WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation

Jongheon Jeong[2*†] Yang Zou[1*] Taewan Kim[1]
Dongqing Zhang[1] Avinash Ravichandran[1‡] Onkar Dabeer[1]
[1] AWS AI Labs [2] KAIST

Abstract

Visual anomaly classification and segmentation are vital for automating industrial quality inspection. The focus of prior research in the field has been on training custom models for each quality inspection task, which requires task-specific images and annotation. In this paper we move away from this regime, addressing zero-shot and few-normal-shot anomaly classification and segmentation. Recently CLIP, a vision-language model, has shown revolutionary generality with competitive zero-/few-shot performance in comparison to full-supervision. But CLIP falls short on anomaly classification and segmentation tasks. Hence, we propose window-based CLIP (WinCLIP) with (1) a compositional ensemble on state words and prompt templates and (2) efficient extraction and aggregation of window/patch/image-level features aligned with text. We also propose its few-normal-shot extension WinCLIP+, which uses complementary information from normal images. In MVTec-AD (and VisA), without further tuning, WinCLIP achieves $91.8\%/85.1\%$ $(78.1\%/79.6\%)$ *AUROC in zero-shot anomaly classification and segmentation while WinCLIP+ does* $93.1\%/95.2\%$ $(83.8\%/96.4\%)$ *in 1-normal-shot, surpassing state-of-the-art by large margins.*

1. Introduction

Visual anomaly classification (AC) and segmentation (AS) classify and localize defects in industrial manufacturing, respectively, predicting an image or a pixel as normal or anomalous. Visual inspection is a long-tail problem. The objects and their defects vary widely in color, texture, and size across a wide range of industrial domains, including aerospace, automobile, pharmaceutical, and electronics. These result in two main challenges in the field.

First, defects are rare with wide range of variations, leading to a lack of representative anomaly samples in the training data. Consequently, existing works have mainly focused on one-class or unsupervised anomaly detection [2, 7, 8, 20, 29, 31, 53, 59], which only requires normal images. These methods typically fit a model to the normal images and treat any deviations from it as anomalous. When hundreds or thousands of normal images are available, many methods achieve high-accuracy on public benchmarks [3, 8, 31]. But in the few-normal-shot regime, there is still room to improve performance [14, 32, 39, 59], particularly in comparison with the fully-supervised upper bound.

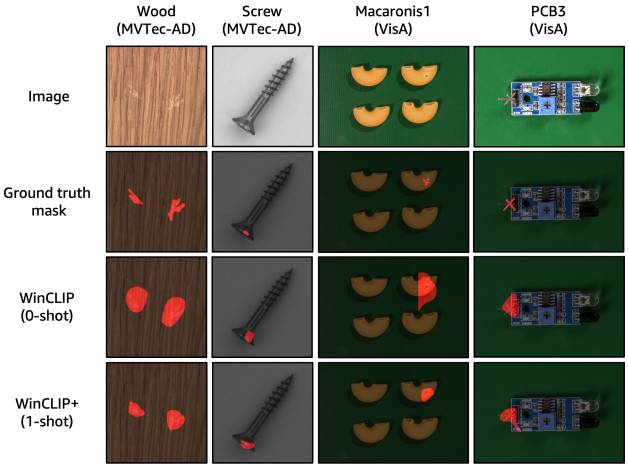


Figure 1. Language guided zero-/one-shot[1] anomaly segmentation from WinCLIP/WinCLIP+. Best viewed in color and zoom in.

Second, prior work has focused on training a bespoke model for each visual inspection task, which is not scalable across the long-tail of tasks. This motivates our interest in zero-shot anomaly classification and segmentation. But many defects are defined with respect to a normal image. For example, a missing component on a circuit board is most easily defined with respect to a normal circuit board with all components present. For such cases, at least a few normal images are needed. So in addition to the zero-shot case, we also consider the case of few-normal-shot anomaly classification and segmentation. Since only few normal images are available, there is no segmentation supervision for localizing anomalies, making this a challenging problem across the long-tail of tasks.

[†] Work done during an Amazon internship.
[*] The authors contributed equally.
[‡] Work done as part of AWS AI Labs.
[1] *few-shot* and *few-normal-shot* are used interchangeably in our case.

WinCLIP：零样本/少样本异常分类与分割

Jongheon Jeong[2*†] Yang Zou[1*] Taewan Kim[1] Dongqing Zhang[1] Avinash Ravichandran[1‡] Onkar Dabeer[1] [1] AWS AI 实验室 [2] 韩国科学技术院

摘要

Visual anomaly classification and segmentation are vital for automating industrial quality inspection. The focus of prior research in the field has been on training custom models for each quality inspection task, which requires task-specific images and annotation. In this paper we move away from this regime, addressing zero-shot and few-normal-shot anomaly classification and segmentation. Recently CLIP, a vision-language model, has shown revolutionary generality with competitive zero-/few-shot performance in comparison to full-supervision. But CLIP falls short on anomaly classification and segmentation tasks. Hence, we propose window-based CLIP (WinCLIP) with (1) a compositional ensemble on state words and prompt templates and (2) efficient extraction and aggregation of window/patch/image-level features aligned with text. We also propose its few-normal-shot extension WinCLIP+, which uses complementary information from normal images. In MVTec-AD (and VisA), without further tuning, WinCLIP achieves 91.8%/85.1% (78.1%/79.6%) *AUROC in zero-shot anomaly classification and segmentation while WinCLIP+ does* 93.1%/95.2% (83.8%/96.4%) *in 1-normal-shot, surpassing state-of-the-art by large margins.*

1. 引言

视觉异常分类（AC）与分割（AS）分别用于工业制造中的缺陷分类与定位，预测图像或像素为正常或异常。视觉检测是一个长尾问题。在航空航天、汽车、制药和电子等广泛的工业领域中，物体及其缺陷在颜色、纹理和尺寸上差异巨大。这导致了该领域面临两大主要挑战。

首先，缺陷种类繁多且罕见，导致在训练数据。因此，现有研究主要集中于单类别或无监督异常检测[2,7,8,20,29,31,53,59]，这类方法仅需正常图像。这些方法通常将模型拟合到正常图像上，并将任何偏离该模型的情况视为异常。当可获得数百或数千张正常图像时，许多方法在公开基准测试中取得了高精度[3,8,31]。但在少样本正常图像的情况下，性能仍有提升空间[14,32,39,59]，尤其是与全监督上限相比。

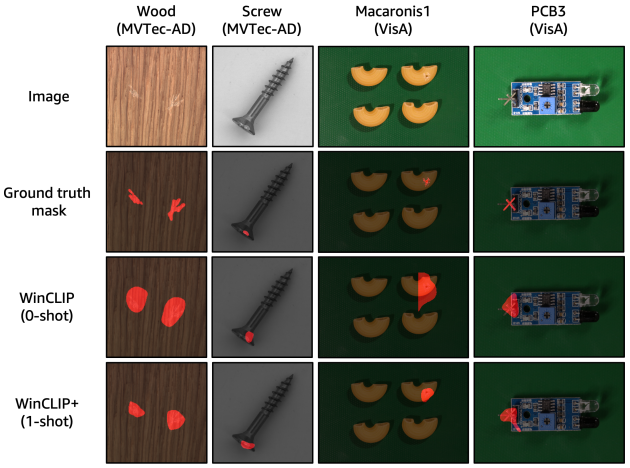


图1. 来自WinCLIP/WinCLIP+的语言引导零样本/单样本[1]异常分割。建议彩色查看并放大。

其次，先前的研究侧重于为每个视觉检测任务训练定制模型，这在任务的长尾分布中不具备可扩展性。这激发了我们对零样本异常分类与分割的兴趣。但许多缺陷的定义依赖于正常图像。例如，电路板上缺失元件的情况，最易通过对比所有元件齐全的正常电路板图像来界定。对于此类情况，至少需要少量正常图像。因此除了零样本场景外，我们还考虑了少样本正常图像的异常分类与分割场景。由于仅能获取少量正常图像，且缺乏用于定位异常的分割监督信号，这使得该问题在长尾任务中具有挑战性。

[†] Work done during an Amazon internship.
[*] The authors contributed equally.
[‡] Work done as part of AWS AI Labs.
[1] *few-shot* and *few-normal-shot* are used interchangeably in our case.

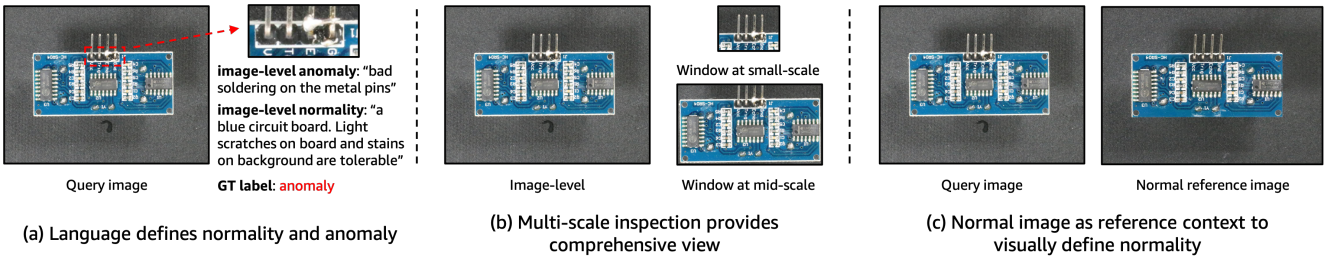


Figure 2. Motivation of language guided visual inspection. (a) Language helps describe and clarify normality and anomaly; (b) Aggregating multi-scale features helps identify local defects; (c) Normal images provide rich referencing content to visually define normality

Vision-language models [1, 18, 27, 36] have shown promise in zero-shot classification tasks. Large-scale training with vision-language annotated pairs learns expressive representations that capture broad concepts. Without additional fine-tuning, text prompts can then be used to extract knowledge from such models for zero-/few-shot transfer to downstream tasks including image classification [27], object detection [11] and segmentation [46]. Since CLIP is one of the few open-source vision-language models, these works build on top of CLIP, benefiting from its generalization ability, and showing competitive low-shot performances in both seen and unseen objects compared to full supervision.

In this paper, we focus on zero-shot and few-normal-shot (1 to 4) regime, which has received limited attention [14, 32, 39]. Our hypothesis is that language is perhaps even more important for zero-shot/few-normal-shot anomaly classification and segmentation. This hypothesis stems from multiple observations. First, "normal" and "anomalous" are states [17] of an object that are context-dependent, and language helps clarify these states. For example, "a hole in a cloth" may be a desirable or undesirable depending upon whether distressed fashion or regular fashion clothes are being manufactured. Language can bring such context and specificity to the broad "normal" and "anomalous" states. Second, language can provide additional information to distinguish defects from acceptable deviations from normality. For example, in Figure 2(a), language provides information on the soldering defect, while minor scratches/stains on background are acceptable. In spite of these advantages, we are not aware of prior work leveraging vision-language models for anomaly classification and segmentation. In this work, with the pre-trained CLIP as a base model, we show and verify our hypothesis that language aids zero-/few-shot anomaly classification/segmentation.

Since CLIP is one of the few open-source vision-language models, we build on top of it. Previously, CLIP-based methods have been applied for zero-shot classification [27]. CLIP can be applied in the same way to anomaly classification, using text prompts for "normal" and "anomalous" as classes. However, we find naïve prompts are not effective (see Table 3). So we improve the naïve baseline with a state-level word ensemble to better describe normal and anomalous states. Another challenge is that CLIP is trained to enforce cross-modal alignment only on the global embeddings of image and text. However, for anomaly segmentation we seek pixel-level classification and it is non-trivial to extract dense visual features aligned with language for zero-shot anomaly segmentation. Therefore, we propose a new *Window-based CLIP* (WinCLIP), which extracts and aggregates the multi-scale features while ensuring vision-language alignment. The multiple scales used are illustrated in Figure 2(b). To leverage normal images available in the few-normal-shot setting, we introduce WinCLIP+, which aggregates complementary information from the language driven WinCLIP and visual cues from the normal reference images, such as the one shown in Figure 2(c). We emphasize that our zero-shot models do not require any tuning for individual cases, and the few-normal-only setup does not use any segmentation annotation, facilitating applicability across a broad range of visual inspection tasks. As a sample, Figure 1 illustrates WinCLIP and WinCLIP+ qualitative results for a few cases.

To summarize, our main contributions are:

- We introduce a compositional prompt ensemble, which improves zero-shot anomaly classification over the naïve CLIP based zero-shot classification.
- Using the pre-trained CLIP model, we propose WinCLIP, that efficiently extract and aggregate multi-scale spatial features aligned with language for zero-shot anomaly segmentation. As far as we know, we are the first to explore language-guided zero-shot anomaly classification and segmentation.
- We propose a simple reference association method, which is applied to multi-scale feature maps for image based few-shot anomaly segmentation. WinCLIP+ combines the language-guided and vision-only methods for few-normal-shot anomaly recognition.
- We show via extensive experiments on MVTec-AD and VisA benchmarks that our proposed methods WinCLIP/WinCLIP+ outperform the state-of-the-art methods in zero-/few-shot anomaly classification and segmentation with large margins.

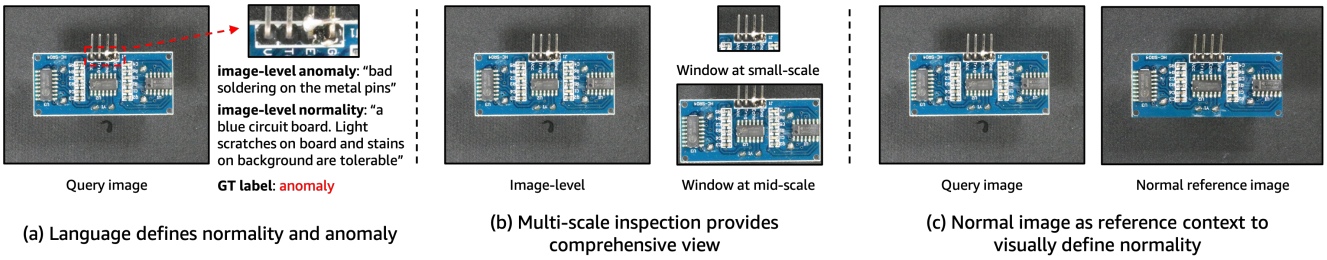


图2. 语言引导视觉检测的动机。(a) 语言有助于描述和澄清正常与异常状态；(b) 聚合多尺度特征有助于识别局部缺陷；(c) 正常图像提供丰富的参考内容，以视觉方式定义正常状态

视觉语言模型[1, 18, 27, 36]在零样本分类任务中展现出潜力。通过大规模视觉语言标注对的训练，这些模型学习了能够捕捉广泛概念的强表征能力。无需额外微调，文本提示即可用于从此类模型中提取知识，实现向下游任务（如图像分类[27]、目标检测[11]和分割[46]）的零样本/少样本迁移。由于CLIP是少数开源的视觉语言模型之一，这些研究均基于CLIP构建，受益于其泛化能力，并在可见与不可见对象的识别中，相比全监督方法展现出具有竞争力的低样本性能。

本文聚焦于零样本和少正常样本（1至4个）机制，该领域目前受到的关注有限[14, 32, 39]。我们的假设是：语言对于零样本/少正常样本的异常分类与分割任务可能具有更关键的作用。这一假设源于多重观察：首先，“正常”与“异常”是物体在特定语境下呈现的状态[17]，而语言有助于澄清这些状态。例如，“布料上的破洞”可能是理想特征也可能属于缺陷，具体取决于生产的是做旧风格时装还是常规时装。语言能为宽泛的“正常”与“异常”状态注入语境信息与具体内涵。其次，语言能提供额外信息以区分缺陷与可接受的正常偏差。例如在图2(a)中，语言明确了焊接缺陷的界定，而背景上的细微划痕/污渍则属于可接受范围。尽管存在这些优势，目前尚未有研究利用视觉-语言模型进行异常分类与分割。本研究以预训练的CLIP为基础模型，通过实验证实了我们的假设：语言能有效辅助零样本/少样本的异常分类与分割任务。

由于CLIP是少数开源的视觉语言模型之一，我们基于它进行构建。此前，基于CLIP的方法已被应用于零样本分类[27]。CLIP可以以同样的方式应用于异常分类，使用“正常”和“异常”的文本提示作为类别。然而，我们发现朴素的提示并不有效（见表3）。因此，我们通过状态级方法改进了朴素基线。

词集成以更好地描述正常和异常状态。另一个挑战在于，CLIP的训练仅针对图像和文本的全局嵌入强制跨模态对齐。然而，对于异常分割，我们需要像素级分类，并且提取与语言对齐的密集视觉特征以实现零样本异常分割并非易事。因此，我们提出了一种新的*Window-basedCLIP* (WinCLIP)，它在确保视觉-语言对齐的同时提取并聚合多尺度特征。所使用的多尺度特征如图2(b)所示。为了利用少样本正常设置中可用的正常图像，我们引入了WinCLIP+，它聚合了来自语言驱动的WinCLIP的互补信息以及来自正常参考图像（如图2(c)所示）的视觉线索。我们强调，我们的零样本模型无需针对个别案例进行任何调整，且少样本正常设置不使用任何分割标注，这有助于在广泛的视觉检测任务中应用。作为示例，图1展示了WinCLIP和WinCLIP+在几个案例中的定性结果。总而言之，我们的主要贡献是：

- 我们引入了组合式提示集成方法，该方法相较于基于CLIP的朴素零样本分类，在零样本异常分类方面取得了改进。
- 利用预训练的CLIP模型，我们提出了Win-CLIP，它能高效提取并聚合与语言对齐的多尺度空间特征，用于零样本异常分割。据我们所知，我们是首个探索语言引导的零样本异常分类与分割的方法。
- 我们提出了一种简单的参考关联方法，该方法应用于多尺度特征图，用于基于图像的少样本异常分割。WinCLIP{v*}结合了语言引导和纯视觉方法，用于少正常样本的异常识别。
- 我们在MVTec-AD和VisA基准上进行了大量实验，结果表明我们提出的方法WinCLIP/WinCLIP+在零样本/少样本异常分类与分割任务中，以显著优势超越了当前最先进的方法。

2. Related work

Vision-language modeling. Among recent successes of large pre-trained vision-language models (VLM) [1, 18, 27], CLIP [27] is the first to perform pre-training on web-scale image-text data, showing unprecedented generality: *e.g.*, its language-driven zero-shot inference, improved both effective robustness [40] and perceptual alignment [10]. Many following VLM works explored large-scale pre-training in different aspects, *e.g.*, scaling up data [18], efficient designs [1,21,47], multi-tasks [22,43], *etc.* To democratize large-scale VLM for the usages in different domains, a billion-scale data LAION-5B [36], a code base of OpenCLIP with pre-trained models [16] are open-sourced. Other works presented CLIP's promise in zero-/few-shot transfer to downstream tasks beyond classification [11,30,41,46]. Good prompt engineering and tuning can non-trivially benefit generalization performances [27, 58]. Moreover, some other works [28, 56, 57] leverage the pre-trained CLIP for language guided detection and segmentation with promising performances.

Anomaly classification and segmentation. Due to the scarcity of anomalies, the major focus has been on one-class methods with many normal images [7, 8, 20, 49, 51, 53]. While the MVTec-AD benchmark [3] is saturated by several works [31, 48, 51], their specific application is hindered due to their unscalable full-normal-shot setup. Recent works [32, 39] explored few-shot setups by leveraging augmentation to expand the small support set for better normality modeling. RegAD [14] further proposed a model-reusing by pre-training an object-agnostic registration network with diverse images to model normality for unseen object, given a few normal samples. Meanwhile, to close the gap between academical and industrial data, Visual Anomaly (VisA) [59] is introduced for a challenging benchmark over MVTec-AD. Additionally, Vision Transformer (ViT) have recently shown its potential in visual inspection [9, 25].

State classification. In some sense, anomaly classification is related to state classification [17] that predicts if an object is normal or anomalous. While the major works in computer vision focus on object, scene, or material recognition [13, 34, 38, 45], state classification aims to differentiate the fine-grained sub-object physical properties or attributes. Several datasets covering generic states/attributes (e.g. tall, crack, red, smooth) over diverse objects and scenes are introduced [15, 17, 23, 50]. Some works [24, 26, 44] built graphs consisting of attributes and objects, of which relationship is learnt by graph neural networks [54].

3. Background

Anomaly classification and segmentation. Given an image $\mathbf{x} \in \mathcal{X}$, both anomaly classification and segmentation (ACS) aim to predict "abnormality" in $\mathbf{x}$. Specifically, we consider anomaly classification (AC) as a binary classification $\mathcal{X} \to \{-, +\}$ where "+" indicates the presence of anomaly in image-level. And anomaly segmentation (AS) is its pixel-level extension to output the location of anomalies via $\mathcal{X} \to \{-, +\}^{h \times w}$ for a certain image with size $h \times w$. In practice, the tasks are often cast into problems of predicting anomaly scores. For example, anomaly classification typically models a mapping $\mathrm{ascore} : \mathcal{X} \to [0, 1]$ so that a binary classification can be performed by thresholding $\mathrm{ascore}(\mathbf{x})$.

Due to the lack of anomalous (or positive) samples in practice, the one-class scenario, where the training data $\mathcal{D} := \{(x_i, -)\}_{i=1}^{K}$ consists of only normal (or negative) samples, has been widely used. In this paper, we follow the one-class protocol, particularly focusing on extreme cases of few-shot ($K = 1$ to 4) and the unexplored zero-shot setups for both AC and AS. And we assume an available list of task-specific texts tags, *e.g.*, for objects and relevant defects.

Zero-shot classification with CLIP. *Contrastive Language Image Pre-training* (CLIP) [27] is a large-scale pre-training method offering a joint vision-language representation. Given million-scale image-text pairs $\{(x_t, s_t)\}_{t=1}^{T}$ from the web, CLIP trains an image encoder f and a text encoder g via contrastive learning [6, 55] to maximize the correlation between $f(x_t)$ and $g(s_t)$ across t in terms of cosine similarity $\langle f(\mathbf{x}), g(\mathbf{s}) \rangle$. Given an input $\mathbf{x}$ and a closed set of free-form texts $S = \{s_1, \cdots, s_k\}$, CLIP can perform zero-shot classification via a k-way categorical distribution:

$$p(\mathbf{s} = s_i | \mathbf{x}; \mathbf{s} \in S) := \frac{\exp(\langle f(\mathbf{x}), g(s_i) \rangle / \tau)}{\sum_{s \in S} \exp(\langle f(\mathbf{x}), g(s) \rangle / \tau)}, \quad (1)$$

where $\tau > 0$ is the temperature hyperparameter.

For a set of class words $C = \{c_1, \cdots, c_k\}$, it has shown that accompanying each label word $c \in C$ with a *prompt template*, *e.g.*, "`a photo of a [c]`", improves accuracy over the case without templates. Moreover, an ensemble of prompt embeddings that aggregates multiple (80) templates *e.g.*, "`a cropped photo of a [c]`", can further boost the performance [27]. Overall, we are essentially "retrieving" the visual knowledge of CLIP through the language interface in appropriate manners. In this paper, we further explore how to extract the knowledge of CLIP in a way more suitable for anomaly recognition.

4. WinCLIP and WinCLIP+

In this section, we first establish a novel binary zero-shot anomaly classification framework with a Compositional Prompt Ensemble to improve CLIP for anomaly classification (Section 4.1). Next, we propose a simple-yet-effective *Window-based CLIP* (WinCLIP) for efficient zero-shot anomaly segmentation (Section 4.2). Lastly, we propose an extension *WinCLIP+* to benefit from few normal reference images, while maintaining the complementary benefits of language-guided predictions (Section 4.3).

2. 相关工作

视觉语言建模。在近期大规模预训练视觉语言模型（VLM）的成功案例中[1, 18, 27]，CLIP[27]率先在互联网规模的图文数据上进行预训练，展现出前所未有的泛化能力：*e.g.*。其基于语言的零样本推理机制，同时提升了有效鲁棒性[40]与感知对齐能力[10]。后续许多VLM研究从不同维度探索了大规模预训练，*e.g.*。包括数据规模扩展[18]、高效架构设计[1,21,47]、多任务学习[22, 43]*etc.*。为促进大规模VLM在不同领域的普及应用，数十亿数据规模的LAION-5B[36]以及包含预训练模型的OpenCLIP代码库[16]均已开源。其他研究展示了CLIP在分类任务之外的下游任务中实现零样本/少样本迁移的潜力[11,30,41,46]。优质的提示工程与微调能显著提升泛化性能[27, 58]。此外，另有研究[28, 56, 57]利用预训练的CLIP模型实现语言引导的检测与分割任务，并取得了优异性能。

异常分类与分割。由于异常样本稀缺，研究重点主要集中于使用大量正常图像的单类方法[7, 8, 20, 49, 51, 53]。虽然MVTec-AD基准测试[3]已被多项研究[31, 48, 51]充分挖掘，但其不可扩展的全正常样本设置限制了具体应用。近期研究[32, 39]通过数据增强扩展小规模支持集以改进正态建模，探索了少样本设置。RegAD[14]进一步提出模型复用方案：利用多样图像预训练与物体无关的配准网络，在给定少量正常样本时对未知物体进行正态建模。同时，为弥合学术与工业数据间的差距，Visual Anomaly (VisA)[59]被提出作为比MVTec-AD更具挑战性的基准测试。此外，视觉Transformer（ViT）近期在视觉检测领域展现出潜力[9, 25]。

状态分类。在某种意义上，异常分类与状态分类相关[17]，后者预测物体是否正常或异常。尽管计算机视觉领域的主要工作集中在物体、场景或材质识别上[13, 34, 38, 45]，状态分类旨在区分细粒度的子物体物理属性或特征。已有研究引入了多个涵盖不同物体和场景的通用状态/属性数据集（例如高、裂纹、红色、光滑）[15, 17, 23, 50]。部分工作[24, 26, 44]构建了由属性和物体组成的图结构，并通过图神经网络学习其关系[54]。

3. 背景

异常分类与分割。给定图像$\mathbf{x} \in \mathcal{X}$，异常分类与分割（ACS）的目标是预测$\mathbf{x}$中的“异常”。具体而言，我们将异常分类（AC）视为二元分类——其中“+”表示图像级别存在异常。而异常分割（AS）是其像素级别的扩展，通过$\mathcal{X} \to \{-,+\}^{h\times w}$为尺寸为$h \times w$的特定图像输出异常位置。在实践中，这些任务通常被转化为预测异常分数的问题。例如，异常分类通常建模一个映射 ascore : $\mathcal{X} \to \to$ [0, 1]，从而可以通过对 ascore(**x**) 进行阈值处理来执行二元分类。

由于实践中缺乏异常（或正）样本，仅包含正常（或负）样本的训练数据 $\mathcal{D} := \{(x_i, -)\}_{i=1}^{K}$ 的单类场景已被广泛使用。本文遵循单类协议，特别关注少样本（$K = 1$ 至 4）的极端情况以及 AC 和 AS 中尚未探索的零样本设置。同时，我们假设存在一个任务特定的文本标签列表 *e.g.*，用于描述对象及相关缺陷。

使用CLIP进行零样本分类。*Contrastive Language Image Pre-training* (CLIP) [27] 是一种大规模预训练方法，提供联合视觉-语言表示。给定来自网络的百万级图像-文本对$\{(x_t, s_t)\}_{t=1}^{T}$，CLIP通过对比学习[6, 55]训练图像编码器f和文本编码器g，以最大化$f(x_t)$和$g(s_t)$在t上的余弦相似度$\langle f(\mathbf{x}), g(\mathbf{s})\rangle$相关性。给定输入$\mathbf{x}$和一组封闭的自由形式文本$S = \{s_1, \cdots, s_k\}$，CLIP可以通过$k$路分类分布进行零样本分类:

$$p(\mathbf{s} = s_i|\mathbf{x}; \mathbf{s} \in S) := \frac{\exp(\langle f(\mathbf{x}), g(s_i)\rangle/\tau)}{\sum_{s\in S}\exp(\langle f(\mathbf{x}), g(s)\rangle/\tau)}, \quad (1)$$

其中 $\tau > 0$ 是温度超参数。

对于一组类别词$C = \{c_1, \cdots, c_k\}$，研究表明，为每个标签词$c \in C$搭配一个*prompt tem-plate*，*e.g.*，例如“一张[c]的照片”，相比不使用模板的情况能提升准确率。此外，通过聚合多个（80个）模板的提示嵌入集合*e.g.*，例如“一张[c]的特写照片”，可以进一步提升性能[27]。总体而言，我们本质上是在通过恰当的方式，从CLIP的语言接口中“检索”其视觉知识。本文将进一步探索如何以更适合异常识别的方式提取CLIP的知识。

4. WinCLIP 与 WinCLIP+

在本节中，我们首先建立了一种新颖的二元零样本异常分类框架，通过组合提示集成来改进CLIP在异常分类上的性能（第4.1节）。接着，我们提出了一种简单而有效的*Window-based CLIP* (WinCLIP)方法，用于高效的零样本异常分割（第4.2节）。最后，我们提出了一种扩展*WinCLIP+*，以利用少量正常参考图像的优势，同时保持语言引导预测的互补益处（第4.3节）。

4.1. Language-driven zero-shot AC

Two-class design. We introduce a binary zero-shot anomaly classification framework *CLIP-AC* by adapting CLIP with two class prompts `[c]` - "`normal [o]`" *vs.* "`anomalous [o]`". `[o]` is an object-level label, *e.g.*, "`bottle`" when available, or simply "`object`". In addition, we also test a one-class design by only using the normal prompt $s_- :=$ "`normal [o]`" to define anomaly score as "$-\langle f(\mathbf{x}), g(s_-)\rangle$". We observe the simple two-class design from CLIP already yields a non-trial performance and outperforms one-class design significantly in experiments (Table 3). This demonstrates (a) CLIP pre-trained by large web dataset provides a powerful representation with good alignment between text and images for anomaly tasks (b) specific definition about anomaly is necessary for good performance.

Compositional prompt ensemble (CPE). Unlike object-level classifiers, CLIP-AC performs classification between two *states* of a given object, *i.e.*, either "normal" or "anomalous", which are subjective with various definitions depending on tasks. For example, "missing transistor" is "anomalous" for a circuit board while "cracked" is "anomalous" for wood. To better define the two abstract states of objects, we propose a Compositional Prompt Ensemble to generate all combinations of pre-defined lists of (a) *state words* per label and (b) *text templates*, rather than freely writing definitions. The state words include common states shared by most objects, *e.g.*, "flawless" for normality/"damaged" for anomaly. Also we can optionally add task-specific state words given prior knowledge of defects, *e.g.*, "bad soldering" on PCB. Moreover, we curate a template list specifically for anomaly tasks *e.g.*, "`a photo of a [c] for visual inspection`". Check details on prompt engineering in supplementary. As in top-left of Figure 4, after getting all the combinations of states and templates, we compute the average of text embeddings per label to represent the normal and anomalous classes. Note that CPE is different from CLIP prompt ensemble that does not explain object labels (*e.g.*, "cat") and only augments templates selected by trial-and-error for object classification, including the ones unsuitable for anomaly tasks, *e.g.*, "`a cartoon [c]`". Thus, the texts from CPE are more aligned with images in CLIP's joint embedding space for anomaly tasks. We denote the zero-shot scoring model with CPE as $\text{ascore}_0 : \mathbb{R}^d \to [0, 1]$ for an image embedding $f(\mathbf{x})$.

Remark. Our two-class design with CPE is a novel approach to define anomaly compared to standard one-class methods [31, 33]. Anomaly detection is an ill-posed problem due to the open-ended nature. Previous methods model normality only by normal images regarding any deviation from normality as anomaly. Such solution is by nature hard to distinguish true anomalies from acceptable deviations from normality, *e.g.*, "scratch on circuit" vs. "tiny yet acceptable scratch". But language can define states in concrete words.

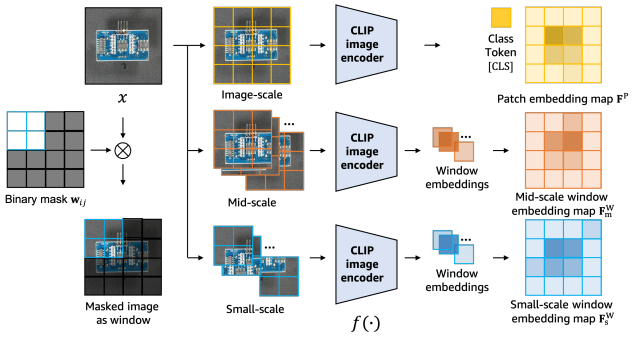


Figure 3. WinCLIP feature extraction in multiple scales of windows through CLIP image encoder, *e.g.*, ViT taking a sequence of (non-masked) patches as input. Window embeddings encode the global information (*e.g.*, from the class token) within each window.

4.2. WinCLIP for zero-shot AS

Given the language guided anomaly scoring model from CPE, we propose Window-based CLIP (WinCLIP) for zero-shot anomaly segmentation to predict pixel-level anomalies. WinCLIP extracts dense visual features with good language alignment and local details for $\mathbf{x}$, followed by applying ascore_0 spatially to obtain the anomaly segmentation map. Specifically, given an image $\mathbf{x}$ of resolution $h \times w$ and an image encoder f, WinCLIP obtains a map of d-dimensional feature map $\mathbf{F}^{\mathtt{W}} \in \mathbb{R}^{h \times w \times d}$ as follows:

1. Generate a set of sliding windows $\{\mathbf{w}_{ij}\}_{ij}$, where each window $\mathbf{w}_{ij} \in \{0, 1\}^{h \times w}$ is a binary mask that is active locally for a $k \times k$ kernel around (i, j).

2. Collect each output embedding $\mathbf{F}^{\mathtt{W}}_{ij}$, computed from the active area of $\mathbf{x}$ after applying each $\mathbf{w}_{ij}$, defined by:

$$\mathbf{F}^{\mathtt{W}}_{ij} := f(\mathbf{x} \odot \mathbf{w}_{ij}), \tag{2}$$

where $\odot$ is the element-wise product (see Figure 3).

Figure 3 illustrates the dense feature extraction of WinCLIP with ViT while it is also applicable to CNN.

In addition, we also explore a natural dense representation candidate, *penultimate feature map*, the last feature map before pooling. Specifically, for patch embedding map $\mathbf{F}^{\mathtt{P}}$ (other than the *class token* `[CLS]`) of ViT-based CLIP, top of Figure 3, we apply ascore_0 patch-wisely for segmentation. However, we observe that such patch-level features are not aligned with the language space, leading to a poor dense predictions (Table 8). We conjecture this is caused by those features have not been directly supervised with language signal in CLIP. Also these patch features have already aggregated the global context due to self-attention, hindering capturing local details for segmentation.

Compared to the penultimate features $\mathbf{F}^{\mathtt{P}}$, we remark dense features from WinCLIP is more aligned with language: *e.g.*, for ViT-based CLIP, all the features in $\mathbf{F}^{\mathtt{W}}$ are now from

4.1. 语言驱动的零样本AC

二分类设计。我们通过调整CLIP，引入一个二元零样本异常分类框架*CLIP-AC*，使用两个类别提示[c]——“正常[o]”*vs.*“异常[o]”。[o]是对象级标签，*e.g.*，例如“瓶子”（当可用时），或简单的“对象”。此外，我们还测试了一种单分类设计，仅使用正常提示s_-：=“正常[o]”，将异常分数定义为“$-\langle f(\mathbf{x}), g(s_-)\rangle$”。我们观察到，CLIP的简单二分类设计已经产生了非平凡的性　　能，并在实验中显著优于单分类设计3)表 这表明（a）通过大型网络数据集预训练的CLIP为异常任务提供了强大的表示能力，实现了文本与图像之间的良好对齐；（b）明确的异常定义对于获得良好性能是必要的。组合提示集成（CPE）。与对象级分类器不同，CLIP-AC在给定对象的两种*states*之间进行分类，*i.e.*，即“正常”或“异常”，这些状态是主观的，其定义因任务而异。例如，“缺失晶体管”对电路板来说是“异常”，而“裂纹”对木材来说是“异常”。为了更好地定义对象的两种抽象状态，我们提出了组合提示集成，以生成预定义列表（a）每个标签的*state words*和（b）*text templates*的所有组合，而不是自由编写定义。状态词包括大多数对象共享的常见状态，*e.g.*，例如“无瑕疵”表示正常/“损坏”表示异常。此外，我们还可以根据缺陷的先验知识选择性地添加任务特定的状态词，*e.g.*，例如PCB上的“不良焊接”。此外，我们专门为异常任务策划了一个模板列表，*e.g.*，例如“用于视觉检查的[c]照片”。提示工程的详细信息请参见补充材料。如图4左上角所示，在获得所有状态和模板的组合后，我们计算每个标签的文本嵌入平均值，以表示正常和异常类别。请注意，CPE与CLIP提示集成不同，后者不解释对象标签（*e.g.*，例如“猫”），并且仅通过试错选择模板进行对象分类，包括那些不适合异常任务的模板，*e.g.*，例如“卡通[c]”。因此，CPE生成的文本在CLIP的联合嵌入空间中与异常任务的图像更对齐。我们将使用CPE的零样本评分模型表示为ascore$_0$：$\mathbb{R}^d \to [0, 1]$，用于图像嵌入$f(\mathbf{x})$。

备注。我们采用CPE的双类别设计是一种新颖的异常定义方法，相较于标准单类别方法[31,33]有所突破。由于开放性的本质，异常检测是一个不适定问题。先前的方法仅通过正常图像来建模正常状态，并将任何偏离正常状态的情况视为异常。这种方法本质上难以区分真实异常与可接受的正常偏差，*e。g。*例如，“电路板上的划痕”与“微小但可接受的划痕”。但语言能够用具体的词汇定义状态。

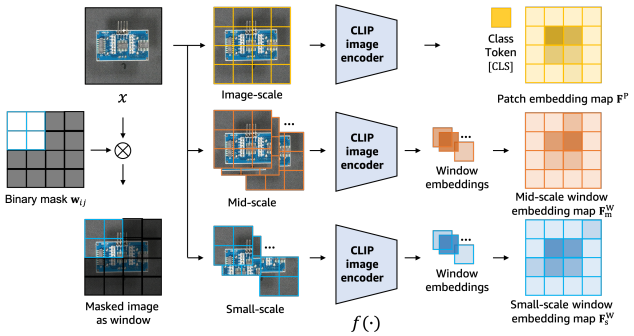


图3. 通过CLIP图像编码器在多个窗口尺度上进行WinCLIP特征提取，*e.g.*。ViT接收一系列（非掩码）图像块作为输入。窗口嵌入编码了每个窗口内的全局信息（*e.g.*。，来自类别标记）。

4.2. 用于零样本AS的WinCLIP

给定CPE的语言引导异常评分模型，我们提出了基于窗口的CLIP（WinCLIP）用于零样本异常分割，以预测像素级异常。WinCLIP为$\mathbf{x}$提取具有良好语言对齐性和局部细节的密集视觉特征，随后在空间上应用评分$_0$以获得异常分割图。具体而言，给定一张分辨率为$h \times w$的图像$\mathbf{x}$和一个图像编码器f，WinCLIP通过以下方式获得一个d维特征图$\mathbf{F}^{\mathrm{W}} \in \mathbb{R}^{h \times w \times d}$的映射：

1. 生成一组滑动窗口 {v*}，其中每个窗口 {v*}0{v*}1{v*} 是一个二值掩码，在 ({v*}) 周围 {v*} 大小的核内局部激活。

2. 收集每个输出嵌入 $\mathbf{F}^{\mathrm{W}}_{ij}$，这些嵌入是在应用每个 $\mathbf{w}_{ij}$ 后，由 $\mathbf{x}$ 的活动区域计算得出的，具体定义为：

$$\mathbf{F}^{\mathrm{W}}_{ij} := f(\mathbf{x} \odot \mathbf{w}_{ij}), \tag{2}$$

其中 $\odot$ 是逐元素乘积（见图3）。

图3展示了WinCLIP结合ViT的密集特征提取过程，该方法同样适用于CNN。

此外，我们还探索了一种自然的密集表示候选——池化前的最后一个特征图*penultimate feature map*。具体来说，对于ViT-based CLIP中除*class token*[CLS])标记外的补丁嵌入图$\mathbf{F}^{\mathrm{P}}$(（见图3顶部），我们采用逐补丁的分数$_0$进行分割。然而，我们观察到此类补丁级特征与语言空间未对齐，导致密集预测效果较差（见表8）。我们推测这是由于CLIP中这些特征未直接受语言信号监督所致。同时，这些补丁特征因自注意力机制已聚合全局上下文，反而阻碍了分割任务中对局部细节的捕捉。

与倒数第二层特征$\mathbf{F}^{\mathrm{P}}$相比，我们注意到WinCLIP提取的密集特征与语言的对应关系更为紧密：*e。g。*对于基于ViT的CLIP，$\mathbf{F}^{\mathrm{W}}$中的所有特征现在均来自

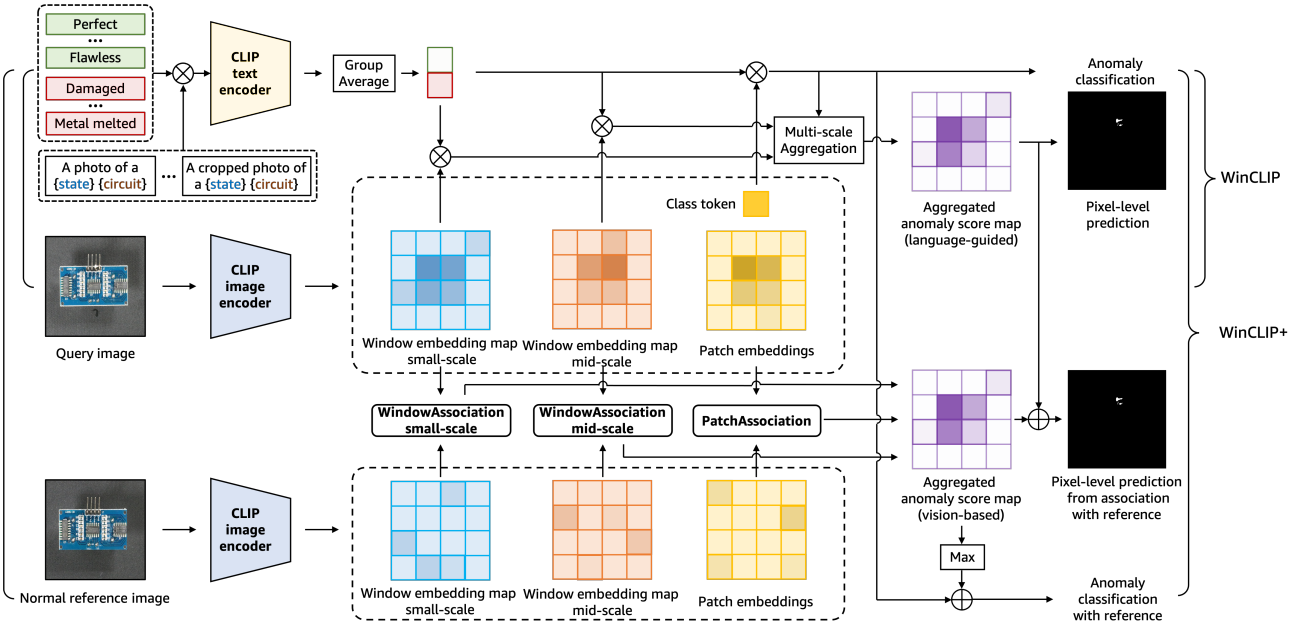


Figure 4. Workflows of WinCLIP/WinCLIP+ (upper/entire pane). Various states and templates are composited and converted to two text embeddings as class prototypes via CLIP text encoder (Section 4.1). The class prototypes are correlated with the multi-scale features from CLIP image encoder (Figure 3) for zero-shot AC/AS in WinCLIP. WinCLIP+ applies the reference association on patch, small-/mid-window (Patch/WindowAssociation) for vision-based anomaly score maps, which are aggregated for few-shot AS/AC with language-guided scores.

class tokens which are directly aligned to texts in CLIP pre-training. Also the features focus more on local details via sliding windows. Lastly, WinCLIP can be efficiently computed, especially with ViT architecture. Concretely, the computation of (2) can directly benefit from just dropping all the masked patches before forwarding them, in a similar manner to masked autoencoder [12].

Harmonic aggregation of windows. For each local window, the zero-shot anomaly score $\mathbf{M}^{\mathrm{W}}_{0,ij}$ is similarity between the window feature $\mathbf{F}^{\mathrm{W}}_{ij}$ and text embeddings from compositional prompt ensemble. This score is distributed to every pixel of the local window. Then at each pixel, we aggregate multiple scores from all overlapping windows to improve segmentation by *harmonic averaging* (3), weighting more on scores towards normality prediction (zero value).

$$\bar{\mathbf{M}}^{\mathrm{W}}_{0,ij} := \left(\frac{1}{\sum_{u,v} (\mathbf{w}_{uv})_{ij}} \sum_{u,v} \frac{(\mathbf{w}_{uv})_{ij}}{\mathbf{M}^{\mathrm{W}}_{0,uv}} \right)^{-1}. \quad (3)$$

Multi-scale aggregation. The kernel size k corresponds to the amount of surrounding context for each location in computing WinCLIP features (2). It controls the balance between local details and global information in segmentation. To capture defects of sizes ranging from small to large scale, we aggregate predictions from multi-scale features: *e.g.*, (a) small-scale (2×2 in patch scales of ViT; corresponds to 32×32 in pixels), (b) mid-scale (3×3 in ViT; 48×48), and (c) image-scale feature (ViT class token capturing image context due to self-attention). We also adopt harmonic averaging for aggregation. Figure 3 illustrates the features on each scale.

4.3. WinCLIP+ with few-normal-shots

For a comprehensive anomaly classification and segmentation, language guided zero-shot approach is not enough as certain defects can only be defined via visual reference rather than only text. For example, "Metal-nut" in MVTec-AD [3] has an anomaly type labeled as "flipped upside-down", which can only be identified relatively from a normal image. To define and recognize the anomalies more precisely, we propose an extension of WinCLIP, *WinCLIP+*, by incorporating K normal reference images $\mathcal{D} := \{(x_i, -)\}_{i=1}^{K}$. WinCLIP+ combines the complementary prediction from both language-guided and visual based approaches for better anomaly classification and segmentation.

We first propose a *reference association* as the key module to incorporate given reference images, which can simply store and retrieve the memory features $\mathbf{R}$ of $\mathcal{D}$ based on the cosine similarity. Given such module and the corresponding (*e.g.*, patch-level[2]) features $\mathbf{F} \in \mathbb{R}^{h \times w \times d}$ extracted from a query image, a prediction $\mathbf{M} \in [0,1]^{h \times w}$ for anomaly segmentation can be made by:

$$\mathbf{M}_{ij} := \min_{r \in \mathbf{R}} \tfrac{1}{2}(1 - \langle \mathbf{F}_{ij}, r \rangle). \quad (4)$$

Then we apply this association module at multiple scales of feature maps that are obtained from WinCLIP (see Fig-

[2]Nevertheless, the module is generally applicable for other scales.

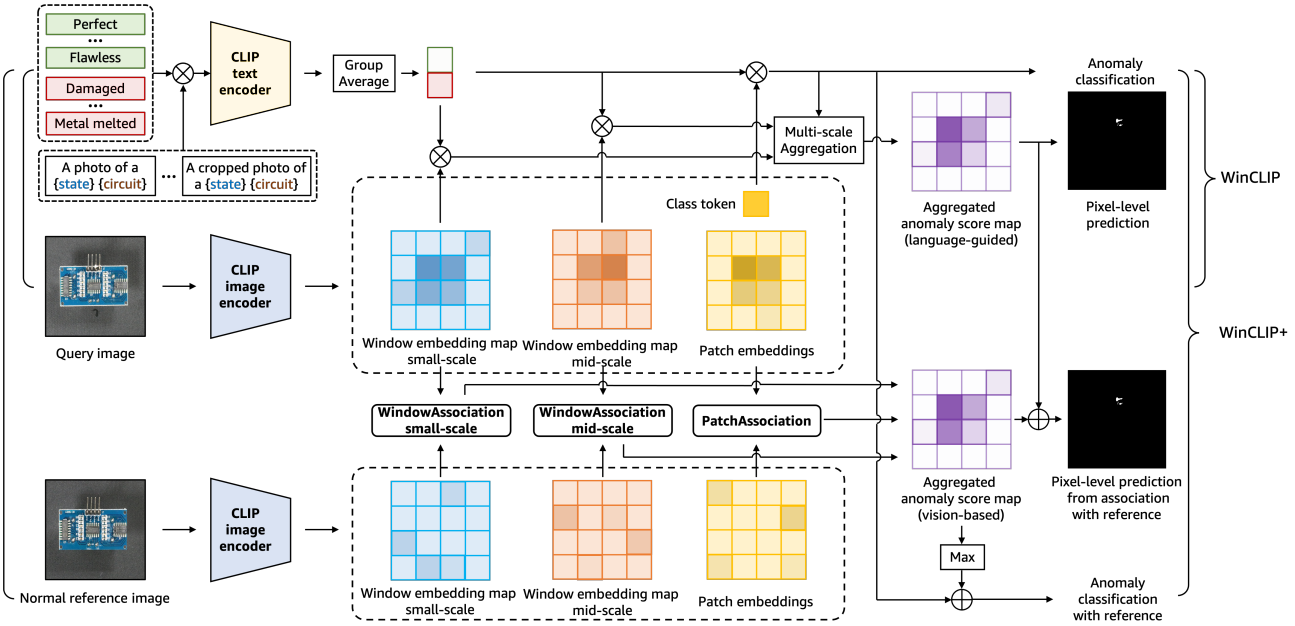


图4. WinCLIP/WinCLIP+ (上/全窗格)的工作流程。通过CLIP文本编码器（第4.1节），将各种状态和模板组合并转换为两个文本嵌入作为类别原型。这些类别原型与CLIP图像编码器提取的多尺度特征（图3）进行关联，以实现WinCLIP中的零样本异常分类/分割。WinCLIP+在补丁、小/中窗口（补丁/窗口关联）上应用参考关联以生成基于视觉的异常得分图，这些得分图与语言引导得分结合，用于少样本异常分割/分类。

类别标记直接与CLIP预训练中的文本对齐。此外，通过滑动窗口机制，特征更侧重于局部细节。最后，WinCLIP能够高效计算，尤其是在ViT架构中。具体而言，计算过程（2）可直接受益于仅丢弃所有掩码补丁再前向传播的方式，这与掩码自编码器[12]的处理方法类似。

窗口的谐波聚合。对于每个局部窗口，零样本异常分数$\mathbf{M}^{\mathtt{W}}_{0,ij}$是窗口特征$\mathbf{F}^{\mathtt{W}}_{ij}$与组合提示集成的文本嵌入之间的相似度。该分数被分配到局部窗口的每个像素上。然后在每个像素处，我们聚合来自所有重叠窗口的多个分数，通过*harmonic averaging* (3)来改进分割，对趋向于正常预测（零值）的分数赋予更高权重。

$$\bar{\mathbf{M}}^{\mathtt{W}}_{0,ij} := \left(\frac{1}{\sum_{u,v} (\mathbf{w}_{uv})_{ij}} \sum_{u,v} \frac{(\mathbf{w}_{uv})_{ij}}{\mathbf{M}^{\mathtt{W}}_{0,uv}} \right)^{-1}. \quad (3)$$

多尺度聚合。卷积核大小 k 对应计算WinCLIP特征时每个位置所考虑的周边上下文范围(2)，它控制着分割任务中局部细节与全局信息之间的平衡。为捕捉从小尺度到大尺度的缺陷，我们聚合了多尺度特征的预测结果：*e.g.*，包括：(a) 小尺度（ViT补丁尺度中的2×2；对应像素尺度32×32）、(b) 中尺度（ViT中的3×3；对应48×48）以及(c) 图像尺度特征（ViT类别令牌捕获的图像上下文）。由于自注意力机制。我们还采用调和平均进行聚合。图3展示了每个尺度上的特征。

4.3. 基于少量正常样本的WinCLIP+

对于全面的异常分类与分割，仅依赖语言引导的零样本方法是不够的，因为某些缺陷只能通过视觉参考而非纯文本来定义。例如，MVTec-AD [3] 中的“金属螺母”有一种标记为“上下翻转”的异常类型，这只能通过正常图像进行相对识别。为了更精确地定义和识别异常，我们提出了WinCLIP的扩展版本*WinCLIP+*，通过引入K正常参考图像$\mathcal{D} := \{(x_i, -)\}_{i=1}^{K}$。WinCLIP+结合了语言引导和视觉基础方法的互补预测，以实现更优的异常分类与分割。

我们首先提出一个*reference association*作为关键模块，用于整合给定的参考图像，该模块能够基于余弦相似度简单地存储和检索$\mathcal{D}$的记忆特征$\mathbf{R}$。给定该模块以及从查询图像中提取的对应(*e.g.*，块级[2])特征$\mathbf{F} \in \mathbb{R}^{h \times w \times d}$，可以通过以下方式生成用于异常分割的预测$\mathbf{M} \in [0,1]^{h \times w}$：

$$\mathbf{M}_{ij} := \min_{r \in \mathbf{R}} \tfrac{1}{2}(1 - \langle \mathbf{F}_{ij}, r \rangle). \quad (4)$$

随后，我们在从WinCLIP获取的多尺度特征图上应用此关联模块（见图-

[2]Nevertheless, the module is generally applicable for other scales.

ure 4 for the overall illustration). Specifically, given few-shot samples, we construct separate reference memories from three different features: (a) WinCLIP features at small-scale $\mathbf{F}^{\mathtt{W}}_{\mathtt{s}}$, (b) those at mid-scale $\mathbf{F}^{\mathtt{W}}_{\mathtt{m}}$, and also (c) from penultimate features $\mathbf{F}^{\mathtt{P}}$ with global context (*e.g.*, the patch tokens in ViT capturing image context due to self-attention). Even though $\mathbf{F}^{\mathtt{P}}$ is not aligned with language, it still useful to define normality and anomaly.

As a result, WinCLIP+ gets three reference memories: $\mathbf{R}^{\mathtt{W}}_{\mathtt{s}}$, $\mathbf{R}^{\mathtt{W}}_{\mathtt{m}}$, and $\mathbf{R}^{\mathtt{P}}$. Then, we average their multi-scale predictions (4) for anomaly segmentation for a given query,

$$\mathbf{M}^{\mathtt{W}} := \tfrac{1}{3}(\mathbf{M}^{\mathtt{P}} + \mathbf{M}^{\mathtt{W}}_{\mathtt{s}} + \mathbf{M}^{\mathtt{W}}_{\mathtt{m}}), \tag{5}$$

and then fusing with our language-guided prediction $\bar{\mathbf{M}}^{\mathtt{W}}_{0}$.

To perform anomaly classification, we combine the maximum value of $\mathbf{M}^{\mathtt{W}}$ and the WinCLIP zero-shot classification score. The two scores have complementary information to collaborative with, specifically (a) one from the spatial features of few-shot references, and (b) the other one from the CLIP knowledge retrieved via language:

$$\text{ascore}_{\mathtt{W}}(\mathbf{x}) := \tfrac{1}{2}\left(\text{ascore}_{0}(f(\mathbf{x})) + \max_{ij} \mathbf{M}^{\mathtt{W}}_{ij}\right). \tag{6}$$

5. Experiments

We perform an array of experiments to evaluate the performance of WinCLIP-based ACS under low-shot regimes, covering recent challenging benchmarks on industrial anomaly classification and segmentation that we are focusing on. We also conduct an extensive ablation study to validate the individual effectiveness of our proposed components. The detailed setups, *e.g.*, pre-processing, metrics, and other implementation details, are given in the supplementary.

Datasets. Our experiments are based on MVTec-AD [3] and VisA [59] datasets. Both benchmarks have diverse subsets of different objects, *e.g.*, capsules, circuit boards. They contain high-resolution images (*e.g.*, 700^2-1024^2 for MVTec-AD, and roughly $1.5\text{K} \times 1\text{K}$ for VisA) of common objects with the full pixel-level annotations.

Evaluation metrics. For classification, we report (a) *Area Under the Receiver Operating Characteristic* (AUROC) following the literature [8,31,49], as well as (b) *Area Under the Precision-Recall curve* (AUPR) and (c) F_1*-score at optimal threshold* (F_1-max) for a clearer view against potential data imbalance [59]). For segmentation, we report (a) *pixel-wise AUROC* (pAUROC) and (b) *Per-Region Overlap* (PRO) [4] scores [8, 20], and (c) *(pixel-wise)* F_1*-max* in a similar manner to the anomaly classification evaluation.

Implementation details. We adopt the CLIP implementation of OpenCLIP[3] and its public pre-trained models in our experiments: namely, we use the LAION-400M [37] based CLIP with ViT-B/16+ [16] unless otherwise noted. We apply WinCLIP with stride 1 on ViT patch embeddings, which is equivalent to stride 16 in pixel-level in case of ViT-B/16+.

[3] https://github.com/mlfoundations/open_clip

5.1. Zero-/few-shot anomaly classification

In Table 1 we compare zero-shot and few-normal-shot anomaly classification results with prior works.

For zero-shot setup, we compare WinCLIP with two prior models: CLIP-AC (first row of Table 1), which is the original CLIP zero-shot classification [27] with labels of the form {"`normal [c]`","`anomalous [c]`"}, and CLIP-AC with the prompt ensemble (second row in Table 1) from [27] engineered for ImageNet [19]. We see that WinCLIP significantly improves over using these naïve adaptations of CLIP on both MVTec-AD and VisA. Section 5.4 presents ablation study on a break-down of this gain.

For the few-normal-shot setup, we see the same trend: WinCLIP+ outperforms prior works by a wide margin across all metrics on both benchmarks. In particular, we improve upon the state-of-the-art PatchCore [31] by 9.7% on 1-shot MVTec-AD and by 5.3% on 1-shot VisA. On MVTec-AD, we note that zero-shot WinCLIP outperforms the few-shot versions of prior works. Furthermore, WinCLIP+ 1/2/4-shot performance is better than WinCLIP 0-shot performance, highlighting the additional value of reference normal images.

5.2. Zero-/few-shot anomaly segmentation

In Table 4 we compare zero-shot and few-normal-shot anomaly segmentation results with prior works. While there are no prior works on zero-shot anomaly segmentation, we adapt two methods developed for other problems to our setup. First, Trans-MM [5] is a recent model interpretation method applicable to Transformers that provides a pixel-level mask. Second, MaskCLIP [57] is a general semantic segmentation model based on CLIP. We see that WinCLIP outperforms both methods by a wide margin on both MVTec-AD and VisA, highlighting that generic adaptations of CLIP do not perform as well as WinCLIP.

For the few-normal-shot setup, we compare with three prior works, which are designed specifically for anomaly localization. We see that WinCLIP+ again outperforms these prior methods across all metrics on both benchmarks, showing the additional value provided by language prompts. In Figure 5, we show qualitative results for a number of objects and defects. We see that in all cases, 1-shot WinCLIP+ provides a mask that is more concentrated on the ground truth compared to prior works. We also see that 1/2/4-normal-shot WinCLIP+ is better than 0-shot WinCLIP, demonstrating the complementary benefits of language driven prediction and visual only based model based on reference normal images.

图4为整体示意图）。具体来说，给定少量样本，我们从三种不同特征构建独立的参考记忆：(a) 小尺度下的WinCLIP特征 $\mathbf{F}_{\mathtt{s}}^{\mathtt{W}}$，(b) 中尺度下的特征 $\mathbf{F}_{\mathtt{m}}^{\mathtt{W}}$，以及 (c) 来自具有全局上下文的倒数第二层特征 $\mathbf{F}^{\mathtt{P}}$（*e.g.*，即ViT中通过自注意力捕获图像上下文的补丁标记）。尽管 $\mathbf{F}^{\mathtt{P}}$ 未与语言对齐，它仍有助于定义正常与异常。

因此，WinCLIP+获得了三个参考记忆：$\mathbf{R}_{\mathtt{s}}^{\mathtt{W}}$、$\mathbf{R}_{\mathtt{m}}^{\mathtt{W}}$和 $\mathbf{R}^{\mathtt{P}}$。随后，我们对它们的多尺度预测（4）进行平均，以对给定查询进行异常分割，

$$\mathbf{M}^{\mathtt{W}} := \tfrac{1}{3}(\mathbf{M}^{\mathtt{P}} + \mathbf{M}_{\mathtt{s}}^{\mathtt{W}} + \mathbf{M}_{\mathtt{m}}^{\mathtt{W}}), \tag{5}$$

然后与我们的语言引导预测 $\bar{\mathbf{M}}_0^{\mathtt{W}}$ 融合。

为了进行异常分类，我们结合了 $\mathbf{M}^{\mathtt{W}}$ 的最大值和WinCLIP零样本分类分数。这两个分数具有互补信息，可协同工作，具体而言：(a) 一个来自少样本参考的空间特征，(b) 另一个来自通过语言检索的CLIP知识：

$$\mathrm{ascore}_{\mathtt{W}}(\mathbf{x}) := \tfrac{1}{2}\left(\mathrm{ascore}_0(f(\mathbf{x})) + \max_{ij} \mathbf{M}_{ij}^{\mathtt{W}}\right). \tag{6}$$

5. 实验

我们进行了一系列实验，以评估基于WinCLIP的ACS在低样本情况下的性能，涵盖了我们在工业异常分类和分割领域关注的近期具有挑战性的基准测试。我们还进行了广泛的消融研究，以验证我们提出的各个组件的有效性。详细的设置、*e.g.*、预处理、指标及其他实现细节均在补充材料中提供。

数据集。我们的实验基于MVTec-AD [3] 和 VisA [59] 数据集。这两个基准测试集包含多样化的不同物体子集，*e.g.*，如胶囊、电路板。它们包含常见物体的高分辨率图像（*e.g.*，MVTec-AD为700^2-1024^2，VisA约为1.5 K $\times$ 1K），并带有完整的像素级标注。

评估指标。对于分类任务，我们参照文献[8,31,49]报告（a）*Area Under the Receiver Operating Characteristic* (AUROC)，同时为更清晰反映潜在数据不平衡问题[59]，还报告（b）*Area Under the Precision-Recall curve* (AUPR)和（c）F_1*-score at optimal threshold* (F_1-max)。对于分割任务，我们以类似异常分类评估的方式报告（a）*pixel-wise AUROC* (pAUROC)和（b）*Per-Region Overlap* (PRO)分数[4,8,20]，以及（c）*(pixel-wise)* F_1*-max*。

实现细节。我们采用OpenCLIP[3]的CLIP实现及其公开预训练模型。

[3] https://github.com/mlfoundations/open_clip

实验设置如下：除非另有说明，我们默认使用基于LAION-400M [37]的CLIP模型，其视觉编码器为ViT-B/16+ [16]。我们在ViT的补丁嵌入上以步幅1应用WinCLIP，这对应于ViT-B/16+在像素级别的步幅16。

5.1. 零样本/少样本异常分类

在表1中，我们将零样本和少正常样本异常分类结果与先前工作进行了比较。

在零样本设置中，我们将WinCLIP与两个先前模型进行比较：CLIP-AC（表1第一行），即原始CLIP零样本分类[27]，其标签形式为{“正常[c]”，“异常[c]”}；以及采用[27]中为ImageNet[19]设计的提示集成方法的CLIP-AC（表1第二行）。我们发现，在MVTec-AD和VisA数据集上，WinCLIP相较于这些对CLIP的朴素适配方法均有显著提升。第5.4节将通过消融实验详细解析这种增益的来源。

在少样本正常设置中，我们观察到相同的趋势：WinCLIP+ 在两个基准测试的所有指标上均大幅超越先前工作。具体而言，我们在 1-shot MVTec-AD 上将现有最优方法 PatchCore [31] 提升了 9.7%，在 1-shot VisA 上提升了 5.3%。在 MVTec-AD 上，我们注意到零样本 WinCLIP 的表现优于先前工作的少样本版本。此外，WinCLIP+ 的 1/2/4-shot 性能优于 WinCLIP 的零样本性能，这凸显了参考正常图像带来的附加价值。

5.2. 零样本/少样本异常分割

在表4中，我们将零样本和少样本正常样本异常分割结果与先前工作进行了比较。虽然目前尚无针对零样本异常分割的先前研究，但我们调整了两种为其他问题开发的方法以适应我们的设置。首先，Trans-MM [5] 是一种适用于Transformer的最新模型解释方法，可提供像素级掩码。其次，MaskCLIP [57] 是一种基于CLIP的通用语义分割模型。我们发现，WinCLIP在MVTec-AD和VisA数据集上均大幅优于这两种方法，这表明CLIP的通用适配方案不如WinCLIP表现优异。

在少样本正常图像设置中，我们与三种专门针对异常定位设计的先前工作进行了比较。我们发现WinCLIP+在两个基准测试的所有指标上再次优于这些先前方法，显示出语言提示带来的附加价值。在图5中，我们展示了多类物体与缺陷的定性结果。可见在所有案例中，单样本WinCLIP+生成的掩码相较于先前工作更集中于真实异常区域。我们还观察到1/2/4样本正常图像下的WinCLIP+优于零样本WinCLIP，这证明了基于语言驱动的预测与基于正常参考图像的纯视觉模型具有互补优势。

Anomaly Classification		MVTec-AD			VisA		
Setup	Method	AUROC	AUPR	F_1-max	AUROC	AUPR	F_1-max
0-shot	CLIP-AC [27]	74.0±0.0	89.1±0.0	88.5±0.0	59.3±0.0	67.0±0.0	74.4±0.0
	+ Prompt ens. [27]	74.1±0.0	89.5±0.0	87.8±0.0	58.2±0.0	66.4±0.0	74.0±0.0
	WinCLIP (ours)	**91.8±0.0**	**96.5±0.0**	**92.9±0.0**	**78.1±0.0**	**81.2±0.0**	**79.0±0.0**
1-shot	SPADE [7]	81.0±2.0	90.6±0.8	90.3±0.8	79.5±4.0	82.0±3.3	80.7±1.9
	PaDiM [8]	76.6±3.1	88.1±1.7	88.2±1.1	62.8±5.4	68.3±4.0	75.3±1.2
	PatchCore [31]	83.4±3.0	92.2±1.5	90.5±1.5	79.9±2.9	82.8±2.3	81.7±1.6
	WinCLIP+ (ours)	**93.1±2.0**	**96.5±0.9**	**93.7±1.1**	**83.8±4.0**	**85.1±4.0**	**83.1±1.7**
2-shot	SPADE [7]	82.9±2.6	91.7±1.2	91.1±1.0	80.7±5.0	82.3±4.3	81.7±2.5
	PaDiM [8]	78.9±3.1	89.3±1.7	89.2±1.1	67.4±5.1	71.6±3.8	75.7±1.8
	PatchCore [31]	86.3±3.3	93.8±1.7	92.0±1.5	81.6±4.0	84.8±3.2	82.5±1.8
	WinCLIP+ (ours)	**94.4±1.3**	**97.0±0.7**	**94.4±0.8**	**84.6±2.4**	**85.8±2.7**	**83.0±1.4**
4-shot	SPADE [7]	84.8±2.5	92.5±1.2	91.5±0.9	81.7±3.4	83.4±2.7	82.1±2.1
	PaDiM [8]	80.4±2.5	90.5±1.6	90.2±1.2	72.8±2.9	75.6±2.2	78.0±1.2
	PatchCore [31]	88.8±2.6	94.5±1.5	92.6±1.6	85.3±2.1	87.5±2.1	84.3±1.3
	WinCLIP+ (ours)	**95.2±1.3**	**97.3±0.6**	**94.7±0.8**	**87.3±1.8**	**88.8±1.8**	**84.2±1.6**

Table 1. Comparison of anomaly classification (AC) performance on MVTec-AD and VisA benchmarks. We report the mean and standard deviation over 5 random seeds for each measurement. Bold indicates the best performance.

Methods	Setup	AC	AS
WinCLIP (ours)	0-shot	91.8	85.1
WinCLIP+ (ours)	1-shot	93.1	95.2
WinCLIP+ (ours)	4-shot	95.2	96.2
DifferNet [32]	16-shot	87.3	-
TDG [39]	10-shot	78.0	-
RegAD-L [14]	2-shot	81.5	93.3
RegAD [14]	4 + agg.	88.2	95.8
MKD [35]	full-shot	87.7	90.7
P-SVDD [49]	full-shot	92.1	95.7
CutPaste [20]	full-shot	95.2	96.0
PatchCore [31]	full-shot	99.6	98.2

Table 2. Comparison with existing many-shot ACS methods in AUROC (or pixel-) on MVTec-AD.

Method	AUROC	AUPR	F_1-max
(a) One-class	34.2	68.9	83.5
Two-class	74.0	89.1	88.5
(b) + State ens.	89.8	95.6	92.2
(c) + Prompt ens.	90.8	96.1	92.5
(d) + Multi-crop	91.8	96.5	92.9

Table 3. Comparison of AC performance on MVTec-AD across WinCLIP ablations in AC (Section 4.1).

5.3. Comparison with many-shot methods

In Table 2 we compare our zero-/few-shot results with full-shot results of several prior works on MVTec-AD. Our 4-shot WinCLIP+ is competitive with CutPaste [20], a recent method that utilizes the *full-shot* samples for model tuning. Also, our 0-shot WinCLIP outperforms recent few-shot methods in AC, such as DifferNet [32] and TDG [39], even compared to their results with more than 10-shots. Recently, a new setup of aggregated few-shot is proposed [14], where one is free to use all the training samples but for the target class which is restricted to k-shot. Our 4-shot WinCLIP+ outperforms RegAD's aggregated 4-shot [14] performance.

5.4. Ablation study

We perform component-wise analysis on MVTec-AD [3]. A further study, *e.g.*, comparison with CLIP-based PatchCore, effect of different backbones, discussion on failure cases, *etc.*, can be found in the supplementary material.

WinCLIP for AC: In Table 3, we report the individual effect of components that constitute our zero-shot AC model. Firstly, we observe (a) the textual supervision for the word "`anomalous`" is crucial to achieve a reasonable performance ("One-class"; Section 4.1), suggesting the effectiveness of CLIP knowledge about "abnormality". Next, we confirm that having a diversity in both (b) state-level and (c) prompt-level texts are the key source of gains. And we remark the proposed state ensemble as a more significant component. Finally, we observe (d) applying multi-crop prediction [13] could also yield a minor improvement.

WinCLIP for AS: Table 8 validates not only the efficiency of WinCLIP to extract local features for zero-shot AS, but also the effectiveness of multi-scale and harmonic averaging to boost the results. To this end, we consider the following additional baselines that also extract patch-level features: (*i*) *Patch-token* (Section 4.2): it takes the patch features at the last layer, and (*ii*) *Image tiling*: it first performs dense "tiling" on an image and then obtains "tile" embeddings for segmentation by forwarding each tile with resizing. Overall, the comparison shows that patch-tokens are not aligned with language despite its fast inference time, while "Image tiling" makes a significant computational overhead although it does benefit from their local features. WinCLIP achieves accelerated inference due to its window-based computation of local features, with even better performance. Also based on the multi-scale study, we observe that segmentation benefits from both features with image-level, and middle/local context. Note that the scores from last patch embeddings of ViT encodes global context thanks to self-attention, which contributes to a comprehensive localization in WinCLIP.

WinCLIP+ for AC and AS: We ablate on different factors to define WinCLIP+ scores for AC (6) and AS (5) respectively. For AC, from Table 5, we clearly remark the effectiveness of ascore_0 upon $\max \mathbf{M}^{\mathtt{W}}$. Interestingly, we observe ascore_0 is beneficial even in higher-shot regimes where $\max \mathbf{M}^{\mathtt{W}}$ can be better, confirming their complementary effects. For AS, in Table 6, we notice the effect of adding $\mathbf{M}^{\mathtt{W}}_{\mathtt{m}}$ (or $\mathbf{M}^{\mathtt{W}}_{\mathtt{s}}$) upon $\mathbf{M}^{\mathtt{P}}$, *i.e.*, the prediction from WinCLIP features: apart from the good performance of $\mathbf{M}^{\mathtt{P}}$, $\mathbf{M}^{\mathtt{W}}$ could still provide useful information from its local-awareness.

WinCLIP with task-specific defects: As mentioned in Section 4.1, besides using the generic state words and tem-

Anomaly Classification		MVTec-AD			VisA		
Setup	Method	AUROC	AUPR	F_1-max	AUROC	AUPR	F_1-max
0-shot	CLIP-AC [27]	74.0±0.0	89.1±0.0	88.5±0.0	59.3±0.0	67.0±0.0	74.4±0.0
	+ Prompt ens. [27]	74.1±0.0	89.5±0.0	87.8±0.0	58.2±0.0	66.4±0.0	74.0±0.0
	WinCLIP (ours)	**91.8±0.0**	**96.5±0.0**	**92.9±0.0**	**78.1±0.0**	**81.2±0.0**	**79.0±0.0**
1-shot	SPADE [7]	81.0±2.0	90.6±0.8	90.3±0.8	79.5±4.0	82.0±3.3	80.7±1.9
	PaDiM [8]	76.6±3.1	88.1±1.7	88.2±1.1	62.8±5.4	68.3±4.0	75.3±1.2
	PatchCore [31]	83.4±3.0	92.2±1.5	90.5±1.5	79.9±2.9	82.8±2.3	81.7±1.6
	WinCLIP+ (ours)	**93.1±2.0**	**96.5±0.9**	**93.7±1.1**	**83.8±4.0**	**85.1±4.0**	**83.1±1.7**
2-shot	SPADE [7]	82.9±2.6	91.7±1.2	91.1±1.0	80.7±5.0	82.3±4.3	81.7±2.5
	PaDiM [8]	78.9±3.1	89.3±1.7	89.2±1.1	67.4±5.1	71.6±3.8	75.7±1.8
	PatchCore [31]	86.3±3.3	93.8±1.7	92.0±1.5	81.6±4.0	84.8±3.2	82.5±1.8
	WinCLIP+ (ours)	**94.4±1.3**	**97.0±0.7**	**94.4±0.8**	**84.6±2.4**	**85.8±2.7**	**83.0±1.4**
4-shot	SPADE [7]	84.8±2.5	92.5±1.2	91.5±0.9	81.7±3.4	83.4±2.7	82.1±2.1
	PaDiM [8]	80.4±2.5	90.5±1.6	90.2±1.2	72.8±2.9	75.6±2.2	78.0±1.2
	PatchCore [31]	88.8±2.6	94.5±1.5	92.6±1.6	85.3±2.1	87.5±2.1	84.3±1.3
	WinCLIP+ (ours)	**95.2±1.3**	**97.3±0.6**	**94.7±0.8**	**87.3±1.8**	**88.8±1.8**	**84.2±1.6**

表1. MVTec-AD与VisA基准测试中异常分类（AC）性能对比。各项测量结果均基于5个随机种子的平均值与标准差报告。**粗体**表示最佳性能。

Methods	Setup	AC	AS
WinCLIP (ours)	0-shot	91.8	85.1
WinCLIP+ (ours)	1-shot	93.1	95.2
WinCLIP+ (ours)	4-shot	95.2	96.2
DifferNet [32]	16-shot	87.3	-
TDG [39]	10-shot	78.0	-
RegAD-L [14]	2-shot	81.5	93.3
RegAD [14]	4 + agg.	88.2	95.8
MKD [35]	full-shot	87.7	90.7
P-SVDD [49]	full-shot	92.1	95.7
CutPaste [20]	full-shot	95.2	96.0
PatchCore [31]	full-shot	99.6	98.2

表2. 在MVTec-AD数据集上基于AUROC（或像素级）与现有多样本ACS方法的比较。

Method	AUROC	AUPR	F_1-max
(a) One-class	34.2	68.9	83.5
Two-class	74.0	89.1	88.5
(b) + State ens.	89.8	95.6	92.2
(c) + Prompt ens.	90.8	96.1	92.5
(d) + Multi-crop	91.8	96.5	92.9

表3. 在MVTec-AD数据集上，WinCLIP消融实验在异常分类（第4.1节）中的性能对比。

5.3. 与多样本方法的比较

在表2中，我们将零样本/少样本结果与先前多项工作在MVTec-AD上的全样本结果进行了比较。我们的4样本WinCLIP+与CutPaste [20]（一种利用*full-shot*样本进行模型调优的最新方法）具有竞争力。此外，我们的零样本WinCLIP在异常分类（AC）方面超越了最近的少样本方法（如DifferNet [32]和TDG [39]），即使与它们使用超过10样本的结果相比也表现更优。最近，有研究提出了聚合少样本的新设置[14]，该设置允许使用除目标类别外的所有训练样本（目标类别被限制为k样本）。我们的4样本WinCLIP+超越了RegAD的聚合4样本[14]性能。

5.4. 消融研究

我们对MVTec-AD [3]进行了逐组件分析。进一步的*e.g.*研究、与基于CLIP的PatchCore的对比、不同主干网络的效果分析、失败案例的讨论以及*etc.*等内容可在补充材料中找到。

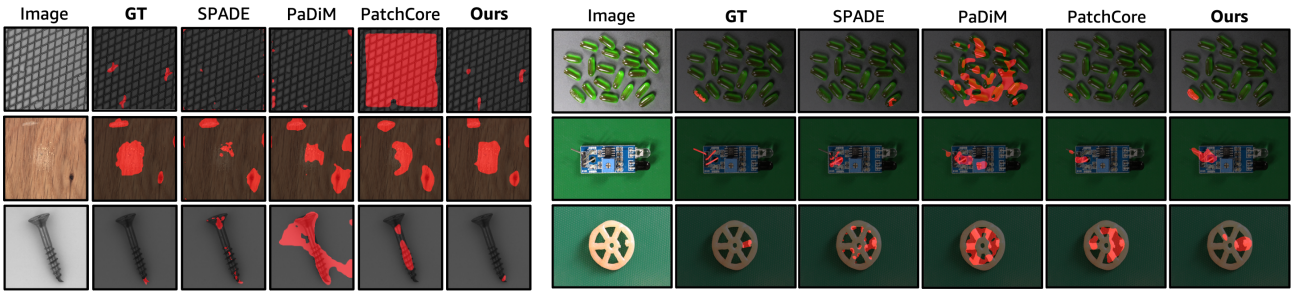
WinCLIP用于异常分类：在表3中，我们报告了构成零样本异常分类模型的各组件独立效果。首先，我们观察到（a）针对“异常”一词的文本监督对实现合理性能至关重要（“单类”；第4.1节），这表明CLIP关于“异常性”知识的有效性。其次，我们证实（b）状态层面与（c）提示层面文本的多样性是性能提升的关键来源，并指出所提出的状态集成是更具显著性的组件。最后，我们注意到（d）应用多裁剪预测[13]也能带来小幅改进。

WinCLIP for AS: 表8不仅验证了效率

WinCLIP提取局部特征用于零样本异常分割的有效性，不仅体现在其基础能力上，还通过多尺度与谐波平均策略进一步提升了效果。为此，我们考虑了以下同样提取块级特征的补充基线：(*i*) *Patch-token* (第4.2节)：采用最后一层的块特征；以及(*ii*) *Image tiling*：先对图像进行密集“分块”，再通过调整尺寸后逐块前传获得分割所需的“块”嵌入。总体而言，对比表明块标记虽推理速度快，却未与语言对齐；而“图像分块”虽能利用局部特征带来收益，却产生了显著的计算开销。WinCLIP基于窗口的局部特征计算实现了加速推理，同时获得了更优性能。基于多尺度研究我们还观察到，分割任务同时受益于图像级特征与中层/局部上下文特征。需注意，ViT最后一层块嵌入的得分因自注意力机制而编码了全局上下文，这为WinCLIP实现全面定位提供了重要支撑。

WinCLIP+ 用于AC和AS：我们通过消融不同因素来分别定义WinCLIP+在AC（6）和AS（5）上的评分。对于AC，从表5可以明显看出ascore_0相较于$\max \mathbf{M}^{\mathtt{W}}$的有效性。有趣的是，我们观察到$\text{ascore}_0$即使在$\max \mathbf{M}^{\mathtt{W}}$可能表现更优的高样本量情况下也有益处，这证实了它们的互补效应。对于AS，在表6中，我们注意到在$\mathbf{M}^{\mathtt{P}}$、i、e基础上添加$\mathbf{M}^{\mathtt{W}}_{\mathtt{m}}$ (或$\mathbf{M}^{\mathtt{W}}_{\mathtt{s}}$)的效果：除了$\mathbf{M}^{\mathtt{P}}$的良好性能外，$\mathbf{M}^{\mathtt{W}}$仍能通过其局部感知能力提供有用信息。

WinCLIP存在特定任务缺陷：如第4.1节所述，除了使用通用状态词和tem-



(a) MVTec-AD (1-shot) (b) VisA (1-shot)

Figure 5. Qualitative comparison of 1-shot anomaly segmentation results on MVTec-AD and VisA benchmarks.

Anomaly Segmentation		MVTec-AD			VisA		
Setup	Method	pAUROC	PRO	F_1-max	pAUROC	PRO	F_1-max
0-shot	Trans-MM [5]	57.5±0.0	21.9±0.0	12.1±0.0	49.4±0.0	10.2±0.0	3.1±0.0
	MaskCLIP [57]	63.7±0.0	40.5±0.0	18.5±0.0	60.9±0.0	27.3±0.0	7.3±0.0
	WinCLIP (ours)	**85.1±0.0**	**64.6±0.0**	**31.7±0.0**	**79.6±0.0**	**56.8±0.0**	**14.8±0.0**
1-shot	SPADE [7]	91.2±0.4	83.9±0.7	42.4±1.0	95.6±0.4	84.1±1.6	35.5±2.2
	PaDiM [8]	89.3±0.9	73.3±2.0	40.2±2.1	89.9±0.8	64.3±2.4	17.4±1.7
	PatchCore [31]	92.0±1.0	79.7±2.0	50.4±2.1	95.4±0.6	80.5±2.5	38.0±1.9
	WinCLIP+ (ours)	**95.2±0.5**	**87.1±1.2**	**55.9±2.7**	**96.4±0.4**	**85.1±2.1**	**41.3±2.3**
2-shot	SPADE [7]	92.0±0.3	85.7±0.7	44.5±1.0	96.2±0.4	85.7±1.1	40.5±3.7
	PaDiM [8]	91.3±0.7	78.2±1.8	43.7±1.5	92.0±0.7	70.1±2.6	21.1±2.4
	PatchCore [31]	93.3±0.6	82.3±1.3	53.0±1.7	96.1±0.5	82.6±2.3	41.0±3.9
	WinCLIP+ (ours)	**96.0±0.3**	**88.4±0.9**	**58.4±1.7**	**96.8±0.3**	**86.2±1.4**	**43.5±3.3**
4-shot	SPADE [7]	92.7±0.3	87.0±0.5	46.2±1.3	96.6±0.3	87.3±0.8	43.6±3.6
	PaDiM [8]	92.6±0.7	81.3±1.9	46.1±1.8	93.2±0.5	72.6±1.9	24.6±1.8
	PatchCore [31]	94.3±0.5	84.3±1.6	55.0±1.9	96.8±0.3	84.9±1.4	43.9±3.1
	WinCLIP+ (ours)	**96.2±0.3**	**89.0±0.8**	**59.5±1.8**	**97.2±0.2**	**87.6±0.9**	**47.0±3.0**

Table 4. Comparison of anomaly segmentation (AS) performance on MVTec-AD and VisA benchmarks. We report the mean and standard deviation over 5 random seeds for each measurement. Bold indicates the best performance.

WinCLIP+ (AC)		# shots (AUROC)			
max $\mathbf{M}^{\mathrm{w}}$	ascore_0	1	2	4	8
✓	✗	87.9	91.0	<u>92.6</u>	<u>94.5</u>
✗	✓	<u>91.8</u>	<u>91.8</u>	91.8	91.8
✓	✓	**93.1**	**94.4**	**95.2**	**96.3**

Table 5. k-shot AC ablations: MVTec-AD. Bold/underline indicate the best/runner-up.

WinCLIP+ (AS)			# shots (pAUROC)			
$\mathbf{M}^{\mathrm{p}}$	$\mathbf{M}^{\mathrm{w}}_{\mathrm{m}}$	$\mathbf{M}^{\mathrm{w}}_{\mathrm{s}}$	1	2	4	8
✓	✗	✗	94.5	94.8	95.4	95.8
✓	✓	✗	<u>95.1</u>	<u>95.7</u>	**96.3**	**96.6**
✓	✓	✓	**95.2**	**96.0**	<u>96.2</u>	<u>96.5</u>

Table 6. k-shot AS ablations: MVTec-AD. Bold/underline indicate the best/runner-up.

Method	PCB2	PCB4	Pipe fryum	Mean
WinCLIP	51.2	79.6	69.7	78.1
+ specific states	**56.5**	**82.7**	**70.4**	**78.9**

Table 7. Ablation on specific states: VisA.

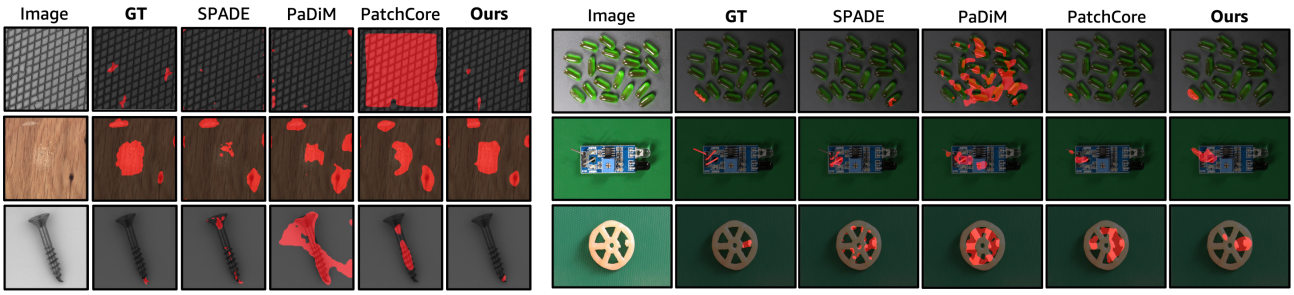
Method	pAUROC	PRO	F_1-max	Time (ms)
Patch-token	22.4	2.3	8.0	**95.5±18.8**
Image tiling	77.9	57.5	25.5	1442.1±62.2
WinCLIP (ours)	**85.1**	**64.6**	**31.7**	389.4±18.5
w/o image-scale	82.0	63.0	29.5	378.6±20.2
w/o mid-scale	84.0	61.6	30.5	<u>190.7</u>±13.9
w/o small-scale	<u>84.7</u>	<u>63.6</u>	<u>30.6</u>	265.4±15.9
w/o Harmonic avg.	81.5	60.5	27.3	279.9±22.8

Table 8. Comparison of AS performance on MVTec-AD and its per-image inference time, measured at Amazon EC2 G4dn instances.

plates (Fig. 6 of supplementary) to cover common cases, our compositional prompt ensemble also supports task-specific state words, *e.g.*, "missing part" on PCB/"burnt" pipe fryum; both VisA and MVTec-AD release specific defect types. Ablation study in Table 7 shows that specific state words further improve zero-shot classification in VisA by 0.8% average AUROC with 5.3% gain on the challenging PCB2.

6. Conclusion

We propose a novel framework to define normality and anomaly via both fine-grained textual definitions and normal reference images for comprehensive anomaly classification and segmentation. First, we show that the CLIP pre-trained on large-scale web data provides a powerful representation with good alignment between texts and images for anomaly recognition tasks. The compositional prompt ensemble defines the normality and anomaly in text and helps to distill knowledge from the pre-trained CLIP for better zero-shot anomaly recognition. WinCLIP efficiently aggregates multi-scale features with image-text alignment from window and image-level to perform zero-shot segmentation. Moreover, given a few normal samples, vision based reference association provides complementary information about the two states to language definitions, leading to few-shot WinCLIP+. In recent benchmarks, WinCLIP and WinCLIP+ outperform state-of-the-arts in zero-/few-shot setups with considerable margins. We believe our work will bring values complementary to standard one-class methods. For further improvement, vision-language pre-training with industrial domain data is a promising direction that is left as a future work.



(a) MVTec-AD (1-shot) (b) VisA (1-shot)

图5. MVTec-AD与VisA基准测试中单样本异常分割结果的定性比较。

Anomaly Segmentation		MVTec-AD			VisA		
Setup	Method	pAUROC	PRO	F_1-max	pAUROC	PRO	F_1-max
0-shot	Trans-MM [5]	57.5±0.0	21.9±0.0	12.1±0.0	49.4±0.0	10.2±0.0	3.1±0.0
	MaskCLIP [57]	63.7±0.0	40.5±0.0	18.5±0.0	60.9±0.0	27.3±0.0	7.3±0.0
	WinCLIP (ours)	**85.1±0.0**	**64.6±0.0**	**31.7±0.0**	**79.6±0.0**	**56.8±0.0**	**14.8±0.0**
1-shot	SPADE [7]	91.2±0.4	83.9±0.7	42.4±1.0	95.6±0.4	84.1±1.6	35.5±2.2
	PaDiM [8]	89.3±0.9	73.3±2.0	40.2±2.1	89.9±0.8	64.3±2.4	17.4±1.7
	PatchCore [31]	92.0±1.0	79.7±2.0	50.4±2.1	95.4±0.6	80.5±2.5	38.0±1.9
	WinCLIP+ (ours)	**95.2±0.5**	**87.1±1.2**	**55.9±2.7**	**96.4±0.4**	**85.1±2.1**	**41.3±2.3**
2-shot	SPADE [7]	92.0±0.3	85.7±0.7	44.5±1.0	96.2±0.4	85.7±1.1	40.5±3.7
	PaDiM [8]	91.3±0.7	78.2±1.8	43.7±1.5	92.0±0.7	70.1±2.6	21.1±2.4
	PatchCore [31]	93.3±0.6	82.3±1.3	53.0±1.7	96.1±0.5	82.6±2.3	41.0±3.9
	WinCLIP+ (ours)	**96.0±0.3**	**88.4±0.9**	**58.4±1.7**	**96.8±0.3**	**86.2±1.4**	**43.5±3.3**
4-shot	SPADE [7]	92.7±0.3	87.0±0.5	46.2±1.3	96.6±0.3	87.3±0.8	43.6±3.6
	PaDiM [8]	92.6±0.7	81.3±1.9	46.1±1.8	93.2±0.5	72.6±1.9	24.6±1.8
	PatchCore [31]	94.3±0.5	84.3±1.6	55.0±1.9	96.8±0.3	84.9±1.4	43.9±3.1
	WinCLIP+ (ours)	**96.2±0.3**	**89.0±0.8**	**59.5±1.8**	**97.2±0.2**	**87.6±0.9**	**47.0±3.0**

表4. MVTec-AD与VisA基准测试上的异常分割（AS）性能对比。各项指标均报告5个随机种子的平均值与标准差。**粗体**表示最佳性能。

WinCLIP+ (AC)		# shots (AUROC)			
max $\mathbf{M}^{\mathrm{w}}$	ascore_0	1	2	4	8
✓	✗	87.9	91.0	92.6	94.5
✗	✓	91.8	91.8	91.8	91.8
✓	✓	**93.1**	**94.4**	**95.2**	**96.3**

表5. k-样本AC消融实验：MVTec-AD数据集。**粗体**/下划线分别表示最优/次优结果。

WinCLIP+ (AS)			# shots (pAUROC)			
$\mathbf{M}^{\mathrm{p}}$	$\mathbf{M}_{\mathrm{m}}^{\mathrm{w}}$	$\mathbf{M}_{\mathrm{s}}^{\mathrm{w}}$	1	2	4	8
✓	✗	✗	94.5	94.8	95.4	95.8
✓	✓	✗	95.1	95.7	**96.3**	**96.6**
✓	✓	✓	**95.2**	**96.0**	96.2	96.5

表6. k-样本AS消融实验：MVTec-AD。粗体/下划线分别表示最佳/次佳结果。

Method	PCB2	PCB4	Pipe fryum	Mean
WinCLIP	51.2	79.6	69.7	78.1
+ specific states	**56.5**	**82.7**	**70.4**	**78.9**

表7. 特定状态消融研究：VisA。

Method	pAUROC	PRO	F_1-max	Time (ms)
Patch-token	22.4	2.3	8.0	**95.5±18.8**
Image tiling	77.9	57.5	25.5	1442.1±62.2
WinCLIP (ours)	**85.1**	**64.6**	**31.7**	389.4±18.5
w/o image-scale	82.0	63.0	29.5	378.6±20.2
w/o mid-scale	84.0	61.6	30.5	190.7±13.9
w/o small-scale	84.7	63.6	30.6	265.4±15.9
w/o Harmonic avg.	81.5	60.5	27.3	279.9±22.8

表8. 在亚马逊EC2 G4dn实例上测得的MVTec-AD异常检测性能对比及其单图推理时间。

板件（补充材料图6）覆盖了常见情况，我们的组合提示集成还支持特定任务的状态词，*e.g.*，例如PCB上的“缺失部分”/“烧焦”的管道炸物；VisA和MVTec-AD均发布了具体的缺陷类型。表7中的消融研究表明，特定状态词进一步将VisA中的零样本分类平均AUROC提升了0.8%，在具有挑战性的PCB2上获得了5.3%的增益。

6. 结论

我们提出了一种新颖的框架，通过细粒度的文本定义和正常用于全面异常分类和分割的参考图像。首先，我们展示了在大规模网络数据上预训练的CLIP为异常识别任务提供了强大的表示能力，实现了文本与图像之间的良好对齐。组合提示集成定义了文本中的正常与异常状态，并有助于从预训练的CLIP中提取知识，以实现更好的零样本异常识别。WinCLIP通过聚合从窗口级到图像级的多尺度特征与图文对齐，高效执行零样本分割。此外，在给定少量正常样本的情况下，基于视觉的参考关联为语言定义提供了关于两种状态的补充信息，从而催生了少样本WinCLIP+。在最近的基准测试中，WinCLIP与WinCLIP+在零样本/少样本设置下以显著优势超越了现有最优方法。我们相信这项工作将为标准单类方法带来互补价值。为进一步提升性能，基于工业领域数据的视觉-语言预训练是一个有前景的方向，留待未来探索。

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: A visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 2, 3

[2] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *International Journal of Computer Vision*, 130(4):947–969, 2022. 1, 17

[3] 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*, pages 9592–9600, 2019. 1, 3, 5, 6, 7

[4] 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*, pages 4183–4192, 2020. 6

[5] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 397–406, 2021. 6, 8

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020. 3

[7] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 1, 3, 7, 8, 13

[8] 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*, pages 475–489. Springer, 2021. 1, 3, 6, 7, 8, 13

[9] Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34:7068–7081, 2021. 3

[10] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 2021. https://distill.pub/2021/multimodal-neurons. 3

[11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *International Conference on Learning Representations*, 2021. 2, 3

[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 5

[13] Geoffrey E Hinton, Alex Krizhevsky, and Ilya Sutskever. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25(1106-1114):1, 2012. 3, 7

[14] Chaoqin Huang, Haoyan Guan, Aofan Jiang, Ya Zhang, Michael Spratlin, and Yanfeng Wang. Registration based few-shot anomaly detection. In *European Conference on Computer Vision*, 2022. 1, 2, 3, 7

[15] Drew A Hudson and Christopher D Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6700–6709, 2019. 3

[16] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. *OpenCLIP*. Zenodo, July 2021. 3, 6, 12

[17] Phillip Isola, Joseph J Lim, and Edward H Adelson. Discovering states and transformations in image collections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1383–1391, 2015. 2, 3

[18] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR, 2021. 2, 3

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 6

[20] 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*, pages 9664–9674, 2021. 1, 3, 6, 7

[21] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 3

[22] Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. Unified-IO: A unified model for vision, language, and multi-modal tasks. *arXiv preprint arXiv:2206.08916*, 2022. 3

[23] M Mancini, MF Naeem, Y Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *34th IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2021. 3

[24] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Learning graph embeddings for open world compositional zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[25] 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 *30th IEEE/IES International Symposium on Industrial Electronics (ISIE)*, June 2021. 3

参考文献

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds 等。Flamingo：一种用于少样本学习的视觉语言模型。*arXiv preprint arXiv:2204.14198*，2022年。2, 3[2] Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, Carsten Steger。超越凹痕与划痕：无监督异常检测与定位中的逻辑约束。*International Journal of Computer Vision*，130(4):947–969，2022年。1, 17[3] Paul Bergmann, Michael Fauser, David Sattlegger, Carsten Steger。MVTec AD —— 一个用于无监督异常检测的综合性真实世界数据集。收录于 *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*，第9592–9600页，2019年。1, 3, 5, 6, 7[4] Paul Bergmann, Michael Fauser, David Sattlegger, Carsten Steger。未受指导的学生：基于判别性潜在嵌入的师生异常检测。收录于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第4183–4192页，2020年。6[5] Hila Chefer, Shir Gur, Lior Wolf。用于解释双模态及编码器-解码器 Transformer 的通用注意力模型可解释性。收录于 *Proceedings of the IEEE/CVF International Conference on Computer Vision*，第397–406页，2021年。6, 8[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffrey Hinton。一种用于视觉表示对比学习的简单框架。收录于 *International Conference on Machine Learning*，第1597–1607页。PMLR，2020年。3[7] Niv Cohen, Yedid Hoshen。基于深度金字塔对应的子图像异常检测。*arXiv preprint arXiv:2005.02357*，2020年。1, 3, 7, 8, 13[8] Thomas Defard, Aleksandr Setkov, Angelique Loesch, Romaric Audigier。PaDiM：一种用于异常检测与定位的块分布建模框架。收录于 *International Conference on Pattern Recognition*，第475–489页。Springer，2021年。1, 3, 6, 7, 8, 13[9] Stanislav Fort, Jie Ren, Balaji Lakshminarayanan。探索分布外检测的极限。*Advances in Neural Information Processing Systems*，34:7068–7081，2021年。3[10] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, Chris Olah。人工神经网络中的多模态神经元。*Distill*，2021年。https://distill.pub/2021/multimodal-neurons。3[11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, Yin Cui。通过视觉与语言知识蒸馏实现开放词汇目标检测。收录于 *International Conference on Learning Representations*，2021年。2, 3[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, Ross Girshick。掩码自编码器是可扩展的视觉学习器。收录于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第16000–16009页，2022年。5[13] Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever。使用深度卷积神经网络进行 ImageNet 分类。*Advances in Neural Information Processing Systems*，第25卷（第1106-1114页）：第1页，2012年。第3、7页 [14] 黄超勤、管浩岩、蒋傲凡、张亚、Michael Spratlin、王延峰。基于配准的小样本异常检测。收录于*European Conference on Computer Vision*，2022年。第1、2、3、7页 [15] Drew A Hudson与Christopher D Manning。GQA：一个用于真实世界视觉推理与组合问答的新数据集。收录于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第6700–6709页，2019年。第3页 [16] Gabriel Ilharco、Mitchell Wortsman、Ross Wightman、Cade Gordon、Nicholas Carlini、Rohan Taori、Achal Dave、Vaishaal Shankar、Hongseok Namkoong、John Miller、Hannaneh Hajishirzi、Ali Farhadi、Ludwig Schmidt。*OpenCLIP*。Zenodo，2021年7月。第3、6、12页 [17] Phillip Isola、Joseph J Lim、Edward H Adelson。在图像集合中发现状态与变换。收录于 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*，第1383–1391页，2015年。第2、3页 [18] 贾超、杨寅飞、夏晔、陈一婷、Zarana Parekh、Hieu Pham、Quoc Le、Sung Yun-Hsuan、李臻、Tom Duerig。利用噪声文本监督扩展视觉与视觉-语言表征学习。收录于 *International Confer-ence on Machine Learning*，第4904–4916页。PMLR，2021年。第2、3页 [19] Alex Krizhevsky、Ilya Sutskever、Geoffrey E Hinton。使用深度卷积神经网络进行ImageNet分类。*Communications of the ACM*，第60卷（第6期）：第84–90页，2017年。第6页 [20] 李春良、Kihyuk Sohn、Jinsung Yoon、Tomas Pfister。CutPaste：用于异常检测与定位的自监督学习。收录于*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第9664–9674页，2021年。第1、3、6、7页 [21] 李俊南、Ramprasaath Selvaraju、Akhilesh Gotmare、Shafiq Joty、熊才明、Steven Chu Hong Hoi。先对齐后融合：基于动量蒸馏的视觉与语言表征学习。*Advances in neural informationprocessing systems*，第34卷：第9694–9705页，2021年。第3页 [22] 卢嘉森、Christopher Clark、Rowan Zellers、Roozbeh Mottaghi、Aniruddha Kembhavi。Unified-IO：一个面向视觉、语言及多模态任务的统一模型。*arXiv preprintarXiv:2206.08916*，2022年。第3页 [23] M Mancini、MF Naeem、Y Xian、Zeynep Akata。开放世界组合零样本学习。收录于*34th IEEE Conference on Computer Vision and Pattern Recognition*。IEEE，2021年。第3页 [24] Massimiliano Mancini、Muhammad Ferjad Naeem、Yongqin Xian、Zeynep Akata。学习开放世界组合零样本学习的图嵌入。*IEEE Transactions on Pattern Analysis and Machine Intelligence*，2022年。第3页 [25] Pankaj Mishra、Riccardo Verk、Daniele Fornasier、Claudio Piciarelli、Gian Luca Foresti。VT-ADL：一种用于图像异常检测与定位的视觉Transformer网络。收录于 *30th IEEE/IES International Symposium on Industrial Electronics (ISIE)*，2021年6月。第3页

[26] MF Naeem, Y Xian, F Tombari, and Zeynep Akata. Learning graph embeddings for compositional zero-shot learning. In *34th IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2021. 3

[27] 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*, pages 8748–8763. PMLR, 2021. 2, 3, 6, 7

[28] 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 Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[29] Nicolae-Catalin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised predictive convolutional attentive block for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 3

[31] 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*, pages 14318–14328, 2022. 1, 3, 4, 6, 7, 8, 13

[32] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but DifferNet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1907–1916, 2021. 1, 2, 3, 7

[33] Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Lucas Deecke, Shoaib A. Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4393–4402, 2018. 4

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 3

[35] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban, and Hamid R Rabiee. Multiresolution knowledge distillation for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14902–14912, 2021. 7

[36] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 2, 3

[37] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. LAION-400M: Open dataset of CLIP-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 6

[38] Lavanya Sharan, Ce Liu, Ruth Rosenholtz, and Edward H Adelson. Recognizing materials using perceptually inspired features. *International Journal of Computer Vision*, 103(3):348–371, 2013. 3

[39] Shelly Sheynin, Sagie Benaim, and Lior Wolf. A hierarchical transformation-discriminating generative model for few shot anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8495–8504, 2021. 1, 2, 3, 7

[40] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599, 2020. 3

[41] Yoad Tewel, Yoav Shalev, Idan Schwartz, and Lior Wolf. ZeroCap: Zero-shot image-to-text generation for visual-semantic arithmetic. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17918–17928, 2022. 3

[42] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 12

[43] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. OFA: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 3

[44] Max Welling and Thomas N Kipf. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 3

[45] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE, 2010. 3

[46] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 2, 3

[47] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. Unified contrastive learning in image-text-label space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19163–19173, 2022. 3

[48] Minghui Yang, Peng Wu, Jing Liu, and Hui Feng. MemSeg: A semi-supervised method for image surface defect detection using differences and commonalities. *arXiv preprint arXiv:2205.00908*, 2022. 3

[26] MF Naeem, Y Xian, F Tombari 与 Zeynep Akata。面向组合零样本学习的图嵌入学习。发表于 *34th IEEE Conference on Computer Vision and Pattern Recognition*。IEEE，2021年。3[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark 等。从自然语言监督中学习可迁移的视觉模型。发表于 *International Conference on Machine Learning*，第8748–8763页。PMLR，2021年。2, 3, 6, 7[28] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou 与 Jiwen Lu。DenseCLIP：基于上下文感知提示的语言引导密集预测。发表于 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*，2022年。3[29] Nicolae-Catalin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund 与 Mubarak Shah。用于异常检测的自监督预测卷积注意力块。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，2022年。1[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser 与 Björn Ommer。基于潜在扩散模型的高分辨率图像合成，2021年。3[31] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox 与 Peter Gehler。迈向工业异常检测的完全召回。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第14318–14328页，2022年。1, 3, 4, 6, 7, 8, 13[32] Marco Rudolph, Bastian Wandt 与 Bodo Rosenhahn。Same same but DifferNet：基于标准化流的半监督缺陷检测。发表于 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*，第1907–1916页，2021年。1, 2, 3, 7[33] Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Lucas Deecke, Shoaib A. Siddiqui, Alexander Binder, Emmanuel Müller 与 Marius Kloft。深度单类分类。发表于 *Proceedings of the 35th International Conference on Machine Learning*，第80卷，第4393–4402页，2018年。4[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein 等。ImageNet大规模视觉识别挑战赛。*International Journal of Computer Vision*，115(3):211–252，2015年。3[35] Mohammadreza Salehi, Niousha Sadjadi, Soroosh Baselizadeh, Mohammad H Rohban 与 Hamid R Rabiee。用于异常检测的多分辨率知识蒸馏。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第14902–14912页，2021年。7[36] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk 与 Jenia Jitsev。LAION-5B：用于训练下一代图文模型的大规模开放数据集。发表于 *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*，2022年。2, 3[37] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, 和 Aran Komatsuzaki。LAION-400M：包含4亿个经CLIP筛选的图像-文本对的开放数据集。*arXiv preprint arXiv:2111.02114*，2021年。6[38] Lavanya Sharan, Ce Liu, Ruth Rosenholtz, 和 Edward H Adelson。使用感知启发特征识别材料。*International Journal of Computer Vision*，103(3):348–371，2013年。3[39] Shelly Sheynin, Sagie Benaim, 和 Lior Wolf。一种用于少样本异常检测的层次化变换判别生成模型。发表于 *Proceedings of the IEEE/CVF International Conference on Computer Vision*，第8495–8504页，2021年。1, 2, 3, 7[40] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, 和 Ludwig Schmidt。测量图像分类中对自然分布偏移的鲁棒性。*Advances in Neural Information Processing Systems*，33:18583–18599，2020年。3[41] Yoad Tewel, Yoav Shalev, Idan Schwartz, 和 Lior Wolf。ZeroCap：用于视觉语义算术的零样本图像到文本生成。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第17918–17928页，2022年。3[42] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, 和 Hervé Jégou。通过注意力训练数据高效的图像变换器与蒸馏。发表于 *International Conference on Machine Learning*，第10347–10357页。PMLR，2021年。12[43] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, 和 Hongxia Yang。OFA：通过简单的序列到序列学习框架统一架构、任务和模态。发表于 *International Conference on Machine Learn- ing*，第23318–23340页。PMLR，2022年。3[44] Max Welling 和 Thomas N Kipf。使用图卷积网络进行半监督分类。发表于 *International Conference on Learning Representations*，2017年。3[45] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, 和 Antonio Torralba。SUN数据库：从修道院到动物园的大规模场景识别。发表于 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*，第3485–3492页。IEEE，2010年。3[46] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, 和 Xiang Bai。一个基于预训练视觉语言模型进行开放词汇语义分割的简单基线。发表于 *European Conference on Computer Vi- sion*，第736–753页。Springer，2022年。2, 3[47] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, 和 Jianfeng Gao。图像-文本-标签空间中的统一对比学习。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第19163–19173页，2022年。3[48] Minghui Yang, Peng Wu, Jing Liu, 和 Hui Feng。MemSeg：一种利用差异性和共性进行图像表面缺陷检测的半监督方法。*arXiv preprint arXiv:2205.00908*，2022年。3

[49] Jihun Yi and Sungroh Yoon. Patch SVDD: Patch-level SVDD for anomaly detection and segmentation. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 3, 6, 7

[50] Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 192–199, 2014. 3

[51] 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. 3

[52] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In Edwin R. Hancock Richard C. Wilson and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press, September 2016. 13

[53] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. DRAEM – A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8330–8339, 2021. 1, 3

[54] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 6(1):1–23, 2019. 3

[55] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. *arXiv preprint arXiv:2010.00747*, 2020. 3

[56] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16793–16803, 2022. 3

[57] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from CLIP. In *European Conference on Computer Vision*, volume 3, page 8, 2022. 3, 6, 8

[58] 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, 2022. 3

[59] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. SPot-the-Difference self-supervised pre-training for anomaly detection and segmentation. In *Proceedings of the European Conference on Computer Vision*, 2022. 1, 3, 6, 12

[49] Jihun Yi 和 Sungroh Yoon。Patch SVDD：用于异常检测与分割的块级SVDD。发表于 *Proceedings of the Asian Conference on Computer Vision*，2020年。3, 6, 7[50] Aron Yu 和 Kristen Grauman。基于局部学习的细粒度视觉比较。发表于 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*，第192–199页，2014年。3[51] Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, 和 Liwei Wu。Fastflow：基于二维归一化流的无监督异常检测与定位。*arXiv preprint arXiv:2111.07677*，2021年。3[52] Sergey Zagoruyko 和 Nikos Komodakis。宽残差网络。载于 Edwin R. Hancock, Richard C. Wilson 和 William A. P. Smith 编辑的 *Proceedings of the British Machine Vi- sion Conference (BMVC)*，第87.1–87.12页。BMVA Press，2016年9月。13[53] Vitjan Zavrtanik, Matej Kristan, 和 Danijel Skočaj。DRAEM – 一种用于表面异常检测的判别性训练重建嵌入。发表于 *Proceedings of the IEEE/CVF International Conference on Computer Vision*，第8330–8339页，2021年。1, 3[54] Si Zhang, Hanghang Tong, Jiejun Xu, 和 Ross Maciejewski。图卷积网络：一项全面综述。*Computational Social Networks*，6(1):1–23，2019年。3[55] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, 和 Curtis P Langlotz。从配对图像和文本进行医学视觉表示的对比学习。*arXiv preprint arXiv:2010.00747*，2020年。3[56] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, 等。Regionclip：基于区域的语言-图像预训练。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第16793–16803页，2022年。3[57] Chong Zhou, Chen Change Loy, 和 Bo Dai。从CLIP中提取免费密集标签。发表于 *European Conference on Com- puter Vision*，第3卷，第8页，2022年。3, 6, 8[58] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, 和 Ziwei Liu。为视觉语言模型学习提示。*International Journal of Computer Vision*，130(9):2337–2348，2022年。3[59] Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, 和 Onkar Dabeer。SPot-the-Difference：用于异常检测与分割的自监督预训练。发表于 *Proceedings of the European Conference on Computer Vision*，2022年。1, 3, 6, 12

Supplementary Material

WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation

A. Experimental details

Compositional prompt ensemble. Figure 6 provides a detailed list of prompts we adopt to perform compositional prompt ensemble proposed in Section 4.1 of the main text. Recall that we consider two levels of prompts: *i.e.*, (a) state-level, and (b) template level. A complete prompt can be composed by replacing the token `[c]` in a template-level prompt with one of state-level prompt, either from the normal or anomaly states. Each of the state-level prompt takes an object-level label `[o]`. In our experiments, we use the object name words available for both MVTec-AD and VisA per dataset to replace `[o]`.

Data pre-processing. For CLIP-based models, including our proposed WinCLIP and WinCLIP+, we apply the data pre-processing pipeline given in OpenCLIP [16] for both MVTec-AD and VisA datasets to minimize potential train-test discrepancy. Specifically, it performs a channel-wise standardization with the pre-computed mean `[0.48145466, 0.4578275, 0.40821073]` and standard deviation `[0.26862954, 0.26130258, 0.27577711]` after normalizing each RGB image into $[0, 1]$, followed by a bicubic re-sizing based on the `Pillow` implementation. By default, we make the input resolution to be 240 for the shorter edge from the re-sizing, to be compatible with ViT-B/16+ in our experiments. This re-sizing policy also applies to other baseline models for fairer comparisons, although we keep the remaining parts of their original data pre-processing pipelines. In addition, similar policy can also be used in other backbones with input of different resolutions.

Evaluation metrics. Although the AUROC is a good metric for balanced dataset, it provides an inflated view of model performance in imbalanced dataset, especially in anomaly segmentation where the normal pixels dominate anomalies. This is also discussed by Zou et al. [59]. F_1-max is computed from the precision and recall for the anomalous samples at the optimal threshold, which is a more straightforward metric to measure the upper bound of anomaly prediction performance across thresholds. Thus we acknowledge that the low-shot anomaly segmentation is still not solved since our best model only achieves $< 60\%$ F_1-max for both MVTec-AD and VisA, even though WinCLIP+ achieves $> 95\%$ pixel-AUROC. In addition, our WinCLIP and WinCLIP+ outperform all the compared methods in terms of all these metrics on the setups, demonstrating the effectiveness of the proposed methods.

Other implementation details. (*i*) The ViT-B/16+ architecture [16], that we mainly adopt in our experiments, is a modification of ViT-B/16 [42] with (a) an increased dimension in both image ($768 \rightarrow 896$) and text ($512 \rightarrow 640$) embeddings, as well

(a) *State*-level (normal)

- `c := "[o]"`
- `c := "flawless [o]"`
- `c := "perfect [o]"`
- `c := "unblemished [o]"`
- `c := "[o] without flaw"`
- `c := "[o] without defect"`
- `c := "[o] without damage"`

(b) *State*-level (anomaly)

- `c := "damaged [o]"`
- `c := "[o] with flaw"`
- `c := "[o] with defect"`
- `c := "[o] with damage"`

(c) *Template*-level

- `"a cropped photo of the [c]."`
- `"a cropped photo of a [c]."`
- `"a close-up photo of a [c]."`
- `"a close-up photo of the [c]."`
- `"a bright photo of a [c]."`
- `"a bright photo of the [c]."`
- `"a dark photo of the [c]."`
- `"a dark photo of a [c]."`
- `"a jpeg corrupted photo of a [c]."`
- `"a jpeg corrupted photo of the [c]."`
- (cont'd) `"a blurry photo of the [c]."`
- `"a blurry photo of a [c]."`
- `"a photo of a [c]."`
- `"a photo of the [c]."`
- `"a photo of a small [c]."`
- `"a photo of the small [c]."`
- `"a photo of a large [c]."`
- `"a photo of the large [c]."`
- `"a photo of the [c] for visual inspection."`
- `"a photo of a [c] for visual inspection."`
- `"a photo of the [c] for anomaly detection."`
- `"a photo of a [c] for anomaly detection."`

Figure 6. Lists of multi-level prompts considered in this paper to construct compositional prompt ensemble.

补充材料

WinCLIP：零样本/少样本异常分类与分割

A. 实验细节

组合提示集成。图6详细列出了我们采用的主文本第4.1节中提出的组合提示集成方法所使用的提示。回顾我们考虑的两个提示层次：*i.e.*，(a) 状态级和(b) 模板级。完整的提示可以通过将模板级提示中的标记[c]替换为状态级提示（来自正常或异常状态之一）来构成。每个状态级提示都包含一个对象级标签[o]。在我们的实验中，我们使用MVTec-AD和VisA数据集中各自可用的对象名称词来替换[o]。

数据预处理。对于基于CLIP的模型，包括我们提出的WinCLIP和WinCLIP+，我们对MVTec-AD和VisA数据集均采用OpenCLIP [16]中给出的数据预处理流程，以最小化潜在的训练-测试差异。具体而言，在将每张RGB图像归一化至[0, 1]后，使用预先计算的均值[0.48145466, 0.4578275, 0.40821073]和标准差[0.26862954, 0.26130258, 0.27577711]进行通道级标准化，随后基于Pillow实现进行双三次重采样调整尺寸。默认情况下，我们将重采样后较短边的输入分辨率设置为240，以兼容实验中使用的ViT-B/16+。尽管我们保留了其他基线模型原始数据预处理流程的其余部分，但此尺寸调整策略同样应用于这些模型，以确保更公平的比较。此外，类似策略也可用于其他具有不同分辨率输入的骨干网络。

评估指标。尽管AUROC对于平衡数据集是一个很好的指标，但在不平衡数据集中，尤其是在正常像素占主导地位的异常分割任务中，它会夸大模型性能的表现。Zou等人[59]也对此进行了讨论。{v*}-max是根据异常样本在最佳阈值下的精确率和召回率计算得出的，这是一个更直接的指标，用于衡量跨阈值异常预测性能的上限。因此，我们承认少样本异常分割问题仍未解决，因为即使在WinCLIP{v*}达到95%像素AUROC的情况下，我们的最佳模型在MVTec-AD和VisA数据集上仅实现了约60%的{v*}-max。此外，我们的WinCLIP和WinCLIP{v*}在所有设置下的这些指标上均优于所有对比方法，证明了所提出方法的有效性。

其他实现细节。(*i*) 我们实验中主要采用的 ViT-B/16+ 架构 [16] 是对 ViT-B/16 [42] 的改进，包括 (a) 同时提升图像（768 → 896）与文本（512 → 640）嵌入的维度，以及

(a) *State*级（正常）

- c := "[哦]"
- c := "完美无瑕 [o]"
- c := "完美 [o]"
- c := "毫无瑕疵 [o]"
- c := "[o] 毫无瑕疵"
- c := "[o] 毫无瑕疵"
- c := "[o] 无损伤"

(b) *State*级（异常）

- c := "受损 [o]"
- c := "[o] 带有瑕疵"
- c := "[o] 带有缺陷"
- c := "[o] 带有损伤"

(c) *Template*级

- 一张[c]的裁剪照片。
- 一张[c]的裁剪照片。
- 一张[c]的特写照片。
- 一张[c]的特写照片。
- 一张明亮的[c]照片。
- 一张明亮的[c]照片。
- 一张[c]的暗色照片。
- 一张[c]的暗色照片。
- 一张[c]的jpeg损坏照片。
- 一张[c]的jpeg损坏照片。
- (续) "一张模糊的[c]照片。"
- 一张模糊的[c]照片。
- 一张[c]的照片。
- 一张[c]的照片。
- 一张小型[c]的照片。
- 一张小型[c]的照片。
- 一张大型[c]的照片。
- 一张大型[c]的照片。
- 一张用于视觉检查的[c]照片。
- 一张用于视觉检查的[c]照片。
- 一张用于异常检测的[c]照片。
- 一张用于异常检测的[c]照片。

图6. 本文为构建组合式提示集成所考虑的多层级提示列表。

MVTec-AD (few-shot)			Anomaly classification			Anomaly segmentation		
Setup	Method	Backbone	pAUROC	PRO	F_1-max	AUROC	AUPR	F_1-max
1-shot	PatchCore [31]	WRN-50-2	83.4±3.0	92.2±1.5	90.5±1.5	92.0±1.0	79.7±2.0	50.4±2.1
	PatchCore (hidden)	ViT-B/16+	79.9±4.8	88.6±2.7	88.7±1.1	91.8±1.0	74.1±2.3	47.6±2.6
	PatchCore (last)		83.3±3.8	90.7±2.1	89.8±1.4	92.3±0.9	74.5±2.2	47.7±2.9
	WinCLIP+ (ours)		**92.7**±1.9	**96.7**±0.7	**93.5**±1.0	**95.2**±0.5	**87.1**±1.2	**55.9**±2.7
2-shot	PatchCore [31]	WRN-50-2	86.3±3.3	93.8±1.7	92.0±1.5	93.3±0.6	82.3±1.3	53.0±1.7
	PatchCore (hidden)	ViT-B/16+	84.1±2.9	90.7±1.9	90.2±1.2	93.5±0.7	77.9±1.8	51.4±2.1
	PatchCore (last)		86.5±2.5	92.3±1.4	91.1±1.6	93.1±0.9	76.8±2.0	49.8±2.2
	WinCLIP+ (ours)		**94.0**±1.7	**97.0**±0.7	**94.0**±1.0	**96.0**±0.3	**88.4**±0.9	**58.4**±1.7
4-shot	PatchCore [31]	WRN-50-2	88.8±2.6	94.5±1.5	92.6±1.6	94.3±0.5	84.3±1.6	55.0±1.9
	PatchCore (hidden)	ViT-B/16+	87.5±3.1	92.5±1.8	91.7±1.4	94.7±0.6	81.2±1.6	54.4±1.9
	PatchCore (last)		89.9±2.2	94.2±1.4	92.7±1.1	94.0±0.6	78.9±1.6	52.2±1.5
	WinCLIP+ (ours)		**94.8**±1.5	**97.5**±0.7	**94.2**±0.9	**96.2**±0.3	**89.0**±0.8	**59.5**±1.8

Table 8. Comparison of few-shot performances on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement. Bold indicates the best performance.

MVTec-AD (zero-shot)			
Model	Size	AC	AS
RN50	224^2	79.8	65.6
RN101	224^2	79.2	62.4
RN50x4	288^2	81.9	71.3
RN50x16	384^2	82.3	65.3
ViT-B/16	224^2	86.1	71.1
ViT-B/16+	240^2	90.8	85.1
ViT-L/14	224^2	86.1	64.4

Table 9. Comparison of WinCLIP performance in AUROC (for AC) and pAUROC (for AS) on zero-shot MVTec-AD, across different CLIP backbone architectures.

as in (b) the input resolution ($224^2 \rightarrow 240^2$; $196 \rightarrow 225$ tokens); (*ii*) We note that CLIP models require the square-shaped resolution, *e.g.*, 240^2 for ViT-B/16+, to be compatible with the attention layers inside. Although the MVTec-AD benchmark already consists of square images, most of images in the VisA benchmark are non-squared (*e.g.*, 1500×1000) and simply taking a crop can affect the anomaly status of the given images. In this respect, to enable CLIP-based models properly handle non-squared images in our experiments, we perform a simple "image tiling" scheme. Specifically, for such non-squared images, we first extract multiple overlapping (squared) "tiles" of size the shorter edge L_s, by taking a sliding window across the longer edge. Then we average the predictions from the tiles to get the final (either in image- and pixel-level) prediction. The stride for the sliding is set to $0.8 \cdot L_s$ at most, *i.e.*, the tiles have overlaps with its neighbors at least in $0.2 \cdot L_s$; (*iii*) In addition, for the baseline results, we use our re-implementation of SPADE [7] and PaDiM [8], and adopt the official implementation of PatchCore[4] in our experiments.

B. Additional results on ablation study

Comparison with CLIP-based PatchCore: PatchCore [31], a current state-of-the-art considered in our experiments, is originally based on the internal features of convolutional network: *e.g.*, WideResNet-50-2 (WRN-50-2) [52] pre-trained on ImageNet. In Table 8, we test whether PatchCore can further benefit from the CLIP-based backbone that our WinCLIP+ is based on. Specifically, we additionally consider two variants of PatchCore that take the patch-token features of CLIP-based ViT-B/16+ backbone, one from (a) the 6^{th}- and 9^{th}-layer of ViT (which corresponds to `block2` and `block3` in ResNet-like models as considered by [31]; "*hidden*"), and the other one from (b) the last layer of ViT ("*last*"). Overall, we have the following observations. First, in case of the ViT-B/16+ backbone, PatchCore performs better with the last layer, which is in contrast to the cases of convolutional backbones. Second, compared to the original PatchCore, the CLIP-based variants achieve no better performances. Third, WinCLIP+ significantly outperforms "PatchCore (last)" where our WinCLIP+ also utilizes the last patch-token features, namely as referred as $\mathbf{F}^{\text{P}}$ (Section 4.3 of the main text). The results confirm the effectiveness of (a) our simple association-based module over a more sophisticated PatchCore[5] in ViT, and (b) the WinCLIP features $\mathbf{F}^{\text{W}}$.

Effect of different CLIP backbones: Table 9, on the other hand, explores the effect of different CLIP architectures to the WinCLIP zero-shot performance. Specifically, on zero-shot setups, we compare AUROC (and pixel-AUROC) from WinCLIP in AC (and AS) testing over the CLIP pre-trained models available at OpenCLIP,[6] including our default choice of ViT-B/16+. To apply WinCLIP for ResNet-based backbones, we notice that the CLIP implementation of ResNet architectures incorporates an attention layer to perform the feature pooling, namely as *attention pooling*, similar to ViT-based architectures. In this respect, for the CLIP-ResNet models, we apply our window-based inference to perform zero-shot AS from the convolutional feature map before the attention pooling, in the same way of applying WinCLIP for ViTs. Here, we remark that the effective patch size of each pixel on the last feature map (before the pooling) of ResNet-based models is designed to be 32 (the downsampling

[4] https://github.com/amazon-science/patchcore-inspection

[5] Technically, PatchCore incorporates several techniques upon a patch-level memory scheme, *e.g.*, local patch aggregation, clustering and score re-weighting.

[6] https://github.com/mlfoundations/open_clip

MVTec-AD (few-shot)			Anomaly classification			Anomaly segmentation		
Setup	Method	Backbone	pAUROC	PRO	F_1-max	AUROC	AUPR	F_1-max
1-shot	PatchCore [31]	WRN-50-2	83.4±3.0	92.2±1.5	90.5±1.5	92.0±1.0	79.7±2.0	50.4±2.1
	PatchCore (hidden)	ViT-B/16+	79.9±4.8	88.6±2.7	88.7±1.1	91.8±1.0	74.1±2.3	47.6±2.6
	PatchCore (last)		83.3±3.8	90.7±2.1	89.8±1.4	92.3±0.9	74.5±2.2	47.7±2.9
	WinCLIP+ (ours)		**92.7±1.9**	**96.7±0.7**	**93.5±1.0**	**95.2±0.5**	**87.1±1.2**	**55.9±2.7**
2-shot	PatchCore [31]	WRN-50-2	86.3±3.3	93.8±1.7	92.0±1.5	93.3±0.6	82.3±1.3	53.0±1.7
	PatchCore (hidden)	ViT-B/16+	84.1±2.9	90.7±1.9	90.2±1.2	93.5±0.7	77.9±1.8	51.4±2.1
	PatchCore (last)		86.5±2.5	92.3±1.4	91.1±1.6	93.1±0.9	76.8±2.0	49.8±2.2
	WinCLIP+ (ours)		**94.0±1.7**	**97.0±0.7**	**94.0±1.0**	**96.0±0.3**	**88.4±0.9**	**58.4±1.7**
4-shot	PatchCore [31]	WRN-50-2	88.8±2.6	94.5±1.5	92.6±1.6	94.3±0.5	84.3±1.6	55.0±1.9
	PatchCore (hidden)	ViT-B/16+	87.5±3.1	92.5±1.8	91.7±1.4	94.7±0.6	81.2±1.6	54.4±1.9
	PatchCore (last)		89.9±2.2	94.2±1.4	92.7±1.1	94.0±0.6	78.9±1.6	52.2±1.5
	WinCLIP+ (ours)		**94.8±1.5**	**97.5±0.7**	**94.2±0.9**	**96.2±0.3**	**89.0±0.8**	**59.5±1.8**

表8. MVTec-AD上少样本性能对比。我们报告了每个测量在5个随机种子上的平均值和标准差。**粗体**表示最佳性能。

MVTec-AD (zero-shot)			
Model	Size	AC	AS
RN50	224^2	79.8	65.6
RN101	224^2	79.2	62.4
RN50x4	288^2	81.9	71.3
RN50x16	384^2	82.3	65.3
ViT-B/16	224^2	86.1	71.1
ViT-B/16+	240^2	90.8	85.1
ViT-L/14	224^2	86.1	64.4

表9. 在不同CLIP骨干架构上，WinCLIP在零样本MVTec-AD数据集上AUROC（用于异常分类）和pAUROC（用于异常分割）的性能对比。

如(b)中所述，输入分辨率（$224^2 \rightarrow 240^2$；196 → 225个标记）；(*ii*) 我们注意到CLIP模型需要方形分辨率，*e.g.*，例如ViT-B/16+的240^2，以兼容内部的注意力层。尽管MVTec-AD基准测试已包含方形图像，但VisA基准测试中的大多数图像为非方形（*e.g.*，例如1500 × 1000），简单地裁剪可能影响给定图像的异常状态。为此，为使基于CLIP的模型在我们的实验中能正确处理非方形图像，我们采用了一种简单的“图像分块”方案。具体而言，对于此类非方形图像，我们首先沿较长边滑动窗口，提取多个重叠的（方形）“块”，其尺寸为较短边长度L_s。然后对来自各块的预测结果进行平均，以获得最终（图像级和像素级）预测。滑动步长最多设置为$0.8 \cdot L_s$，*i.e.*，即各块与相邻块至少重叠$0.2 \cdot L_s$；(*iii*) 此外，对于基线结果，我们使用自行重新实现的SPADE[7]和PaDiM[8]，并在实验中采用PatchCore[4]的官方实现。

B. 消融研究的补充结果

与基于CLIP的PatchCore对比：PatchCore[31]作为我们实验中考虑的当前最先进方法，其原始设计基于卷积网络内部特征：*e.g.*——即在ImageNet上预训练的WideResNet-50-2（WRN-50-2）[52]。在表8中，我们测试了PatchCore是否能从我们WinCLIP+所基于的CLIP骨干网络中进一步获益。具体而言，我们额外考虑了两种PatchCore变体，它们采用基于CLIP的ViT-B/16+骨干网络的补丁令牌特征：一种来自(a) ViT的第6^{th}层和第9^{th}层（对应[31]所考虑的类ResNet模型中的block2和block3；记为“*hidden*”），另一种来自(b) ViT的最后一层（记为“*last*”）。总体而言，我们得到以下观察结果：首先，对于ViT-B/16+骨干网络，PatchCore在最后一层表现更佳，这与卷积骨干网络的情况相反；其次，相较于原始PatchCore，基于CLIP的变体未能获得更好的性能；第三，WinCLIP+显著优于“PatchCore（最后一层）”——尽管我们的WinCLIP+同样利用了最后一层补丁令牌特征（即正文$\mathbf{F}^{\text{P}}$ (第4.3节)所述）。这些结果证实了以下有效性：(a) 在ViT中，我们简单的基于关联的模块优于更复杂的PatchCore[5]；(b) WinCLIP特征$\mathbf{F}^{\text{W}}$的优越性。

不同CLIP主干网络的影响：另一方面，表9探讨了不同CLIP架构对WinCLIP零样本性能的影响。具体而言，在零样本设置下，我们比较了WinCLIP在AC（和AS）测试中基于OpenCLIP提供的CLIP预训练模型（包括我们的默认选择ViT-B/16）所获得的AUROC（和像素级AUROC）。为了将WinCLIP应用于基于ResNet的主干网络，我们注意到ResNet架构的CLIP实现包含一个注意力层来执行特征池化，即*attention pooling*，类似于基于ViT的架构。在这方面，对于CLIP-ResNet模型，我们采用基于窗口的推理方法，在注意力池化之前的卷积特征图上执行零样本AS，这与将WinCLIP应用于ViT的方式相同。在此，我们指出，基于ResNet的模型在最后特征图（池化之前）上每个像素的有效补丁大小被设计为32（下采样

[4] https://github.com/amazon-science/patchcore-inspection

[5] Technically, PatchCore incorporates several techniques upon a patch-level memory scheme, *e.g.*, local patch aggregation, clustering and score re-weighting.

[6] https://github.com/mlfoundations/open_clip

rate), which is larger than those of ViTs we test, *e.g.*, of 16. Overall, we observe that ViT-based models generally show better performance compared to ResNets, in both AC and AS. The particular gap in AS is possibly due to the bigger patch sizes in ResNets, which can result in more blurry outputs. Still, we observe the performance benefits from larger models or resolutions in both types of architecture.

C. Additional qualitative results

In Figure 7-10, we provide further qualitative results obtained from our (zero-shot) WinCLIP and (few-shot) WinCLIP+ for anomaly segmentation, both in MVTec-AD and VisA considered in our experiments. Specifically, we report MVTec-AD results in Figure 7 and 8, and VisA results in Figure 9 and 10.

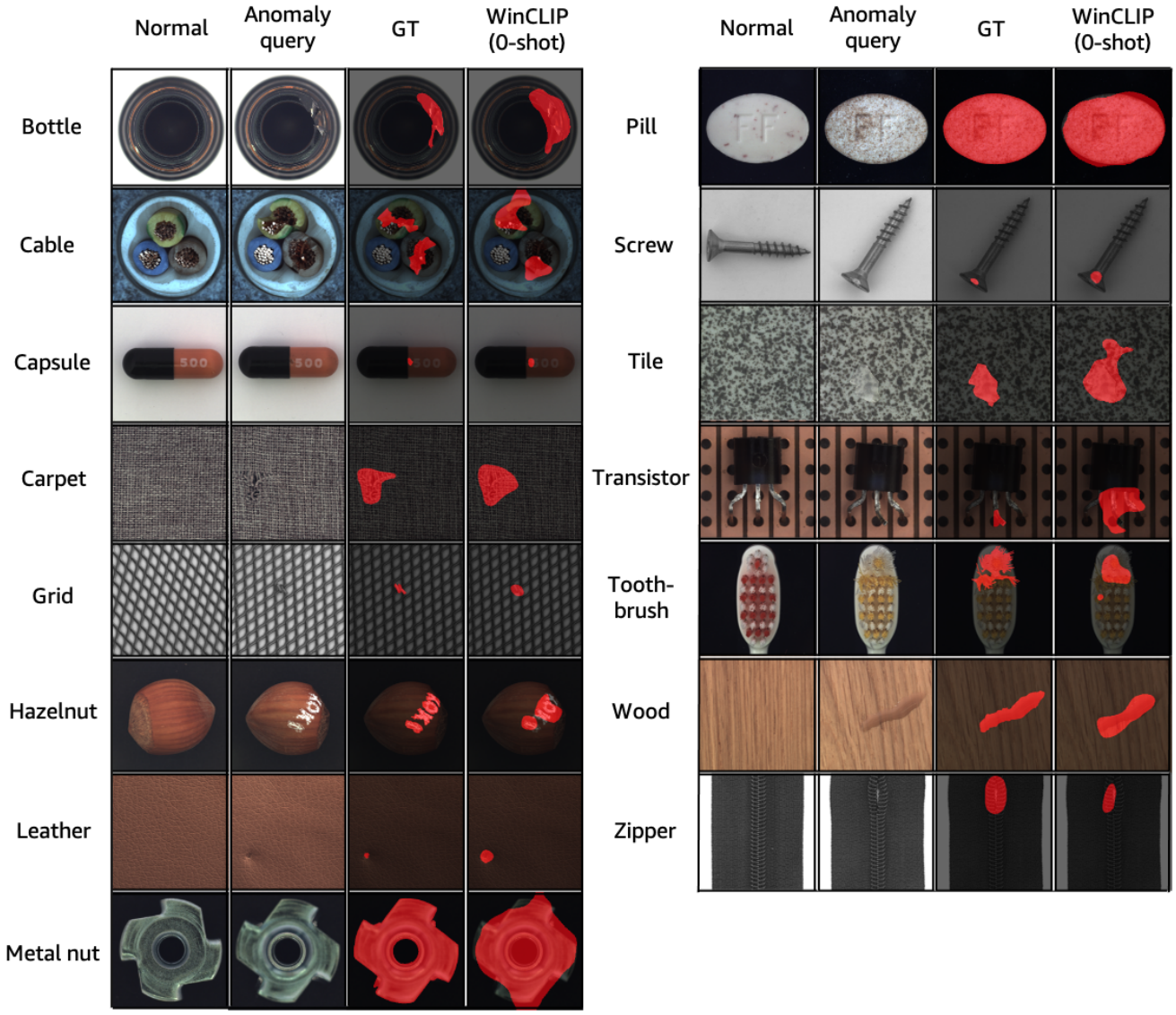


Figure 7. Additional qualitative results from WinCLIP (0-shot), tested on MVTec-AD.

率），这比我们测试的ViTs模型（*e.g.*，为16）要高。总体而言，我们观察到基于ViT的模型在AC和AS方面通常表现出比ResNets更好的性能。AS中的明显差距可能是由于ResNets中较大的补丁尺寸，这可能导致输出更加模糊。尽管如此，我们观察到在两种架构类型中，更大模型或更高分辨率都能带来性能提升。

C. 其他定性结果

在图7-10中，我们展示了从（零样本）WinCLIP和（少样本）WinCLIP+获得的异常分割进一步定性结果，这些结果均基于我们实验中使用的MVTec-AD和VisA数据集。具体而言，图7和图8展示了MVTec-AD的结果，图9和图10展示了VisA的结果。

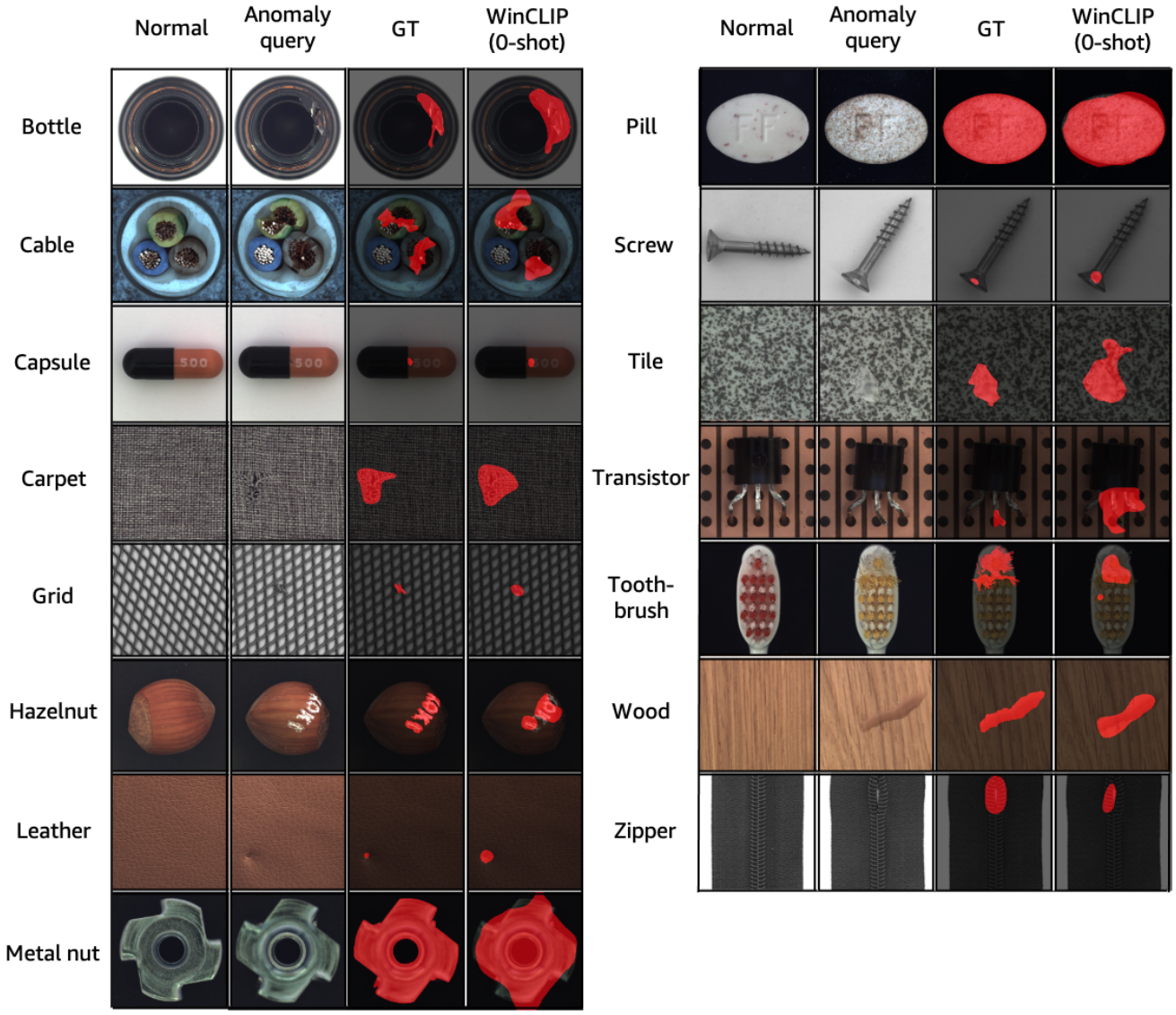


图7. 来自WinCLIP（零样本）在MVTec-AD上测试的额外定性结果。

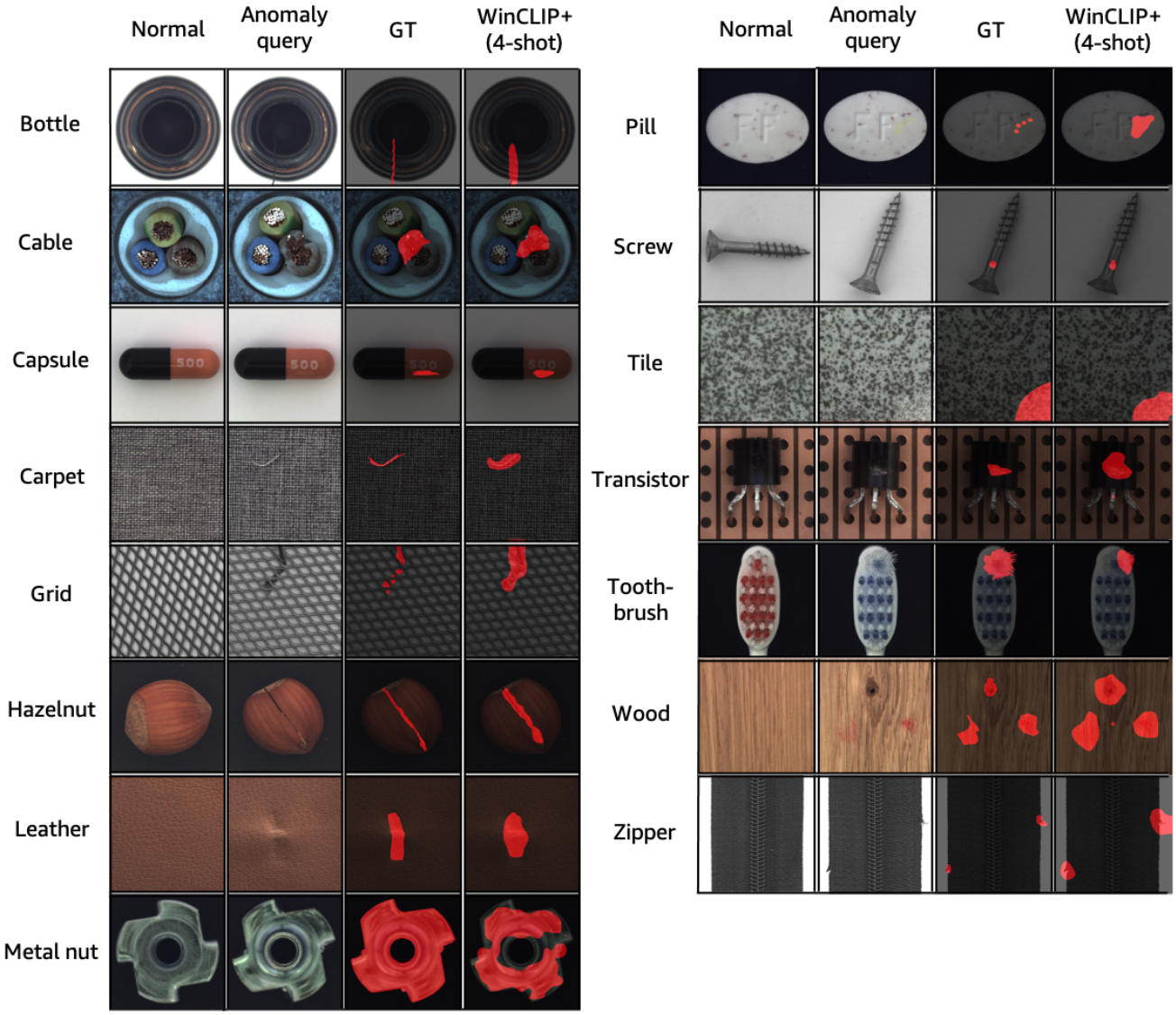


Figure 8. Additional qualitative results from few-shot WinCLIP+ (4-shot), tested on MVTec-AD.

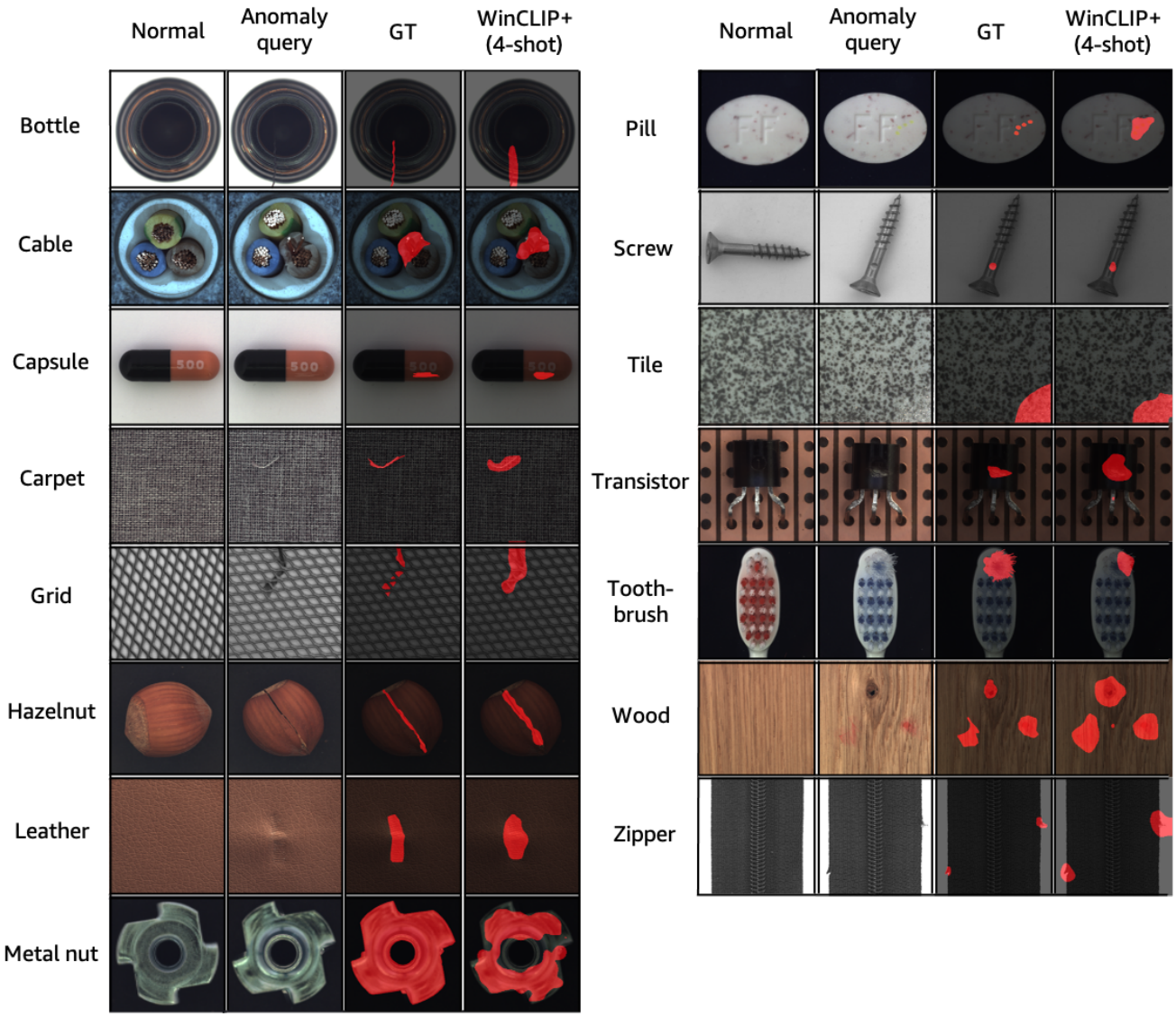


图8. 在MVTec-AD上测试的少样本WinCLIP+ (4-shot)的额外定性结果。

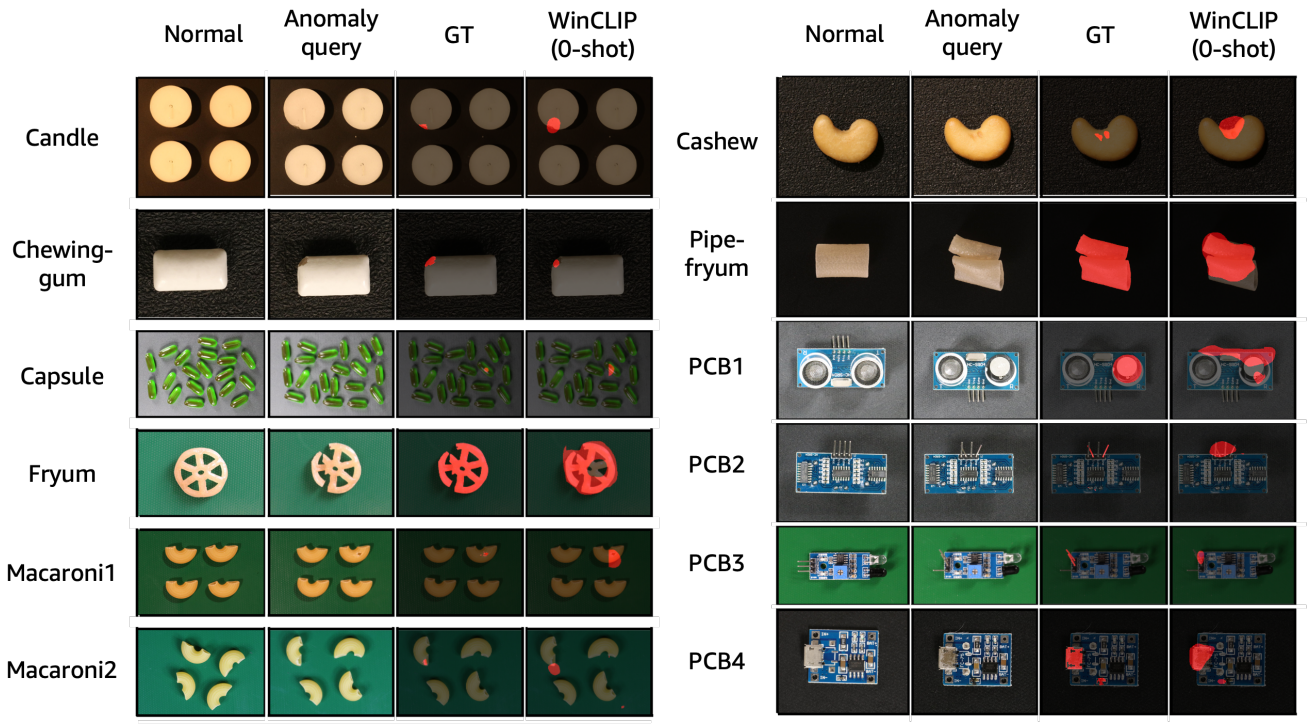


Figure 9. Additional qualitative results from WinCLIP (0-shot), tested on VisA.

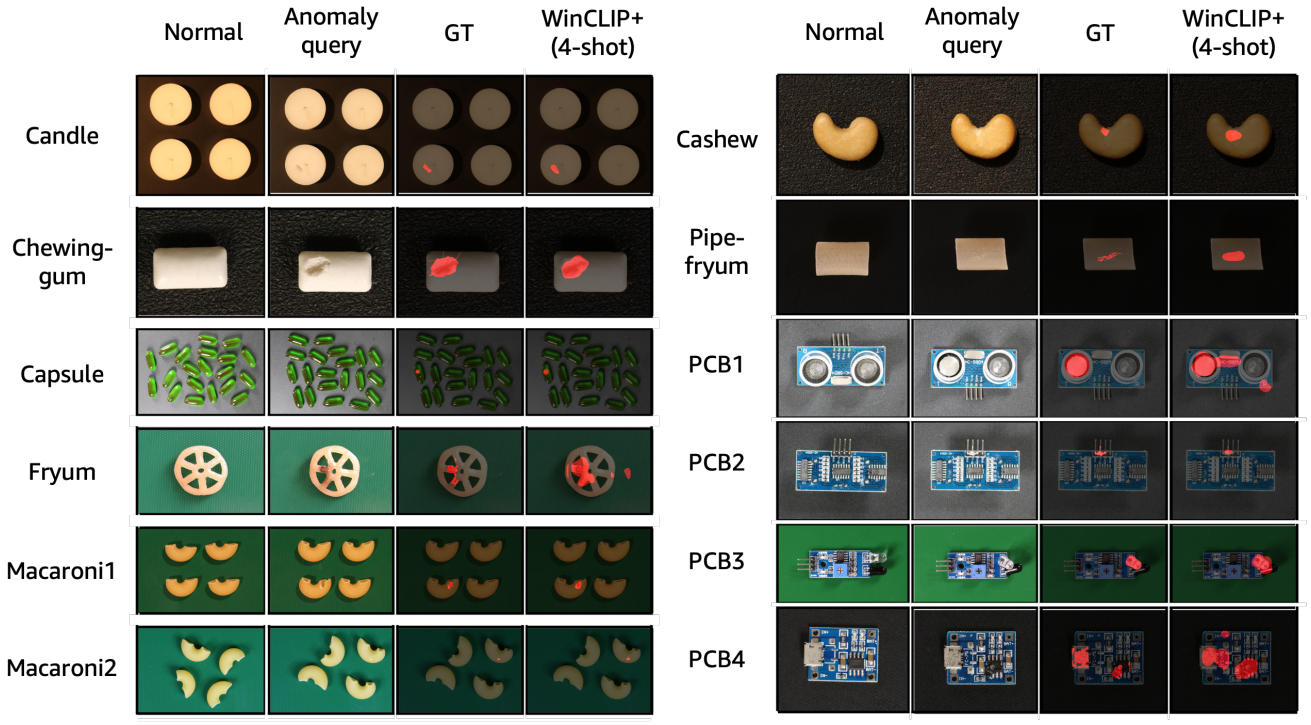


Figure 10. Additional qualitative results from few-shot WinCLIP+ (4-shot), tested on VisA.

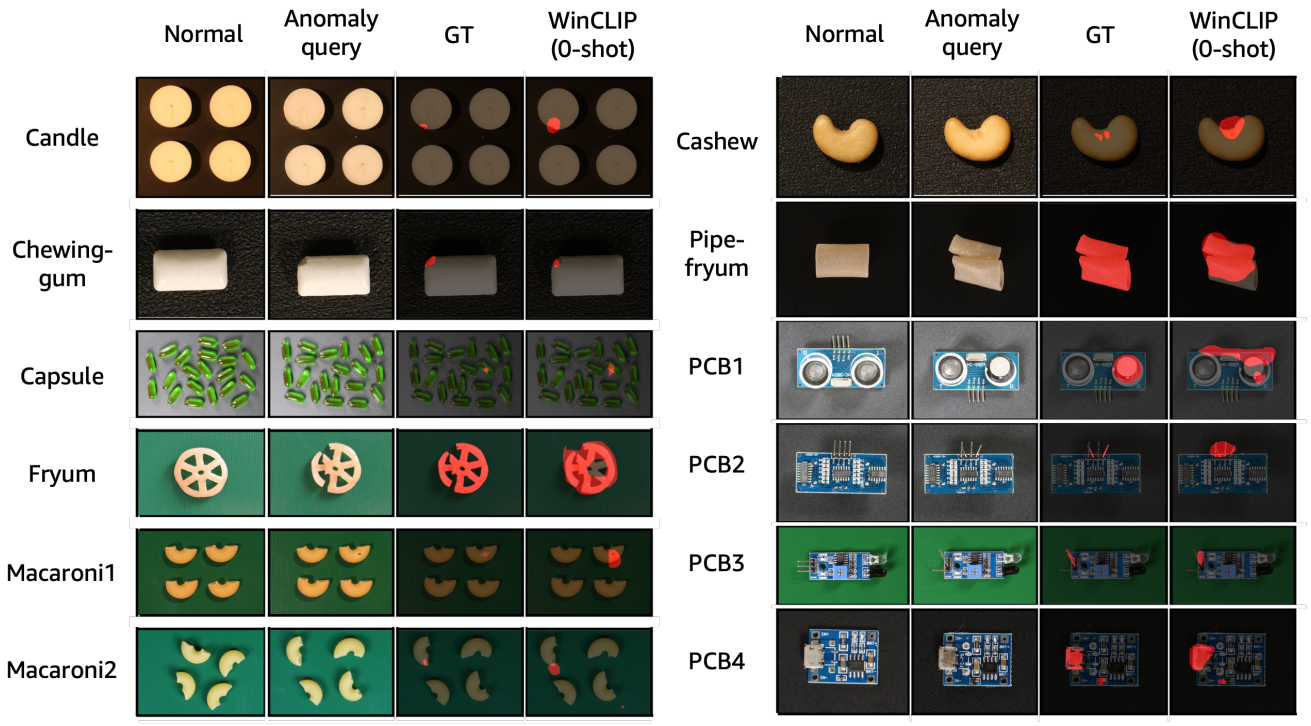


图9. 在VisA数据集上测试的WinCLIP（零样本）附加定性结果。

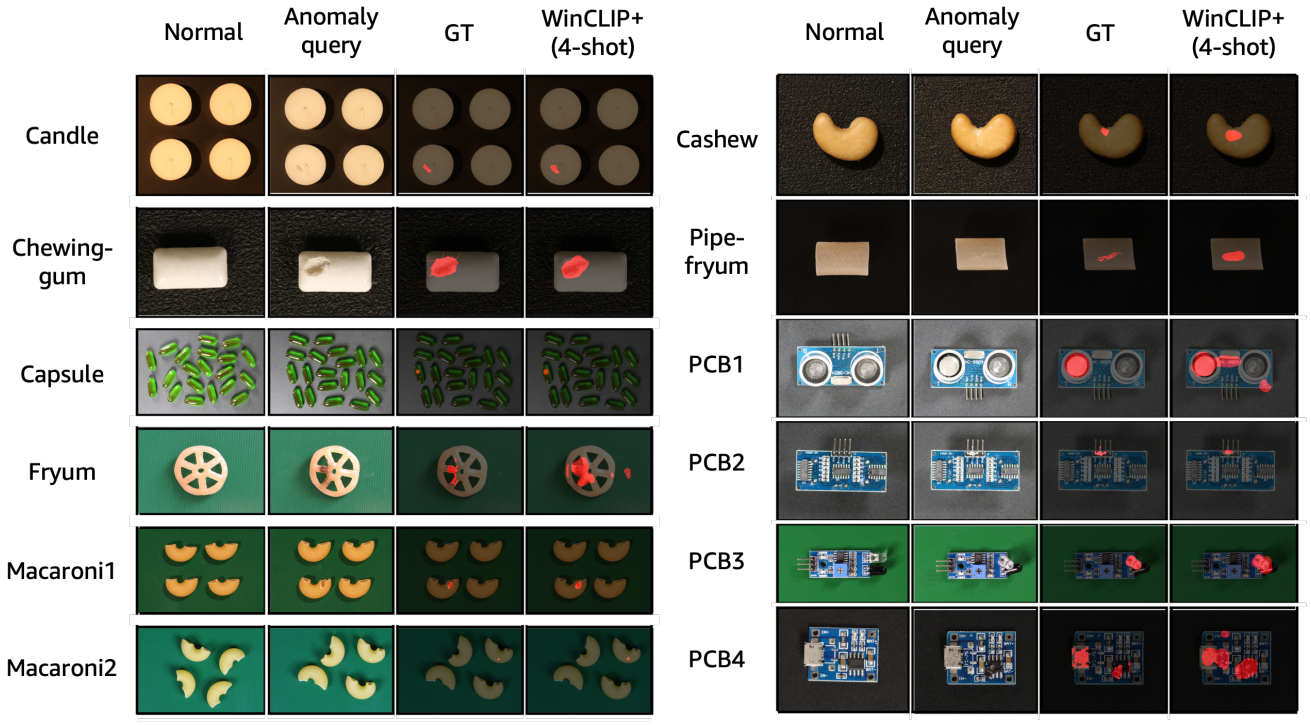


图10. 在VisA数据集上测试的少样本WinCLIP+ (4样本)的额外定性结果。

Failure cases. We present some failure examples from both MVTec-AD and VisA for language driven zero-shot WinCLIP in Figure 11. Note that the normal images are shown just for better illustration and are not used in model prediction. The first major factor causing the failure is the logical anomaly [2] illustrated in Figure 11(a), *e.g.*, misplaced axis in cable, missing text on capsule, missing capacitor in PCB1 and bent component in PCB3. Such type of anomalies need to be clarified by normal reference images while language might be not sufficient. The issues are alleviated by our few-normal-shot WinCLIP+. The second major factor refers to tiny defect illustrated in Figure 11(b), such as the ones in carpet, wood, capsule, macaroni1. We conjecture that spatial features with more local details might improve these cases, which is left for future exploration. The third major factor is the irrelevant deviation from normality that are not defects of interests illustrated in Figure 11(c), *e.g.*, the tiny red/white dots in pill/hazelnut, extra ingredient on cashew, designed holes and acceptable scratches in PCB2. We hypothesize that more clarification on these deviation and a pre-trained model with better understanding on these states might alleviate the problem. Lastly, although WinCLIP can roughly localize anomalies such as the cases in bottle, tile, PCB4 and fryum, it makes some errors around the true positives, illustrated in Figure 11(d). However, we argue this is minor as the rough anomaly localization is sufficient to explain where the defects are for visual inspection.

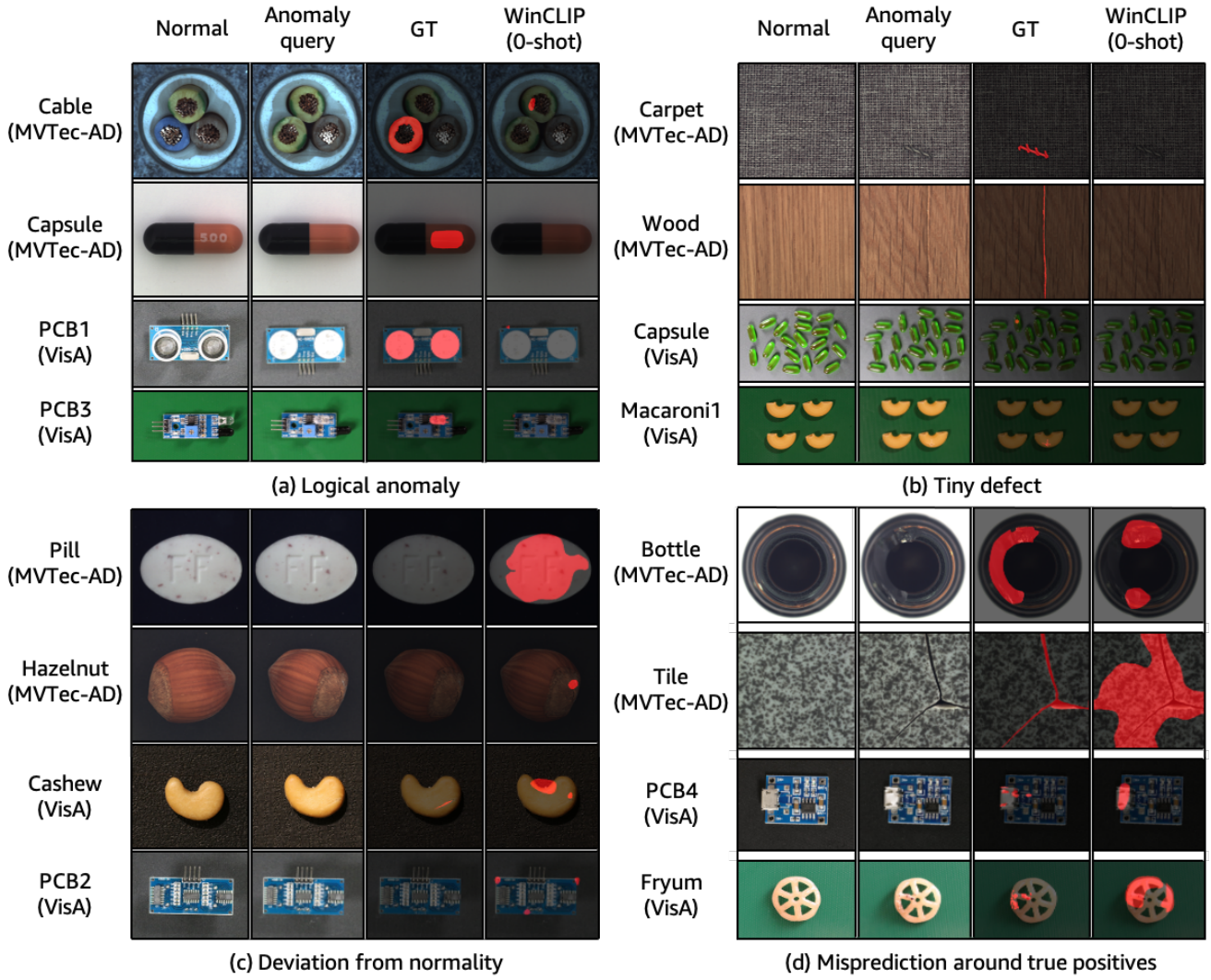


Figure 11. Curated illustrations of failure cases from zero-shot WinCLIP.

失败案例。我们在图11中展示了MVTec-AD和VisA数据集上语言驱动零样本WinCLIP的一些失败示例。请注意，正常图像仅用于更清晰的展示，并未用于模型预测。导致失败的第一个主要因素是逻辑异常[2]，如图11(a)所示，*e.g.*，包括电缆中错位的轴线、胶囊上缺失的文字、PCB1中缺失的电容器以及PCB3中弯曲的元件。此类异常需要正常参考图像来明确判断，而语言描述可能不够充分。我们的少量正常样本WinCLIP+缓解了这些问题。第二个主要因素涉及微小缺陷，如图11(b)所示的地毯、木材、胶囊、通心粉1中的案例。我们推测包含更多局部细节的空间特征可能改善这些情况，这留待未来探索。第三个主要因素是与正常状态无关但并非目标缺陷的偏差，如图11(c)所示，*e.g.*，包括药片/榛子上的微小红/白点、腰果上的额外附着物、PCB2上设计的孔洞和可接受的划痕。我们假设对这些偏差进行更明确的说明，以及使用能更好理解这些状态的预训练模型，可能缓解该问题。最后，尽管WinCLIP能大致定位异常（如瓶子、瓷砖、PCB4和fryum中的案例），但在真实异常区域周边仍存在一些误差，如图11(d)所示。然而我们认为这影响较小，因为粗略的异常定位已足以说明缺陷位置以供人工检测。

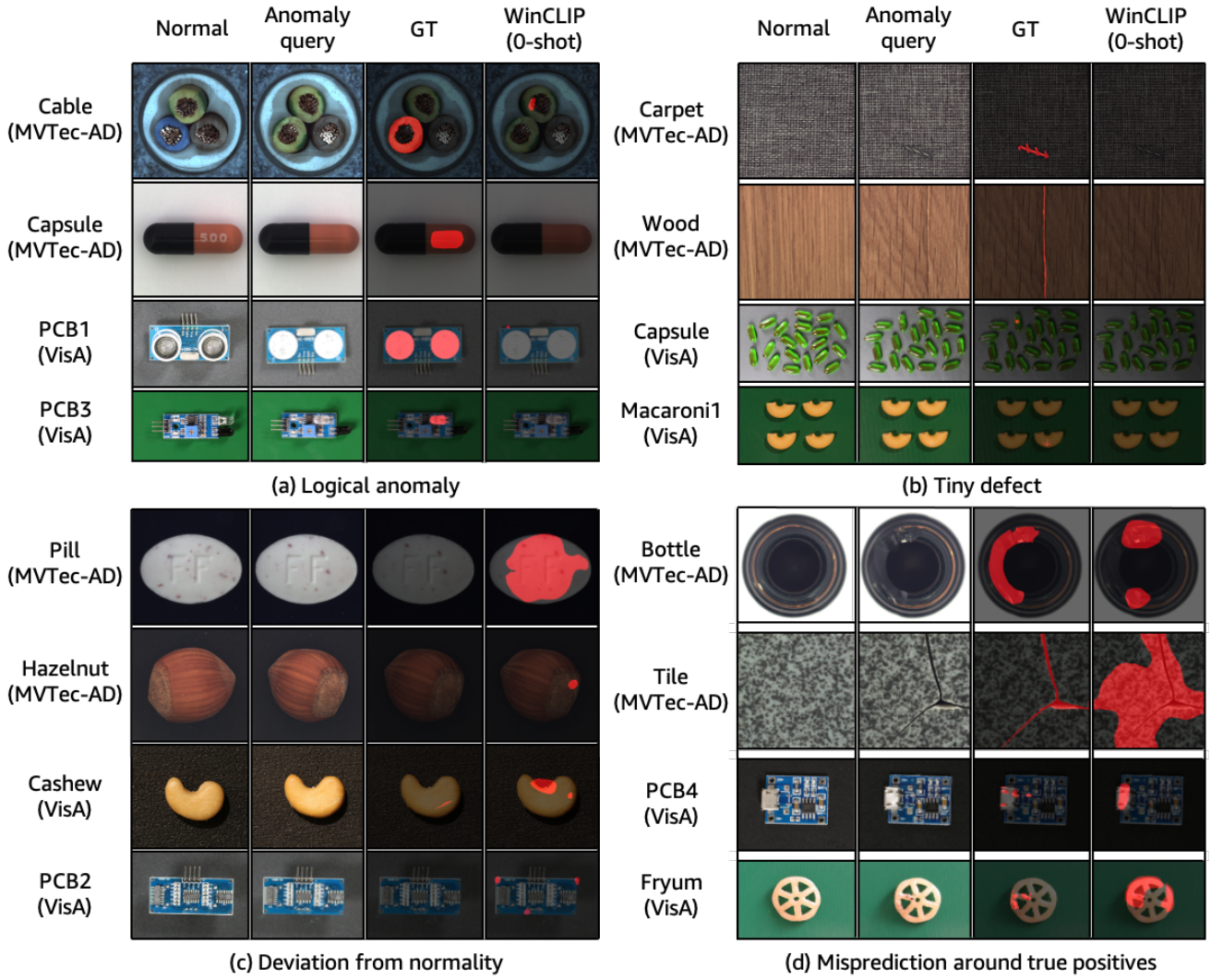


图11. 零样本WinC失败案例的精选图示

唇部。

D. Detailed quantitative results

In this section, we report the detailed, subset-level performance values for the evaluation metrics provided in Table 1 and 4 of the main text. Specifically, we report MVTec-AD results in Table 10-15 and VisA results in Table 16-21.

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	99.2±0.0	98.7±0.6	97.4±0.7	99.4±0.4	98.2±0.9	99.5±0.1	98.5±1.0	99.2±0.3	99.3±0.3	99.5±0.2	98.8±0.2	99.2±0.3	99.3±0.4
Cable	86.5±0.0	71.2±3.3	57.7±4.6	88.8±4.2	88.9±1.9	76.2±5.2	62.3±5.9	91.0±2.7	88.4±0.7	83.4±3.1	70.0±6.1	91.0±2.7	90.9±0.9
Capsule	72.9±0.0	70.2±3.0	57.7±7.3	67.8±2.9	72.3±6.8	70.9±6.1	64.3±3.0	72.8±7.0	77.3±8.8	78.9±5.5	65.2±2.5	72.8±7.0	82.3±8.9
Carpet	100.0±0.0	98.1±0.2	96.6±1.0	95.3±0.8	99.8±0.3	98.3±0.4	97.8±0.5	96.6±0.5	99.8±0.3	98.6±0.2	97.9±0.4	96.6±0.5	100.0±0.0
Grid	98.8±0.0	40.0±6.8	54.2±6.7	63.6±10.3	99.5±0.3	41.3±3.6	67.2±4.2	67.7±8.3	99.4±0.2	44.6±6.6	68.1±3.8	67.7±8.3	99.6±0.1
Hazelnut	93.9±0.0	95.8±1.3	88.3±2.6	88.3±2.7	97.5±1.4	96.2±2.1	90.8±0.8	93.2±3.8	98.3±0.7	98.4±1.3	91.9±1.2	93.2±3.8	98.4±0.4
Leather	100.0±0.0	100.0±0.0	97.5±0.7	97.3±0.7	99.9±0.0	100.0±0.0	97.5±0.9	97.9±0.7	99.9±0.0	100.0±0.0	98.5±0.2	97.9±0.7	100.0±0.0
Metal nut	97.1±0.0	71.0±2.2	53.0±3.8	73.4±2.9	98.7±0.8	77.0±7.9	54.8±3.8	77.7±8.5	99.4±0.2	77.8±5.7	60.7±5.2	77.7±8.5	99.5±0.2
Pill	79.1±0.0	86.5±3.1	61.3±3.8	81.9±2.8	91.2±2.1	84.8±0.9	59.1±6.4	82.9±2.9	92.3±0.7	86.7±0.3	54.9±2.7	82.9±2.9	92.8±1.0
Screw	83.3±0.0	46.7±2.5	55.0±2.5	44.4±4.6	86.4±0.9	46.6±2.2	54.0±4.4	49.0±3.8	86.0±2.1	50.5±5.4	50.0±4.1	49.0±3.8	87.9±1.2
Tile	100.0±0.0	99.9±0.1	92.2±2.2	99.0±0.9	99.9±0.0	99.9±0.1	93.3±1.1	98.5±1.0	99.9±0.2	100.0±0.0	93.1±0.6	98.5±1.0	99.9±0.1
Toothbrush	87.5±0.0	71.7±2.6	82.5±1.2	83.3±3.8	92.2±4.9	78.6±3.2	87.6±4.2	85.9±3.5	97.5±1.6	78.8±5.2	89.2±2.5	85.9±3.5	96.7±2.6
Transistor	88.0±0.0	77.2±2.0	73.3±6.0	78.1±6.9	83.4±3.8	81.3±3.7	72.8±6.3	90.0±4.3	85.3±1.7	81.4±2.1	82.4±6.5	90.0±4.3	85.7±2.5
Wood	99.4±0.0	98.8±0.3	96.1±1.2	97.8±0.3	99.9±0.1	99.2±0.4	96.9±0.5	98.3±0.6	99.9±0.1	98.9±0.6	97.0±0.2	98.3±0.6	99.8±0.3
Zipper	91.5±0.0	89.3±1.9	85.8±2.7	92.3±0.5	88.8±5.9	93.3±2.9	86.3±2.6	94.0±2.1	94.0±1.4	95.1±1.3	88.3±2.0	94.0±2.1	94.5±0.5
Mean	**91.8±0.0**	81.0±2.0	76.6±3.1	83.4±3.0	**93.1±2.0**	82.9±2.6	78.9±3.1	86.3±3.3	**94.4±1.3**	84.8±2.5	80.4±2.5	88.8±2.6	**95.2±1.3**

Table 10. Comparison of anomaly classification (AC) performance in terms of class-wise AUROC on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUPR	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	99.8±0.0	99.6±0.1	99.2±0.2	99.8±0.1	99.4±0.3	99.8±0.0	99.6±0.3	99.8±0.1	99.8±0.1	99.9±0.0	99.7±0.0	99.8±0.1	99.8±0.1
Cable	91.2±0.0	79.6±2.3	64.9±3.8	93.8±2.2	93.2±1.1	84.5±3.1	69.6±6.6	95.1±1.3	92.9±0.6	88.8±1.9	76.1±5.6	97.1±0.7	94.4±0.3
Capsule	91.5±0.0	91.2±0.9	86.9±2.2	89.4±2.0	91.6±2.7	91.6±2.1	88.4±0.8	91.0±2.9	93.3±3.6	94.4±1.9	87.8±0.8	94.9±1.1	95.1±3.3
Carpet	100.0±0.0	99.4±0.0	99.0±0.2	98.7±0.2	99.9±0.1	99.5±0.1	99.4±0.1	99.0±0.1	99.9±0.1	99.6±0.1	99.4±0.1	98.8±0.2	100.0±0.0
Grid	99.6±0.0	66.9±2.1	75.0±3.3	81.1±4.9	99.9±0.1	68.3±2.1	82.5±2.3	84.1±4.0	99.8±0.1	68.8±4.2	83.0±1.8	86.4±4.0	99.9±0.0
Hazelnut	96.9±0.0	97.9±0.6	93.3±1.7	92.9±2.2	98.6±0.7	98.0±1.1	94.1±0.5	96.0±2.0	99.1±0.4	99.1±0.7	94.8±0.6	97.0±1.2	99.1±0.2
Leather	100.0±0.0	100.0±0.0	99.2±0.2	99.1±0.2	100.0±0.0	100.0±0.0	99.2±0.3	99.3±0.2	100.0±0.0	100.0±0.0	99.6±0.1	99.6±0.1	100.0±0.0
Metal nut	99.3±0.0	91.7±0.8	82.0±2.7	91.0±1.1	99.7±0.2	93.7±2.4	82.2±1.4	92.3±4.0	99.9±0.0	94.1±1.8	85.5±1.7	97.0±2.6	99.9±0.1
Pill	95.7±0.0	97.0±0.8	88.3±1.3	96.5±0.6	98.3±0.5	96.5±0.4	87.9±2.6	96.6±0.7	98.6±0.1	97.0±0.2	87.0±1.2	96.9±0.4	98.6±0.2
Screw	93.1±0.0	71.3±1.8	78.1±1.0	71.4±2.3	94.2±0.6	71.0±1.4	77.3±1.3	72.9±3.4	94.1±1.5	73.7±2.4	75.7±2.8	71.8±1.9	94.9±0.8
Tile	100.0±0.0	100.0±0.0	97.2±0.7	99.6±0.3	100.0±0.0	100.0±0.0	97.6±0.4	99.4±0.4	100.0±0.1	100.0±0.0	97.6±0.2	99.6±0.1	100.0±0.0
Toothbrush	95.6±0.0	88.3±0.6	93.7±0.5	93.5±1.4	96.7±2.0	90.8±1.3	95.2±1.6	94.1±1.4	99.0±0.6	91.3±2.6	95.8±0.7	94.8±0.7	98.7±1.1
Transistor	87.1±0.0	76.2±1.7	66.2±7.5	77.7±5.5	79.0±4.0	81.6±3.4	69.0±6.5	89.3±3.9	80.7±2.3	80.3±2.6	77.6±8.4	84.5±9.0	80.7±3.2
Wood	99.8±0.0	99.6±0.1	98.8±0.3	99.3±0.1	100.0±0.0	99.7±0.1	99.0±0.1	99.5±0.2	100.0±0.0	99.7±0.2	99.1±0.0	99.5±0.2	99.9±0.1
Zipper	97.5±0.0	96.9±0.5	95.5±0.9	97.2±0.3	96.8±1.8	98.2±0.8	95.4±1.0	97.8±1.0	98.3±0.4	98.6±0.4	96.2±0.8	99.1±0.7	98.5±0.2
Mean	**96.5±0.0**	90.6±0.8	88.1±1.7	92.2±1.5	**96.5±0.9**	91.7±1.2	89.3±1.7	93.8±1.7	**97.0±0.7**	92.5±1.2	90.5±1.6	94.5±1.5	**97.3±0.6**

Table 11. Comparison of anomaly classification (AC) performance in terms of class-wise AUPR on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	97.6±0.0	97.8±0.8	96.3±1.2	98.3±0.6	96.5±1.3	98.7±0.4	97.1±1.1	97.5±0.6	97.7±0.7	98.6±0.3	97.9±0.4	97.9±0.8	97.8±0.6
Cable	84.5±0.0	79.6±2.3	77.2±1.1	85.2±3.6	86.1±1.3	80.4±1.7	78.7±1.2	86.1±2.4	85.2±0.7	83.8±2.5	81.1±1.1	91.3±1.0	87.2±0.6
Capsule	91.4±0.0	92.0±0.6	91.0±0.2	92.0±1.0	91.6±0.7	92.1±0.4	92.1±0.9	93.6±0.6	92.1±0.7	92.7±0.3	92.8±0.9	94.3±0.3	92.5±0.5
Carpet	99.4±0.0	96.5±0.2	95.1±0.5	94.9±0.5	99.2±0.8	96.6±0.3	96.5±0.4	95.3±0.5	99.3±0.7	96.9±0.3	96.6±0.3	94.3±0.8	99.9±0.2
Grid	98.2±0.0	84.5±0.3	84.5±0.3	86.2±1.1	98.9±0.4	84.8±0.3	85.3±0.9	86.9±2.3	99.1±0.0	84.8±0.5	85.0±0.5	87.5±2.0	99.1±0.0
Hazelnut	89.7±0.0	92.4±1.3	87.4±1.6	87.0±1.4	94.7±2.3	93.2±2.8	89.3±1.1	91.0±3.7	95.6±1.6	95.9±2.0	90.0±1.7	92.8±1.2	96.2±1.0
Leather	100.0±0.0	99.9±0.2	96.2±0.9	95.9±0.7	99.5±0.0	100.0±0.0	96.6±1.3	95.7±0.9	99.7±0.2	100.0±0.0	97.9±0.2	97.5±0.7	99.8±0.2
Metal nut	96.3±0.0	90.1±0.6	90.1±0.3	91.4±1.2	97.7±1.0	90.5±1.1	90.0±0.3	91.9±0.9	98.4±0.5	90.6±0.9	90.3±0.4	93.6±1.4	98.5±0.6
Pill	91.6±0.0	93.5±0.4	91.6±0.0	91.9±0.3	93.8±0.7	93.4±0.3	91.7±0.1	92.0±0.3	94.3±0.4	93.6±0.5	91.7±0.1	92.1±0.3	94.1±0.4
Screw	87.4±0.0	85.3±0.0	85.6±0.3	85.8±0.2	88.5±0.3	85.6±0.2	85.5±0.1	85.7±0.2	89.0±0.6	85.8±0.7	85.7±0.3	86.8±0.6	89.6±0.7
Tile	99.4±0.0	99.2±0.5	90.7±2.0	97.5±1.2	98.9±0.2	99.2±0.3	91.5±1.3	96.9±1.5	99.2±0.3	99.4±0.0	91.0±0.9	97.5±0.4	99.2±0.3
Toothbrush	87.9±0.0	85.6±1.3	85.4±1.3	88.9±2.2	94.1±1.9	87.1±1.4	90.1±3.1	90.8±1.6	96.7±1.8	86.5±1.8	90.6±2.1	92.6±2.2	96.8±2.3
Transistor	79.5±0.0	70.3±1.7	66.4±4.7	70.6±7.2	75.1±3.1	73.4±3.4	65.9±3.7	85.3±6.1	75.9±2.4	72.3±2.7	74.8±7.9	78.3±11.5	76.6±2.8
Wood	98.3±0.0	97.0±0.8	94.3±1.1	96.1±0.3	99.4±0.3	97.7±1.0	94.9±0.3	96.6±0.9	99.5±0.4	97.6±0.7	94.8±0.5	96.5±0.9	99.2±0.9
Zipper	92.9±0.0	91.0±0.9	91.2±0.8	95.3±0.5	92.1±2.5	93.5±1.8	92.2±0.6	95.4±0.5	94.4±0.3	94.7±0.8	92.5±0.8	96.5±0.3	94.7±0.4
Mean	**92.9±0.0**	90.3±0.8	88.2±1.1	90.5±1.5	**93.7±1.1**	91.1±1.0	89.2±1.1	92.0±1.5	**94.4±0.8**	91.5±0.9	90.2±1.2	92.6±1.6	**94.7±0.8**

Table 12. Comparison of anomaly classification (AC) performance in terms of class-wise F_1-max on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

D. 详细定量结果

在本节中，我们报告了主文表1和表4中评估指标的详细子集级性能值。具体而言，我们在表10-15中报告MVTec-AD结果，在表16-21中报告VisA结果。

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	99.2±0.0	98.7±0.6	97.4±0.7	99.4±0.4	98.2±0.9	99.5±0.1	98.5±1.0	99.2±0.3	99.3±0.3	99.5±0.2	98.8±0.2	99.2±0.3	99.3±0.4
Cable	86.5±0.0	71.2±3.3	57.7±4.6	88.8±4.2	88.9±1.9	76.2±5.2	62.3±5.9	91.0±2.7	88.4±0.7	83.4±3.1	70.0±6.1	91.0±2.7	90.9±0.9
Capsule	72.9±0.0	70.2±3.0	57.7±7.3	67.8±2.9	72.3±6.8	70.9±6.1	64.3±3.0	72.8±7.0	77.3±8.8	78.9±5.5	65.2±2.5	72.8±7.0	82.3±8.9
Carpet	100.0±0.0	98.1±0.2	96.6±1.0	95.3±0.8	99.8±0.3	98.3±0.4	97.8±0.5	96.6±0.5	99.8±0.3	98.6±0.2	97.9±0.4	96.6±0.5	100.0±0.0
Grid	98.8±0.0	40.0±6.8	54.2±6.7	63.6±10.3	99.5±0.3	41.3±3.6	67.2±4.2	67.7±8.3	99.4±0.2	44.6±6.6	68.1±3.8	67.7±8.3	99.6±0.1
Hazelnut	93.9±0.0	95.8±1.3	88.3±2.6	88.3±2.7	97.5±1.4	96.2±2.1	90.8±0.8	93.2±3.8	98.3±0.7	98.4±1.3	91.9±1.2	93.2±3.8	98.4±0.4
Leather	100.0±0.0	100.0±0.0	97.5±0.7	97.3±0.7	99.9±0.0	100.0±0.0	97.5±0.9	97.9±0.7	99.9±0.0	100.0±0.0	98.5±0.2	97.9±0.7	100.0±0.0
Metal nut	97.1±0.0	71.0±2.2	53.0±3.8	73.4±2.9	98.7±0.8	77.0±7.9	54.8±3.8	77.7±8.5	99.4±0.2	77.8±5.7	60.7±5.2	77.7±8.5	99.5±0.2
Pill	79.1±0.0	86.5±3.1	61.3±3.8	81.9±2.8	91.2±2.1	84.8±0.9	59.1±6.4	82.9±2.9	92.3±0.7	86.7±0.3	54.9±2.7	82.9±2.9	92.8±1.0
Screw	83.3±0.0	46.7±2.5	55.0±2.5	44.4±4.6	86.4±0.9	46.6±2.2	54.0±4.4	49.0±3.8	86.0±2.1	50.5±5.4	50.0±4.1	49.0±3.8	87.9±1.2
Tile	100.0±0.0	99.9±0.1	92.2±2.2	99.0±0.9	99.9±0.0	99.9±0.1	93.3±1.1	98.5±1.0	99.9±0.2	100.0±0.0	93.1±0.6	98.5±1.0	99.9±0.1
Toothbrush	87.5±0.0	71.7±2.6	82.5±1.2	83.3±3.8	92.2±4.9	78.6±3.2	87.6±4.2	85.9±3.5	97.5±1.6	78.8±5.2	89.2±2.5	85.9±3.5	96.7±2.6
Transistor	88.0±0.0	77.2±2.0	73.3±6.0	78.1±6.9	83.4±3.8	81.3±3.7	72.8±6.3	90.0±4.3	85.3±1.7	81.4±2.1	82.4±6.5	90.0±4.3	85.7±2.5
Wood	99.4±0.0	98.8±0.3	96.1±1.2	97.8±0.3	99.9±0.1	99.2±0.4	96.9±0.5	98.3±0.6	99.9±0.1	98.9±0.6	97.0±0.2	98.3±0.6	99.8±0.3
Zipper	91.5±0.0	89.3±1.9	85.8±2.7	92.3±0.5	88.8±5.9	93.3±2.9	86.3±2.6	94.0±2.1	94.0±1.4	95.1±1.3	88.3±2.0	94.0±2.1	94.5±0.5
Mean	**91.8±0.0**	81.0±2.0	76.6±3.1	83.4±3.0	**93.1±2.0**	82.9±2.6	78.9±3.1	86.3±3.3	**94.4±1.3**	84.8±2.5	80.4±2.5	88.8±2.6	**95.2±1.3**

表10. 在MVTec-AD数据集上基于各类别AUROC的异常分类（AC）性能比较。我们报告了每个测量在5个随机种子下的平均值和标准差。

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUPR	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	99.8±0.0	99.6±0.1	99.2±0.2	99.8±0.1	99.4±0.3	99.8±0.0	99.6±0.3	99.8±0.1	99.8±0.1	99.9±0.0	99.7±0.0	99.8±0.1	99.8±0.1
Cable	91.2±0.0	79.6±2.3	64.9±3.8	93.8±2.2	93.2±1.1	84.5±3.1	69.6±6.6	95.1±1.3	92.9±0.6	88.8±1.9	76.1±5.6	97.1±0.7	94.4±0.3
Capsule	91.5±0.0	91.2±0.9	86.9±2.2	89.4±2.0	91.6±2.7	91.6±2.1	88.4±0.8	91.0±2.9	93.3±3.6	94.4±1.9	87.8±0.8	94.9±1.1	95.1±3.3
Carpet	100.0±0.0	99.4±0.0	99.0±0.2	98.7±0.2	99.9±0.1	99.5±0.1	99.4±0.1	99.0±0.1	99.9±0.1	99.6±0.1	99.4±0.1	98.8±0.2	100.0±0.0
Grid	99.6±0.0	66.9±2.1	75.0±3.3	81.1±4.9	99.9±0.1	68.3±2.1	82.5±2.3	84.1±4.0	99.8±0.1	68.8±4.2	83.0±1.8	86.4±4.0	99.9±0.0
Hazelnut	96.9±0.0	97.9±0.6	93.3±1.7	92.9±2.2	98.6±0.7	98.0±1.1	94.1±0.5	96.0±2.0	99.1±0.4	99.1±0.7	94.8±0.6	97.0±1.2	99.1±0.2
Leather	100.0±0.0	100.0±0.0	99.2±0.2	99.1±0.2	100.0±0.0	100.0±0.0	99.2±0.3	99.3±0.2	100.0±0.0	100.0±0.0	99.6±0.1	99.6±0.1	100.0±0.0
Metal nut	99.3±0.0	91.7±0.8	82.0±2.7	91.0±1.1	99.7±0.2	93.7±2.4	82.2±1.4	92.3±4.0	99.9±0.0	94.1±1.8	85.5±1.7	97.0±2.6	99.9±0.1
Pill	95.7±0.0	97.0±0.8	88.3±1.3	96.5±0.6	98.3±0.5	96.5±0.4	87.9±2.6	96.6±0.7	98.6±0.1	97.0±0.2	87.0±1.2	96.9±0.4	98.6±0.2
Screw	93.1±0.0	71.3±1.8	78.1±1.0	71.4±2.3	94.2±0.6	71.0±1.4	77.3±1.3	72.9±3.4	94.1±1.5	73.7±2.4	75.7±2.8	71.8±1.9	94.9±0.8
Tile	100.0±0.0	100.0±0.0	97.2±0.7	99.6±0.3	100.0±0.0	100.0±0.0	97.6±0.4	99.4±0.4	100.0±0.1	100.0±0.0	97.6±0.2	99.6±0.1	100.0±0.0
Toothbrush	95.6±0.0	88.3±0.6	93.7±0.5	93.5±1.4	96.7±2.0	90.8±1.3	95.2±1.6	94.1±1.4	99.0±0.6	91.3±2.6	95.8±0.7	94.8±0.7	98.7±1.1
Transistor	87.1±0.0	76.2±1.7	66.2±7.5	77.7±5.5	79.0±4.0	81.6±3.4	69.0±6.5	89.3±3.9	80.7±2.3	80.3±2.6	77.6±8.4	84.5±9.0	80.7±3.2
Wood	99.8±0.0	99.6±0.1	98.8±0.3	99.3±0.1	100.0±0.0	99.7±0.1	99.0±0.1	99.5±0.2	100.0±0.0	99.7±0.2	99.1±0.0	99.5±0.2	99.9±0.1
Zipper	97.5±0.0	96.9±0.5	95.5±0.9	97.2±0.3	96.8±1.8	98.2±0.8	95.4±1.0	97.8±1.0	98.3±0.4	98.6±0.4	96.2±0.8	99.1±0.7	98.5±0.2
Mean	**96.5±0.0**	90.6±0.8	88.1±1.7	92.2±1.5	**96.5±0.9**	91.7±1.2	89.3±1.7	93.8±1.7	**97.0±0.7**	92.5±1.2	90.5±1.6	94.5±1.5	**97.3±0.6**

表11. 在MVTec-AD数据集上基于各类别AUPR的异常分类（AC）性能对比。我们报告了每个测量结果在5个随机种子下的平均值和标准差。

MVTec-AD (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	97.6±0.0	97.8±0.8	96.3±1.2	98.3±0.6	96.5±1.3	98.7±0.4	97.1±1.1	97.5±0.6	97.7±0.7	98.6±0.3	97.9±0.4	97.9±0.8	97.8±0.6
Cable	84.5±0.0	79.6±2.3	77.2±1.1	85.2±3.6	86.1±1.3	80.4±1.7	78.7±1.2	86.1±2.4	85.2±0.7	83.8±2.5	81.1±1.1	91.3±1.0	87.2±0.6
Capsule	91.4±0.0	92.0±0.6	91.0±0.2	92.0±1.0	91.6±0.7	92.1±0.4	92.1±0.9	93.6±0.6	92.1±0.7	92.7±0.3	92.8±0.9	94.3±0.3	92.5±0.5
Carpet	99.4±0.0	96.5±0.2	95.1±0.5	94.9±0.5	99.2±0.8	96.6±0.3	96.5±0.4	95.3±0.5	99.3±0.7	96.9±0.3	96.6±0.3	94.3±0.8	99.9±0.2
Grid	98.2±0.0	84.5±0.3	84.5±0.3	86.2±1.1	98.9±0.4	84.8±0.3	85.3±0.9	86.9±2.3	99.1±0.0	84.8±0.5	85.0±0.5	87.5±2.0	99.1±0.0
Hazelnut	89.7±0.0	92.4±1.3	87.4±1.6	87.0±1.4	94.7±2.3	93.2±2.8	89.3±1.1	91.0±3.7	95.6±1.6	95.9±2.0	90.0±1.7	92.8±1.2	96.2±1.0
Leather	100.0±0.0	99.9±0.2	96.2±0.9	95.9±0.7	99.5±0.0	100.0±0.0	96.6±1.3	95.7±0.9	99.7±0.2	100.0±0.0	97.9±0.2	97.5±0.7	99.8±0.2
Metal nut	96.3±0.0	90.1±0.6	90.1±0.3	91.4±1.2	97.7±1.0	90.5±1.1	90.0±0.3	91.9±0.9	98.4±0.5	90.6±0.9	90.3±0.4	93.6±1.4	98.5±0.6
Pill	91.6±0.0	93.5±0.4	91.6±0.0	91.9±0.3	93.8±0.7	93.4±0.3	91.7±0.1	92.0±0.3	94.3±0.4	93.6±0.5	91.7±0.1	92.1±0.3	94.1±0.4
Screw	87.4±0.0	85.3±0.0	85.6±0.3	85.8±0.2	88.5±0.3	85.6±0.2	85.5±0.1	85.7±0.2	89.0±0.6	85.8±0.7	85.7±0.3	86.8±0.6	89.6±0.7
Tile	99.4±0.0	99.2±0.5	90.7±2.0	97.5±1.2	98.9±0.2	99.2±0.3	91.5±1.3	96.9±1.5	99.2±0.3	99.4±0.0	91.0±0.9	97.5±0.4	99.2±0.3
Toothbrush	87.9±0.0	85.6±1.3	85.4±1.3	88.9±2.2	94.1±1.9	87.1±1.4	90.1±3.1	90.8±1.6	96.7±1.8	86.5±1.8	90.6±2.1	92.6±2.2	96.8±2.3
Transistor	79.5±0.0	70.3±1.7	66.4±4.7	70.6±7.2	75.1±3.1	73.4±3.4	65.9±3.7	85.3±6.1	75.9±2.4	72.3±2.7	74.8±7.7	78.3±11.5	76.6±2.8
Wood	98.3±0.0	97.0±0.8	94.3±1.1	96.1±0.3	99.4±0.3	97.7±1.0	94.9±0.3	96.6±0.9	99.5±0.4	97.6±0.7	94.8±0.5	96.5±0.9	99.2±0.9
Zipper	92.9±0.0	91.0±0.9	91.2±0.8	95.3±0.5	92.1±2.5	93.5±1.8	92.2±0.6	95.4±0.5	94.4±0.3	94.7±0.8	92.5±0.8	96.5±0.3	94.7±0.4
Mean	**92.9±0.0**	90.3±0.8	88.2±1.1	90.5±1.5	**93.7±1.1**	91.1±1.0	89.2±1.1	92.0±1.5	**94.4±0.8**	91.5±0.9	90.2±1.2	92.6±1.6	**94.7±0.8**

表12. 在MVTec-AD数据集上基于类别级F_1-max的异常分类（AC）性能对比。我们报告了每个测量指标在5个随机种子下的平均值和标准差。

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
pAUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	89.5±0.0	95.3±0.2	96.1±0.5	97.9±0.1	97.5±0.2	95.7±0.2	96.9±0.1	98.1±0.0	97.7±0.1	96.1±0.0	97.1±0.1	98.2±0.0	97.8±0.0
Cable	77.0±0.0	86.4±0.2	88.4±1.2	95.5±0.8	93.8±0.6	87.4±0.3	90.0±0.8	96.4±0.3	94.3±0.4	88.2±0.2	92.1±0.4	97.5±0.3	94.9±0.1
Capsule	86.9±0.0	96.3±0.2	94.5±0.6	95.6±0.4	94.6±0.8	96.7±0.1	95.2±0.5	96.5±0.4	96.4±0.3	97.0±0.2	96.2±0.4	96.8±0.6	96.2±0.5
Carpet	95.4±0.0	98.2±0.0	97.8±0.2	98.4±0.1	99.4±0.0	98.3±0.0	98.2±0.0	98.5±0.1	99.3±0.0	98.4±0.0	98.4±0.0	98.6±0.1	99.3±0.0
Grid	82.2±0.0	80.7±1.3	70.2±2.8	58.8±4.9	96.8±1.0	83.5±1.0	70.8±2.0	62.6±3.2	97.7±0.8	87.2±1.1	77.0±1.8	69.4±1.3	98.0±0.2
Hazelnut	94.3±0.0	97.2±0.1	95.4±0.7	95.8±0.6	98.5±0.2	97.6±0.1	96.8±0.3	96.3±0.6	98.7±0.1	97.7±0.1	97.2±0.2	97.6±0.1	98.8±0.0
Leather	96.7±0.0	99.1±0.0	98.5±0.1	98.8±0.2	99.3±0.0	99.1±0.0	98.7±0.1	99.0±0.1	99.3±0.0	99.1±0.0	98.8±0.0	99.1±0.0	99.3±0.0
Metal nut	61.0±0.0	83.8±0.7	74.6±1.1	89.3±1.4	90.0±0.6	85.8±1.1	80.3±2.1	94.6±1.4	91.4±0.4	87.1±0.7	82.7±3.9	95.9±1.8	92.9±0.4
Pill	80.0±0.0	89.4±0.4	84.8±1.0	93.1±1.1	96.4±0.3	89.9±0.2	87.3±0.7	94.2±0.3	97.0±0.2	90.7±0.2	88.9±0.5	94.8±0.4	97.1±0.0
Screw	89.6±0.0	94.8±0.2	83.3±0.7	89.6±0.5	94.5±0.4	95.6±0.4	89.8±0.8	90.0±0.7	95.2±0.3	96.4±0.4	90.8±0.2	91.3±1.0	96.0±0.5
Tile	77.6±0.0	91.7±0.3	84.1±1.1	94.1±0.5	96.3±0.2	92.0±0.1	87.7±0.2	94.4±0.2	96.5±0.1	92.2±0.1	88.9±0.3	94.6±0.1	96.6±0.1
Toothbrush	86.9±0.0	94.6±0.6	97.3±0.3	97.3±0.4	97.8±0.1	96.2±0.3	97.7±0.3	97.5±0.2	98.1±0.1	97.0±0.6	98.4±0.2	98.4±0.4	98.4±0.5
Transistor	74.7±0.0	71.4±1.3	90.2±2.8	84.9±2.7	85.0±1.8	72.8±0.9	92.3±2.1	89.6±0.9	88.3±1.0	73.4±0.7	94.0±2.7	90.7±1.4	88.5±1.2
Wood	93.4±0.0	93.4±0.1	90.7±0.4	92.7±0.9	94.6±1.0	93.8±0.1	91.9±0.1	93.2±0.7	95.3±0.4	93.9±0.1	92.2±0.1	93.5±0.3	95.4±0.2
Zipper	91.6±0.0	94.9±0.3	93.9±0.8	97.4±0.4	93.9±0.8	95.8±0.2	95.4±0.3	98.0±0.1	94.1±0.7	96.2±0.1	96.1±0.2	98.1±0.1	94.2±0.4
Mean	**85.1**±**0.0**	91.2±0.4	89.3±0.9	92.0±1.0	**95.2**±**0.5**	92.0±0.3	91.3±0.7	93.3±0.6	**96.0**±**0.3**	92.7±0.3	92.6±0.7	94.3±0.5	**96.2**±**0.3**

Table 13. Comparison of anomaly segmentation (AS) performance in terms of class-wise pixel-AUROC on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
PRO	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	76.4±0.0	91.1±0.4	89.8±0.8	93.5±0.3	91.2±0.4	91.8±0.5	91.7±0.2	93.9±0.3	91.8±0.3	92.5±0.1	92.2±0.2	94.0±0.2	91.6±0.2
Cable	42.9±0.0	63.5±0.7	59.1±3.2	84.7±1.0	72.5±2.3	66.7±0.9	66.5±2.8	88.5±0.9	74.7±2.3	69.5±0.4	74.2±1.8	91.7±0.6	77.0±1.1
Capsule	62.1±0.0	92.7±0.4	80.0±2.0	83.9±0.9	85.6±2.7	93.4±0.3	82.3±2.1	86.6±1.0	90.6±0.6	94.1±0.6	85.7±1.3	87.8±1.9	90.1±1.5
Carpet	84.1±0.0	96.1±0.0	92.9±0.3	93.3±0.3	97.4±0.4	96.2±0.0	93.9±0.2	93.7±0.4	97.3±0.3	96.3±0.0	94.4±0.2	93.9±0.4	97.0±0.2
Grid	57.0±0.0	67.7±1.9	41.2±4.6	21.7±9.5	90.5±2.7	72.1±1.5	45.1±3.6	23.7±3.8	92.8±2.5	78.0±1.5	55.5±3.4	30.4±4.6	93.6±0.6
Hazelnut	81.6±0.0	94.9±0.3	85.7±1.9	88.3±1.3	93.7±0.9	95.6±0.2	89.4±0.9	89.8±1.3	94.2±0.3	95.6±0.1	90.4±0.7	92.0±0.3	94.2±0.3
Leather	91.1±0.0	98.7±0.0	95.6±0.2	95.2±1.0	98.6±0.0	98.8±0.0	96.2±0.2	95.9±0.3	98.3±0.4	98.8±0.0	96.3±0.1	96.4±0.1	98.0±0.4
Metal nut	31.8±0.0	73.4±1.1	38.1±1.6	66.7±2.9	84.7±1.1	78.1±1.8	48.2±5.0	79.6±4.2	86.7±0.8	81.2±1.4	54.0±8.8	83.8±5.5	89.4±0.1
Pill	65.0±0.0	92.8±0.3	78.9±0.6	89.5±1.6	93.5±0.2	93.3±0.2	84.3±0.4	91.6±0.5	94.5±0.2	93.9±0.2	86.6±0.4	92.5±0.4	94.6±0.3
Screw	68.5±0.0	85.0±0.8	51.6±1.7	68.1±1.3	82.3±1.1	87.2±1.2	69.5±2.1	69.0±2.1	84.1±0.5	89.5±1.3	72.3±0.8	72.4±3.1	86.3±1.8
Tile	51.2±0.0	84.2±0.4	66.7±1.5	82.5±1.1	89.4±0.4	84.6±0.2	71.9±0.5	82.5±0.5	89.6±0.4	84.9±0.1	73.6±0.9	83.0±0.1	89.9±0.3
Toothbrush	67.7±0.0	83.5±1.3	82.1±1.5	79.0±2.4	85.3±1.0	87.4±1.1	83.3±2.6	81.0±0.7	84.7±1.4	89.0±1.1	87.1±1.7	85.5±3.0	86.0±3.3
Transistor	43.4±0.0	55.3±2.0	70.3±7.0	70.9±4.6	65.0±1.8	57.6±1.4	76.5±5.5	78.8±1.5	68.6±1.1	58.5±0.7	82.2±7.4	79.5±2.8	69.0±1.1
Wood	74.1±0.0	92.9±0.1	86.5±0.6	87.1±1.0	91.0±0.6	93.1±0.1	88.0±0.2	86.8±1.4	91.8±0.6	93.2±0.1	88.4±0.2	87.7±0.4	91.7±0.3
Zipper	71.7±0.0	86.8±0.6	81.7±2.0	91.2±1.1	86.0±1.7	89.0±0.4	85.6±0.7	92.8±0.4	86.4±1.6	90.1±0.2	87.2±0.8	93.4±0.2	86.9±0.7
Mean	**64.6**±**0.0**	83.9±0.7	73.3±2.0	79.7±2.0	**87.1**±**1.2**	85.7±0.7	78.2±1.8	82.3±1.3	**88.4**±**0.9**	87.0±0.5	81.3±1.9	84.3±1.6	**89.0**±**0.8**

Table 14. Comparison of anomaly segmentation (AS) performance in terms of class-wise PRO on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	58.1±0.0	61.5±0.3	68.2±1.9	74.8±0.4	72.8±0.8	62.7±0.4	70.7±0.4	75.1±0.1	73.2±0.9	64.3±0.3	71.4±0.4	75.0±0.2	73.3±0.6
Cable	19.7±0.0	25.9±1.2	27.4±1.8	59.8±1.4	49.4±3.3	28.5±0.8	29.5±1.6	62.2±1.0	51.2±1.3	30.2±0.4	34.5±1.1	65.5±1.1	54.7±1.1
Capsule	21.7±0.0	37.5±3.5	27.1±2.8	32.3±2.1	29.7±7.8	39.6±3.0	33.1±2.6	37.9±4.5	43.5±1.4	40.8±3.4	37.0±2.0	39.0±6.3	40.7±4.9
Carpet	49.7±0.0	67.1±0.2	62.4±0.5	67.3±0.4	73.3±1.5	67.6±0.2	62.6±0.2	67.0±0.7	72.9±1.3	68.1±0.2	62.9±0.2	67.4±0.3	72.0±0.7
Grid	18.6±0.0	17.0±1.0	9.4±2.1	5.5±2.2	50.7±4.5	18.9±1.1	13.1±1.5	5.2±1.2	53.4±3.8	23.1±1.6	18.0±1.9	10.0±5.3	52.7±1.5
Hazelnut	37.6±0.0	62.8±0.8	47.7±3.3	50.1±3.9	68.9±2.6	65.1±0.3	57.1±0.7	53.6±3.7	70.5±1.7	65.2±0.6	58.0±1.3	60.8±1.5	71.0±0.3
Leather	39.7±0.0	55.6±0.1	52.3±0.8	58.6±0.4	58.0±0.7	55.8±0.6	52.8±0.2	58.8±0.3	57.5±0.6	55.5±0.1	52.5±0.2	58.8±0.3	56.3±1.0
Metal nut	32.4±0.0	46.4±1.1	38.2±0.9	55.1±2.6	59.4±1.7	48.7±1.4	44.5±2.0	70.4±4.8	62.7±1.5	50.4±0.9	47.5±4.4	74.8±6.7	67.4±1.6
Pill	17.6±0.0	29.6±0.8	25.3±0.6	54.5±4.0	64.7±1.8	31.2±0.4	28.8±1.2	59.4±1.7	67.8±0.5	33.0±0.5	32.7±1.1	61.7±1.6	67.9±0.4
Screw	13.5±0.0	11.6±1.0	3.5±0.1	6.4±0.4	22.2±2.8	14.7±2.3	5.9±0.3	6.5±0.4	22.4±2.8	20.1±5.2	6.4±0.2	7.4±0.5	30.1±4.3
Tile	32.6±0.0	57.3±0.5	42.2±1.4	64.2±1.4	71.2±0.4	58.0±0.2	47.3±0.4	64.4±0.8	71.9±0.6	58.4±0.2	48.8±0.6	65.0±0.1	72.2±0.6
Toothbrush	17.1±0.0	40.0±2.5	59.5±3.8	63.4±3.2	62.7±3.6	46.9±1.8	62.4±2.7	61.5±2.4	65.8±2.2	51.0±3.7	65.0±1.4	64.9±0.5	69.4±4.6
Transistor	30.5±0.0	21.4±1.9	41.6±8.1	48.2±5.6	39.1±3.5	23.2±1.4	47.5±7.7	54.6±1.5	45.6±2.3	23.8±0.8	54.0±10.5	55.7±2.6	46.6±2.2
Wood	51.5±0.0	56.4±0.3	46.6±0.7	52.9±1.1	65.2±1.4	57.0±0.1	47.2±0.2	52.9±1.8	65.8±0.6	57.1±0.4	47.7±0.3	53.3±0.7	65.1±0.5
Zipper	34.4±0.0	45.1±0.3	51.3±2.5	62.5±2.4	50.6±3.9	48.7±0.5	53.1±1.1	65.3±0.7	50.9±4.5	51.2±0.6	55.2±1.7	65.1±0.5	52.8±2.7
Mean	**31.7**±**0.0**	42.4±1.0	40.2±2.1	50.4±2.1	**55.9**±**2.7**	44.5±1.0	43.7±1.5	53.0±1.7	**58.4**±**1.7**	46.2±1.3	46.1±1.8	55.0±1.9	**59.5**±**1.8**

Table 15. Comparison of anomaly segmentation (AS) performance in terms of class-wise F_1-max on MVTec-AD. We report the mean and standard deviation over 5 random seeds for each measurement.

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
pAUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	89.5±0.0	95.3±0.2	96.1±0.5	97.9±0.1	97.5±0.2	95.7±0.2	96.9±0.1	98.1±0.0	97.7±0.1	96.1±0.0	97.1±0.1	98.2±0.0	97.8±0.0
Cable	77.0±0.0	86.4±0.2	88.4±1.2	95.5±0.8	93.8±0.6	87.4±0.3	90.0±0.8	96.4±0.3	94.3±0.4	88.2±0.2	92.1±0.4	97.5±0.3	94.9±0.1
Capsule	86.9±0.0	96.3±0.2	94.5±0.6	95.6±0.4	94.6±0.8	96.7±0.1	95.2±0.5	96.5±0.4	96.4±0.3	97.0±0.2	96.2±0.4	96.8±0.6	96.2±0.5
Carpet	95.4±0.0	98.2±0.0	97.8±0.2	98.4±0.1	99.4±0.0	98.3±0.0	98.2±0.0	98.5±0.1	99.3±0.0	98.4±0.0	98.4±0.0	98.6±0.1	99.3±0.0
Grid	82.2±0.0	80.7±1.3	70.2±2.8	58.8±4.9	96.8±1.0	83.5±1.0	70.8±2.0	62.6±3.2	97.7±0.8	87.2±1.1	77.0±1.8	69.4±1.3	98.0±0.2
Hazelnut	94.3±0.0	97.2±0.1	95.4±0.7	95.8±0.6	98.5±0.2	97.6±0.1	96.8±0.3	96.3±0.6	98.7±0.1	97.7±0.1	97.2±0.2	97.6±0.1	98.8±0.0
Leather	96.7±0.0	99.1±0.0	98.5±0.1	98.8±0.2	99.3±0.0	99.1±0.0	98.7±0.1	99.0±0.1	99.3±0.0	99.1±0.0	98.8±0.0	99.1±0.0	99.3±0.0
Metal nut	61.0±0.0	83.8±0.7	74.6±1.1	89.3±1.4	90.0±0.6	85.8±1.1	80.3±2.1	94.6±1.4	91.4±0.4	87.1±0.7	82.7±3.9	95.9±1.8	92.9±0.4
Pill	80.0±0.0	89.4±0.4	84.8±1.0	93.1±1.1	96.4±0.3	89.9±0.2	87.3±0.7	94.2±0.3	97.0±0.2	90.7±0.2	88.9±0.5	94.8±0.4	97.1±0.0
Screw	89.6±0.0	94.8±0.2	83.3±0.7	89.6±0.5	94.5±0.4	95.6±0.4	89.8±0.8	90.0±0.7	95.2±0.3	96.4±0.4	90.8±0.2	91.3±1.0	96.0±0.5
Tile	77.6±0.0	91.7±0.3	84.1±1.1	94.1±0.5	96.3±0.2	92.0±0.1	87.7±0.2	94.4±0.2	96.5±0.1	92.2±0.1	88.9±0.3	94.6±0.1	96.6±0.1
Toothbrush	86.9±0.0	94.6±0.6	97.3±0.3	97.3±0.4	97.8±0.1	96.2±0.3	97.7±0.3	97.5±0.2	98.1±0.1	97.0±0.6	98.4±0.2	98.4±0.4	98.4±0.5
Transistor	74.7±0.0	71.4±1.3	90.2±2.8	84.9±2.7	85.0±1.8	72.8±0.9	92.3±2.1	89.6±0.9	88.3±1.0	73.4±0.7	94.0±2.7	90.7±1.4	88.5±1.2
Wood	93.4±0.0	93.4±0.1	90.7±0.4	92.7±0.9	94.6±1.0	93.8±0.1	91.9±0.1	93.2±0.7	95.3±0.4	93.9±0.1	92.2±0.1	93.5±0.3	95.4±0.2
Zipper	91.6±0.0	94.9±0.3	93.9±0.8	97.4±0.4	93.9±0.8	95.8±0.2	95.4±0.3	98.0±0.1	94.1±0.7	96.2±0.1	96.1±0.2	98.1±0.1	94.2±0.4
Mean	**85.1±0.0**	91.2±0.4	89.3±0.9	92.0±1.0	**95.2±0.5**	92.0±0.3	91.3±0.7	93.3±0.6	**96.0±0.3**	92.7±0.3	92.6±0.7	94.3±0.5	**96.2±0.3**

表13. MVTec-AD数据集上各类别像素-AUROC的异常分割（AS）性能对比。各项测量结果均基于5个随机种子计算均值与标准差。

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
PRO	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	76.4±0.0	91.1±0.4	89.8±0.8	93.5±0.3	91.2±0.4	91.8±0.5	91.7±0.2	93.9±0.3	91.8±0.3	92.5±0.1	92.2±0.2	94.0±0.2	91.6±0.2
Cable	42.9±0.0	63.5±0.7	59.1±3.2	84.7±1.0	72.5±2.3	66.7±0.9	66.5±2.8	88.5±0.9	74.7±2.3	69.5±0.4	74.2±1.8	91.7±0.6	77.0±1.1
Capsule	62.1±0.0	92.7±0.4	80.0±2.0	83.9±0.9	85.6±2.7	93.4±0.3	82.3±2.1	86.6±1.0	90.6±0.6	94.1±0.6	85.7±1.3	87.8±1.9	90.1±1.5
Carpet	84.1±0.0	96.1±0.0	92.9±0.3	93.3±0.3	97.4±0.4	96.2±0.0	93.9±0.2	93.7±0.4	97.3±0.3	96.3±0.0	94.4±0.2	93.9±0.4	97.0±0.2
Grid	57.0±0.0	67.7±1.9	41.2±4.6	21.7±9.5	90.5±2.7	72.1±1.5	45.1±3.6	23.7±3.8	92.8±2.5	78.0±1.5	55.5±3.4	30.4±4.6	93.6±0.6
Hazelnut	81.6±0.0	94.9±0.3	85.7±1.9	88.3±1.3	93.7±0.9	95.6±0.2	89.4±0.9	89.8±1.3	94.2±0.3	95.6±0.1	90.4±0.7	92.0±0.3	94.2±0.3
Leather	91.1±0.0	98.7±0.0	95.6±0.2	95.2±1.0	98.6±0.0	98.8±0.0	96.2±0.2	95.9±0.3	98.3±0.4	98.8±0.0	96.3±0.1	96.4±0.1	98.0±0.4
Metal nut	31.8±0.0	73.4±1.1	38.1±1.6	66.7±2.9	84.7±1.1	78.1±1.8	48.2±5.0	79.6±4.2	86.7±0.8	81.2±1.4	54.0±8.8	83.8±5.5	89.4±0.1
Pill	65.0±0.0	92.8±0.3	78.9±0.6	89.5±1.6	93.5±0.2	93.3±0.2	84.3±0.4	91.6±0.5	94.5±0.2	93.9±0.2	86.6±0.4	92.5±0.4	94.6±0.3
Screw	68.5±0.0	85.0±0.8	51.6±1.7	68.1±1.3	82.3±1.1	87.2±1.2	69.5±2.1	69.0±2.1	84.1±0.5	89.5±1.3	72.3±0.8	72.4±3.1	86.3±1.8
Tile	51.2±0.0	84.2±0.4	66.7±1.5	82.5±1.1	89.4±0.4	84.6±0.2	71.9±0.5	82.5±0.5	89.6±0.4	84.9±0.1	73.6±0.9	83.0±0.1	89.9±0.3
Toothbrush	67.7±0.0	83.5±1.3	82.1±1.5	79.0±2.4	85.3±1.0	87.4±1.1	83.3±2.6	81.0±0.7	84.7±1.4	89.0±1.1	87.1±1.7	85.5±3.0	86.0±3.3
Transistor	43.4±0.0	55.3±2.0	70.3±7.0	70.9±4.6	65.0±1.8	57.6±1.4	76.5±5.5	78.8±1.5	68.6±1.1	58.5±0.7	82.2±7.4	79.5±2.8	69.0±1.1
Wood	74.1±0.0	92.9±0.1	86.5±0.6	87.1±1.0	91.0±0.6	93.1±0.1	88.0±0.2	86.8±1.4	91.8±0.6	93.2±0.1	88.4±0.2	87.7±0.4	91.7±0.3
Zipper	71.7±0.0	86.8±0.6	81.7±2.0	91.2±1.1	86.0±1.7	89.0±0.4	85.6±0.7	92.8±0.4	86.4±1.6	90.1±0.2	87.2±0.8	93.4±0.2	86.9±0.7
Mean	**64.6±0.0**	83.9±0.7	73.3±2.0	79.7±2.0	**87.1±1.2**	85.7±0.7	78.2±1.8	82.3±1.3	**88.4±0.9**	87.0±0.5	81.3±1.9	84.3±1.6	**89.0±0.8**

表14. 在MVTec-AD数据集上基于类别PRO指标的异常分割（AS）性能对比。我们报告了每个测量在5个随机种子下的平均值和标准差。

MVTec-AD (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Bottle	58.1±0.0	61.5±0.3	68.2±1.9	74.8±0.4	72.8±0.8	62.7±0.4	70.7±0.4	75.1±0.1	73.2±0.9	64.3±0.3	71.4±0.4	75.0±0.2	73.3±0.6
Cable	19.7±0.0	25.9±1.2	27.4±1.8	59.8±1.4	49.4±3.3	28.5±0.8	29.5±1.6	62.2±1.0	51.2±1.3	30.2±0.4	34.5±1.1	65.5±1.1	54.7±1.1
Capsule	21.7±0.0	37.5±3.5	27.1±2.8	32.3±2.1	29.7±7.8	39.6±3.0	33.1±2.6	37.9±4.5	43.5±1.4	40.8±3.4	37.0±2.0	39.0±6.3	40.7±4.9
Carpet	49.7±0.0	67.1±0.2	62.4±0.5	67.3±0.4	73.3±1.5	67.6±0.2	62.6±0.2	67.0±0.7	72.9±1.3	68.1±0.2	62.9±0.2	67.4±0.3	72.0±0.7
Grid	18.6±0.0	17.0±1.0	9.4±2.1	5.5±2.2	50.7±4.5	18.9±1.1	13.1±1.5	5.2±1.2	53.4±3.8	23.1±1.6	18.0±1.9	10.0±5.3	52.7±1.5
Hazelnut	37.6±0.0	62.8±0.8	47.7±3.3	50.1±3.9	68.9±2.6	65.1±0.3	57.1±0.7	53.6±3.7	70.5±1.7	65.2±0.6	58.0±1.3	60.8±1.5	71.0±0.3
Leather	39.7±0.0	55.6±0.1	52.3±0.8	58.6±0.4	58.0±0.7	55.8±0.6	52.8±0.2	58.8±0.3	57.5±0.6	55.5±0.1	52.5±0.2	58.8±0.3	56.3±1.0
Metal nut	32.4±0.0	46.4±1.1	38.2±0.9	55.1±2.6	59.4±1.7	48.7±1.4	44.5±2.0	70.4±4.8	62.7±1.5	50.4±0.9	47.5±4.4	74.8±6.7	67.4±1.6
Pill	17.6±0.0	29.6±0.8	25.3±0.6	54.5±4.0	64.7±1.8	31.2±0.4	28.8±1.2	59.4±1.7	67.8±0.5	33.0±0.5	32.7±1.1	61.7±1.6	67.9±0.4
Screw	13.5±0.0	11.6±1.0	3.5±0.1	6.4±0.4	22.2±2.8	14.7±2.3	5.9±0.3	6.5±0.4	22.4±2.8	20.1±5.2	6.4±0.2	7.4±0.5	30.1±4.3
Tile	32.6±0.0	57.3±0.5	42.2±1.4	64.2±1.4	71.2±0.4	58.0±0.2	47.3±0.4	64.4±0.8	71.9±0.6	58.4±0.2	48.8±0.6	65.0±0.1	72.2±0.6
Toothbrush	17.1±0.0	40.0±2.5	59.5±3.8	63.4±3.2	62.7±3.6	46.9±1.8	62.4±2.7	61.5±2.4	65.8±2.2	51.0±3.7	65.0±1.4	64.9±0.5	69.4±4.6
Transistor	30.5±0.0	21.4±1.9	41.6±8.1	48.2±5.6	39.1±3.5	23.2±1.4	47.5±7.7	54.6±1.5	45.6±2.3	23.8±0.8	54.0±10.5	55.7±2.6	46.6±2.2
Wood	51.5±0.0	56.4±0.3	46.6±0.7	52.9±1.1	65.2±1.4	57.0±0.1	47.2±0.2	52.9±1.8	65.8±0.6	57.1±0.4	47.7±0.3	53.3±0.7	65.1±0.5
Zipper	34.4±0.0	45.1±0.3	51.3±2.5	62.5±2.4	50.6±3.9	48.7±0.5	53.1±1.1	65.3±0.7	50.9±4.5	51.2±0.6	55.2±1.7	65.1±0.5	52.8±2.7
Mean	**31.7±0.0**	42.4±1.0	40.2±2.1	50.4±2.1	**55.9±2.7**	44.5±1.0	43.7±1.5	53.0±1.7	**58.4±1.7**	46.2±1.3	46.1±1.8	55.0±1.9	**59.5±1.8**

表15. MVTec-AD数据集上各类别F_1-max异常分割性能对比。我们报告了每个测量指标在5个随机种子下的平均值和标准差。

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	95.4±0.0	86.1±5.6	70.8±4.1	85.1±1.4	93.4±1.4	91.3±3.3	75.8±2.1	85.3±1.5	94.8±1.0	92.8±2.1	77.5±1.6	87.8±0.8	95.1±0.3
Capsules	85.0±0.0	73.3±7.5	51.0±7.8	60.0±7.6	85.0±3.1	71.7±11.2	51.7±4.6	57.8±5.4	84.9±0.8	73.4±7.1	52.7±3.4	63.4±5.4	86.8±1.7
Cashew	92.1±0.0	95.9±1.1	62.3±9.9	89.5±4.4	94.0±0.4	97.3±1.4	74.6±3.6	93.6±0.6	94.3±0.5	96.4±1.3	77.7±3.2	93.0±1.5	95.2±0.8
Chewinggum	96.5±0.0	92.1±2.0	69.9±4.9	97.3±0.3	97.6±0.8	93.4±1.0	82.7±2.1	97.8±0.6	97.3±0.8	93.5±1.4	83.5±3.7	98.3±0.3	97.7±0.3
Fryum	80.3±0.0	81.1±4.0	58.3±5.9	75.0±4.8	88.5±1.9	90.5±3.9	69.2±9.0	83.4±2.4	90.5±0.4	92.9±1.6	71.2±5.9	88.6±1.3	90.8±0.5
Macaroni1	76.2±0.0	66.0±10.5	62.1±4.6	68.0±3.4	82.9±1.5	69.1±8.2	62.2±5.0	75.6±4.6	83.3±1.9	65.8±1.2	65.9±3.9	82.9±2.7	85.2±0.9
Macaroni2	63.7±0.0	55.8±6.1	47.5±5.9	55.6±4.6	70.2±0.9	58.3±4.4	50.8±2.9	57.3±5.6	71.8±2.0	56.7±3.2	55.0±2.9	61.7±1.8	70.9±2.2
PCB1	73.6±0.0	87.2±2.3	76.2±1.2	78.9±1.1	75.6±23.0	86.7±1.1	62.4±10.8	71.5±20.0	76.7±5.2	83.4±8.5	82.6±1.5	84.7±6.7	88.3±1.7
PCB2	51.2±0.0	73.5±3.7	61.2±2.0	81.5±0.8	62.2±3.9	70.3±8.1	66.8±2.0	84.3±1.7	62.6±3.7	71.7±7.0	73.5±2.4	84.3±1.0	67.5±2.6
PCB3	73.4±0.0	72.2±1.0	51.4±12.2	82.7±2.3	74.1±1.1	75.8±5.7	67.3±3.8	84.8±1.2	78.8±1.9	79.0±4.1	65.9±1.9	87.0±1.1	83.3±1.7
PCB4	79.6±0.0	93.4±1.3	76.1±3.6	93.9±2.8	85.2±8.9	86.1±8.2	69.3±13.7	94.3±3.2	82.3±9.9	95.4±2.3	85.4±2.0	95.6±1.6	87.6±8.0
Pipe fryum	69.7±0.0	77.9±3.2	66.7±2.2	90.7±1.7	97.2±1.1	78.1±3.0	75.3±1.8	93.5±1.3	98.0±0.6	79.3±0.9	82.9±2.2	96.4±0.7	98.5±0.4
Mean	**78.1±0.0**	79.5±4.0	62.8±5.4	79.9±2.9	**83.8±4.0**	80.7±5.0	67.4±5.1	81.6±4.0	**84.6±2.4**	81.7±3.4	72.8±2.9	85.3±2.1	**87.3±1.8**

Table 16. Comparison of anomaly classification (AC) performance in terms of class-wise AUROC on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUPR	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	95.8±0.0	86.5±4.3	69.2±3.9	86.6±2.3	93.6±1.5	90.7±3.2	72.8±1.0	86.8±1.7	95.1±1.1	92.6±1.9	72.5±1.1	88.9±1.1	95.3±0.4
Capsules	90.9±0.0	79.4±4.9	63.4±5.7	72.3±5.3	89.9±2.5	79.9±5.8	63.4±2.0	73.6±4.7	88.9±0.7	81.1±4.5	63.0±2.3	78.4±3.1	91.5±1.4
Cashew	96.4±0.0	97.9±0.4	78.2±5.7	94.6±2.0	97.2±0.2	98.6±0.6	86.1±2.2	96.9±0.3	97.3±0.2	98.3±0.6	88.4±2.0	96.5±0.7	97.7±0.4
Chewinggum	98.6±0.0	96.4±0.9	79.8±3.6	98.9±0.1	99.0±0.3	97.1±0.4	89.5±1.9	99.1±0.2	98.9±0.3	97.1±0.6	88.5±3.2	99.3±0.1	99.0±0.1
Fryum	90.1±0.0	89.8±1.8	74.5±2.9	87.6±2.4	94.7±1.0	94.5±2.3	81.0±5.4	92.1±1.3	95.8±0.2	95.8±1.0	81.5±3.0	95.0±0.6	96.0±0.3
Macaroni1	75.8±0.0	61.9±11.2	60.4±2.9	67.8±3.4	84.9±1.2	64.5±9.5	63.1±4.3	74.9±5.2	84.7±1.5	60.2±2.7	64.9±2.1	82.1±3.5	86.5±0.6
Macaroni2	60.3±0.0	52.7±4.2	51.7±5.0	54.9±3.2	68.4±1.8	55.9±3.1	52.7±1.5	57.2±2.6	70.4±1.8	51.9±2.3	54.9±2.5	60.2±3.0	69.6±2.8
PCB1	78.4±0.0	84.9±3.7	68.6±2.4	72.1±2.5	76.5±19.0	83.8±2.1	60.4±7.7	72.6±16.4	78.3±4.3	83.2±7.2	77.4±2.9	81.0±9.2	87.7±1.7
PCB2	49.2±0.0	74.9±2.9	63.3±1.2	84.4±0.4	64.9±3.3	71.7±6.6	68.9±2.6	86.6±1.1	65.8±4.0	74.2±5.0	75.0±1.7	86.2±1.0	71.3±3.4
PCB3	76.5±0.0	75.5±2.1	52.3±10.8	84.6±1.5	73.5±1.6	78.3±5.2	65.2±3.8	86.1±0.5	80.9±1.6	81.0±3.6	64.5±2.4	88.3±1.1	84.8±1.8
PCB4	77.7±0.0	92.9±1.6	74.7±2.6	92.8±3.1	78.5±15.5	81.9±11.2	67.6±11.9	93.2±3.4	72.5±16.2	94.8±2.9	84.0±2.0	94.9±1.2	85.6±8.9
Pipe fryum	82.3±0.0	88.3±2.0	79.2±1.5	95.4±0.6	98.6±0.5	88.1±1.7	84.5±1.7	96.8±0.7	99.0±0.3	88.8±1.0	89.8±1.7	98.3±0.3	99.2±0.2
Mean	**81.2±0.0**	82.0±3.3	68.3±4.0	82.8±2.3	**85.1±4.0**	82.3±4.3	71.6±3.8	84.8±3.2	**85.8±2.7**	83.4±2.7	75.6±2.2	87.5±2.1	**88.8±1.8**

Table 17. Comparison of anomaly classification (AC) performance in terms of class-wise AUPR on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	89.4±0.0	80.4±6.1	72.0±2.0	79.5±0.6	87.8±1.2	85.5±2.9	74.8±2.9	78.7±0.8	89.1±1.3	86.9±2.4	76.7±1.3	80.5±0.9	88.9±1.0
Capsules	83.9±0.0	81.0±2.4	77.5±0.8	77.9±1.5	84.9±2.0	80.4±3.9	77.2±0.4	77.2±0.3	85.4±0.6	79.5±1.8	77.2±0.3	77.3±0.6	86.0±0.9
Cashew	88.4±0.0	94.8±1.8	80.8±0.7	89.6±3.6	90.7±0.7	95.5±2.2	82.4±1.2	92.3±0.5	90.9±0.7	95.7±0.9	82.5±1.2	91.1±2.1	91.6±1.3
Chewinggum	94.8±0.0	89.7±2.2	83.8±2.0	95.9±0.8	95.6±0.9	90.5±1.2	86.7±0.8	97.0±0.5	95.4±0.6	91.3±1.5	87.9±0.8	97.4±0.6	95.7±0.5
Fryum	82.7±0.0	85.3±2.1	80.6±0.8	82.9±1.7	87.2±1.4	90.9±1.8	82.7±2.1	84.7±1.4	88.4±0.6	91.9±1.7	82.9±1.8	86.6±0.6	88.9±0.8
Macaroni1	74.2±0.0	71.9±2.0	69.2±2.3	70.4±1.9	76.2±1.4	72.8±3.1	68.8±1.8	74.3±2.1	76.7±2.0	70.8±1.2	70.0±1.6	78.9±1.4	78.2±1.2
Macaroni2	69.8±0.0	68.1±0.8	67.1±0.2	67.6±0.7	72.3±1.1	68.2±1.2	67.1±0.4	67.6±1.2	73.9±0.9	67.9±0.6	68.4±1.0	68.8±0.8	73.1±1.6
PCB1	71.0±0.0	85.5±0.2	80.3±0.8	84.5±0.4	81.3±6.6	85.8±0.2	71.4±5.4	78.3±6.7	73.2±3.7	81.2±6.4	83.1±0.6	85.6±1.8	83.1±2.2
PCB2	67.1±0.0	70.9±1.9	68.6±1.4	75.9±0.8	67.2±0.3	71.5±2.7	69.2±0.6	78.1±2.1	67.3±0.3	71.1±3.2	72.0±2.3	79.2±1.9	67.7±0.6
PCB3	71.0±0.0	70.2±1.4	67.7±0.9	76.7±2.5	73.5±1.5	73.3±3.6	69.9±1.1	78.9±1.0	73.9±1.3	75.5±3.3	69.0±0.7	80.7±0.5	77.0±1.4
PCB4	74.9±0.0	87.5±1.7	74.5±2.2	90.4±2.7	86.1±2.1	83.1±5.9	74.6±3.9	91.3±4.1	86.8±3.8	90.6±2.1	81.0±1.6	92.2±3.4	84.6±7.0
Pipe fryum	80.7±0.0	82.7±0.7	81.2±0.5	89.2±2.4	94.4±0.7	82.7±1.4	83.1±0.6	91.4±1.1	95.4±0.8	82.8±0.6	85.4±0.8	93.9±1.2	95.6±0.7
Mean	**79.0±0.0**	80.7±1.9	75.3±1.2	81.7±1.6	**83.1±1.7**	81.7±2.5	75.7±1.8	82.5±1.8	**83.0±1.4**	82.1±2.1	78.0±1.2	84.3±1.3	**84.2±1.6**

Table 18. Comparison of anomaly classification (AC) performance in terms of class-wise F_1-max on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	95.4±0.0	86.1±5.6	70.8±4.1	85.1±1.4	93.4±1.4	91.3±3.3	75.8±2.1	85.3±1.5	94.8±1.0	92.8±2.1	77.5±1.6	87.8±0.8	95.1±0.3
Capsules	85.0±0.0	73.3±7.5	51.0±7.8	60.0±7.6	85.0±3.1	71.7±11.2	51.7±4.6	57.8±5.4	84.9±0.8	73.4±7.1	52.7±3.4	63.4±5.4	86.8±1.7
Cashew	92.1±0.0	95.9±1.1	62.3±9.9	89.5±4.4	94.0±0.4	97.3±1.4	74.6±3.6	93.6±0.6	94.3±0.5	96.4±1.3	77.7±3.2	93.0±1.5	95.2±0.8
Chewinggum	96.5±0.0	92.1±2.0	69.9±4.9	97.3±0.3	97.6±0.8	93.4±1.0	82.7±2.1	97.8±0.6	97.3±0.8	93.5±1.4	83.5±3.7	98.3±0.3	97.7±0.3
Fryum	80.3±0.0	81.1±4.0	58.3±5.9	75.0±4.8	88.5±1.9	90.5±3.9	69.2±9.0	83.4±2.4	90.5±0.4	92.9±1.6	71.2±5.9	88.6±1.3	90.8±0.5
Macaroni1	76.2±0.0	66.0±10.5	62.1±4.6	68.0±3.4	82.9±1.5	69.1±8.2	62.2±5.0	75.6±4.6	83.3±1.9	65.8±1.2	65.9±3.9	82.9±2.7	85.2±0.9
Macaroni2	63.7±0.0	55.8±6.1	47.5±5.9	55.6±4.6	70.2±0.9	58.3±4.4	50.8±2.9	57.3±5.6	71.8±2.0	56.7±3.2	55.0±2.9	61.7±1.8	70.9±2.2
PCB1	73.6±0.0	87.2±2.3	76.2±1.2	78.9±1.1	75.6±23.0	86.7±1.1	62.4±10.8	71.5±20.0	76.7±5.2	83.4±8.5	82.6±1.5	84.7±6.7	88.3±1.7
PCB2	51.2±0.0	73.5±3.7	61.2±2.0	81.5±0.8	62.2±3.9	70.3±8.1	66.8±2.0	84.3±1.7	62.6±3.7	71.7±7.0	73.5±2.4	84.3±1.0	67.5±2.6
PCB3	73.4±0.0	72.2±1.0	51.4±12.2	82.7±2.3	74.1±1.1	75.8±5.7	67.3±3.8	84.8±1.2	78.8±1.9	79.0±4.1	65.9±1.9	87.0±1.1	83.3±1.7
PCB4	79.6±0.0	93.4±1.3	76.1±3.6	93.9±2.8	85.2±8.9	86.1±8.2	69.3±13.7	94.3±3.2	82.3±9.9	95.4±2.3	85.4±2.0	95.6±1.6	87.6±8.0
Pipe fryum	69.7±0.0	77.9±3.2	66.7±2.2	90.7±1.7	97.2±1.1	78.1±3.0	75.3±1.8	93.5±1.3	98.0±0.6	79.3±0.9	82.9±2.2	96.4±0.7	98.5±0.4
Mean	**78.1±0.0**	79.5±4.0	62.8±5.4	79.9±2.9	**83.8±4.0**	80.7±5.0	67.4±5.1	81.6±4.0	**84.6±2.4**	81.7±3.4	72.8±2.9	85.3±2.1	**87.3±1.8**

表16. 在VisA数据集上按类别AUROC衡量的异常分类（AC）性能对比。我们报告了每个测量在5个随机种子下的平均值和标准差。

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
AUPR	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	95.8±0.0	86.5±4.3	69.2±3.9	86.6±2.3	93.6±1.5	90.7±3.2	72.8±1.0	86.8±1.7	95.1±1.1	92.6±1.9	72.5±1.1	88.9±1.1	95.3±0.4
Capsules	90.9±0.0	79.4±4.9	63.4±5.7	72.3±5.3	89.9±2.5	79.9±5.8	63.4±2.0	73.6±4.7	88.9±0.7	81.1±4.5	63.0±2.3	78.4±3.1	91.5±1.4
Cashew	96.4±0.0	97.9±0.4	78.2±5.7	94.6±2.0	97.2±0.2	98.6±0.6	86.1±2.2	96.9±0.3	97.3±0.2	98.3±0.6	88.4±2.0	96.5±0.7	97.7±0.4
Chewinggum	98.6±0.0	96.4±0.9	79.8±3.6	98.9±0.1	99.0±0.3	97.1±0.4	89.5±1.9	99.1±0.2	98.9±0.3	97.1±0.6	88.5±3.2	99.3±0.1	99.0±0.1
Fryum	90.1±0.0	89.8±1.8	74.5±2.9	87.6±2.4	94.7±1.0	94.5±2.3	81.0±5.4	92.1±1.3	95.8±0.2	95.8±1.0	81.5±3.0	95.0±0.6	96.0±0.3
Macaroni1	75.8±0.0	61.9±11.2	60.4±2.9	67.8±3.4	84.9±1.2	64.5±9.5	63.1±4.3	74.9±5.2	84.7±1.5	60.2±2.7	64.9±2.1	82.1±3.5	86.5±0.6
Macaroni2	60.3±0.0	52.7±4.2	51.7±5.0	54.9±3.2	68.4±1.8	55.9±3.1	52.7±1.5	57.2±2.6	70.4±1.8	51.9±2.3	54.9±2.5	60.2±3.0	69.6±2.8
PCB1	78.4±0.0	84.9±3.7	68.6±2.4	72.1±2.5	76.5±19.0	83.8±2.1	60.4±7.7	72.6±16.4	78.3±4.3	83.2±7.2	77.4±2.9	81.0±9.2	87.7±1.7
PCB2	49.2±0.0	74.9±2.9	63.3±1.2	84.4±0.4	64.9±3.3	71.7±6.6	68.9±2.6	86.6±1.1	65.8±4.0	74.2±5.0	75.0±1.7	86.2±1.0	71.3±3.4
PCB3	76.5±0.0	75.5±2.1	52.3±10.8	84.6±1.5	73.5±1.6	78.3±5.2	65.2±3.8	86.1±0.5	80.9±1.6	81.0±3.6	64.5±2.4	88.3±1.1	84.8±1.8
PCB4	77.7±0.0	92.9±1.6	74.7±2.6	92.8±3.1	78.5±15.5	81.9±11.2	67.6±11.9	93.2±3.4	72.5±16.2	94.8±2.9	84.0±2.0	94.9±1.2	85.6±8.9
Pipe fryum	82.3±0.0	88.3±2.0	79.2±1.5	95.4±0.6	98.6±0.5	88.1±1.7	84.5±1.7	96.8±0.7	99.0±0.3	88.8±1.0	89.8±1.7	98.3±0.3	99.2±0.2
Mean	**81.2±0.0**	82.0±3.3	68.3±4.0	82.8±2.3	**85.1±4.0**	82.3±4.3	71.6±3.8	84.8±3.2	**85.8±2.7**	83.4±2.7	75.6±2.2	87.5±2.1	**88.8±1.8**

表17. 在VisA数据集上按类别AUPR衡量的异常分类（AC）性能对比。我们报告了每个测量在5个随机种子下的平均值和标准差。

VisA (AC)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	89.4±0.0	80.4±6.1	72.0±2.0	79.5±0.6	87.8±1.2	85.5±2.9	74.8±2.9	78.7±0.8	89.1±1.3	86.9±2.4	76.7±1.3	80.5±0.9	88.9±1.0
Capsules	83.9±0.0	81.0±2.4	77.5±0.8	77.9±1.5	84.9±2.0	80.4±3.9	77.2±0.4	77.2±0.3	85.4±0.6	79.5±1.8	77.2±0.3	77.3±0.6	86.0±0.9
Cashew	88.4±0.0	94.8±1.8	80.8±0.7	89.6±3.6	90.7±0.7	95.5±2.2	82.4±1.2	92.3±0.5	90.9±0.7	95.7±0.9	82.5±1.2	91.1±2.1	91.6±1.3
Chewinggum	94.8±0.0	89.7±2.2	83.8±2.0	95.9±0.8	95.6±0.9	90.5±1.2	86.7±0.8	97.0±0.5	95.4±0.6	91.3±1.5	87.9±0.8	97.4±0.6	95.7±0.5
Fryum	82.7±0.0	85.3±2.1	80.6±0.8	82.9±1.7	87.2±1.4	90.9±1.8	82.7±2.1	84.7±1.4	88.4±0.6	91.9±1.7	82.9±1.8	86.6±0.6	88.9±0.8
Macaroni1	74.2±0.0	71.9±2.0	69.2±2.3	70.4±1.9	76.2±1.4	72.8±3.1	68.8±1.8	74.3±2.1	76.7±2.0	70.8±1.2	70.0±1.6	78.9±1.4	78.2±1.2
Macaroni2	69.8±0.0	68.1±0.8	67.1±0.2	67.6±0.7	72.3±1.1	68.2±1.2	67.1±0.4	67.6±1.2	73.9±0.9	67.9±0.6	68.4±1.0	68.8±0.8	73.1±1.6
PCB1	71.0±0.0	85.5±0.2	80.3±0.8	84.5±0.4	81.3±6.6	85.8±0.2	71.4±5.4	78.3±6.7	73.2±3.7	81.2±6.4	83.1±0.6	85.6±1.8	83.1±2.2
PCB2	67.1±0.0	70.9±1.9	68.6±1.4	75.9±0.8	67.2±0.3	71.5±2.7	69.2±0.6	78.1±2.1	67.3±0.3	71.1±3.2	72.0±2.3	79.2±1.9	67.7±0.6
PCB3	71.0±0.0	70.2±1.4	67.7±0.9	76.7±2.5	73.5±1.5	73.3±3.6	69.9±1.1	78.9±1.0	73.9±1.3	75.5±3.3	69.0±0.7	80.7±0.5	77.0±1.4
PCB4	74.9±0.0	87.5±1.7	74.5±2.2	90.4±2.7	86.1±2.1	83.1±5.9	74.6±3.9	91.3±4.1	86.8±3.8	90.6±2.1	81.0±1.6	92.2±3.4	84.6±7.0
Pipe fryum	80.7±0.0	82.7±0.7	81.2±0.5	89.2±2.4	94.4±0.7	82.7±1.4	83.1±0.6	91.4±1.1	95.4±0.8	82.8±0.6	85.4±0.8	93.9±1.2	95.6±0.7
Mean	**79.0±0.0**	80.7±1.9	75.3±1.2	81.7±1.6	**83.1±1.7**	81.7±2.5	75.7±1.8	82.5±1.8	**83.0±1.4**	82.1±2.1	78.0±1.2	84.3±1.3	**84.2±1.6**

T表18. 在VisA数据集上基于类别F_1-max的异常分类（AC）性能对比。我们报告了每个测量指标在5个随机种子下的平均值和标准差。

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
pAUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	88.9±0.0	97.9±0.3	91.7±2.2	97.2±0.2	97.4±0.2	98.1±0.2	94.9±0.8	97.7±0.3	97.7±0.1	98.2±0.1	95.4±0.2	97.9±0.1	97.8±0.2
Capsules	81.6±0.0	95.5±0.5	70.9±1.1	93.2±0.9	96.4±0.6	96.5±0.9	75.7±1.7	94.0±0.2	96.8±0.3	97.7±0.1	79.1±0.7	94.8±0.5	97.1±0.2
Cashew	84.7±0.0	95.9±0.5	95.5±0.6	98.1±0.1	98.5±0.2	95.9±0.4	96.4±0.4	98.2±0.2	98.5±0.1	95.9±0.3	97.2±0.3	98.3±0.2	98.7±0.0
Chewinggum	93.3±0.0	96.0±0.4	90.1±0.4	96.9±0.3	98.6±0.1	96.0±0.3	93.1±0.7	96.6±0.1	98.6±0.1	95.7±0.3	94.4±0.5	96.8±0.1	98.5±0.1
Fryum	88.5±0.0	93.5±0.3	93.3±0.6	93.3±0.5	96.4±0.3	93.9±0.2	94.1±0.6	94.0±0.3	97.0±0.2	94.4±0.1	95.0±0.4	94.2±0.2	97.1±0.1
Macaroni1	70.9±0.0	97.9±0.2	89.4±0.9	95.2±0.4	96.4±0.6	98.5±0.2	91.7±0.3	96.0±1.3	96.5±0.7	98.8±0.1	93.5±0.5	97.0±0.3	97.0±0.2
Macaroni2	59.3±0.0	94.1±1.0	86.4±1.1	89.1±1.6	96.8±0.4	95.2±0.4	90.1±0.8	90.2±1.9	96.8±0.6	96.4±0.2	90.2±0.3	93.9±0.3	97.3±0.3
PCB1	61.2±0.0	94.7±0.4	89.9±0.3	96.1±1.5	96.6±0.6	96.5±1.5	90.6±0.6	97.6±0.9	97.0±0.9	96.8±1.5	93.2±1.5	98.1±1.0	98.1±0.9
PCB2	71.6±0.0	95.1±0.2	90.9±1.4	95.4±0.2	93.0±0.4	95.7±0.1	93.9±0.9	96.0±0.3	93.9±0.2	96.3±0.0	93.7±1.0	96.6±0.2	94.6±0.4
PCB3	85.3±0.0	96.0±0.1	93.9±0.3	96.2±0.3	94.3±0.3	96.6±0.1	95.1±0.5	97.1±0.1	95.1±0.2	96.9±0.0	95.7±0.1	97.4±0.2	95.8±0.1
PCB4	94.4±0.0	92.0±0.6	89.6±0.6	95.6±0.6	94.0±0.9	92.8±0.3	90.7±0.9	96.2±0.4	95.6±0.3	94.1±0.2	92.1±0.5	97.0±0.2	96.1±0.3
Pipe fryum	75.4±0.0	98.4±0.2	97.2±0.6	98.8±0.2	98.3±0.2	98.7±0.1	98.1±0.4	99.1±0.1	98.5±0.2	98.8±0.0	98.5±0.1	99.1±0.0	98.7±0.1
Mean	**79.6±0.0**	95.6±0.4	89.9±0.8	95.4±0.6	**96.4±0.4**	96.2±0.4	92.0±0.7	96.1±0.5	**96.8±0.3**	96.6±0.3	93.2±0.5	96.8±0.3	**97.2±0.2**

Table 19. Comparison of anomaly segmentation (AS) performance in terms of class-wise pixel-AUROC on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
PRO	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	83.5±0.0	95.6±0.5	81.5±5.3	92.6±0.4	94.0±0.4	95.6±0.4	87.3±1.2	93.4±0.6	94.2±0.2	95.7±0.1	88.3±0.7	94.1±0.4	94.4±0.2
Capsules	35.3±0.0	83.1±1.1	30.6±1.1	66.6±4.5	73.6±3.5	85.4±3.1	38.4±3.7	67.9±2.3	75.9±1.9	89.0±1.2	43.3±2.0	69.0±3.2	77.0±1.4
Cashew	76.4±0.0	89.8±1.1	73.4±2.1	90.8±0.2	91.1±0.8	90.4±0.5	78.4±2.7	91.4±1.0	90.4±0.6	90.4±0.6	81.2±2.8	92.1±0.3	91.3±0.9
Chewinggum	70.4±0.0	73.9±1.2	58.1±0.6	78.2±1.3	91.0±0.5	73.8±1.1	63.7±2.4	78.0±0.4	90.9±0.7	72.7±0.9	67.2±1.8	79.3±0.8	91.0±0.4
Fryum	77.4±0.0	83.7±1.2	71.1±1.6	78.7±2.3	89.1±1.0	84.5±0.9	71.2±0.8	81.4±2.8	89.3±0.2	86.2±0.9	73.2±1.3	81.0±1.2	89.7±0.5
Macaroni1	34.3±0.0	92.0±0.6	62.2±4.4	83.4±1.3	84.6±2.3	93.9±0.8	71.8±2.4	86.2±4.6	85.2±1.4	95.1±0.4	76.6±2.1	89.6±0.7	86.8±0.8
Macaroni2	21.4±0.0	80.0±3.3	54.9±3.6	66.0±3.0	89.3±2.4	81.7±1.5	65.6±3.4	67.2±6.5	88.6±1.7	86.0±0.8	65.9±1.5	78.3±0.9	90.5±1.3
PCB1	26.3±0.0	81.3±5.7	63.9±1.8	79.0±10.7	82.5±6.0	87.2±2.3	68.4±4.1	86.1±1.7	83.8±5.0	88.0±2.7	70.2±3.3	88.1±2.6	87.9±2.1
PCB2	37.2±0.0	83.7±0.6	64.4±3.8	80.9±0.5	73.6±1.5	85.5±1.0	72.9±3.4	82.9±1.8	76.2±0.9	87.0±0.5	71.9±2.6	83.7±1.0	78.0±1.3
PCB3	56.1±0.0	84.3±1.0	69.0±1.2	78.1±2.0	79.5±2.5	86.1±0.6	74.0±2.3	82.2±1.1	82.3±1.8	87.7±0.6	77.2±0.8	84.4±1.9	84.2±1.0
PCB4	80.4±0.0	66.9±2.0	59.1±1.8	77.9±3.1	76.6±4.1	69.3±1.1	62.6±3.6	79.5±4.8	81.7±1.2	74.7±1.0	67.9±2.6	83.5±2.5	84.2±0.7
Pipe fryum	82.3±0.0	94.3±0.5	83.9±0.8	93.6±0.5	96.1±0.6	95.0±0.2	86.9±0.9	94.5±0.4	96.2±0.6	95.0±0.3	88.7±1.3	95.0±0.5	96.6±0.2
Mean	**56.8±0.0**	84.1±1.6	64.3±2.4	80.5±2.5	**85.1±2.1**	85.7±1.1	70.1±2.6	82.6±2.3	**86.2±1.4**	87.3±0.8	72.6±1.9	84.9±1.4	**87.6±0.9**

Table 20. Comparison of anomaly segmentation (AS) performance in terms of class-wise PRO on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	22.5±0.0	37.1±0.8	19.6±2.6	40.9±1.0	42.7±1.7	37.6±0.5	21.6±1.4	40.4±1.0	42.2±0.8	38.3±1.0	21.6±0.7	41.0±1.0	43.0±0.9
Capsules	9.2±0.0	28.5±7.8	2.3±0.2	35.7±6.8	58.2±1.3	37.4±8.4	3.0±0.4	37.8±5.7	57.0±3.7	48.0±2.0	3.9±0.4	47.0±3.0	59.8±1.8
Cashew	13.2±0.0	51.1±1.4	35.1±4.9	60.4±1.1	59.5±2.1	52.8±1.2	40.8±4.0	60.3±0.4	60.5±2.4	54.1±0.5	47.2±2.9	60.7±0.4	62.3±1.1
Chewinggum	41.1±0.0	58.7±1.1	19.7±2.0	64.5±1.0	65.3±0.5	59.9±0.5	29.5±6.4	63.9±0.4	64.8±0.9	59.5±0.6	37.8±4.4	64.4±0.6	65.2±0.2
Fryum	22.1±0.0	34.0±1.5	32.3±1.5	37.2±1.4	50.8±1.8	36.6±2.1	36.5±3.6	41.1±2.9	54.8±1.7	40.3±1.8	44.5±1.3	44.6±2.9	56.5±0.6
Macaroni1	7.0±0.0	28.5±3.4	4.8±0.7	16.5±2.6	34.1±1.7	39.2±3.5	5.5±1.2	20.0±8.2	33.2±1.9	37.6±5.8	7.0±0.7	21.6±1.9	33.8±0.9
Macaroni2	1.0±0.0	6.7±2.8	1.9±0.5	2.7±0.7	34.4±3.0	8.6±1.8	2.4±0.1	5.1±2.3	29.9±3.4	18.3±3.0	2.4±0.3	10.9±1.7	35.1±2.5
PCB1	2.4±0.0	16.6±1.1	8.6±0.7	30.7±3.2	25.9±2.6	34.8±19.9	8.9±0.5	47.7±19.7	34.6±16.2	37.1±20.8	13.9±3.6	55.9±20.3	50.9±20.4
PCB2	4.7±0.0	35.0±1.9	9.8±2.2	33.3±1.0	18.7±1.5	39.2±1.1	16.6±2.5	33.3±0.6	24.0±1.1	42.6±1.0	15.8±3.0	33.6±0.3	27.8±1.9
PCB3	10.3±0.0	43.9±1.6	18.1±0.3	36.6±0.6	31.2±6.7	44.9±0.7	20.6±1.6	37.1±0.2	37.1±2.8	47.5±0.7	22.4±1.3	37.3±0.1	42.5±1.1
PCB4	32.0±0.0	30.7±1.0	13.3±1.5	34.8±1.4	22.8±2.9	35.0±3.4	15.5±1.6	40.9±4.3	30.6±2.4	39.2±5.2	19.8±2.5	44.1±4.4	31.9±3.0
Pipe fryum	12.3±0.0	55.5±1.9	43.3±3.6	62.6±2.6	51.8±2.0	59.3±1.4	52.9±5.5	64.5±0.6	53.6±2.6	61.1±0.7	58.2±0.8	65.0±0.5	55.1±1.1
Mean	**14.8±0.0**	35.5±2.2	17.4±1.7	38.0±1.9	**41.3±2.3**	40.5±3.7	21.1±2.4	41.0±3.9	**43.5±3.3**	43.6±3.6	24.6±1.8	43.9±3.1	**47.0±3.0**

Table 21. Comparison of anomaly segmentation (AS) performance in terms of class-wise F_1-max on VisA. We report the mean and standard deviation over 5 random seeds for each measurement.

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
pAUROC	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	88.9±0.0	97.9±0.3	91.7±2.2	97.2±0.2	97.4±0.2	98.1±0.2	94.9±0.8	97.7±0.3	97.7±0.1	98.2±0.1	95.4±0.2	97.9±0.1	97.8±0.2
Capsules	81.6±0.0	95.5±0.5	70.9±1.1	93.2±0.9	96.4±0.6	96.5±0.9	75.7±1.7	94.0±0.2	96.8±0.3	97.7±0.1	79.1±0.7	94.8±0.5	97.1±0.2
Cashew	84.7±0.0	95.9±0.5	95.5±0.6	98.1±0.1	98.5±0.2	95.9±0.4	96.4±0.4	98.2±0.2	98.5±0.1	95.9±0.3	97.2±0.3	98.3±0.2	98.7±0.0
Chewinggum	93.3±0.0	96.0±0.4	90.1±0.4	96.9±0.3	98.6±0.1	96.0±0.3	93.1±0.7	96.6±0.1	98.6±0.1	95.7±0.3	94.4±0.5	96.8±0.1	98.5±0.1
Fryum	88.5±0.0	93.5±0.3	93.3±0.6	93.3±0.5	96.4±0.3	93.9±0.2	94.1±0.6	94.0±0.3	97.0±0.2	94.4±0.1	95.0±0.4	94.2±0.2	97.1±0.1
Macaroni1	70.9±0.0	97.9±0.2	89.4±0.9	95.2±0.4	96.4±0.6	98.5±0.2	91.7±0.3	96.0±1.3	96.5±0.7	98.8±0.1	93.5±0.5	97.0±0.3	97.0±0.2
Macaroni2	59.3±0.0	94.1±1.0	86.4±1.1	89.1±1.6	96.8±0.4	95.2±0.4	90.1±0.8	90.2±1.9	96.8±0.6	96.4±0.2	90.2±0.3	93.9±0.3	97.3±0.3
PCB1	61.2±0.0	94.7±0.4	89.9±0.3	96.1±1.5	96.6±0.6	96.5±1.5	90.6±0.6	97.6±0.9	97.0±0.9	96.8±1.5	93.2±1.5	98.1±1.0	98.1±0.9
PCB2	71.6±0.0	95.1±0.2	90.9±1.4	95.4±0.2	93.0±0.4	95.7±0.1	93.9±0.9	96.0±0.3	93.9±0.2	96.3±0.0	93.7±1.0	96.6±0.2	94.6±0.4
PCB3	85.3±0.0	96.0±0.1	93.9±0.3	96.2±0.3	94.3±0.3	96.6±0.1	95.1±0.5	97.1±0.1	95.1±0.2	96.9±0.0	95.7±0.1	97.4±0.2	95.8±0.1
PCB4	94.4±0.0	92.0±0.6	89.6±0.6	95.6±0.6	94.0±0.9	92.8±0.3	90.7±0.9	96.2±0.4	95.6±0.3	94.1±0.2	92.1±0.5	97.0±0.2	96.1±0.3
Pipe fryum	75.4±0.0	98.4±0.2	97.2±0.6	98.8±0.2	98.3±0.2	98.7±0.1	98.1±0.4	99.1±0.1	98.5±0.2	98.8±0.0	98.5±0.1	99.1±0.0	98.7±0.1
Mean	**79.6±0.0**	95.6±0.4	89.9±0.8	95.4±0.6	**96.4±0.4**	96.2±0.4	92.0±0.7	96.1±0.5	**96.8±0.3**	96.6±0.3	93.2±0.5	96.8±0.3	**97.2±0.2**

T表 19. VisA数据集上各类别像素-AUROC的异常分割（AS）性能对比。我们报告了每个测量指标在5个随机种子下的平均值和标准差。

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
PRO	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	83.5±0.0	95.6±0.5	81.5±5.3	92.6±0.4	94.0±0.4	95.6±0.4	87.3±1.2	93.4±0.6	94.2±0.2	95.7±0.1	88.3±0.7	94.1±0.4	94.4±0.2
Capsules	35.3±0.0	83.1±1.1	30.6±1.1	66.6±4.5	73.6±3.5	85.4±3.1	38.4±3.7	67.9±2.3	75.9±1.9	89.0±1.2	43.3±2.0	69.0±3.2	77.0±1.4
Cashew	76.4±0.0	89.8±1.1	73.4±2.1	90.8±0.2	91.1±0.8	90.4±0.5	78.4±2.7	91.4±1.0	90.4±0.6	90.4±0.6	81.2±2.8	92.1±0.3	91.3±0.9
Chewinggum	70.4±0.0	73.9±1.2	58.1±0.6	78.2±1.3	91.0±0.5	73.8±1.1	63.7±2.4	78.0±0.4	90.9±0.7	72.7±0.9	67.2±1.8	79.3±0.8	91.0±0.4
Fryum	77.4±0.0	83.7±1.2	71.1±1.6	78.7±2.3	89.1±1.0	84.5±0.9	71.2±0.8	81.4±2.8	89.3±0.2	86.2±0.9	73.2±1.3	81.0±1.2	89.7±0.5
Macaroni1	34.3±0.0	92.0±0.6	62.2±4.4	83.4±1.3	84.6±2.3	93.9±0.8	71.8±2.4	86.2±4.6	85.2±1.4	95.1±0.4	76.6±2.1	89.6±0.7	86.8±0.8
Macaroni2	21.4±0.0	80.0±3.3	54.9±3.6	66.0±3.0	89.3±2.4	81.7±1.5	65.6±3.4	67.2±6.5	88.6±1.7	86.0±0.8	65.9±1.5	78.3±0.9	90.5±1.3
PCB1	26.3±0.0	81.3±5.7	63.9±1.8	79.0±10.7	82.5±6.0	87.2±2.3	68.4±4.1	86.1±1.7	83.8±5.0	88.0±2.7	70.2±3.3	88.1±2.6	87.9±2.1
PCB2	37.2±0.0	83.7±0.6	64.4±3.8	80.9±0.5	73.6±1.5	85.5±1.0	72.9±3.4	82.9±1.8	76.2±0.9	87.0±0.5	71.9±2.6	83.7±1.0	78.0±1.3
PCB3	56.1±0.0	84.3±1.0	69.0±1.2	78.1±2.0	79.5±2.5	86.1±0.6	74.0±2.3	82.2±1.1	82.3±1.8	87.7±0.6	77.2±0.8	84.4±1.9	84.2±1.0
PCB4	80.4±0.0	66.9±2.0	59.1±1.8	77.9±3.1	76.6±4.1	69.3±1.1	62.6±3.6	79.5±4.8	81.7±1.2	74.7±1.0	67.9±2.6	83.5±2.5	84.2±0.7
Pipe fryum	82.3±0.0	94.3±0.5	83.9±0.8	93.6±0.5	96.1±0.6	95.0±0.2	86.9±0.9	94.5±0.4	96.2±0.6	95.0±0.3	88.7±1.3	95.0±0.5	96.6±0.2
Mean	**56.8±0.0**	84.1±1.6	64.3±2.4	80.5±2.5	**85.1±2.1**	85.7±1.1	70.1±2.6	82.6±2.3	**86.2±1.4**	87.3±0.8	72.6±1.9	84.9±1.4	**87.6±0.9**

表20. VisA数据集上基于类别PRO的异常分割（AS）性能对比。各项测量结果均报告5个随机种子的平均值与标准差。

VisA (AS)	$K=0$	$K=1$				$K=2$				$K=4$			
F_1-max	**WinCLIP**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**	SPADE	PaDiM	PatchCore	**WinCLIP+**
Candle	22.5±0.0	37.1±0.8	19.6±2.6	40.9±1.0	42.7±1.7	37.6±0.5	21.6±1.4	40.4±1.0	42.2±0.8	38.3±1.0	21.6±0.7	41.0±1.0	43.0±0.9
Capsules	9.2±0.0	28.5±7.8	2.3±0.2	35.7±6.8	58.2±1.3	37.4±8.4	3.0±0.4	37.8±5.7	57.0±3.7	48.0±2.0	3.9±0.4	47.0±3.0	59.8±1.8
Cashew	13.2±0.0	51.1±1.4	35.1±4.9	60.4±1.1	59.5±2.1	52.8±1.2	40.8±4.0	60.3±0.4	60.5±2.4	54.1±0.5	47.2±2.9	60.7±0.4	62.3±1.1
Chewinggum	41.1±0.0	58.7±1.1	19.7±2.0	64.5±1.0	65.3±0.5	59.9±0.5	29.5±6.4	63.9±0.4	64.8±0.9	59.5±0.6	37.8±4.4	64.4±0.6	65.2±0.2
Fryum	22.1±0.0	34.0±1.5	32.3±1.5	37.2±1.4	50.8±1.8	36.6±2.1	36.5±3.6	41.1±2.9	54.8±1.7	40.3±1.8	44.5±1.3	44.6±2.9	56.5±0.6
Macaroni1	7.0±0.0	28.5±3.4	4.8±0.7	16.5±2.6	34.1±1.7	39.2±3.5	5.5±1.2	20.0±8.2	33.2±1.9	37.6±5.8	7.0±0.7	21.6±1.9	33.8±0.9
Macaroni2	1.0±0.0	6.7±2.8	1.9±0.5	2.7±0.7	34.4±3.0	8.6±1.8	2.4±0.1	5.1±2.3	29.9±3.4	18.3±3.0	2.4±0.3	10.9±1.7	35.1±2.5
PCB1	2.4±0.0	16.6±1.1	8.6±0.7	30.7±3.2	25.9±2.6	34.8±19.9	8.9±0.5	47.7±19.7	34.6±16.2	37.1±20.8	13.9±3.6	55.9±20.3	50.9±20.4
PCB2	4.7±0.0	35.0±1.9	9.8±2.2	33.3±1.0	18.7±1.5	39.2±1.1	16.6±2.5	33.3±0.6	24.0±1.1	42.6±1.0	15.8±3.0	33.6±0.3	27.8±1.9
PCB3	10.3±0.0	43.9±1.6	18.1±0.3	36.6±0.6	31.2±6.7	44.9±0.7	20.6±1.6	37.1±0.2	37.1±2.8	47.5±0.7	22.4±1.3	37.3±0.1	42.5±1.1
PCB4	32.0±0.0	30.7±1.0	13.3±1.5	34.8±1.4	22.8±2.9	35.0±3.4	15.5±1.6	40.9±4.3	30.6±2.4	39.2±5.2	19.8±2.5	44.1±4.4	31.9±3.0
Pipe fryum	12.3±0.0	55.5±1.9	43.3±3.6	62.6±2.6	51.8±2.0	59.3±1.4	52.9±5.5	64.5±0.6	53.6±2.6	61.1±0.7	58.2±0.8	65.0±0.5	55.1±1.1
Mean	**14.8±0.0**	35.5±2.2	17.4±1.7	38.0±1.9	**41.3±2.3**	40.5±3.7	21.1±2.4	41.0±3.9	**43.5±3.3**	43.6±3.6	24.6±1.8	43.9±3.1	**47.0±3.0**

表21. 在VisA数据集上基于类别F_1-max的异常分割（AS）性能比较。我们报告了每个测量在5个随机种子上的平均值和标准差。