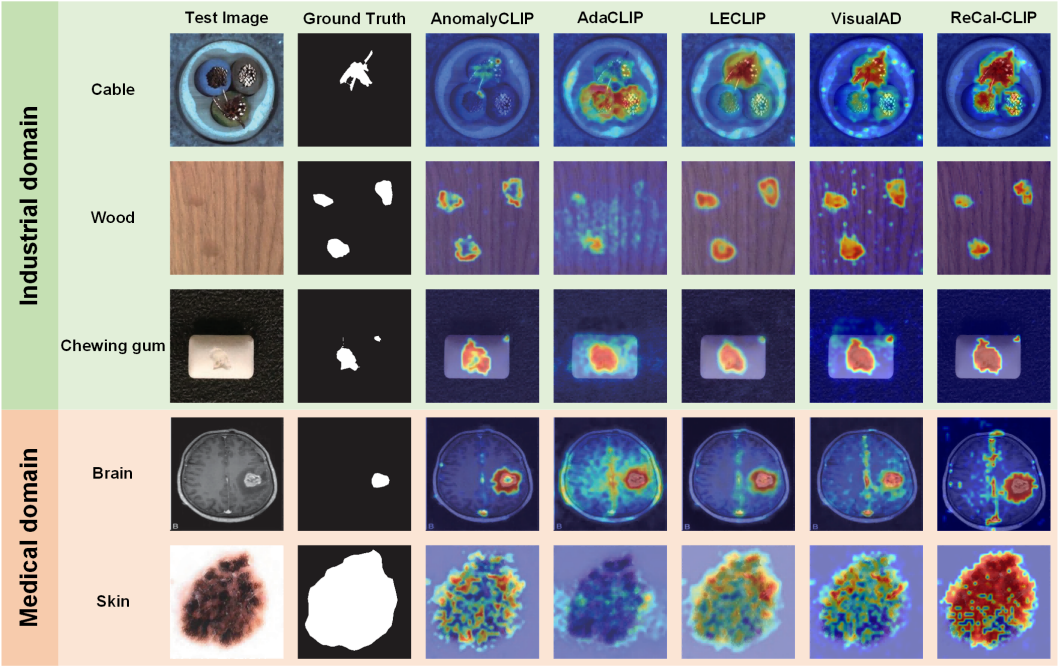


Figure 3: Qualitative anomaly localization comparison across industrial and medical target domains. ReCal-CLIP produces more complete and spatially coherent responses for local industrial defects and clearer lesion-focused activations in medical cases.

Table 2: Module ablation on MVTec-to-VisA. Checkmarks indicate enabled modules. IDE, SUS, and STR are the Hybrid Recalibration branches; SCPA and SDPA denote token aggregation and patch alignment. P/I denote pixel/image-level metrics; best results are **bold**.

Method						MVTec → VisA			
Base	IDE	SUS	STR	SCPA	SDPA	P-AUROC	P-AUPRO	I-AUROC	I-AP
✓				✓	✓	92.7	78.5	76.9	79.5
✓	✓			✓	✓	96.2	87.9	85.2	86.8
✓		✓		✓	✓	95.8	88.7	84.2	86.3
✓			✓	✓	✓	95.4	85.4	87.2	88.8
✓		✓	✓	✓	✓	95.0	86.8	85.3	86.4
✓	✓		✓	✓	✓	95.3	87.8	85.6	86.9
✓	✓	✓		✓	✓	**96.3**	89.6	84.5	86.0
✓	✓	✓	✓		✓	95.8	89.5	86.6	87.7
✓	✓	✓	✓	✓		95.9	89.2	85.5	86.3
✓	✓	✓	✓	✓	✓	96.0	**91.1**	**87.5**	**89.2**

the same source-target protocol, without target-domain optimization, so they evaluate transfer rather than domain-specific fitting. Qualitative maps in Figure 3 further show that ReCal-CLIP produces more continuous responses on industrial defects and clearer lesion-focused activations on medical cases, supporting the design motivation of improving image-side patch tokens before anomaly scoring.

4.3 Ablation Study

We conduct ablations on MVTec-to-VisA to assess the token-recalibration chain. As shown in Table 2, removing all three Hybrid Recalibration branches reduces pixel AUROC/AUPRO from

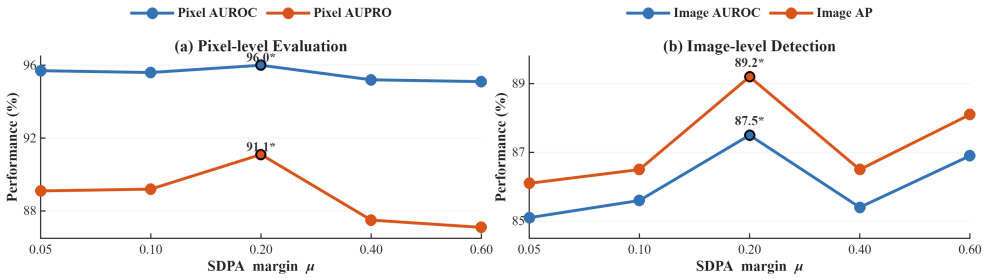


Figure 4: Sensitivity of the SDPA margin μ on MVTec-to-VisA. The default $\mu = 0.2$ gives 96.0/91.1 pixel AUROC/AUPRO and 87.5/89.2 image AUROC/AP.

Table 3: Sensitivity of the SDPA sample number K_0 on MVTec-to-VisA.

K_0	P-AUROC	P-AUPRO	I-AUROC	I-AP
32	95.5	88.6	86.1	87.0
64	95.4	88.2	85.3	86.3
128	95.6	88.7	85.9	87.3
256	**96.0**	**91.1**	**87.5**	**89.2**
512	95.6	88.9	86.5	87.9

96.0/91.1 to 92.7/78.5, the largest degradation among the tested configurations. This result identifies Hybrid Recalibration as the dominant empirical contributor in this ablation, although it does not by itself establish a unique underlying mechanism.

The branch ablations show non-identical empirical effects. IDE has the largest impact on localization, with its removal lowering P-AUPRO to 86.8, whereas removing STR lowers I-AUROC/I-AP to 84.5/86.0. Removing SUS changes both localization overlap and image-level precision. Apart from the branch-free configuration, pixel AUROC varies only slightly. SCPA and SDPA provide smaller gains: removing them lowers P-AUPRO by 1.6 and 1.9 points, respectively, and removing SDPA lowers I-AUROC/I-AP by 2.0/2.9 points. These observations support the utility of the complete configuration but do not imply that SDPA semantically purifies features or eliminates source-domain bias.

The default values $\mu = 0.2$ and $K_0 = 256$ are part of the source-domain configuration fixed before target-domain evaluation. Figure 4 and Table 3 examine performance around this preset. On VisA, the tested metrics peak jointly at $\mu = 0.2$, while both smaller and larger sample budgets give lower results than $K_0 = 256$. These observations show that the fixed configuration lies in a favorable region of the tested ranges, but should not be interpreted as evidence for semantic purification or complete domain-bias removal.

5 Conclusion

We presented ReCal-CLIP, a CLIP-based framework for cross-domain zero-shot anomaly detection. ReCal-CLIP adjusts image-side patch tokens before anomaly scoring through anomaly-suspect routing, Hybrid Recalibration, lightweight SCPA, and training-only SDPA. Experiments on industrial benchmarks, together with supplementary medical-domain evaluation, show strong detection and localization performance. The ablation and sensitivity studies provide empirical support for the proposed token-recalibration pipeline under the evaluated source–target protocol; they do not establish a stronger causal interpretation of SDPA. Future work will explore more adaptive token neighborhoods for harder domain shifts. This may further improve localization under subtle and heterogeneous anomalies.

AI USE STATEMENT

Generative AI tools assisted methodological discussion, code implementation, experiment execution and auditing, statistical analysis, language polishing, manuscript organization, and LaTeX formatting. Numerical results were computed by executable pipelines from recorded inputs rather than synthesized from manuscript tables. Automated checks do not substitute for independent author verification. The authors take responsibility for the final manuscript, including its claims, implementation, data use, and results.

REPRODUCIBILITY STATEMENT

Section 3 specifies the ReCal-CLIP method, scoring procedure, and training objective. Appendix A identifies inherited text-side components and details their implementation; Appendix B documents the visual execution path, auxiliary losses, feature transformations, multi-view scoring, and evaluation-time smoothing. Appendix C records the datasets, metadata, and source–target protocol, while Appendix D provides the training and reporting configuration, including the three-epoch schedule and fixed-seed repetitions. Appendix E reports per-category results. These disclosures support reproduction of the reported protocol; the ablations and sensitivity analyses are empirical evidence under that protocol, not a theoretical guarantee, a multi-seed uncertainty estimate, or proof of a unique causal mechanism.

ETHICS STATEMENT

This work uses publicly available industrial anomaly-detection and medical-imaging benchmarks under their respective terms and licenses. No new human participants, private records, or personally identifying information were collected. The medical datasets are used only for methodological evaluation and not for clinical diagnosis. Dataset biases, annotation errors, and domain shifts may affect performance; any real-world industrial or medical deployment therefore requires domain-specific validation, safety review, and appropriate human oversight.

REFERENCES

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.

Jorge Bernal, F. Javier Sánchez, and Fernando Vilariño. Towards automatic polyp detection with a polyp appearance model. *Pattern Recognition*, 45(9):3166–3182, 2012.

Jorge Bernal, F. Javier Sánchez, Gloria Fernández-Esparrach, Debora Gil, Cristina Rodríguez, and Fernando Vilariño. WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized Medical Imaging and Graphics*, 43:99–111, 2015.

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. *arXiv preprint arXiv:2305.17382*, 2023.

Xuhai Chen, Jiangning Zhang, Guanzhong Tian, Haoyang He, Wuhao Zhang, Yabiao Wang, Chengjie Wang, and Yong Liu. CLIP-AD: A language-guided staged dual-path model for zero-shot anomaly detection. In *International Joint Conference on Artificial Intelligence*, pp. 17–33. Springer, 2024.

Sasank Chilamkurthy, Rohit Ghosh, Swetha Tanamala, Mustafa Biviji, Norbert G Campeau, Vasantha Kumar Venugopal, Vidur Mahajan, Pooja Rao, and Prashant Warier. Deep learning algorithms

for detection of critical findings in head ct scans: A retrospective study. *The Lancet*, 392(10162): 2388–2396, 2018.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3606–3613, 2014.

Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, and Allan Halpern. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration. *arXiv preprint arXiv:1902.03368*, 2019.

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.

Zheng Fang, Xiaoyang Wang, Haocheng Li, Jiejie Liu, Qiugui Hu, and Jimin Xiao. Fastrecon: Few-shot industrial anomaly detection via fast feature reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17481–17490, 2023.

Xinru Guo, Lijuan Song, and Zirui Ma. CHSA-CLIP: Zero-shot anomaly detection via cross-modal hierarchical semantic alignment. *Expert Systems with Applications*, 330:133094, 2026.

Ahmed Hamada. Br35H: Brain tumor detection 2020. `https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection`, 2020.

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.

Ming Hu, Yongsheng Huo, Mingyu Dou, Jianfu Yin, Peng Zhao, Yao Wang, Cong Hu, Bingliang Hu, and Quan Wang. FB-CLIP: Fine-grained zero-shot anomaly detection with foreground-background disentanglement. *arXiv preprint arXiv:2603.19608*, 2026.

Shihua Huang, Zhichao Lu, Ran Cheng, and Cheng He. FaPN: Feature-aligned pyramid network for dense image prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 864–873, 2021.

Jiin Im, Yongho Son, and Je Hyeong Hong. FUN-AD: Fully unsupervised learning for anomaly detection with noisy training data. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 9447–9456. IEEE, 2025.

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.

Stepan Jezek, Martin Jonak, Radim Burget, Pavel Dvorak, and Martin Skotak. Deep learning-based defect detection of metal parts: Evaluating current methods in complex conditions. In *2021 International Conference on Ultra Modern Telecommunications and Control Systems and Workshops*, pp. 66–71. IEEE, 2021.

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 *International Conference on Multimedia Modeling*, pp. 451–462. Springer, 2020.

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.

Dongfang Liu, Yiming Cui, Liqi Yan, Christos Mousas, Baijian Yang, and Yingjie Chen. Densernet: Weakly supervised visual localization using multi-scale feature aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6101–6109, 2021.

Yuyao Liu, Qingyong Li, Zhehong Wang, Jien Kato, Jie Zhang, and Wen Wang. LECLIP: Boosting zero-shot anomaly detection with local enhanced CLIP. *IEEE Transactions on Instrumentation and Measurement*, 2025.

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*, pp. 1–6. IEEE, 2021.

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 *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. DenseCLIP: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18082–18091, 2022.

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.

Maryam Sultana, Arif Mahmood, Sajid Javed, and Soon Ki Jung. Unsupervised deep context prediction for background estimation and foreground segmentation. *Machine Vision and Applications*, 30(3):375, 2019.

Xiaoming Sun, Yuxiang Zhang, Yanni Dong, and Bo Du. Contrastive self-supervised learning-based background reconstruction for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–12, 2025.

Xian Tao, Xinyi Gong, Xin Zhang, Shaohua Yan, and Chandranath Adak. Deep learning for unsupervised anomaly localization in industrial images: A survey. *IEEE Transactions on Instrumentation and Measurement*, 71:1–21, 2022.

Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip S. Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):8052–8072, 2022.

Lian Xu, Wanli Ouyang, Mohammed Bennamoun, Farid Boussaid, and Dan Xu. Multi-class token transformer for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4310–4319, 2022.

Zhiwei Yang, Yucong Meng, Kexue Fu, Shuo Wang, and Zhijian Song. More: Class patch attention needs regularization for weakly supervised semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 9400–9408, 2025.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. FastFlow: Unsupervised anomaly detection and localization via 2d normalizing flows. *arXiv preprint arXiv:2111.07677*, 2021.

Jing Zhang, Shaofu Lin, Lei Ding, and Lorenzo Bruzzone. Multi-scale context aggregation for semantic segmentation of remote sensing images. *Remote Sensing*, 12(4):701, 2020.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2022.

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

Jiawen Zhu, Yew-Soon Ong, Chunhua Shen, and Guansong Pang. Fine-grained abnormality prompt learning for zero-shot anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22241–22251, 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.

Zuo Zuo, Zongze Wu, Badong Chen, and Xiaopin Zhong. A reconstruction-based feature adaptation for anomaly detection with self-supervised multi-scale aggregation. In *ICASSP 2024–2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5840–5844. IEEE, 2024.

A TEXT-SIDE IMPLEMENTATION DETAILS

This appendix records the text-side implementation used by the released ReCal-CLIP code. The object-agnostic normal/anomalous prompt-learning setting follows AnomalyCLIP Zhou et al. (2024); the token selector, projection MLP, and three-path pooling are retained from the underlying FB-CLIP Hu et al. (2026) implementation. These inherited components provide the shared text-prototype scoring path, not a new text-side contribution of ReCal-CLIP. In the module ablation (Table 2), that path remains in place while the proposed visual-side branches, SCPA, or training-only SDPA are varied. The differences therefore support the contribution of the added mechanisms under the evaluated configuration, without isolating a text-side gain or establishing an exclusive causal explanation.

A.1 PROMPT PARAMETERIZATION

For the single class name "object," the prompt learner constructs a normal state prompt "object without defect" and an anomalous state prompt "object with defect." Each state is preceded by E trainable context token embeddings, giving

$$P_n = [V_1, \ldots, V_E, \text{object without defect}], \qquad P_a = [W_1, \ldots, W_E, \text{object with defect}], \tag{21}$$

where the two context sequences are independent trainable parameters. The notation omits the fixed start token, terminal punctuation, EOT token, and padding. The start/suffix embeddings are obtained from the frozen CLIP embedding table; these embeddings and the tokenizer-produced token IDs are stored as buffers. In the default configuration, $E = 12$ and $L = 77$. The assembled embeddings are passed to the text transformer, while token IDs identify the EOT position.

A.2 TEXT ENCODING AND POOLING

Given an assembled prompt embedding matrix $P \in \mathbb{R}^{L \times d_t}$, the implementation adds CLIP positional embeddings, applies the CLIP text transformer, and applies its final layer normalization:

$$Z = \mathrm{LN}_{\mathrm{final}}\big(\mathrm{Transformer}(P + E_{\mathrm{pos}})\big), \qquad Z \in \mathbb{R}^{L \times d_t}. \tag{22}$$

The executed code forms three complementary pooled representations. First, it selects the end-of-text (EOT) position k_{eot} from the token IDs and applies the frozen CLIP text projection W_t:

$$u_{\mathrm{eot}} = Z_{k_{\mathrm{eot}}} W_t. \tag{23}$$

Second, it averages all L sequence positions, including special-token and padding positions, and applies the same projection:

$$u_{\mathrm{mean}} = \left(\frac{1}{L} \sum_{l=1}^{L} Z_l \right) W_t. \tag{24}$$

There is no CLS pooling path in this text encoder. Third, a trainable token selector produces one scalar logit per text position:

$$q_l = \mathrm{ReLU}(Z_l W_1 + b_1) W_2 + b_2, \tag{25}$$

$$a_l = \mathrm{softmax}_l(q_l), \qquad u_{\mathrm{pool}} = \sum_{l=1}^{L} a_l Z_l, \tag{26}$$

where W_1 maps d_t to $\lfloor d_t/2 \rfloor$ and W_2 maps this hidden representation to one scalar. The softmax covers all L positions without a padding mask; the selector performs soft weighting, not hard token selection. The attention-pooled representation is mapped to the CLIP embedding space by a separate projection MLP:

$$u_{\mathrm{attn}} = \mathrm{ReLU}(u_{\mathrm{pool}} W_3 + b_3) W_4 + b_4, \tag{27}$$

where the code uses $d_t \to d_t \to 768$ dimensions. The three paths are combined with fixed coefficients rather than dynamically normalized fusion weights:

$$u = u_{\mathrm{mean}} + 0.5 u_{\mathrm{attn}} + 0.5 u_{\mathrm{eot}}, \qquad \mathbf{t} = \frac{u}{\|u\|_2}. \tag{28}$$

The final normalization is applied by the enhanced text-encoding wrapper before text–image similarity computation. The resulting normal and anomalous vectors are stacked as $\mathbf{T} = [\mathbf{t}_n; \mathbf{t}_a]$.

A.3 TRAINABLE AND FROZEN PARAMETERS

During source-domain training, the optimized parameters are the normal/anomalous context embeddings, the token selector, the attention-branch projection MLP, the confidence-gated residual recalibration module, and the multi-layer scoring coefficients `scale_weights`. The CLIP token embedding, positional embeddings, text transformer, final normalization, text projection W_t, and visual backbone remain frozen. The three text-pooling paths use the fixed coefficients in Equation (28). Training and evaluation use the same pooling and normalization path; evaluation performs no parameter updates.

B VISUAL-SIDE AND EVALUATION DETAILS

This appendix supplements the visual-side and evaluation steps whose code-level details are not needed for the main methodological narrative. The equations below describe the executed implementation, including the distinction between training-only quantities and quantities used to produce the reported maps.

Executed STR aggregation. For each patch position i, the STR branch unfolds a stride-one 5×5 neighborhood $\mathcal{N}_5(i)$ with two-position zero padding. We regard an out-of-grid slot as $\mathbf{x}_j = \mathbf{0}$ and $m_j = 0$, matching the implementation's padded feature and routing tensors. The routed suspect and context features are

$$\mathbf{x}^a_{ij} = m_j \mathbf{x}_j, \qquad \mathbf{x}^n_{ij} = (1 - m_j)\mathbf{x}_j, \qquad j \in \mathcal{N}_5(i). \tag{29}$$

Let $\bar{\mathbf{x}}^a_i = 25^{-1} \sum_{j \in \mathcal{N}_5(i)} \mathbf{x}^a_{ij}$. The implementation assigns context stability and suspect information-richness logits

$$d^n_{ij} = \cos(\mathbf{x}^n_{ij}, \mathbf{x}_0)\, \|\mathbf{x}^n_{ij}\|_2, \tag{30}$$

$$d^a_{ij} = \big[1 - \cos(\mathbf{x}^a_{ij}, \mathbf{x}_0)\big]\, \|\mathbf{x}^a_{ij} - \bar{\mathbf{x}}^a_i\|_2. \tag{31}$$

The two sets of logits are independently normalized over all 25 neighborhood slots,

$$\pi^a_{ij} = \mathrm{softmax}_j(d^a_{ij}), \qquad \pi^n_{ij} = \mathrm{softmax}_j(d^n_{ij}), \tag{32}$$

and the routed local summaries are

$$\mathbf{z}^a_i = \frac{\sum_{j \in \mathcal{N}_5(i)} \pi^a_{ij} \mathbf{x}^a_{ij}}{\max\Big(\sum_{j \in \mathcal{N}_5(i)} m_j, 10^{-8}\Big)}, \tag{33}$$

$$\mathbf{z}^n_i = \frac{\sum_{j \in \mathcal{N}_5(i)} \pi^n_{ij} \mathbf{x}^n_{ij}}{\max\Big(\sum_{j \in \mathcal{N}_5(i)} (1 - m_j), 10^{-8}\Big)}, \tag{34}$$

$$\mathbf{u}^L_i = \mathbf{z}^a_i + \mathbf{z}^n_i. \tag{35}$$

The CLS token is copied unchanged before this patch-grid operation. The resulting STR sequence subsequently enters the same confidence-gated residual adapter used by the other branches; no separate concatenation-based A_L module is used in the executed path.

B.1 SEMANTIC CONSISTENCY LOSS

The term denoted by $\mathcal{L}^{it}_{reg}$ in the training objective is implemented as a semantic consistency loss. Let $\mathbf{T} \in \mathbb{R}^{2 \times d}$ contain the normalized normal and anomalous text features, and let $\mathbf{V} \in \mathbb{R}^{B \times L \times d}$ be the projected token features from the last selected visual layer. The implementation first averages the L visual tokens, rather than computing a separate loss for every patch:

$$\bar{\mathbf{v}}_b = \frac{1}{L} \sum_{i=1}^{L} \mathbf{V}_{b,i}, \qquad \hat{\mathbf{v}}_b = \frac{\bar{\mathbf{v}}_b}{\|\bar{\mathbf{v}}_b\|_2}, \qquad \hat{\mathbf{t}}_c = \frac{\mathbf{T}_c}{\|\mathbf{T}_c\|_2}. \tag{36}$$

With temperature $\tau_s = 0.07$, the two-class similarities and probabilities are

$$q_{b,c} = \frac{\hat{\mathbf{v}}_b^\top \hat{\mathbf{t}}_c}{\tau_s}, \qquad p_{b,c} = \mathrm{softmax}_c(q_{b,c}). \tag{37}$$

The loss combines an entropy term and an absolute normal–anomalous separation term:

$$\mathcal{L}_{ent} = -\frac{1}{B}\sum_{b=1}^{B}\sum_{c\in\{n,a\}} p_{b,c}\log(p_{b,c}+\epsilon), \tag{38}$$

$$\mathcal{L}_{mar} = \frac{1}{B}\sum_{b=1}^{B}\max\left(0, 1-|q_{b,a}-q_{b,n}|\right), \tag{39}$$

$$\mathcal{L}_{sem} = 0.15\left(\mathcal{L}_{ent} + 0.5\mathcal{L}_{mar}\right). \tag{40}$$

Here $\epsilon = 10^{-8}$ is used inside the logarithm. This loss is added to the image-level focal loss, pixel-level focal and Dice losses, and (when enabled) the SDPA loss during source-domain training. The evaluation loop ignores the returned loss and performs no optimization; the forward function may still compute and return it under `no_grad` for interface compatibility.

B.2 Context-residual feature transformation

After Hybrid Recalibration and before projection, SCPA, and text-prototype scoring, the code receives one token set for every selected visual layer and recalibration branch. If there is more than one such set, it forms a shared context estimate from the first $\lfloor L/2 \rfloor$ sequence positions of every set, including the CLS token at position zero, and concatenates these subsets along the token dimension. For the resulting tensor $\mathbf{U}$, the context estimate is the fixed mean–max combination

$$\mathbf{b} = 0.5\ \mathrm{Mean}_{tokens}(\mathbf{U}) + 0.5\ \mathrm{Max}_{tokens}(\mathbf{U}). \tag{41}$$

The same vector is expanded to all positions of each processed token set. For a token $\mathbf{u}_i$ and its expanded context $\mathbf{b}_i$, the code computes

$$\mathbf{r}_i = \mathbf{u}_i - \mathbf{b}_i, \qquad \alpha_i = 1 - \cos(\mathbf{u}_i, \mathbf{b}_i), \tag{42}$$

and uses the similarity-weighted residual in the visual token update:

$$\mathbf{u}'_i = 0.5\,\mathbf{u}_i + 0.5\,\alpha_i\mathbf{r}_i. \tag{43}$$

The transformed token $\mathbf{u}'_i$ is projected by the frozen visual projection and sent to the subsequent scoring path. The residual magnitude is not added separately to the normal or anomalous logits. If only one token set is present, the implementation keeps that recalibrated set unchanged and does not construct a context residual.

B.3 Multi-layer, multi-branch scoring

For each selected layer, Hybrid Recalibration iterates through the configured branches in order; consequently, the flattened token-set order is layer-major and branch-minor. For the default five layers and three branches, this produces fifteen scoring views. Each projected view is row-wise normalized, processed by SCPA, and compared with the two text features:

$$\mathbf{Q}_{\ell,v} = \frac{\mathrm{SCPA}(\mathbf{U}'_{\ell,v})\mathbf{T}^{\top}}{0.07}, \tag{44}$$

where SCPA uses the fixed 0.5, 0.3, and 0.2 weights already specified in the main text. The learnable layer coefficients are normalized with a softmax. If $k = \ell V + v$ indexes the flattened views and V is the number of branches, the actual score weight is

$$\omega_{\ell,v} = \mathrm{softmax}(\boldsymbol{\gamma})_{\ell}\frac{3}{V}, \qquad \mathbf{Q} = \sum_{\ell,v}\omega_{\ell,v}\mathbf{Q}_{\ell,v}, \tag{45}$$

where $\boldsymbol{\gamma}$ is the parameter vector named `scale_weights`. The factor $3/V$ preserves the score scale when the number of enabled branches differs from the default three.

The aggregated logits $\mathbf{Q}$ are passed directly to a two-class softmax. The image-level score is taken from the image token before map smoothing, whereas the patch probabilities are reshaped into the anomaly map. Context-residual information affects these logits only through the transformed visual features $\mathbf{U}'_{\ell,v}$ above; no separate reconstruction score is added to either class channel.

B.4 GAUSSIAN SMOOTHING DURING EVALUATION

Gaussian smoothing is applied only in the evaluation loop. After inference produces a patch anomaly map, the map is bilinearly resized to the standard evaluation resolution and filtered independently for each image using a Gaussian filter with the command-line parameter σ (default $\sigma = 4$):

$$\mathbf{M}_{eval} = G_\sigma * \mathrm{Resize}(\mathbf{M}_{raw}). \tag{46}$$

Pixel AUROC and AUPRO are computed from $\mathbf{M}_{eval}$. The image-level prediction is collected directly from the model's image-token score before this filtering, so Gaussian smoothing does not affect image AUROC or image AP, does not update model parameters, and is not used during source-domain training.

Evaluation metrics. Image AUROC is the area under the true-positive-rate versus false-positive-rate curve obtained by thresholding the image-token anomaly scores against binary image labels. Image average precision (AP) summarizes the precision–recall curve for anomalous images as the weighted mean of precision at successive recall increments. Pixel AUROC applies the same ROC definition to the flattened anomaly maps and binary masks within each evaluated category. Pixel AUPRO averages the fraction of correctly detected pixels within each connected anomalous region at each threshold, plots this per-region overlap (PRO) against the false-positive rate on normal pixels, and integrates over the sampled thresholds with false-positive rate below 0.3. In the evaluation implementation, 200 threshold steps are sampled over the observed anomaly-score range and the retained false-positive-rate coordinates are rescaled to $[0, 1]$ before trapezoidal integration. Metrics are computed per category and then arithmetically averaged across categories; reported percentages are the corresponding scores multiplied by 100.

C DETAILED DATASET DESCRIPTIONS

This appendix provides detailed descriptions of the public datasets used in our experiments. The statistics below follow the standardized dataset preparation used by closely related cross-domain anomaly-detection studies. For MVTec AD and VisA, which serve as source domains in the two reported transfer directions, the counts are the actual items in the complete official labeled test/evaluation splits used for source training. For datasets used only as targets, the counts describe their evaluation pools. Pixel-level masks are used for source supervision only in the corresponding source run and otherwise only for target evaluation.

C.1 DATASET OVERVIEW

We evaluate ReCal-CLIP on five industrial benchmarks and six medical benchmarks. The industrial datasets provide both image-level anomaly labels and pixel-level defect masks. The medical datasets have complementary annotation regimes: HeadCT and Br35H provide image-level labels only, whereas ISIC, CVC-ClinicDB, CVC-ColonDB, and Kvasir provide pixel-level lesion annotations. Consequently, image-level AUROC/AP are reported for HeadCT and Br35H, while pixel-level AUROC/AUPRO are reported for the medical localization benchmarks.

C.2 INDUSTRIAL DATASETS

MVTec AD. MVTec AD Bergmann et al. (2019) is a real-world industrial anomaly-detection benchmark containing 15 categories, including ten object categories and five texture categories: carpet, bottle, hazelnut, leather, cable, capsule, grid, pill, transistor, metal nut, screw, toothbrush, zipper, tile, and wood. It covers structural and surface defects under diverse inspection conditions. For the MVTec→target experiments, we use its complete official labeled test split as the source: 1,725 images, comprising 467 normal and 1,258 anomalous images. Anomalous images provide pixel-level defect masks; during source training these masks supervise the pixel objective and define SDPA patch groups. When MVTec AD is evaluated as a target, none of these target images or annotations is used for optimization, and the source domain is VisA.

VisA. VisA Zou et al. (2022) is a large-scale industrial benchmark containing 12 object categories: candle, capsules, cashew, chewing gum, fryum, macaroni1, macaroni2, pcb1, pcb2, pcb3, pcb4, and

Table 4: Statistics and evaluation roles of the datasets used in ReCal-CLIP. MVTec AD and VisA counts are also the actual source-training counts in their respective transfer directions. The medical polyp datasets contain only polyp-positive images in the adopted benchmark pool and are therefore used for pixel-level localization rather than image-level classification. ISIC counts are omitted because the standardized source in the code release does not provide unified normal/anomalous counts in its metadata.

Domain	Dataset	Categories	Normal images	Anomalous images	Pixel masks	Evaluation
Industrial	MVTec AD	15	467	1,258	Yes	Image + pixel
	VisA	12	962	1,200	Yes	Image + pixel
	MPDD	6	176	282	Yes	Image + pixel
	BTAD	3	451	290	Yes	Image + pixel
	DTD-Synthetic	12	357	947	Yes	Image + pixel
Medical	HeadCT	1	100	100	No	Image only
	Br35H	1	1,500	1,500	No	Image only
	ISIC	1	–	–	Yes	Pixel only
	CVC-ClinicDB	1	0	612	Yes	Pixel only
	CVC-ColonDB	1	0	380	Yes	Pixel only
	Kvasir	1	0	1,000	Yes	Pixel only

pipe fryum. It includes complex objects with multiple components and defects captured under varying viewpoints, illumination conditions, and backgrounds. For the reverse VisA→MVTec experiment, we use its complete official labeled evaluation split as the source: 2,162 images, comprising 962 normal and 1,200 anomalous images. Anomalous images provide pixel-level masks used for the pixel objective and SDPA patch grouping during that source run. In the main MVTec→VisA experiment, VisA is instead an unseen target and its images and annotations are not used for optimization or model selection.

MPDD. MPDD Jezek et al. (2021) is designed for defect detection on painted metal parts under realistic manufacturing conditions. It contains six categories: bracket black, bracket brown, bracket white, connector, metal plate, and tubes. The benchmark includes variations in object position, scale, and appearance, together with structural and surface anomalies. The labeled pool used in our metadata contains 176 normal and 282 anomalous images, all accompanied by pixel-level defect annotations.

BTAD. BTAD Mishra et al. (2021) contains high-resolution images from three industrial product categories, represented as "01", "02", and "03" in the released class mapping. The labeled benchmark pool contains 451 normal and 290 anomalous images. The accompanying pixel-level masks enable both image-level detection and pixel-level localization.

DTD-Synthetic. DTD-Synthetic is the synthetic texture-anomaly benchmark used in our experiments. It contains 12 texture categories: Woven_001, Woven_127, Woven_104, Stratified_154, Blotchy_099, Woven_068, Woven_125, Marbled_078, Perforated_037, Mesh_114, Fibrous_183, and Matted_069. Its labeled benchmark pool contains 357 normal and 947 anomalous images. The synthetic anomalous regions are supplied with pixel-level masks, allowing both image-level and pixel-level evaluation. This benchmark is used specifically for texture anomaly detection and should not be confused with using the original 47-category Describable Textures Dataset as a standard image-classification benchmark Cimpoi et al. (2014).

C.3 Medical datasets

HeadCT and Br35H. HeadCT Chilamkurthy et al. (2018) is organized as a single binary anomaly-detection task with 100 normal and 100 anomalous images in the adopted benchmark pool. Br35H Hamada (2020) is a balanced brain MRI dataset containing 1,500 normal and 1,500 tumor-containing images. These datasets provide image-level labels but no pixel-level lesion masks in the

evaluation protocol. They are therefore used exclusively for image-level anomaly recognition, and no localization metrics are reported for them.

ISIC. ISIC Codella et al. (2019) is a dermoscopic skin-lesion benchmark with lesion-region annotations. We use it as a medical pixel-localization target. The provided masks distinguish lesion regions from surrounding skin, and the target images and masks are used only for evaluation rather than for source-domain optimization.

CVC-ClinicDB, CVC-ColonDB, and Kvasir. CVC-ClinicDB Bernal et al. (2015) contains 612 colonoscopy frames with colorectal polyps, CVC-ColonDB Bernal et al. (2012) contains 380 polyp-positive colonoscopy frames, and Kvasir Jha et al. (2020) contains 1,000 gastrointestinal endoscopy images with manually annotated polyp masks. The adopted benchmark pools contain no normal images for these three datasets; all images are polyp-positive and are therefore evaluated only at the pixel level. The masks distinguish lesion regions from non-lesion background pixels, making these datasets suitable for evaluating cross-domain anomaly localization.

C.4 Source–target protocol and metadata

For the main transfer direction, ReCal-CLIP is trained on all 1,725 items of the complete labeled MVTec AD test split and evaluated directly on VisA, MPDD, BTAD, DTD-Synthetic, and the medical target datasets. For the MVTec AD target result, the direction is reversed: the model is trained on all 2,162 items of the complete labeled VisA evaluation split and evaluated on MVTec AD. The source split is represented under the `train` key of the corresponding source-domain `meta.json`; "train" here denotes its role in our cross-domain protocol rather than an original anomaly-free training split. Source and target roles are run-specific and disjoint: the dataset designated as target contributes no image, label, mask, or statistic to parameter updates, checkpoint selection, or test-time adaptation.

The dataloader expects each dataset root to contain a `meta.json` grouped by class. Each item contains `img_path`, `mask_path`, `cls_name`, and `anomaly`. Normal items have `anomaly=0` and receive an all-zero mask in the loader. Anomalous items have `anomaly=1` and point to their pixel mask. During source-domain training, image labels and masks supervise the image/pixel objectives and define the normal and anomalous patch sets for the training-only SDPA objective. If a source image does not contain both patch types after mask resizing, it contributes zero to SDPA, while the other training losses remain active. During target-domain evaluation, the metadata are used only to load images and annotations for scoring.

The source-domain metadata therefore contain 467 normal and 1,258 anomalous MVTec AD records for MVTec-source runs, and 962 normal and 1,200 anomalous VisA records for the reverse run. Their image labels and anomaly masks are used consistently by the supervised source-domain objectives, including SDPA. The corresponding metadata-generation procedure and source sample lists accompany the code release so that the source protocol can be reproduced exactly.

D Training and Reporting Protocol

This section records the detailed configuration used for training and evaluation. Unless otherwise specified, ReCal-CLIP uses the frozen CLIP ViT-L/14@336px backbone, resizes input images to 518×518, and extracts multi-level visual tokens from layers $\{1, 6, 12, 18, 24\}$. The CLIP image and text encoders remain frozen; only the learnable prompt parameters and the lightweight Hybrid Recalibration components are optimized. We train for 3 epochs using a batch size of 2, the Adam optimizer, and a learning rate of 3×10^{-5} for the main MVTec-source experiments. All experiments are conducted on a single NVIDIA GeForce RTX 4080 SUPER GPU using PyTorch 2.8.0. SDPA uses a maximum sample number of $K_0 = 256$, loss weight $\lambda_{sdpa} = 0.05$, and margin $\mu = 0.2$. SCPA uses fixed aggregation weights 0.5, 0.3, and 0.2 for the 1×1, 3×3, and 5×5 branches, respectively. The evaluation-time Gaussian filter uses the default $\sigma = 4$ described above.

For ReCal-CLIP and its ablations, we repeat the complete training-and-evaluation procedure three times. Each run starts from the same pretrained CLIP initialization, uses the fixed random seed 42 and the same configuration, and produces its own trained checkpoint, which is then evaluated

under the same protocol. In practice, we launch the same training command separately for the three runs and report the arithmetic mean of their evaluation metrics. Since all runs use the same seed, this protocol is not a multi-seed uncertainty estimate, and we do not claim statistical significance. The three-epoch schedule, data split, preprocessing, and all other settings are held fixed within each comparison group.

All model-selection decisions are made from the source-domain training protocol described above. The default configuration, including $\mu = 0.2$ and $K_0 = 256$, is fixed before target-domain evaluation and retained across targets and comparison runs. Target-domain images, labels, and masks are not used to update parameters or to select a target-specific checkpoint. The target domain is used only for evaluation as described in Section C. Per-category results and their dataset-level aggregation follow the same evaluation configuration, including the resizing and pixel-map smoothing rules stated in this appendix.

E Per-category Results

Tables 5–9 report the latest per-category results for the industrial benchmarks. The columns are pixel AUROC, pixel AUPRO, image AUROC, and image AP, respectively; all values are percentages. The final row of each table is the arithmetic mean over the categories in that dataset. The medical benchmarks contain only one evaluation category in the adopted protocol, so their entries in Table 10 are copied from the corresponding rows of the main comparison table rather than recomputed as an additional multi-category aggregate.

Table 5: Per-category results on MVTec AD.

Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
bottle	93.7	85.7	95.7	98.6
cable	80.0	68.3	79.4	88.1
capsule	96.3	92.6	93.7	98.7
carpet	99.6	96.8	100	100
grid	98.4	85.1	98.7	99.6
hazelnut	97.9	86.3	95.6	97.8
leather	99.4	97.9	100	100
metal_nut	66.4	70.4	75.3	94.7
pill	91.2	94.1	86.6	97.2
screw	97.9	92.7	84.4	94.1
tile	97.9	92.9	99.4	99.7
toothbrush	93.9	86.0	96.1	98.5
transistor	71.7	54.0	88.5	85.8
wood	98.2	95.6	98.7	99.6
zipper	98.3	86.4	98.7	99.7
MEAN	**92.1**	**85.6**	**92.7**	**96.8**

Figure 5 shows category-wise changes rather than uniform improvement: most displayed changes are positive, while some categories show negative changes. The latter identify cases where the evaluated method does not outperform the comparison on the plotted metric; the figure alone does not establish whether domain shift, defect size, category composition, or another factor caused them. The category-level variation thus qualifies the aggregate gains and motivates further failure-case analysis rather than a causal explanation for individual decreases.

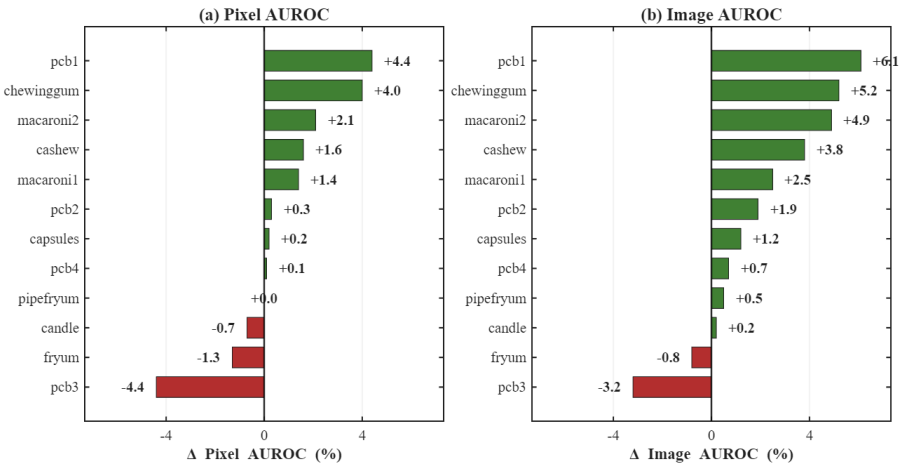


Figure 5: Category-wise performance changes on MVTec-to-VisA. Green and red bars indicate positive and negative changes, respectively.

Table 6: Per-category results on VisA.

Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
candle	98.7	96.7	93.8	95.1
capsules	96.7	91.3	92.8	96.3
cashew	94.5	96.8	93.8	97.3
chewinggum	99.4	93.7	97.9	99.1
fryum	95.9	94.0	96.2	98.1
macaroni1	99.3	97.4	86.8	88.7
macaroni2	97.7	92.6	73.4	71.9
pcb1	93.6	83.9	77.7	81.1
pcb2	92.9	78.6	76.3	76.4
pcb3	89.7	79.0	68.9	72.7
pcb4	95.4	91.4	94.1	93.8
pipe_fryum	97.9	97.1	98.4	99.3
MEAN	**96.0**	**91.1**	**87.5**	**89.2**

Table 7: Per-category results on MPDD.

Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
bracket_black	98.1	93.3	73.3	85.2
bracket_brown	95.3	78.7	63.1	79.1
bracket_white	99.4	96.9	93.6	94.9
connector	97.2	88.9	82.6	71.3
metal_plate	92.6	80.1	86.8	95.3
tubes	98.3	93.3	98.1	99.2
MEAN	**96.8**	**88.5**	**82.9**	**87.5**

Table 8: Per-category results on BTAD.

Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
01	93.6	73.6	93.7	98.4
02	96.6	73.9	91.4	98.1
03	97.3	86.8	98.2	95.2
MEAN	**95.8**	**78.1**	**94.4**	**97.2**

Table 9: Per-category results on DTD-Synthetic.

Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
woven_001	99.7	98.1	100	100
woven_127	94.9	93.4	95.3	97.9
woven_104	98.0	92.8	99.7	99.9
Stratified_154	99.6	96.8	99.8	100
Blotchy_099	99.2	95.4	99.3	99.8
woven_068	98.5	94.0	95.8	97.6
woven_125	99.5	96.4	100	100
Marbled_078	99.1	95.2	99.3	99.8
Perforated_037	96.3	88.4	94.4	98.2
Mesh_114	98.1	89.4	92.3	98.2
Fibrous_183	99.1	96.5	99.7	99.9
Matted_069	98.5	82.2	97.1	99.3
MEAN	**98.4**	**93.2**	**97.7**	**99.2**

Table 10: Single-category medical benchmark results. Values are copied from the corresponding medical rows in Table 1; "—" denotes unavailable metrics.

Dataset	Category	Pixel AUROC	Pixel AUPRO	Image AUROC	Image AP
HeadCT	single category	—	—	93.9	93.9
Br35H	single category	—	—	97.3	97.5
ISIC	single category	94.9	89.0	—	—
CVC-ClinicDB	single category	87.2	74.5	—	—
CVC-ColonDB	single category	83.4	74.8	—	—
Kvasir	single category	84.8	48.8	—	—