

Chun-Liang Li*, Kihyuk Sohn*, Jinsung Yoon, Tomas Pfister
 Google Cloud AI Research
 {chunliang, kihyuks, jinsungyoon, tpfister}@google.com

Abstract

We aim at constructing a high performance model for defect detection that detects unknown anomalous patterns of an image without anomalous data. To this end, we propose a two-stage framework for building anomaly detectors using normal training data only. We first learn self-supervised deep representations and then build a generative one-class classifier on learned representations. We learn representations by classifying normal data from the CutPaste, a simple data augmentation strategy that cuts an image patch and pastes at a random location of a large image. Our empirical study on MVTec anomaly detection dataset demonstrates the proposed algorithm is general to be able to detect various types of real-world defects. We bring the improvement upon previous arts by 3.1 AUCs when learning representations from scratch. By transfer learning on pretrained representations on ImageNet, we achieve a new state-of-the-art 96.6 AUC. Lastly, we extend the framework to learn and extract representations from patches to allow localizing defective areas without annotations during training.

1. Introduction

Anomaly detection aims to detect an instance containing anomalous and defective patterns that are different from those seen in normal instances. Many problems from different vision applications are anomaly detection, including manufacturing defect detection [9, 5], medical image analysis [50, 48], and video surveillance [2, 31, 53]. Unlike a typical supervised classification problem, anomaly detection faces unique challenges. First, due to the nature of the problem, it is difficult to obtain a large amount of anomalous data, either labeled or unlabeled. Second, the difference between normal and anomalous patterns are often *fine-grained* as defective areas might be small and subtle in high-resolution images.

Due to limited access to anomalous data, constructing an anomaly detector is often conducted under semi-supervised or one-class classification settings using normal data only.

*Equal contributions.

Since the distribution of anomaly patterns is unknown in advance, we train models to learn patterns of normal instances and determine anomaly if the test example is not represented well by these models. For example, an autoencoder that is trained to reconstruct normal data is used to declare anomalies when the data reconstruction error is high. Generative models declare anomalies when the probability density is below a certain threshold. However, the anomaly score defined as an aggregation of pixel-wise reconstruction error or probability densities lacks to capture a high-level semantic information [42, 37].

Alternative methods using high-level learned representations have shown more effective for anomaly detection. For example, deep one-class classifier [46] demonstrates an effective end-to-end trained one-class classifiers parameterized by deep neural networks. It outperforms its shallow counterparts, such as one-class SVMs [49] and reconstruction-based approaches such as autoencoders [34]. In self-supervised representation learning, predicting geometric transformations of an image [20, 24, 4], such as rotation or translation, and contrastive learning [54, 52] have shown to be successful in distinguishing normal data from outliers. However, most existing works focus on detecting semantic outliers (e.g., visual objects from different classes) from object-centric natural images. In Section 4.1, we show these methods do not generalize well in detecting fine-grained anomalous patterns as in defect detection.

In this work, we tackle a one-class defect detection problem, a special case of image anomaly detection, where various forms of unknown anomalous patterns present locally in the high-resolution images. We follow the two-stage framework [52], where we first learn self-supervised representations by solving a proxy task, then build a generative one-class classifier on learned representations to distinguish data with anomalous patterns from normal ones. Our innovation is at designing a novel proxy task for self-supervised learning of representations. Specifically, we formulate a proxy classification task between normal training data and the ones augmented by the *CutPaste*, the proposed data augmentation strategy that cuts an image patch and pastes at a random location of an image. *CutPaste* augmentation is mo-

CutPaste：用于异常检测与定位的自监督学习

李春亮*、孙奇旭*、Jinsung Yoon、Tomas Pfister Google
Cloud AI 研究院

{chunliang,kihyuk,jinsungyoon,tpfister}@google.com

摘要

We aim at constructing a high performance model for defect detection that detects unknown anomalous patterns of an image without anomalous data. To this end, we propose a two-stage framework for building anomaly detectors using normal training data only. We first learn self-supervised deep representations and then build a generative one-class classifier on learned representations. We learn representations by classifying normal data from the CutPaste, a simple data augmentation strategy that cuts an image patch and pastes at a random location of a large image. Our empirical study on MVTec anomaly detection dataset demonstrates the proposed algorithm is general to be able to detect various types of real-world defects. We bring the improvement upon previous arts by 3.1 AUCs when learning representations from scratch. By transfer learning on pretrained representations on ImageNet, we achieve a new state-of-the-art **96.6** AUC. Lastly, we extend the framework to learn and extract representations from patches to allow localizing defective areas without annotations during training.

1. 引言

异常检测旨在识别包含与正常实例中不同、具有异常和缺陷模式的实例。从制造缺陷检测[9, 5]、医学图像分析[50, 48]到视频监控[2, 31, 53]，许多视觉应用中的问题都属于异常检测范畴。与典型的监督分类问题不同，异常检测面临独特挑战：首先，由于问题本质，难以获取大量异常数据（无论是否标注）；其次，正常与异常模式之间的差异往往*fine-grained*，因为在高分辨率图像中缺陷区域可能既微小又难以察觉。

由于异常数据的获取途径有限，构建异常检测器通常仅在正常数据下以半监督或单分类设置进行。

*Equal contributions.

由于异常模式的分布是未知的，我们训练模型来学习正常实例的模式，并在测试样本不能被这些模型很好地表示时判定为异常。例如，训练用于重构正常数据的自编码器会在数据重构误差较高时判定异常；生成模型则在概率密度低于特定阈值时判定异常。然而，基于像素级重构误差或概率密度聚合定义的异常评分难以捕捉高层次语义信息[42, 37]。

利用高层学习表征的替代方法在异常检测中已显示出更高的效能。例如，深度单类分类器[46]展示了通过深度神经网络参数化的端到端训练单类分类器的有效性。其性能优于浅层模型（如单类支持向量机[49]）以及基于重构的方法（如自动编码器[34]）。在自监督表征学习中，预测图像的几何变换（如旋转或平移）[20, 24, 4]与对比学习[54, 52]已被证明能有效区分正常数据与异常值。然而，现有研究大多聚焦于从以物体为中心的自然图像中检测语义异常（如来自不同类别的视觉对象）。在第4.1节中，我们将展示这些方法在缺陷检测场景下检测细粒度异常模式时泛化能力不足。

在本工作中，我们解决单类别缺陷检测问题——这是图像异常检测的一种特殊案例，即高分辨率图像局部区域可能呈现各种未知的异常模式。我们遵循两阶段框架[52]：首先通过求解代理任务来学习自监督表征，然后在学习到的表征上构建生成式单类别分类器，从而区分具有异常模式的数据与正常数据。我们的创新在于设计了一种新颖的自监督表征学习代理任务。具体而言，我们构建了正常训练数据与经过CutPaste增强数据之间的代理分类任务——该数据增强策略通过裁剪图像区块并随机粘贴至图像其他位置来实现。CutPaste增强方法旨在

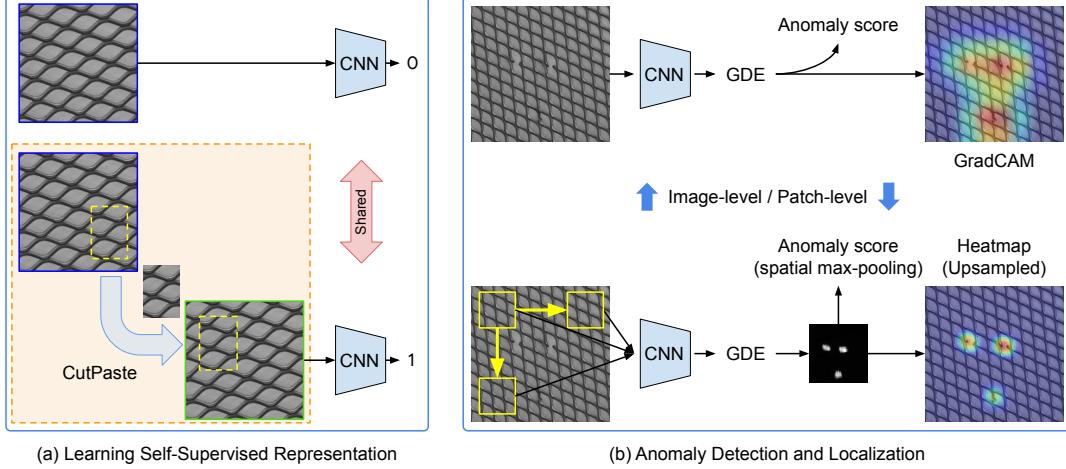


Figure 1: An overview of our method for anomaly detection and localization. (a) A deep network (CNN) is trained to distinguish images from normal (blue) and augmented (green) data distributions by *CutPaste* (orange dotted box), which cuts a small rectangular region (yellow dotted box) from normal data and pastes it at random location. Representations are trained either from the whole image or local patches. (b, top) An image-level representation makes a holistic decision for anomaly detection and is used to localize defect via GradCAM [51]. (b, bottom) A patch-level representation extracts dense features from local patches to produce anomaly score map, which is then max-pooled for detection or upsampled for localization [32].

tivated to produce a spatial *irregularity* to serve as a coarse approximation of real defects, which we have no access at training. Rectangular patches of different sizes, aspect ratios, and rotation angles are pasted to generate diverse augmentations. Although *CutPaste* augmented samples (Figure 2(e)) are easily distinguishable from real defects and thus might be a crude approximation of a real anomaly distribution, we show that representations learned by detecting irregularity introduced by *CutPaste* augmentations generalize well on detecting real defects.

We evaluate our methods on MVTec anomaly detection dataset [5], a real-world industrial visual inspection benchmark. By learning deep representations from scratch, we achieve **95.2 AUC** on image-level anomaly detection, which outperforms existing works [25, 61] by at least 3.1 AUC. Moreover, we report state-of-the-art **96.6 image-level AUC** by transfer learning from an ImageNet pretrained model. Moreover, we explain how learned representations could be used to localize the defective areas in high-resolution images. Without using any anomaly data, a simple patch model extension can achieve **96.0 pixel-level localization AUC**, which improves upon previous state-of-the-art [61] (95.7 AUC). We conduct an extensive study using different types of augmentation and proxy tasks to show the effectiveness of *CutPaste* augmentations for self-supervised representation learning on unknown defect detection.

2. A Framework for Anomaly Detection

In this section, we present our anomaly detection framework for high-resolution image with defects in local regions. Following [54], we adopt a two-stage framework for

building an anomaly detector, where in the first stage we learn deep representations from normal data and then construct an one-class classifier using learned representations. Subsequently, in Section 2.1, we present a novel method for learning self-supervised representations by predicting *CutPaste* augmentation, and extend to learning and extracting representations from local patches in Section 2.4.

2.1. Self-Supervised Learning with *CutPaste*

Defining good pretext tasks is essential for self-supervised representation learning. While popular methods including rotation prediction [19] and contrastive learning [60, 12] have been studied in the context of semantic one-class classification [20, 24, 4, 54, 52], our study in Section 4.1 shows that naively applying existing methods, such as rotation prediction or contrastive learning, is sub-optimal for detecting local defects as we will show in Section 4.1.

We conjecture that geometric transformations [20, 24, 4], such as rotations and translations, are effective in learning representation of semantic concepts (e.g., objectness), but less of regularity (e.g., continuity, repetition). As shown in Figure 2(b), anomalous patterns of defect detection typically include irregularities such as cracks (bottle, wood) or twists (toothbrush, grid). Our aim is to design an augmentation strategy creating local irregular patterns. Then we train the model to identify these local irregularity with the hope that it can generalize to unseen real defects at test time.

A popular augmentation method that could create a local irregularity in image is *Cutout* [18] (Figure 2(c)), which wipes out a randomly selected small rectangular area of an image. *Cutout* is found to be a useful data augmenta-

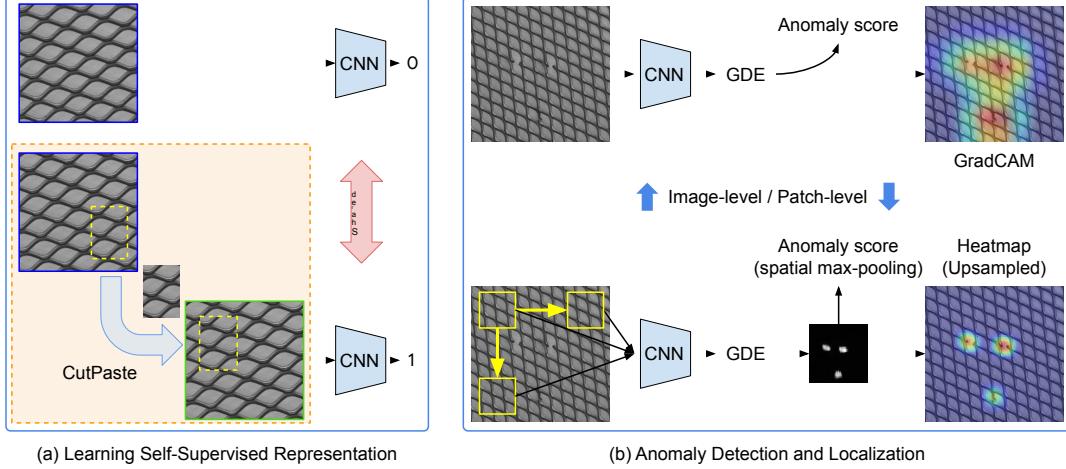


图1：我们的异常检测与定位方法概览。(a)通过CutPaste(橙色虚线框)从正常数据中截取小型矩形区域(黄色虚线框)并随机粘贴至其他位置，训练深度网络(CNN)区分正常(蓝色)与增强(绿色)数据分布的图像。表征训练可基于完整图像或局部图像块进行。(b上图)图像级表征对异常检测进行整体判断，并通过GradCAM[51]实现缺陷定位。(b下图)图像块级表征从局部区域提取密集特征生成异常得分图，随后通过最大池化实现检测或上采样实现定位[32]。

我们尝试生成空间化的*irregularity*作为真实缺陷的粗略近似，因为在训练阶段无法获取真实缺陷数据。通过粘贴不同尺寸、长宽比和旋转角度的矩形块，我们生成了多样化的增强样本。尽管CutPaste增强样本(图2(e))与真实缺陷容易区分，可能只是真实异常分布的粗略近似，但我们发现通过检测CutPaste增强引入的异常所学习到的表征，能够有效泛化到真实缺陷的检测任务中。

我们在MVTec异常检测数据集[5]上评估了所提出的方法，这是一个真实工业视觉检测基准。通过从零开始学习深度表征，我们在图像级异常检测中实现了95.2%的AUC，较现有工作[25,61]至少提升3.1个AUC点。此外，通过ImageNet预训练模型的迁移学习，我们取得了当前最优的96.6%图像级AUC。我们还阐释了如何利用学习到的表征定位高分辨率图像中的缺陷区域。在不使用任何异常数据的情况下，简单的补丁模型扩展即可实现96.0%的像素级定位AUC，较先前最优方法[61](95.7 AUC)有所提升。我们通过多种数据增强方式和代理任务进行广泛研究，证明了CutPaste增强技术在未知缺陷检测的自监督表征学习中的有效性。

2. 异常检测框架

在本节中，我们针对局部区域存在缺陷的高分辨率图像提出异常检测框架。遵循[54]的研究思路，我们采用两阶段框架进行

构建异常检测器，在第一阶段我们从正常数据中学习深度表征，随后利用学习到的表征构建单分类器。接着在2.1节中，我们提出通过预测剪切-粘贴数据增强来学习自监督表征的新方法，并在2.4节延伸至从局部图像块中学习并提取表征。

2.1. 使用CutPaste的自监督学习

定义合适的预训练任务对于自监督表征学习至关重要。尽管旋转预测[19]和对比学习[60,12]等流行方法已在语义单类别分类[20,24,4,54,52]背景下得到研究，但我们在第4.1节中的研究表明，如第4.1节将展示的，直接套用现有方法(如旋转预测或对比学习)来检测局部缺陷并非最优解。

我们推测，几何变换[20, 24, 4](如旋转和平移)能有效学习语义概念(如物体性)的表征，但对规律性(如连续性、重复性)的表征作用有限。如图2(b)所示，缺陷检测中的异常模式通常包含不规则特征，例如裂纹(瓶子、木材)或扭曲(牙刷、网格)。我们的目标是设计一种能生成局部不规则模式的增强策略，通过训练模型识别这些局部不规则性，以期在测试时能泛化到未见过的真实缺陷。

一种能在图像中制造局部不规则性的流行增强方法是Cutout[18](图2(c))，该方法会擦除图像中随机选取的小矩形区域。研究发现Cutout是一种有效的数据增强手段。

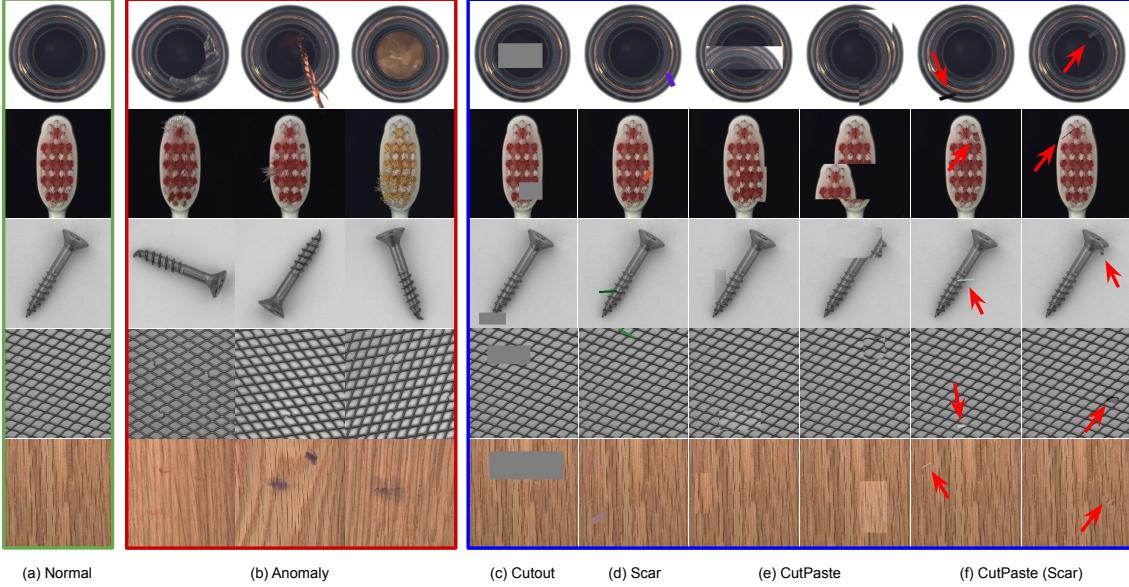


Figure 2: Visualization of (a, green) normal, (b, red) anomaly, and (c–h, blue) augmented normal samples from bottle, toothbrush, screw, grid, and wood classes of MVTec anomaly detection dataset [5]. Augmented normal samples are generated by baseline augmentations including (c) Cutout and (d) Scar, and our proposed (e) CutPaste and (f) CutPaste (Scar). We use red arrows in (f) to highlight the pasted patch of scar shape, a thin rectangle with rotation.

tion that enforces *invariance*, leading to improved accuracy on multi-class classification tasks. In contrast, we start by *discriminating* Cutout images from the normal ones. At first glance, the task seems easy to solve by well-crafted low level image filters. Surprisingly, as we will show in Section 4, without the hindsight of knowing this, a deep convolution network does not learn these shortcuts. Using Cutout in the algorithm design for defect detection can also be found in [32, 57]. We can make the task harder by randomly choosing colors and the scale as shown in Figure 2(d) to avoid naive shortcut solutions.

To further prevent learning naive decision rules for discriminating augmented images and encouraging the model to learn to detect irregularity, we propose the *CutPaste* augmentation as follows:

1. Cut a small rectangular area of variable sizes and aspect ratios from a normal training image.
2. Optionally, we rotate or jitter pixel values in the patch.
3. Paste a patch back to an image at a random location.

We show the CutPaste augmentation process in the orange dotted box of Figure 1 and more examples in Figure 2(e). Following the idea of rotation prediction [19], we define the training objective of the proposed self-supervised representation learning as follows:

$$\mathcal{L}_{\text{CP}} = \mathbb{E}_{x \in \mathcal{X}} \{ \text{CE}(g(x), 0) + \text{CE}(g(\text{CP}(x)), 1) \} \quad (1)$$

where \mathcal{X} is the set of normal data, $\text{CP}(\cdot)$ is a CutPaste augmentation and g is a binary classifier parameterized by deep

networks. $\text{CE}(\cdot, \cdot)$ refers to a cross-entropy loss. In practice, data augmentations, such as translation or color jitter, are applied before feeding x into g or CP.

2.2. CutPaste Variants

CutPaste-Scar. A special case of Cutout called “scar” using a long-thin rectangular box of random color, as in Figure 2(d), is proposed in [16] for defect detection. Similarly, in addition to original CutPaste using a large rectangular patch, we propose a *CutPaste-Scar* using a scar-like (long-thin) rectangular box filled with an image patch (Figure 2(f)).

Multi-Class Classification. While CutPaste (large patch) and CutPaste-Scar share a similarity, the shapes of an image patch of two augmentations are very different. Empirically, they have their own advantages on different types of defects. To leverage the strength of both scales in the training, we formulate a finer-grained 3-way classification task among normal, CutPaste and CutPaste-Scar by treating CutPaste variants as two separate classes. Detailed study will be presented in Section 5.2.

Similarity between CutPaste and real defects. The success of CutPaste may be understood from outlier exposure [23], where we generate the pseudo anomalies (CutPaste) during the training. Apart from using natural images as in [23], CutPaste creates examples preserving more local structures of the normal examples (i.e., the pasted patch is

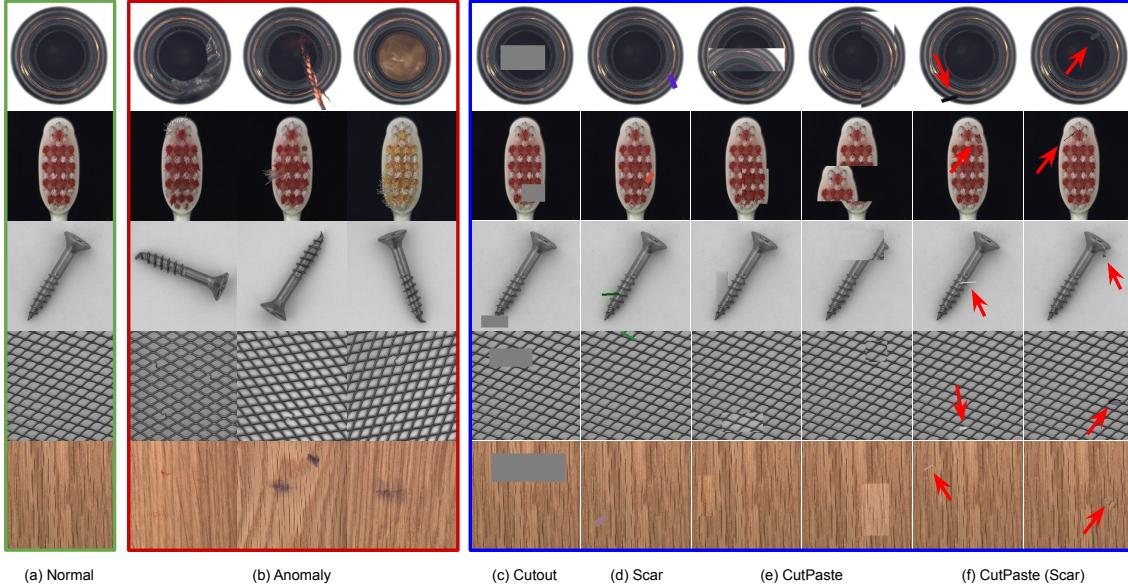


图2：MVTec异常检测数据集[5]中瓶子、牙刷、螺丝、网格和木材类别的(a,绿色)正常样本、(b,红色)异常样本及(c-h,蓝色)增强正常样本可视化。增强正常样本通过基线增强方法生成，包括(c)Cutout和(d)Scar，以及我们提出的(e)CutPaste与(f)CutPaste(Scar)。我们在(f)中使用红色箭头突出显示疤痕形状的粘贴补丁——一个经过旋转的细长矩形。

强制 $invariance$ 的约束，在多类别分类任务中提高了准确性。相比之下，我们首先从正常图像中 *discriminating*截取区域。乍看之下，通过精心设计的低级图像滤波器似乎很容易解决这个任务。但令人惊讶的是，正如我们将在第4节展示的，在不知情的情况下，深度卷积网络并不会学习这些捷径。在缺陷检测算法设计中使用Cutout的方法也可见于[32, 57]。我们可以通过随机选择颜色和比例（如图2(d)所示）来增加任务难度，从而避免简单的捷径解决方案。

为了进一步防止学习区分增强图像的简单决策规则，并鼓励模型学习检测不规则性，我们提出如下 *CutPaste*增强方法：

1. 从正常训练图像中截取一个尺寸和长宽比可变的矩形区域。
2. 可选择对该图像块进行旋转或像素值抖动处理。
3. 将图像块随机粘贴回原图像的某个位置。

我们在图1的橙色虚线框中展示了CutPaste增强过程，并在图2(e)中展示了更多示例。遵循旋转预测[19]的思想，我们将提出的自监督表征学习的训练目标定义如下：

$$\mathcal{L}_{CP} = \mathbb{E}_{x \in \mathcal{X}} \{ \text{CE}(g(x), 0) + \text{CE}(g(CP(x)), 1) \} \quad (1)$$

其中 \mathcal{X} 是正常数据的集合， $CP(\cdot)$ 是CutPaste增强方法， g 是由深度神经网络参数化的二元分类器

网络。 $\text{CE}(\cdot, \cdot)$ 指的是交叉熵损失。在实践中，数据增强（如平移或颜色抖动）会在将 x 输入 g 或 CP 之前应用。

2.2. CutPaste变体

CutPaste-Scar。文献[16]提出了一种称为“疤痕”的Cutout特殊形式，采用随机颜色的细长矩形框（如图2(d)所示）进行缺陷检测。类似地，我们在原始使用大矩形补丁的CutPaste基础上，提出采用CutPaste-Scar方法——使用填充图像补丁的疤痕状（细长型）矩形框（如图2(f)所示）。

多类别分类。虽然CutPaste（大尺寸块）与CutPaste-Scar存在相似性，但两种数据增强方法生成的图像块形状差异显著。实验表明，它们在不同类型的缺陷检测中各具优势。为在训练中兼顾两种尺度方法的优势，我们通过将CutPaste变体视为两个独立类别，构建了更细粒度的三向分类任务（正常样本、CutPaste样本与CutPaste-Scar样本）。详细研究将在第5.2节展开。

CutPaste与真实缺陷之间的相似性。CutPaste的成功可以从异常暴露[23]的角度理解——我们在训练过程中生成伪异常样本（即CutPaste）。与[23]中直接使用自然图像不同，CutPaste通过保留正常样本更多局部结构的方式创建样本（即被粘贴的图像块会保持

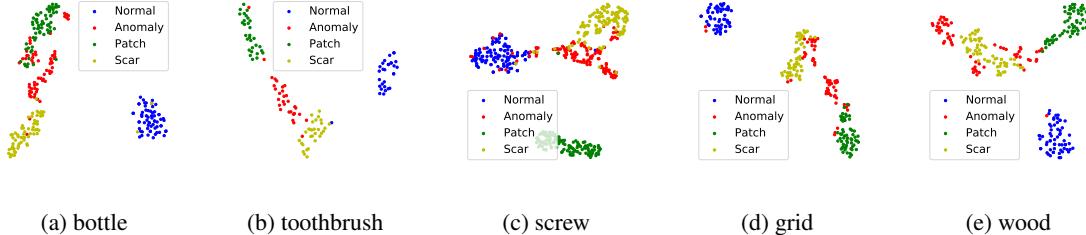


Figure 3: t-SNE visualization of representations of models trained with 3-way CutPaste prediction task. We plot embeddings of normal (blue), anomaly (red), and augmented normal by CutPaste (“Patch”, green) and CutPaste-scar (“Scar”, yellow).

from the same domain), which is more challenging for the model to learn to find this *irregularity*.

On the other hand, CutPaste does look similar to some real defects. A natural question is if the success of CutPaste is from a good mimic of real defects. In Figure 3, we show the t-SNE plots of the representations from the trained model. Clearly, the CutPaste examples are almost not overlapped with real defect examples (anomaly), but the learned representation is able to distinguish between normal example, different CutPaste augmented samples and real defects. It suggests (1) CutPaste is still not a perfect simulation of real defects and (2) learning on it to find irregularity generalizes well on unseen anomalies.

2.3. Computing Anomaly Score

There exist various ways to compute anomaly scores via one-class classifiers. In this work, we build generative classifiers like kernel density estimator [52] or Gaussian density estimator [43], on representations f . Below, we explain how to compute anomaly scores and the trade-offs.

Although nonparametric KDE is free from distribution assumptions, it requires many examples for accurate estimation [58] and could be computationally expensive. With limited normal training examples for defect detection, we consider a simple parametric Gaussian density estimator (GDE) whose log-density is computed as follows:

$$\log p_{\text{gde}}(x) \propto \left\{ -\frac{1}{2} (f(x) - \mu)^\top \Sigma^{-1} (f(x) - \mu) \right\} \quad (2)$$

where μ and Σ are learned from normal training data.¹

2.4. Localization with Patch Representation

While we present a method for learning a holistic representation of an image, learning a representation of an image patch would be preferred if we want to localize defective regions [38, 6, 61] in addition to image-level detection. By learning and extracting representations from an image

¹We note that a mixture of Gaussian, which is a middle ground between KDE and GDE, can also be used for more expressive density modeling. We do not observe significant performance gain empirically.

patch, we can build an anomaly detector that is able to compute the score of an image patch, which then can be used to localize the defective area.

CutPaste prediction is readily applicable to learn a patch representation – all we need to do at training is to crop a patch before applying CutPaste augmentation. Similar to Equation (1), the training objective can be written as:

$$\mathbb{E}_{x \in \mathcal{X}} \{ \mathbb{CE}(g(c(x)), 0) + \mathbb{CE}(g(\text{CP}(c(x))), 1) \} \quad (3)$$

where $c(x)$ crops a patch at random location of x . At test time, we extract embeddings from all patches with a given stride. For each patch, we evaluate its anomaly score and use a Gaussian smoothing to propagate the score to every pixel [32]. In Section 4.2, we visualize a heatmap using patch-level detector for defect localization, along with that of an image-level detector using visual explanation techniques such as GradCAM [51].

3. Related Work

Anomaly detection under one-class classification setting, where we assume only the normal data is given during the training, has been widely studied [49, 56, 46, 63, 66, 13, 42, 36, 27]. Recent success of self-supervised learning in computer vision [39, 19, 8, 60, 40, 12, 21] has also been demonstrated effective for one-class classification and anomaly detection. One major family is by predicting geometric transformations [20, 24, 4], such as rotation, translation or flips. The other family includes variants of contrastive learning with geometric augmentations [54, 52]. However, the success has been limited to semantic anomaly detection benchmarks, such as CIFAR-10 [28] or ImageNet [17], and as we show in Section 4.1, methods relying on geometric transformations perform poorly on defect detection benchmarks.

Because of practical applications, such as industrial inspection or medical diagnosis, defect detection [9, 5] has received lots of attention. The initial steps have been taken with methods including autoencoding [9, 7, 25, 59], generative adversarial networks [48, 3], using pretrained models on ImageNet [38, 45, 6, 14, 43, 44], and self-supervised learning by solving different proxy tasks with augmentations [61, 47, 57, 15]. The proposed CutPaste prediction

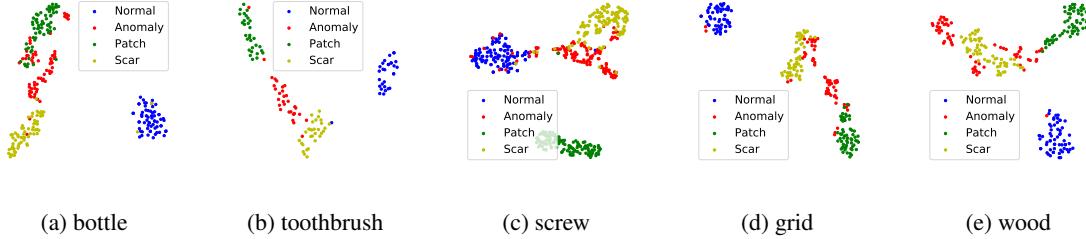


图3：采用三向CutPaste预测任务训练所得模型表征的t-SNE可视化。我们绘制了正常样本（蓝色）、异常样本（红色）以及通过CutPaste（“拼贴”绿色）和CutPaste-scar（“疤痕”黄色）增强的正常样本的嵌入表示。

来自同一领域），这对模型学习找到这个*irregularity*更具挑战性。

另一方面，CutPaste方法确实与某些真实缺陷相似。这就引出一个自然的问题：CutPaste的成功是否源于对真实缺陷的良好模拟？在图3中，我们展示了训练模型表征的t-SNE分布图。可以明显看出，CutPaste样本与真实缺陷样本（异常）几乎不存在重叠区域，但学到的表征能够有效区分正常样本、不同的CutPaste增强样本以及真实缺陷。这表明：(1) CutPaste仍非对真实缺陷的完美模拟；(2) 基于该方法学习不规则性，能够对未见过的异常样本表现出良好的泛化能力。

2.3. 异常分数计算

通过单类别分类器计算异常分数存在多种方法。在本工作中，我们基于表征向量 f 构建生成式分类器，例如核密度估计器[52]或高斯密度估计器[43]。下文将阐述如何计算异常分数及其权衡关系。

尽管非参数KDE无需分布假设，但需要大量样本才能进行精确估计[58]，且计算成本较高。由于缺陷检测可用的正常训练样本有限，我们采用一种简单的参数化高斯密度估计器（GDE），其对数密度计算如下：

$$\log p_{\text{gde}}(x) \propto \left\{ -\frac{1}{2}(f(x) - \mu)^\top \Sigma^{-1} (f(x) - \mu) \right\} \quad (2)$$

其中 μ 和 Σ 是从正常训练数据中学习得到的。¹

2.4. 基于局部块表示的位置识别

尽管我们提出了一种学习图像整体表示的方法，但若要在图像级检测之外定位缺陷区域[38, 6, 61]，学习图像块的表示会是更优选择。通过从图像中学习并提取表示

¹We note that a mixture of Gaussian, which is a middle ground between KDE and GDE, can also be used for more expressive density modeling. We do not observe significant performance gain empirically.

通过图像块，我们可以构建一个异常检测器，能够计算图像块的得分，进而用于定位缺陷区域。

CutPaste预测方法可直接用于学习补丁表示——在训练时我们只需在应用CutPaste增强前裁剪出图像补丁。类似于公式(1)，训练目标可表示为：

$$\mathbb{E}_{x \in \mathcal{X}} \{ \mathbb{C}\mathbb{E}(g(c(x)), 0) + \mathbb{C}\mathbb{E}(g(\text{CP}(c(x))), 1) \} \quad (3)$$

其中 $c(x)$ 在 x 的随机位置裁剪图像块。测试时，我们以给定步幅提取所有图像块的嵌入向量。针对每个图像块，我们计算其异常分数并使用高斯平滑将分数传播至每个像素[32]。在第4.2节中，我们使用基于图像块的检测器生成缺陷定位热力图，同时配合使用GradCAM[51]等视觉解释技术的图像级检测器进行可视化对比。

3. 相关工作

在单类别分类设置下的异常检测，即假设训练期间仅给定正常数据的情况，已得到广泛研究[49, 56, 46, 6, 3, 66, 13, 42, 36, 27]。自监督学习在计算机视觉领域的最新成果[39, 19, 8, 60, 40, 12, 21]也被证明对单类别分类和异常检测有效。其中主要方法体系之一是通过预测几何变换（如旋转、平移或翻转）来实现[20, 24, 4]；另一体系则包含结合几何增强的对比学习变体[54, 52]。然而这些成功目前主要局限于语义异常检测基准（如CIFAR-10 [28]或ImageNet [17]），正如我们在第4.1节所示，依赖几何变换的方法在缺陷检测基准上表现不佳。

由于工业检测或医疗诊断等实际应用的需求，缺陷检测[9, 5]受到了广泛关注。早期研究通过自编码[9, 7, 25, 59]、生成对抗网络[48, 3]、使用ImageNet预训练模型[38, 45, 6, 14, 43, 44]以及通过数据增强解决不同代理任务的自监督学习[61, 47, 57, 15]等方法迈出了重要步伐。所提出的CutPaste预测

task is not only shown to have strong performance on defect detection, but also amenable to combine with existing methods, such as transfer learning from pretrained models for better performance or patch-based models for more accurate localization, which we demonstrate in Section 4.

3.1. Relation to Other Augmentations

Although Cutout [18] and RandomErasing [65] are similar to CutPaste, they create irregularities by a small rectangular region filled with either zero or uniformly sampled pixel values instead of a structural image patch as CutPaste. Moreover, unlike typical use of augmentations for learning invariant representations, we learn a representation that is *discriminative* to these augmentations.

Scar augmentation [16] (Figure 2(d)) is a special case of Cutout, which uses a long-thin rectangle with random colors. While it demonstrates strong performance, we show that CutPaste with the same scale (Figure 2(f)), which fills a long-thin rectangle by a patch from the same image, improves upon representations trained by predicting Cutout.

CutMix [62], which extracts a rectangular image patch from an image and pastes at random location of another image, is related to CutPaste in terms of pasting operations. One main difference is CutMix leverages existing image labels with MixUp [64] in the objective while CutPaste prediction is a self-supervised learning without the need of image labels. The other difference is CutMix studies standard supervised tasks, while we aim for one-class classification.

[11] presents a denoising autoencoder with patch-swap augmentation as noise process. [26] proposes to learn representations by predicting local augmentations using GAN. Our method is simpler (e.g., no need to train decoder or GAN) while highly performant, thus more practical.

4. Experiments

We conduct most experiments on MVTec Anomaly Detection dataset [5] that contains 10 object and 5 texture categories for anomaly detection. The dataset is composed of normal images for training and both normal and anomaly images with various types of defect for testing. It also provides pixel-level annotations for defective test images. The dataset is relatively small scale in number of images, where the number of training images varies from 60 to 391, posing a unique challenge for learning deep representations.

We follow one-class classification protocol, also known as semi-supervised anomaly detection [10],² where we train a one-class classifier for each category on its respective normal training examples. Following [52], we learn the representations by augmentation prediction from scratch with

²While previous works [5, 6] have used *unsupervised* to describe their settings, it could be misleading as training data is curated to include normal data only.

ResNet-18 [22] plus an MLP projection head on top of average pooling layer followed by the last linear layer. We construct a Gaussian density estimation (GDE) as Equation (2) for anomaly detector based on the top pooled features.

We train a model on 256×256 image. We note that the same training strategy, such as the selection of hyperparameters or data augmentations, is applied to all categories. Detailed settings of training can be found in Appendix A.

4.1. Main Results

We report the anomaly detection performance in Table 1. We run experiments 5 times with different random seeds and report the mean AUC and standard error for each category. We also report the average of mean and standard errors for texture, object, and all categories.

We test representations trained with different proxy tasks of self-supervised learning, including baselines such as rotation [20], Cutout or scar predictions, the proposed CutPaste, CutPaste-Scar predictions, and using both with 3-way classification. We also compare with previous works, including deep one-class classifier (DOCC) [45], uninformed student [5], and patch SVDD [61]. We note that some of these methods use ImageNet pretrained model for transfer learning, either by fine-tuning (DOCC) or distillation (uninformed student). The results are in Table 1.

Rotation prediction is demonstrated to be powerful in semantic anomaly detection [52]. However, it results in unsatisfactory 73.1 AUC in defect detection compared with the Scar prediction (85.0), a Cutout variant. Some failure of rotation prediction is due to the unaligned objects, such as screw shown in Figure 2. For aligned objects, although it performs well on toothbrush, it is sub-optimal on capsule. Detailed ablation study of Cutout variants can be found in Section 5.

CutPaste and CutPaste-Scar, which improve Cutout and Scar prediction by avoiding potential naive solutions, outperform other augmentation predictions with 90.9 and 93.5 AUCs, respectively. With a finer-grained 3-way classification to leverage different scale of CutPaste, we achieve the best 95.2 AUC, which surpasses existing works on learning from scratch, such as P-SVDD [61] (92.1 AUC). The proposed data-driven approach via CutPaste is also better than existing works leveraging pretrained networks, including DOCC [45] (87.9 AUC) with pretrained VGG16 and Uninformed Student [6] (92.5 AUC) with pretrained ResNet18. Last, we further improve the AUC to **96.1** by ensembling anomaly scores from 5 CutPaste (3-way) models.

4.2. Defect Localization

We conduct anomaly localization experiments using our representations trained with 3-way classification task. One challenge to accurate localization of defect is that it is difficult to use a heatmap-style approach for localization as our

该任务不仅在缺陷检测方面表现出强大的性能，而且易于与现有方法结合，例如通过预训练模型进行迁移学习以提升性能，或采用基于补丁的模型实现更精准的定位，我们将在第4节对此进行演示。

3.1. 与其他增强方法的关系

尽管Cutout [18]和RandomErasing [65]与CutPaste类似，但它们是通过填充零或均匀采样像素值的小矩形区域来制造不规则性，而非像CutPaste那样使用结构化图像块。此外，与典型通过增强学习不变表示的方法不同，我们学习的表示对这些增强方法具有*discriminative*特性。

疤痕增强[16]（图2(d)）是Cutout的一种特殊形式，它采用随机颜色的细长矩形。虽然该方法表现出强大性能，但我们证明采用相同尺度的CutPaste（图2(f)）——通过从同图像中提取的补丁填充细长矩形——能够优化基于Cutout预测训练得到的表征。

CutMix [62]从一张图像中提取矩形图像块并随机粘贴到另一张图像的随机位置，其在粘贴操作方面与Cut Paste相关。一个主要区别是CutMix在目标函数中结合了MixUp [64]并利用现有图像标签，而CutPaste预测是一种无需图像标签的自监督学习。另一区别在于CutMix研究标准监督任务，而我们的目标是单类别分类。

[11]提出了一种采用补丁交换增强作为噪声过程的去噪自编码器。[26]提出通过使用GAN预测局部增强来学习表示。我们的方法更简单（例如无需训练解码器或GAN）且性能优异，因此更具实用性。

4. 实验

我们在MVTec异常检测数据集[5]上进行了大部分实验，该数据集包含10个物体类别和5个纹理类别用于异常检测。数据集由用于训练的正常图像，以及同时包含正常图像和具有各类缺陷的异常图像的测试集组成。该数据集还提供了缺陷测试图像的像素级标注。该数据集的图像数量规模相对较小，训练图像数量从60到391张不等，这为学习深度表征带来了独特挑战。

我们遵循单类分类协议（也称为半监督异常检测[10]），²即为每个类别在其各自的正常训练样本上训练单类分类器。依照[52]的方法，我们通过数据增强预测从头学习表征，

²While previous works [5, 6] have used *unsupervised* to describe their settings, it could be misleading as training data is curated to include normal data only.

ResNet-18 [22]加上一个MLP投影头，置于平均池化层之上，随后是最后一个线性层。我们基于顶部池化后的特征构建高斯密度估计（GDE）作为异常检测器，如公式(2)所示。

我们在 256×256 图像上训练模型。需要注意的是，所有类别均采用相同的训练策略，包括超参数选择和数据增强方法。具体训练设置详见附录A。

4.1. 主要结果

我们在表1中报告了异常检测性能。我们使用不同的随机种子进行了5次实验，并报告了每个类别的平均AUC和标准误差。同时我们还报告了纹理、物体及所有类别的平均值与标准误差的汇总结果。

我们测试了通过自监督学习的不同代理任务训练得到的表示，包括基线方法如旋转[20]、Cutout或疤痕预测、提出的CutPaste、CutPaste-Scar预测，以及使用两者进行的三分类任务。我们还与先前的工作进行了比较，包括深度单类分类器（DOCC）[45]、无指导学生模型[5]和局部SVDD[61]。需要指出的是，部分方法使用了ImageNet预训练模型进行迁移学习，具体方式包括微调（DOCC）或蒸馏（无指导学生模型）。实验结果如表1所示。

旋转预测被证明在语义异常检测中非常有效[52]。然而，在缺陷检测任务中，该方法仅取得73.1的AUC值（作为对比，Cutout变体Scar预测的AUC达到85.0），表现不尽如人意。部分旋转预测失败案例源于未对齐物体，如图2所示的螺丝钉。对于已对齐物体，该方法在牙刷上表现良好，但在胶囊类物体上仍非最优。关于Cutout变体的详细消融研究可见第5节。

CutPaste与CutPaste-Scar技术通过规避潜在的简单解决方案，改进了Cutout和Scar预测，分别以90.9和93.5的AUC值超越其他数据增强预测方法。通过采用更细粒度的三分类策略以利用不同尺度的CutPaste，我们实现了最优的95.2 AUC，这一结果超越了P-SVDD[61]（92.1 AUC）等从零开始学习的现有研究。基于CutPaste的数据驱动方法同样优于依赖预训练网络的现有方案：包括采用VGG16预训练模型的DOCC[45]（87.9 AUC）以及使用ResNet18预训练模型的Uninformed Student[6]（92.5 AUC）。最后，我们通过集成5个CutPaste（三分类）模型的异常分数，将AUC进一步提升至96.1。

4.2. 缺陷定位

我们使用通过三分类任务训练得到的表征进行异常定位实验。准确缺陷定位面临的一个挑战是，由于我们的模型特性，难以采用热力图式方法进行精确定位。

Table 1: Anomaly detection performance on MVTec AD dataset [5]. We report AUCs of representations trained to classify CutPaste, CutPaste (scar), both (3-way), and baseline augmentations such as rotation, Cutout, or scar. For comparison, we report those of deep one-class classifier [45], uninformed student [6] and patch-SVDD [61]. We report mean and standard error tested with 5 random seeds. Lastly, we report the AUC using ensemble of 5 CutPaste (3-way) models. The best performing model and those within standard error are bold-faced.

Category	DOCC [45]	U-Student [6]	P-SVDD [61]	Rotation	Cutout	Scar	CutPaste	CutPaste (scar)	CutPaste (3-way)	Ensemble	
texture	carpet	90.6	95.3	92.9	29.7 ± 1.4	35.3 ± 2.3	92.7 ± 0.4	67.9 ± 1.8	94.6 ± 0.6	93.1 ± 1.1	
	grid	52.4	98.7	94.6	60.5 ± 7.0	57.5 ± 3.0	74.4 ± 2.5	99.9 \pm 0.1	95.5 ± 0.3	99.9 \pm 0.1	
	leather	78.3	93.4	90.9	55.2 ± 1.4	67.7 ± 1.5	99.9 \pm 0.1	99.7 ± 0.1	100.0 \pm 0.0	100.0 \pm 0.0	
	tile	96.5	95.8	97.8	70.1 ± 1.9	71.8 ± 4.0	96.7 \pm 0.9	95.9 \pm 1.0	89.4 ± 2.8	93.4 ± 1.0	
	wood	91.6	95.5	96.5	95.8 ± 1.1	92.0 ± 0.8	98.9 \pm 0.2	94.9 ± 0.5	98.7 \pm 0.3	98.6 \pm 0.5	
		average	81.9	95.7	94.5	$\parallel 62.3 \pm 2.6$	64.9 ± 2.3	92.5 ± 0.8	91.7 ± 0.7	95.7 ± 0.8	
<hr/>											
object	bottle	99.6	96.7	98.6	95.0 ± 0.7	88.7 ± 0.8	98.5 ± 0.2	99.2 ± 0.2	98.0 ± 0.5	98.3 ± 0.5	
	cable	90.9	82.3	90.3	85.3 ± 0.8	80.2 ± 1.4	78.3 ± 1.7	87.1 ± 0.8	78.8 ± 2.9	80.6 ± 0.5	
	capsule	91.0	92.8	76.7	71.8 ± 1.4	69.5 ± 1.1	82.9 ± 0.7	87.9 ± 0.7	95.3 \pm 0.8	96.2 \pm 0.5	
	hazelnut	95.0	91.4	92.0	83.6 ± 0.8	69.7 ± 1.3	98.9 \pm 0.2	91.3 ± 0.6	96.7 ± 0.4	97.3 ± 0.3	
	metal nut	85.2	94.0	94.0	72.7 ± 0.5	84.6 ± 0.7	86.9 ± 1.5	96.8 ± 0.5	97.9 ± 0.2	99.3 \pm 0.2	
	pill	80.4	86.7	86.1	79.2 ± 1.4	78.7 ± 0.7	82.2 ± 1.4	93.4 \pm 0.9	85.8 ± 1.3	92.4 \pm 1.3	
	screw	86.9	87.4	81.3	35.8 ± 2.9	17.6 ± 4.4	11.3 ± 2.2	54.4 ± 1.7	83.7 ± 0.7	86.3 ± 1.0	
	toothbrush	96.4	98.6	100.0	99.1 ± 0.2	98.1 ± 0.6	94.8 ± 1.0	99.2 ± 0.2	96.7 ± 0.4	98.3 ± 0.9	
	transistor	90.8	83.6	91.5	88.9 ± 0.4	82.5 ± 1.2	92.0 ± 0.7	96.4 \pm 0.7	91.1 ± 0.6	95.5 \pm 0.5	
	zipper	92.4	95.8	97.9	74.3 ± 1.6	75.7 ± 1.0	86.8 ± 0.9	99.4 \pm 0.1	99.5 \pm 0.1	99.4 \pm 0.2	
		average	90.9	90.9	90.8	$\parallel 78.6 \pm 1.1$	74.5 ± 1.3	81.3 ± 1.1	90.5 ± 0.6	92.4 ± 0.8	94.3 \pm 0.6
<hr/>											
		average	87.9	92.5	92.1	$\parallel 73.1 \pm 1.6$	71.3 ± 1.6	85.0 ± 1.0	90.9 ± 0.7	93.5 ± 0.8	95.2 \pm 0.6
<hr/>											

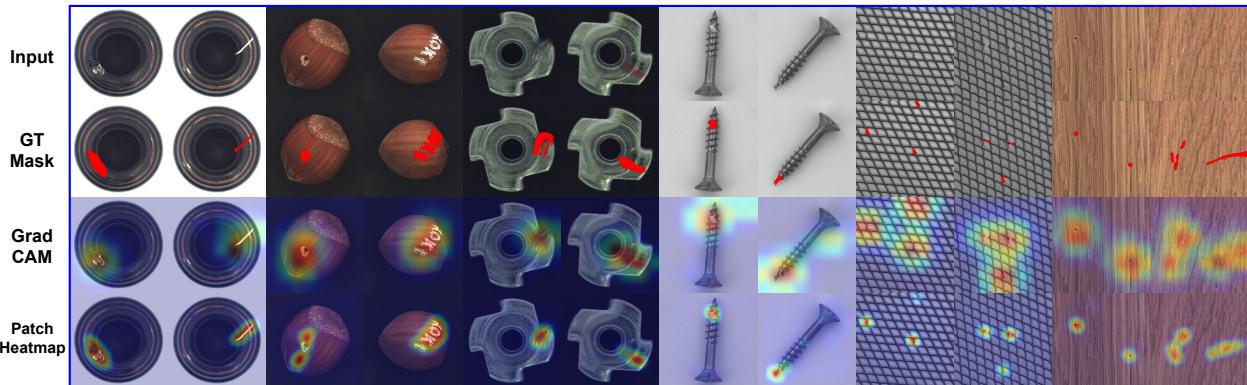


Figure 4: Defect localization on bottle, hazelnut, metal nut, screw, wood and grid classes of MVTec datasets. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector. We provide more examples in Appendix B.

model learns a holistic representation of an image. Instead, we use visual explanation techniques, GradCAM [51], to highlight the area affecting the decision of anomaly detector. We show qualitative results in the second row of Figure 4, which are visually pleasing. We further evaluate the pixel-wise localization AUC, achieving 88.3.

Instead, we learn a representation of an image patch using CutPaste prediction, as in Section 2.4. We train models of 64×64 patches from 256×256 image. At test time, we densely extract anomaly scores with a stride of 4 and propagate the anomaly scores via receptive field upsampling with Gaussian smoothing [32]. We report a localization AUC in Table 2. Our patch-based model achieves **96.0**

AUC. Specifically, our model shows strong performance on texture categories over previous state-of-the-art (96.3 AUC compared to 93.7). We also outperforms the DistAug contrastive learning [52], which only results in 90.4 localization AUC. Finally, we visualize representative samples for localization in Figure 4, showing accurate localization even when defects are tiny. More comprehensive results on defect localization are given in Appendix B.

4.3. Transfer Learning with Pretrained Models

In Section 4.1, we have shown the proposed data-driven approach is better than leveraging pretrained networks, such as DOCC [45] and Uninformed Student [6]. It is consistent

表1：MVTec AD数据集[5]上的异常检测性能。我们报告了经过训练用于分类CutPaste、CutPaste（疤痕）、两者（三分类）以及旋转、Cutout或疤痕等基线数据增强方法的表征AUC值。作为对比，我们同时报告了深度单分类器[45]、无先验信息学生模型[6]和patch-SVDD[61]的结果。所有结果均报告5次随机种子测试的均值及标准误差。最后，我们报告了集成5个CutPaste（三分类）模型的AUC值。最佳性能模型及处于标准误差范围内的结果均以粗体标出。

Category	DOCC [45]	U-Student [6]	P-SVDD [61]	Rotation	Cutout	Scar	CutPaste	CutPaste (scar)	CutPaste (3-way)	Ensemble	
texture	carpet	90.6	95.3	92.9	29.7 ± 1.4	35.3 ± 2.3	92.7 ± 0.4	67.9 ± 1.8	94.6 ± 0.6	93.1 ± 1.1	
	grid	52.4	98.7	94.6	60.5 ± 7.0	57.5 ± 3.0	74.4 ± 2.5	99.9 \pm 0.1	95.5 ± 0.3	99.9 \pm 0.1	
	leather	78.3	93.4	90.9	55.2 ± 1.4	67.7 ± 1.5	99.9 \pm 0.1	99.7 ± 0.1	100.0 \pm 0.0	100.0 \pm 0.0	
	tile	96.5	95.8	97.8	70.1 ± 1.9	71.8 ± 4.0	96.7 \pm 0.9	95.9 \pm 1.0	89.4 ± 2.8	93.4 ± 1.0	
	wood	91.6	95.5	96.5	95.8 ± 1.1	92.0 ± 0.8	98.9 \pm 0.2	94.9 ± 0.5	98.7 \pm 0.3	98.6 \pm 0.5	
		average	81.9	95.7	94.5	62.3 ± 2.6	64.9 ± 2.3	92.5 ± 0.8	91.7 ± 0.7	95.7 ± 0.8	97.0 \pm 0.5
object	bottle	99.6	96.7	98.6	95.0 ± 0.7	88.7 ± 0.8	98.5 ± 0.2	99.2 ± 0.2	98.0 ± 0.5	98.3 ± 0.5	
	cable	90.9	82.3	90.3	85.3 ± 0.8	80.2 ± 1.4	78.3 ± 1.7	87.1 ± 0.8	78.8 ± 2.9	80.6 ± 0.5	
	capsule	91.0	92.8	76.7	71.8 ± 1.4	69.5 ± 1.1	82.9 ± 0.7	87.9 ± 0.7	95.3 \pm 0.8	96.2 \pm 0.5	
	hazelnut	95.0	91.4	92.0	83.6 ± 0.8	69.7 ± 1.3	98.9 \pm 0.2	91.3 ± 0.6	96.7 ± 0.4	97.3 ± 0.3	
	metal nut	85.2	94.0	94.0	72.7 ± 0.5	84.6 ± 0.7	86.9 ± 1.5	96.8 ± 0.5	97.9 ± 0.2	99.3 \pm 0.2	
	pill	80.4	86.7	86.1	79.2 ± 1.4	78.7 ± 0.7	82.2 ± 1.4	93.4 \pm 0.9	85.8 ± 1.3	92.4 \pm 1.3	
	screw	86.9	87.4	81.3	35.8 ± 2.9	17.6 ± 4.4	11.3 ± 2.2	54.4 ± 1.7	83.7 ± 0.7	86.3 ± 1.0	
	toothbrush	96.4	98.6	100.0	99.1 ± 0.2	98.1 ± 0.6	94.8 ± 1.0	99.2 ± 0.2	96.7 ± 0.4	98.3 ± 0.9	
	transistor	90.8	83.6	91.5	88.9 ± 0.4	82.5 ± 1.2	92.0 ± 0.7	96.4 \pm 0.7	91.1 ± 0.6	95.5 \pm 0.5	
	zipper	92.4	95.8	97.9	74.3 ± 1.6	75.7 ± 1.0	86.8 ± 0.9	99.4 \pm 0.1	99.5 \pm 0.1	99.4 \pm 0.2	
		average	90.9	90.9	90.8	78.6 ± 1.1	74.5 ± 1.3	81.3 ± 1.1	90.5 ± 0.6	92.4 ± 0.8	94.3 \pm 0.6
		average	87.9	92.5	92.1	73.1 ± 1.6	71.3 ± 1.6	85.0 ± 1.0	90.9 ± 0.7	93.5 ± 0.8	95.2 \pm 0.6

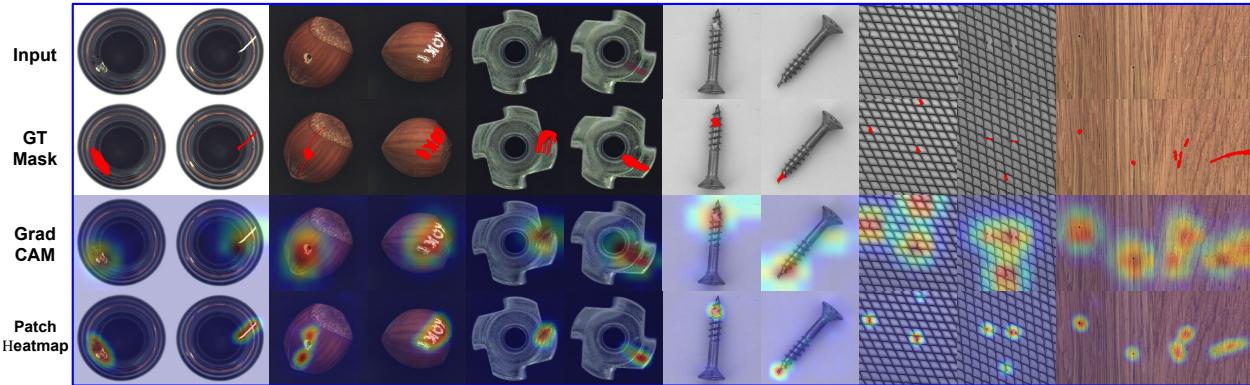


图4：MVTec数据集中瓶装品、榛果、金属螺母、螺丝、木材及网格类别的缺陷定位结果。自上而下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果，以及使用块级检测器的热力图。更多示例见附录B。

模型学习图像的整体表示。相反，我们使用视觉解释技术GradCAM [51]来突出影响异常检测器决策的区域。我们在图4第二行展示了定性结果，这些结果在视觉上令人满意。我们进一步评估了像素级定位AUC，达到了88.3。

相反，我们使用CutPaste预测来学习图像块的表征，如第2.4节所述。我们基于 256×256 图像中的 64×64 图像块训练模型。测试时，我们以步长4密集提取异常分数，并通过高斯平滑[32]进行感受野上采样来传播异常分数。表2中报告了定位AUC结果。我们基于图像块的模型实现了**96.0**

AUC。具体而言，我们的模型在纹理类别上展现出优于以往先进方法的强劲性能（96.3 AUC对比93.7）。我们同样超越了仅获得90.4定位AUC的DistAug对学习[52]。最后，我们在图4中可视化了具有代表性的定位样本，证明即使缺陷极其微小也能实现精准定位。更多关于缺陷定位的完整结果详见附录B。

4.3. 基于预训练模型的迁移学习

在第4.1节中，我们已经证明所提出的数据驱动方法优于利用预训练网络的方法，例如DOCC [45]和Uninformed Student [6]。这一结论具有一致性

Table 2: Pixel-wise localization AUC on MVTec dataset. The best models within standard error are bold-faced.

Category		FCDD [32]	P-SVDD [61]	CutPaste (3-way)
texture	carpet	96	92.6	98.3 \pm 0.0
	grid	91	96.2	97.5 \pm 0.1
	leather	98	97.4	99.5 \pm 0.0
	tile	91	91.4	90.5 \pm 0.2
	wood	88	90.8	95.5 \pm 0.1
average		93	93.7	96.3 \pm 0.1
object	bottle	97	98.1	97.6 \pm 0.1
	cable	90	96.8	90.0 \pm 0.2
	capsule	93	95.8	97.4 \pm 0.1
	hazelnut	95	97.5	97.3 \pm 0.1
	metal nut	94	98.0	93.1 \pm 0.4
	pill	81	95.1	95.7 \pm 0.1
	screw	86	95.7	96.7 \pm 0.1
	toothbrush	94	98.1	98.1 \pm 0.0
	transistor	88	97.0	93.0 \pm 0.2
	zipper	92	95.1	99.3 \pm 0.0
average		91	96.7	95.8 \pm 0.1
average		92	95.7	96.0 \pm 0.1

Table 3: Detection performance on MVTec dataset using representations of EfficientNet (B4) [55] pretrained on ImageNet [17] and fine-tuned by the CutPaste (3-way). The number is bold when it is better than its pretrained or fine-tuned counterpart under the same feature (pool v.s. level-7).

Category	Pool		Level-7		
	Pretrain	Finetune	Pretrain	Finetune	
texture	carpet	98.3	100.0 \pm 0.0	97.6	100.0 \pm 0.0
	grid	96.4	98.8 \pm 0.1	98.2	99.1 \pm 0.0
	leather	100.0	100.0 \pm 0.0	100.0	100.0 \pm 0.0
	tile	99.9	98.9 \pm 0.2	99.9	99.8 \pm 0.2
	wood	99.7	99.8 \pm 0.0	99.6	99.8 \pm 0.0
average		98.9	99.5 \pm 0.0	99	99.7 \pm 0.0
object	bottle	99.8	100.0 \pm 0.0	100.0	100.0 \pm 0.0
	cable	91.2	93.9 \pm 0.1	96.5	96.2 \pm 0.3
	capsule	93	94.3 \pm 0.3	94.7	95.4 \pm 0.1
	hazelnut	96.6	99.7 \pm 0.0	100.0	99.9 \pm 0.0
	metalfnut	94.3	98.7 \pm 0.1	97.7	98.6 \pm 0.0
	pill	81.9	91.3 \pm 0.2	91	93.3 \pm 0.2
	screw	86.3	86 \pm 0.1	92.0	86.6 \pm 0.2
	toothbrush	89.3	92.8 \pm 0.2	90.3	90.7 \pm 0.1
	transistor	94.6	95.6 \pm 0.2	97.5	97.5 \pm 0.2
	zipper	95.6	99.9 \pm 0.0	97	99.9 \pm 0.1
average		92.3	95.2 \pm 0.1	95.7	95.8 \pm 0.1
average		94.5	96.6 \pm 0.1	96.8	97.1 \pm 0.0

with the prior study on semantic anomaly detection [52]. On the other hand, pretrained EfficientNet [55] is found useful for defect detection [43]. As shown in Table 3, without fine-tuning, the representation from the pretrained EfficientNet (B4) results in 94.5 AUC, which is competitive with the proposed CutPaste prediction (95.2 from Table 1).

Here we demonstrate that the proposed self-supervised learning via CutPaste is versatile, which can also be used to improve the pretrained networks to better adapt to the data. We use pretrained EfficientNet (B4) as a backbone, and follow the standard fine-tuning steps to train with the same

Table 4: Detection AUCs of representations trained to predict Cutout, with mean pixel value, with random color, Confetti noise [32], or the proposed CutPaste.

Category	Cutout	Cutout (Mean)	Cutout (Color)	Confetti	CutPaste
texture	64.9 \pm 2.3	65.5 \pm 1.8	70.5 \pm 2.2	80.1 \pm 2.3	91.7 \pm 0.7
object	74.5 \pm 1.3	78.1 \pm 1.1	78.9 \pm 1.0	76.7 \pm 0.3	90.5 \pm 0.6
all	71.3 \pm 1.7	73.9 \pm 1.3	76.1 \pm 1.4	77.8 \pm 1.5	90.9 \pm 0.7

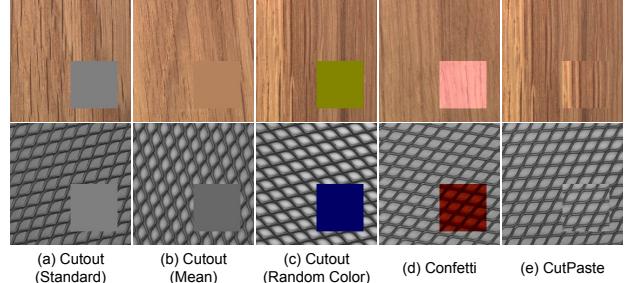


Figure 5: Visual comparison between the proposed Cut-Paste and Cutout variants, including filling with grey color, mean pixel values, random colors and Confetti noise [32].

CutPaste prediction (3-way) task. Detailed settings can be found in Appendix A. We show the results in Table 3. After fine-tuning via CutPaste, we achieve the **new state-of-the-art 96.6 AUC**. Furthermore, CutPaste prediction is a general and useful strategy to adapt to the data for most of the situations. For example, CutPaste improves by a large margin on class `pill` (81.9 → 91.3). For many nearly perfect situations, such as `bottle`, CutPaste is still able to improve by a small margin. Last, as suggested by [30, 43], we investigate the performance of various deep features. We find that level-7 feature shows the best performance, and we further improve the level-7 feature of EfficientNet from 96.8 (pretrained) to **97.1** \pm 0.0 with CutPaste.

5. Ablation Study

We conduct various additional studies to provide deeper insights of the proposed CutPaste. We first compare CutPaste with different Cutout variants in addition to the standard ones reported in Section 4.1. Second, we showcase the representation learned via predicting CutPaste generalizes well to more crafted unseen defects. Last, we compare with the semantic anomaly detection.

5.1. From Cutout to CutPaste

We evaluate the performance of representations trained to predict variants of Cutout augmentations whose areas are filled by grey color (standard), mean pixel values, random color, or image patch from different location, i.e., CutPaste. We also test Confetti noise [32] that jitters a color of a local patch. We show samples from considered augmentations in Figure 5 and report the detection AUCs in Table 4. While

表2：MVTec数据集上的像素级定位AUC。最优及在标准误差范围内的模型已加粗标示。

Category		FCDD [32]	P-SVDD [61]	CutPaste (3-way)
texture	carpet	96	92.6	98.3 \pm 0.0
	grid	91	96.2	97.5 \pm 0.1
	leather	98	97.4	99.5 \pm 0.0
	tile	91	91.4	90.5 \pm 0.2
	wood	88	90.8	95.5 \pm 0.1
average		93	93.7	96.3 \pm 0.1
object	bottle	97	98.1	97.6 \pm 0.1
	cable	90	96.8	90.0 \pm 0.2
	capsule	93	95.8	97.4 \pm 0.1
	hazelnut	95	97.5	97.3 \pm 0.1
	metal nut	94	98.0	93.1 \pm 0.4
	pill	81	95.1	95.7 \pm 0.1
	screw	86	95.7	96.7 \pm 0.1
	toothbrush	94	98.1	98.1 \pm 0.0
	transistor	88	97.0	93.0 \pm 0.2
	zipper	92	95.1	99.3 \pm 0.0
average		91	96.7	95.8 \pm 0.1
average		92	95.7	96.0 \pm 0.1

表3：采用ImageNet[17]预训练并经CutPaste（三向）微调的EfficientNet (B4) [55]表征在MVTec数据集上的检测性能。当同一特征（池化与层级7）下数值优于其预训练或微调对应项时，该数值以粗体标示。

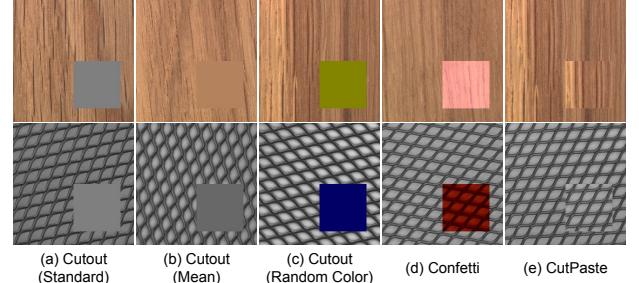
Category	Pool		Level-7		
	Pretrain	Finetune	Pretrain	Finetune	
texture	carpet	98.3	100.0 \pm 0.0	97.6	100.0 \pm 0.0
	grid	96.4	98.8 \pm 0.1	98.2	99.1 \pm 0.0
	leather	100.0	100.0 \pm 0.0	100.0	100.0 \pm 0.0
	tile	99.9	98.9 \pm 0.2	99.9	99.8 \pm 0.2
	wood	99.7	99.8 \pm 0.0	99.6	99.8 \pm 0.0
average		98.9	99.5 \pm 0.0	99	99.7 \pm 0.0
object	bottle	99.8	100.0 \pm 0.0	100.0	100.0 \pm 0.0
	cable	91.2	93.9 \pm 0.1	96.5	96.2 \pm 0.3
	capsule	93	94.3 \pm 0.3	94.7	95.4 \pm 0.1
	hazelnut	96.6	99.7 \pm 0.0	100.0	99.9 \pm 0.0
	metalfnut	94.3	98.7 \pm 0.1	97.7	98.6 \pm 0.0
	pill	81.9	91.3 \pm 0.2	91	93.3 \pm 0.2
	screw	86.3	86 \pm 0.1	92.0	86.6 \pm 0.2
	toothbrush	89.3	92.8 \pm 0.2	90.3	90.7 \pm 0.1
	transistor	94.6	95.6 \pm 0.2	97.5	97.5 \pm 0.2
	zipper	95.6	99.9 \pm 0.0	97	99.9 \pm 0.1
average		92.3	95.2 \pm 0.1	95.7	95.8 \pm 0.1
average		94.5	96.6 \pm 0.1	96.8	97.1 \pm 0.0

与先前关于语义异常检测的研究[52]一致。另一方面，预训练的EfficientNet[55]被证明适用于缺陷检测[43]。如表3所示，在不进行微调的情况下，预训练EfficientNet (B4) 的表征达到了94.5的AUC值，这与提出的CutPaste预测方法（表1中的95.2）具有可比性。

我们在此证明，所提出的基于CutPaste的自监督学习方法具有通用性，它同样可用于改进预训练网络以更好地适应数据。我们以预训练的EfficientNet (B4) 为骨干网络，遵循标准微调步骤进行训练，并保持

表4：用于预测Cutout、均值像素、随机色彩、Confetti噪声[32]或所提CutPaste方法所训练表征的检测AUC值

Category	Cutout	Cutout (Mean)	Cutout (Color)	Confetti	CutPaste
texture	64.9 ± 2.3	65.5 ± 1.8	70.5 ± 2.2	80.1 ± 2.3	91.7 ± 0.7
object	74.5 ± 1.3	78.1 ± 1.1	78.9 ± 1.0	76.7 ± 0.3	90.5 ± 0.6
all	71.3 ± 1.7	73.9 ± 1.3	76.1 ± 1.4	77.8 ± 1.5	90.9 ± 0.7



(a) Cutout (Standard)
(b) Cutout (Mean)
(c) Cutout (Random Color)
(d) Confetti
(e) CutPaste

图5：本文提出的剪切粘贴法与Cutout变体之间的视觉对比，包括灰色填充、平均像素值填充、随机颜色填充以及五彩纸屑噪声填充[32]的对比效果。

CutPaste预测（三分类）任务。详细设置见附录A。我们在表3中展示了结果。通过CutPaste微调后，我们取得了全新的最优性能**96.6** AUC。此外，CutPaste预测是一种通用且有效的策略，能适应大多数场景下的数据特性。例如在药片类别上，CutPaste大幅提升了性能（ $81.9 \rightarrow 91.3$ ）。对于诸如瓶子这类近乎完美的场景，CutPaste仍能实现小幅提升。最后，根据[30,43]的建议，我们研究了不同深度特征的性能。发现第7层特征表现最佳，并将EfficientNet的第7层特征从预训练的96.8提升至CutPaste优化后的**97.1** \pm 0.0。

5. 消融实验

我们进行了各种额外的研究，以更深入地了解所提出的CutPaste方法。首先，除了第4.1节中报告的标准方法外，我们还比较了CutPaste与不同的Cutout变体。其次，我们展示了通过预测CutPaste学到的表示能够很好地推广到更复杂的未见过的缺陷上。最后，我们与语义异常检测进行了比较。

5.1. 从Cutout到CutPaste

我们评估了为预测Cutout增强变体而训练的表示性能，这些增强区域分别用灰色（标准）、平均像素值、随机颜色或来自不同位置的图像块（即CutPaste）进行填充。同时测试了Confetti噪声[32]，该方法会扰动局部图像块的颜色。图5展示了所采用增强方法的样本示例，表4则报告了检测AUC值。尽管

Table 5: Detection AUCs of representations trained with binary classification between normal and the union of CutPaste and CutPaste-Scar examples and 3-way classification among normal, CutPaste and CutPaste-scar examples.

Category	CutPaste	CutPaste (scar)	Binary (Union)	3-Way
texture	91.7 \pm 0.7	95.7 \pm 0.8	97.3 \pm 0.3	97.0 \pm 0.5
object	90.5 \pm 0.6	92.4 \pm 0.8	92.8 \pm 0.5	94.3 \pm 0.6
all	90.9 \pm 0.7	93.5 \pm 0.8	94.3 \pm 0.5	95.2 \pm 0.6

achieving 71.3 AUC that already is significantly better than random guessing, predicting a standard Cutout augmentation is still a simple task and the network may have learned a naive solution from easy proxy task, as discussed in Section 2. By gradually increasing the difficulty of proxy task to avoid known trivial solutions with random color to the patch, or with structures similar to local patterns of the normal data (Confetti noise, CutPaste), the network learns to find irregularity and generalizes better to detect real defects.

5.2. Binary v.s. Finer-Grained Classification

In Table 1, although CutPaste-scar shows better performance on average than CutPaste, there is no clear winner that works the best for all. As there are diverse types of defect in practice, we leverage the strength of both augmentations for representation learning. In Section 2.2, we train a model by solving a 3-way classification task between normal, CutPaste and CutPaste-scar. Alternatively, we train to solve a binary classification task by discriminating normal examples and the union of two augmentations.

The results, along with those of representations trained with CutPaste and CutPaste-scar, are in Table 5. It is clear that using both augmentations improve the performance. Between binary with union of augmentations and 3-way, we observe better detection performance with representations trained by 3-way classification task. A plausible hypothesis on the superiority of 3-way formulation in our case is that it is more natural to model CutPaste and CutPaste-scar augmentations separately than together as there exists a systematic difference between them in the size, shape, and rotation angle of patches.

5.3. CutPaste on Synthetic Anomaly Detection

We further study the generalization of our models to unseen anomalies. Specifically, we test on synthetic anomaly datasets created by patching diverse shape masks to normal data, such as digits [29], square, ellipse, or heart [35], filled with random color or natural images. Samples of synthetic anomalies are shown in Figure 6 and detection results are in Table 6. We first note that these datasets are not trivial – a model trained by predicting Cutout augmentations achieves only 81.5. Our proposed CutPaste (3-way) model performs well on synthetic dataset, achieving 98.3 AUC on average. We highlight that some shapes (e.g., ellipse, heart) or color



Figure 6: Synthetic defects on pill class. From left to right, we use MNIST [29], square, ellipse, heart [35] with random color, and those filled with natural image patches.

Table 6: Detection AUCs on synthetic data. Various shapes, such as digit, square, ellipse, or heart, are patched to normal images with random color or natural images (\dagger).

Dataset	MNIST	Square	Ellipse	Heart	Square \dagger	Ellipse \dagger	Heart \dagger	Avg
Cutout	52.3	90.6	89.3	87.5	86.4	84.0	80.7	81.5
CutPaste	96.1	98.4	98.2	97.9	99.3	99.2	99.0	98.3

statistics inside the patch (e.g., constant color, natural images) are not seen at training, but we can still generalize to these unseen cases.

5.4. Application to Semantic Outlier Detection

We also conduct the semantic anomaly detection experiment on CIFAR-10 [28] following the protocol in [20, 52], where a single class is treated as normal and remaining 9 classes are anomalies. We make a comparison of Cutout, CutPaste and rotation prediction [52]. Cutout results in 60.2 AUC, and CutPaste achieves 69.4 AUC, which significantly improves upon Cutout (60.2). However, these are still far behind that of rotation prediction (91.3 AUC) on CIFAR-10 semantic anomaly detection. On the other hand, in Section 4.1, we have discussed the reversed situation that rotation prediction is much worse than 3-way CutPaste prediction. The results suggest the difference between semantic anomaly detection and defect detection, which needs different algorithm and augmentation designs.

6. Conclusion

We propose a data-driven approach for defect detection and localization. The key to our success is self-supervised learning of representations with CutPaste, a simple yet effective augmentation that encourages the model to find local irregularities. We show superior image-level anomaly detection performance on the real-world dataset. Furthermore, by learning and extracting patch-level representations, we demonstrate state-of-the-art pixel-wise anomaly localization performance. We envision the CutPaste augmentation could be a cornerstone for building a powerful model for semi-supervised and unsupervised defect detection.

Acknowledgment. We thank Yang Feng for sharing the implementation of uninformed student and Sercan Arik for the proofread of our manuscript.

表5：使用正常样本与CutPaste及CutPaste-Scar合并样本间的二元分类、以及正常样本/CutPaste样本/CutPaste-Scar样本间的三元分类所训练表征的检测AUC值。

Category	CutPaste	CutPaste (scar)	Binary (Union)	3-Way
texture	91.7 \pm 0.7	95.7 \pm 0.8	97.3 \pm 0.3	97.0 \pm 0.5
object	90.5 \pm 0.6	92.4 \pm 0.8	92.8 \pm 0.5	94.3 \pm 0.6
all	90.9 \pm 0.7	93.5 \pm 0.8	94.3 \pm 0.5	95.2 \pm 0.6

在达到71.3 AUC（该指标已显著优于随机猜测）的同时，预测标准Cutout增强仍是一项简单任务——如第2节所述，网络可能已从简易代理任务中习得原始解决方案。通过逐步提升代理任务难度以避免已知的琐碎解决方案（如对图像块使用随机着色，或采用与正常数据局部模式相似的结构化噪声：Confetti噪声、CutPaste等），网络能逐渐掌握不规则性检测，从而提升真实缺陷检测的泛化能力。

5.2. 二元与细粒度分类对比

在表1中，虽然CutPaste-scar平均表现优于CutPaste，但并未出现适用于所有情况的最佳方案。由于实践中存在多样化的缺陷类型，我们综合运用两种数据增强方法的优势进行表征学习。在2.2节中，我们通过训练模型完成正常样本、CutPaste样本与CutPaste-scar样本的三分类任务；或者通过区分正常样本与两种增强样本的并集，训练模型解决二分类任务。

结果以及使用CutPaste和CutPaste-scar训练的表征均展示在表5中。可以明显看出，同时使用两种数据增强方法能够提升性能。在采用数据增强并集的二分类与三分支分类之间，我们观察到通过三分支分类任务训练的表征具有更好的检测性能。关于三分支设定在本研究中更具优势的合理假设是：由于CutPaste与CutPaste-scar在补丁尺寸、形状和旋转角度上存在系统性差异，对这两种增强方式分别建模比合并处理更符合其本质特性。

5.3. 合成异常检测中的CutPaste方法

我们进一步研究了模型对未见异常情况的泛化能力。具体而言，我们在合成异常数据集上进行了测试，这些数据集通过将不同形状的掩码（如数字[29]、方形、椭圆或心形[35]）拼接到正常数据上创建，并填充随机颜色或自然图像。合成异常样本如图6所示，检测结果见表6。我们首先注意到这些数据集并非无关紧要——通过预测Cutout增强训练的模型仅达到81.5。我们提出的CutPaste（三向）模型在合成数据集上表现良好，平均达到98.3 AUC。需要强调的是，某些形状（如椭圆、心形）或颜色



图6：药片类别的合成缺陷。从左至右依次为：使用MNIST [29]、随机颜色的正方形、椭圆、心形[35]以及填充自然图像块的对应形状。

表6：合成数据上的检测AUC值。将数字、方形、椭圆或心形等多种形状以随机颜色或自然图像([†])形式嵌入正常图像中。

Dataset	MNIST	Square	Ellipse	Heart	Square [†]	Ellipse [†]	Heart [†]	Avg
Cutout	52.3	90.6	89.3	87.5	86.4	84.0	80.7	81.5
CutPaste	96.1	98.4	98.2	97.9	99.3	99.2	99.0	98.3

补丁内部的统计信息（例如恒定颜色、自然图像）在训练时并未出现，但我们仍能泛化至这些未见情况。

5.4. 在语义异常检测中的应用

我们同样按照[20, 52]中的方案在CIFAR-10[28]上进行了语义异常检测实验——将单个类别视作正常样本，其余9个类别作为异常样本。针对Cutout、CutPaste和旋转预测[52]进行对比，Cutout取得60.2的AUC值，CutPaste达到69.4的AUC值，较Cutout（60.2）实现显著提升。但这两者在CIFAR-10语义异常检测任务上仍远落后于旋转预测的91.3 AUC表现。另一方面，在4.1节我们已讨论过相反情况：旋转预测远逊于三分类CutPaste预测。这些结果揭示了语义异常检测与缺陷检测之间的本质差异，二者需要不同的算法与数据增强设计方案。

6. 结论

我们提出了一种数据驱动的缺陷检测与定位方法。该方法的核心在于通过CutPaste技术实现自监督表征学习——这种简单而有效的增强方法能促使模型发现局部异常。我们在真实数据集上实现了卓越的图像级异常检测性能。此外，通过学习和提取块级表征，我们展示了最先进的像素级异常定位性能。我们预见CutPaste数据增强技术将成为构建半监督和无监督缺陷检测强大模型的基石。

致谢。我们感谢杨峰分享无指导学生的实现代码，并感谢Sercan Arik对我们的手稿进行校对。

References

- [1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *{USENIX}*, 2016. 12
- [2] Amit Adam, Ehud Rivlin, Ilan Shimshoni, and David Reinitz. Robust real-time unusual event detection using multiple fixed-location monitors. *IEEE transactions on pattern analysis and machine intelligence*, 2008. 1
- [3] Samet Akcay, Amir Atappour-Abarghouei, and Toby P Breckon. Gandomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*, 2018. 4
- [4] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. In *ICLR*, 2020. 1, 2, 4
- [5] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad—a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 2019. 1, 2, 3, 4, 5, 6
- [6] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020. 4, 5, 6
- [7] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. *arXiv preprint arXiv:1807.02011*, 2018. 4
- [8] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018. 4
- [9] Diego Carrera, Giacomo Boracchi, Alessandro Foi, and Brendt Wohlberg. Detecting anomalous structures by convolutional sparse models. In *IJCNN*, 2015. 1, 4
- [10] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 2009. 5
- [11] Liang Chen, Paul Bentley, Kensaku Mori, Kazunari Misawa, Michitaka Fujiwara, and Daniel Rueckert. Self-supervised learning for medical image analysis using image context restoration. *Medical image analysis*, 2019. 5
- [12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 2, 4
- [13] Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018. 4
- [14] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 4
- [15] Anne-Sophie Collin and Christophe De Vleeschouwer. Improved anomaly detection by training an autoencoder with skip connections on images corrupted with stain-shaped noise. *arXiv preprint arXiv:2008.12977*, 2020. 4
- [16] daisukelab. Spotting defects! — deep metric learning solution for mvtec anomaly detection dataset. <https://medium.com/Analytics-vidhya/spotting-defects-deep-metric-learning-solution-for-mvtec-anomaly-detection-dataset-c77691beb1eb>. Accessed: 2020-11-12. 3, 5
- [17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 4, 7
- [18] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 2, 5
- [19] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 2, 3, 4
- [20] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NIPS*, 2018. 1, 2, 4, 5, 8
- [21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 4
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5
- [23] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018. 3
- [24] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *NIPS*, 2019. 1, 2, 4
- [25] Chaoqing Huang, Jinkun Cao, Fei Ye, Maosen Li, Ya Zhang, and Cewu Lu. Inverse-transform autoencoder for anomaly detection. *arXiv preprint arXiv:1911.10676*, 2019. 2, 4
- [26] Simon Jenni, Hailin Jin, and Paolo Favaro. Steering self-supervised feature learning beyond local pixel statistics. In *CVPR*, 2020. 5
- [27] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Why normalizing flows fail to detect out-of-distribution data. *arXiv preprint arXiv:2006.08545*, 2020. 4
- [28] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. 4, 8
- [29] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 8
- [30] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NIPS*, 2018. 7
- [31] Yusha Liu, Chun-Liang Li, and Barnabás Póczos. Classifier two sample test for video anomaly detections. In *BMVC*, 2018. 1
- [32] Philipp Liznerski, Lukas Ruff, Robert A Vandermeulen, Billy Joe Franks, Marius Kloft, and Klaus-Robert Müller. Explainable deep one-class classification. *arXiv preprint arXiv:2007.01760*, 2020. 2, 3, 4, 6, 7, 13
- [33] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 12

参考文献

- [1] Martín Abadi, Paul Barham, 陈建民, 赵志峰, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard 等. TensorFlow：面向大规模机器学习的系统。发表于{USENIX}, 2016. 12 [2] Amit Adam, Ehud Rivlin, Ilan S himshoni, Daviv Reinitz. 基于多定点监控器的鲁棒实时异常事件检测. *IEEE transactions on pattern analysis and machine intelligence*, 2008. 1 [3] Samet Akcay, Amir Atapour-Abarghouei, Toby P Breckon. GAN异常检测：通过对抗训练实现半监督异常识别. 发表于ACCV, 2018. 4 [4] Liron Bergman, Yedid Hoshen. 面向通用数据的基于分类的异常检测. 发表于ICLR, 2020. 1,2,4 [5] Paul Bergmann, Michael Fause r, David Sattlegger, Carsten Steger. MVTec AD——面向无监督异常检测的综合性真实场景数据集. 发表于CVPR, 2019. 1,2,3, 4,5,6 [6] Paul Bergmann, Michael Fauser, David Sattlegger, Cars ten Steger. 无先验知识的学生网络：基于判别性潜在嵌入的师生异常检测. 发表于CVPR, 2020. 4,5,6 [7] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, Carsten Steger. 通过结构相似性在自编码器中的应用改进无监督缺陷分割. arXiv preprint arXiv:1807.02011, 2018. 4 [8] Mathilde Caron, Piotr Bojanowski, Armand Joulin, Matthijs Douze. 通过深度聚类实现视觉特征的无监督学习. 发表于ECCV, 2018. 4 [9] Diego Carrera, Giacomo Boracchi, Alessandro Foi, Brendt Wohlberg. 通过卷积稀疏模型检测异常结构. 发表于IJCNN, 2015. 1,4 [10] Varun Chandola, Arindam Banerjee, Vipin Kumar. 异常检测研究综述. ACM computing surveys(CSUR), 2009. 5 [11] Liang C hen, Paul Bentley, Kensaku Mori, 三泽和成, 藤原道孝, Daniel R ueckert. 基于图像上下文重建的医学图像分析自监督学习. Medical image analysis, 2019. 5 [12] 陈霆, Simon Kornblith, M ohammad Norouzi, Geoffrey Hinton. 视觉表征对比学习的简易框架. arXiv preprint arXiv:2002.05709, 2020. 2,4 [13] Hyunsun Choi, Eric Jang, Alexander A Alemi. WAIC及其原理：基于生成集成模型的鲁棒异常检测. arXiv preprint arXiv:1810.01392, 2018. 4 [14] Niv Cohen, Yedid Hoshen. 基于深度金字塔对应关系的子图像异常检测. arXiv preprint arXiv:2005.02357, 2020. 4 [15] Anne-Sophie Collin, Christophe De Vleeschouwer. 通过污迹形噪声图像训练带跳跃连接的自编码器以改进异常检测. arXiv preprint arXiv:2008.12977, 2020. 4 [16] daisukelab. 缺陷检测！——MVTec异常检测的深度度量学习解决方案

数据集。https://medium.com/analytics-vidhya/spotting-defects-deep-metric-learning-solution-for-mvtac-anomaly-detection-dataset-c77691beb1eb。访问日期：2020年11月12日。3, 5 [17] 贾登峰、董伟、理查德·索彻、李立佳、李凯、李飞飞。ImageNet：一个大规模分层图像数据库。发表于CVPR, 2009年。4, 7 [18] 特伦斯·德弗里斯与格雷厄姆·W·泰勒。通过Cutout改进卷积神经网络的正则化。arXiv preprint arXiv:1708.04552, 2017年。2, 5 [19] 斯皮罗斯·吉达里斯、普拉维尔·辛格与尼科斯·科莫达基斯。通过预测图像旋转进行无监督表示学习。发表于ICLR, 2018年。2, 3, 4 [20] 伊扎克·戈兰与兰·埃尔亚尼夫。使用几何变换的深度异常检测。发表于NIPS, 2018年。1, 2, 4, 5, 8 [21] 何恺明、范浩祺、吴育昕、谢赛宁与罗斯·吉尔希克。无监督视觉表示学习的动量对比方法。发表于CVPR, 2020年。4 [22] 何恺明、张祥雨、任少卿与孙剑。深度残差学习在图像识别中的应用。发表于CVPR, 第770-778页, 2016年。5 [23] 丹·亨德里克斯、曼塔斯·马泽卡与托马斯·迪特里希。基于异常暴露的深度异常检测。arXiv preprint arXiv:1812.04606, 2018年。3 [24] 丹·亨德里克斯、曼塔斯·马泽卡、萨拉瓦·卡达夫与宋 Dawn。使用自监督学习提升模型鲁棒性与不确定性。发表于NIPS, 2019年。1, 2, 4 [25] 黄超庆、曹金坤、叶飞、李茂森、张娅与卢策吾。基于逆变换自编码器的异常检测。arXiv preprint arXiv:1911.10676, 2019年。2, 4 [26] 西蒙·詹尼、金海林与保罗·法瓦罗。引导自监督特征学习突破局部像素统计局限。发表于CVPR, 2020年。5 [27] 波利娜·基里琴科、帕维尔·伊兹梅洛夫与安德鲁·戈登·威尔逊。归一化流为何难以检测分布外数据。arXiv preprint arXiv:2006.08545, 2020年。4 [28] 亚历克斯·克里泽夫斯基。基于微小图像的多层特征学习技术报告, 多伦多大学, 2009年。4, 8 [29] 扬·勒昆、Léon Bottou、约书亚·本吉奥与帕特里克·哈夫纳。基于梯度的学习在文档识别中的应用。Proceedings of the IEEE, 1998年。8 [30] 金敏李、基博克李、洪拉克李与申镇宇。检测分布外样本与对抗攻击的简易统一框架。发表于NIPS, 2018年。7 [31] 刘玉沙、李春良与巴纳巴斯·P·茨奥斯。视频异常检测中的分类器双样本检验。发表于BMVC, 2018年。1 [32] 菲利普·利兹内尔斯基、卢卡斯·鲁夫、罗伯特·A·范德默伦、比利·乔·弗兰克斯、马吕斯·克洛夫与克劳斯·罗伯特·M·勒。可解释的深度单分类方法。arXiv preprint arXiv:2007.01760, 2020年。2, 3, 4, 6, 7, 13 [33] 伊利亚·洛什奇洛夫与弗兰克·哈特。带热重启的随机梯度下降法。发表于ICLR, 2017年。12

- [34] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *ICANN*, 2011. 1
- [35] Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. URL <https://github.com/deepmind/dsprites-dataset/>. [Accessed on: 2018-05-08], 2017. 8
- [36] Warren R Morningstar, Cusuh Ham, Andrew G Gallagher, Balaji Lakshminarayanan, Alexander A Alemi, and Joshua V Dillon. Density of states estimation for out-of-distribution detection. *arXiv preprint arXiv:2006.09273*, 2020. 4
- [37] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, and Balaji Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using a test for typicality. *arXiv preprint arXiv:1906.02994*, 2019. 1
- [38] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by cnn-based self-similarity. *Sensors*, 2018. 4
- [39] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 4
- [40] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4
- [41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011. 12
- [42] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In *NIPS*, 2019. 1, 4
- [43] Oliver Rippel, Patrick Mertens, and Dorit Merhof. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. *arXiv preprint arXiv:2005.14140*, 2020. 4, 7
- [44] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but different: Semi-supervised defect detection with normalizing flows. *arXiv preprint arXiv:2008.12577*, 2020. 4
- [45] Lukas Ruff, Jacob R Kauffmann, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *arXiv preprint arXiv:2009.11732*, 2020. 4, 5, 6
- [46] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, 2018. 1, 4
- [47] Mohammadreza Salehi, Ainaz Eftekhar, Niousha Sadjadi, Mohammad Hossein Rohban, and Hamid R Rabiee. Puzzle-ae: Novelty detection in images through solving puzzles. *arXiv preprint arXiv:2008.12959*, 2020. 4
- [48] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *IPMI*, 2017. 1, 4
- [49] Bernhard Schölkopf, Robert C Williamson, Alex J Smola, John Shawe-Taylor, and John C Platt. Support vector method for novelty detection. In *NIPS*, 2000. 1, 4
- [50] Philipp Seeböck, Sebastian Waldstein, Sophie Klimscha, Bianca S Gerendas, René Donner, Thomas Schlegl, Ursula Schmidt-Erfurth, and Georg Langs. Identifying and categorizing anomalies in retinal imaging data. *arXiv preprint arXiv:1612.00686*, 2016. 1
- [51] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 2, 4, 6
- [52] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *ICLR*, 2021. 1, 2, 4, 5, 6, 7, 8
- [53] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *CVPR*, 2018. 1
- [54] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *arXiv preprint arXiv:2007.08176*, 2020. 1, 2, 4
- [55] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 7
- [56] David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 2004. 4
- [57] Tareq Tayeh, Sulaiman Aburakhia, Ryan Myers, and Abdallah Shami. Distance-based anomaly detection for industrial surfaces using triplet networks. *arXiv preprint arXiv:2011.04121*, 2020. 3, 4
- [58] Alexandre B Tsybakov. *Introduction to nonparametric estimation*. Springer Science & Business Media, 2008. 4
- [59] Shashanka Venkataraman, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. *arXiv preprint arXiv:1911.08616*, 2019. 4
- [60] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 2019. 2, 4
- [61] Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. *arXiv preprint arXiv:2006.16067*, 2020. 2, 4, 5, 6, 7
- [62] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 5
- [63] Shuangfei Zhai, Yu Cheng, Weining Lu, and Zhongfei Zhang. Deep structured energy based models for anomaly detection. *arXiv preprint arXiv:1605.07717*, 2016. 4
- [64] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. MixUp: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 5
- [65] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020. 5, 12

[34] Jonathan Masci, Ueli Meier, Dan Ciresan 与 Jürgen Schmidhuber。用于分层特征提取的堆叠卷积自编码器。见*ICANN*, 2011年。1 [35] Loic Matthey, Irina Higgins, Demis Hassabis 与 Alexander Lerchner。dsprites：解耦测试精灵数据集。
URL <https://github.com/deepmind/dsprites-dataset>. [Accessed on: 2018-05-08], 2017年。8 [36] Warren R Morningstar, Cusuh Ham, Andrew G Gallagher, Balaji Lakshminarayanan, Alexander A Alemi 与 Joshua V Dillon。基于态密度估计的分布外检测。arXiv preprint arXiv:2006.09273, 2020年。4 [37] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh 与 Balaji Lakshminarayanan。利用典型性检验检测深度生成模型的分布外输入。arXiv preprint arXiv:1906.02994, 2019年。1 [38] Paolo Napoletano, Flavio Piccoli 与 Raimondo Schettini。基于CNN自相似性的纳米纤维材料异常检测。Sensors, 2018年。4 [39] Mehdi Noroozi 与 Paolo Favaro。通过解决拼图游戏实现视觉表征的无监督学习。见*ECCV*, 2016年。4 [40] Aaron van den Oord, Yazhe Li 与 Oriol Vinyals。基于对比预测编码的表征学习。arXiv preprint arXiv:1807.03748, 2018年。4 [41] F. Pedregosa 等。Scikit-learn: Python中的机器学习工具库。Journal of Machine Learning Research, 2011年。12 [42] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon 与 Balaji Lakshminarayanan。基于似然比的分布外检测。见*NIPS*, 2019年。1,4 [43] Oliver Rippel, Patrick Mertens 与 Dorit Merhof。基于预训练深度特征中正常数据分布建模的异常检测。arXiv preprint arXiv:2005.14140, 2020年。4,7 [44] Marco Rudolph, Bastian Wandt 与 Bodo Rosenhahn。相似但不同：基于标准化流的半监督缺陷检测。arXiv preprint arXiv:2008.12577, 2020年。4 [45] Lukas Ruff 等。深度与浅层异常检测的统一综述。arXiv preprint arXiv:2009.11732, 2020年。4,5,6 [46] Lukas Ruff 等。深度单类分类。见*ICML*, 2018年。1,4 [47] Mohammadreza Salehi 等。Puzzle-AE：通过解谜实现图像新颖性检测。arXiv preprint arXiv:2008.12959, 2020年。4 [48] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth 与 Georg Langs。无监督

使用生成对抗网络进行异常检测以指导标记发现。在*IPMI*, 2017年。1, 4 [49] Bernhard Schölkopf, Robert C Williamson, Alex J Smola, John Shawe-Taylor, 与 John C Platt。支持向量新颖性检测方法。在*NIPS*, 2000年。1, 4 [50] Philipp Seeböck, Sebastian Waldstein, Sophie Klimscha, Bianca S Gerendas, René Donner, Thomas Schlegl, Ursula Schmidt-Erfurth, 与 Georg Langs。视网膜成像数据中异常的识别与分类。
arXiv preprint arXiv:1612.00686, 2016年。1 [51] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, 与 Dhruv Batra。Grad-CAM：通过基于梯度的定位从深度网络获得可视化解释。在*ICCV*, 2017年。2, 4, 6 [52] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, 与 Tomas Pfister。深度单类分类的表征学习与评估。在*ICLR*, 2021年。1, 2, 4, 5, 6, 7, 8 [53] Waqas Sultani, Chen Chen, 与 Mubarak Shah。监控视频中的真实世界异常检测。在*CVPR*, 2018年。1 [54] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, 与 Jinwoo Shin。CSI：基于分布偏移实例对比学习的新颖性检测。arXiv preprint arXiv:2007.08176, 2020年。1, 2, 4 [55] Mingxing Tan 与 Quoc Le。EfficientNet：重新思考卷积神经网络的模型缩放。在*ICML*, 2019年。7 [56] David MJ Tax 与 Robert PW Duin。支持向量数据描述。Machine learning, 2004年。4 [57] Tareq Tayeh, Sulaiman Aburakhia, Ryan Myers, 与 Abdallah Shami。使用三元组网络的工业表面基于距离的异常检测。arXiv preprint arXiv:2011.04121, 2020年。3, 4 [58] Alexandre B Tsybakov。Introduction to nonparametric estimation。Springer Science & Business Media, 2008年。4 [59] S hashanka Venkataraman, Kuan-Chuan Peng, Rajat Vikram Singh, 与 Abhijit Mahalanobis。图像中注意力引导的异常定位。arXiv preprint arXiv:1911.08616, 2019年。4 [60] Mang Ye, Xu Zhang, Pong C Yuen, 与 Shih-Fu Chang。通过不变和扩散实例特征的无监督嵌入学习。在*CVPR*, 2019年。2, 4 [61] Jihun Yi 与 Sungroh Yoon。Patch SVDD：用于异常检测和分割的块级SVDD。arXiv preprint arXiv:2006.16067, 2020年。2, 4, 5, 6, 7 [62] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sangyuk Chun, Junsuk Choe, 与 Youngjoon Yoo。CutMix：训练具有可定位特征的强分类器的正则化策略。在*ICCV*, 2019年。5 [63] Shuangfei Zhai, Yu Cheng, Weining Lu, 与 Zhongfei Zhang。用于异常检测的深度结构化能量模型。
arXiv preprint arXiv:1605.07717, 2016年。4 [64] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, 与 David Lopez-Paz。MixUp：超越经验风险最小化。arXiv preprint arXiv:1710.09412, 2017年。5 [65] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozhi Li, 与 Yi Yang。随机擦除数据增强。在*AAAI*, 2020年。5, 12

- [66] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cris-tian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoen-coding gaussian mixture model for unsupervised anomaly detection. In *ICLR*, 2018. 4

[66] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. 深度自编码高斯混合模型在无监督异常检测中的应用。见 *ICLR*, 2018. 4

A. Details on Experiments

A.1. Experiment with ResNet-18

We train a model on 256×256 image. Model parameters are updated for 65k steps using momentum SGD with the learning rate of 0.03, momentum of 0.9, and the batch size of 64 (or 96 for 3-way). A single cycle of cosine learning rate decay schedule [33] and L2 weight regularization with a coefficient of 0.00003 are used. We apply random translation and color jitters for data augmentation to enhance invariance of representations. We note that the same training strategy, such as the selection of hyperparameters or data augmentations, is applied to different categories. Full range of hyperparameters we studied are provided in below. We use Tensorflow [1], and scikit-learn [41] for GDE implementation.

A.2. Implementation details on CutPaste

Our implementation of the CutPaste augmentation closely follows that of RandomErasing³ [65]. First, we determine the size of the patch by sampling the area ratio between the patch and the full image from $(0.02, 0.15)$. We then determine the aspect ratio by sampling from $(0.3, 1) \cup (1, 3.3)$. The locations of the patch where we cut from and paste to are randomly selected in a way that the entire patch appears in the full image. To construct CutPaste-Scar, we directly sample the width between $[2, 16]$ and the length between $[10, 25]$ in terms of number of pixels. CutPaste-Scar is randomly rotated between $(-45, 45)$ degrees. Before pasting, we apply color jitter,⁴ which applies brightness, contrast, saturation, and hue transformations in sequence with random order, with the maximum intensity of 0.1.

A.3. Ablation Study on Hyperparameters

We study the impact of different values for optimization hyperparameters. Furthermore, we study the impact of CutPaste hyperparameters, such as jitter intensity or the size of the patch. All experiments are done on CutPaste 3-way setting. Below, we enumerate the ranges for hyperparameters we studied. The bold-faced items are default values used for experiments in the main text as well as in this section.

1. Learning rates $\in \{0.1, \mathbf{0.03}, 0.01, 0.003\}$.
2. Number of training epochs $\in \{128, 192, \mathbf{256}, 320, 384\}$.⁵
3. Maximum jitter intensity on patch $\in \{0, \mathbf{0.1}, 0.2, 0.3\}$.
4. Maximum size of patch $\in \{0.1, \mathbf{0.15}, 0.2, 0.3\}$.

We report the mean and standard error of detection AUCs in Table 7 and 8. We observe that our method is fairly robust across different learning rates and number of epochs. When the learning rate becomes too small (≤ 0.003) we observe slow convergence, so it suggests to train longer. The number of training epochs is also an important hyperparameter for semi-supervised anomaly detection as early stopping is particularly difficult. We observe that our method provides a reliable solution when trained for different number of epochs.

For jitter intensity on patch of the CutPaste augmentation, we find it more important for texture categories. Part of the reasons is that the CutPaste augmentation, when applied without jitter augmentation, makes it too difficult to distinguish from the original images, as texture categories contain repetitive patterns. By adding jitter on the patch, the contrast between pasted patch and the surrounding area becomes more apparent and this makes the CutPaste prediction task a bit easier. Similarly, the size of the patch mostly affects the performance on texture categories and we observe the method prefers generally smaller patch sizes.

A.4. Experiment with EfficientNet

We follow the Keras guide⁶ of fine-tuning EfficientNet. We use EfficientNet B4 with batch size 24, which is the limit we tried to fit into single GPU training. We first train only the MLP head for 10 epochs with learning rate of 0.03 while freezing the pretrained backbone. We then fine-tune all layers for 64 epochs with learning rate of 0.0001. We note that the batchnorm layers are kept frozen as suggested by the Keras guide. The other unstated hyperparameters are the same as in Section A.1.

³https://pytorch.org/docs/stable/_modules/torchvision/transforms/transforms.html#RandomErasing

⁴https://pytorch.org/docs/stable/_modules/torchvision/transforms/transforms.html#ColorJitter

⁵Note that, unlike conventional definition for an epoch, we define 256 parameter update steps as one epoch.

⁶https://keras.io/examples/vision/image_classification_efficientnet_fine_tuning/

A. 实验详情

A.1. 使用 ResNet-18 进行实验

我们在 256×256 图像上训练模型。模型参数通过动量SGD优化器更新65k步，学习率为0.03，动量为0.9，批量大小为64（三分类任务中为96）。训练采用单周期余弦学习率衰减策略[33]和系数为0.00003的L2权重正则化。通过随机平移和颜色扰动进行数据增强，以提升表征不变性。需要说明的是，不同类别任务均采用相同的训练策略（包括超参数选择和数据增强方案）。完整超参数研究范围如下所示。我们使用TensorFlow[1]和scikit-learn[41]实现高斯判别器扩展（GDE）。

A.2. CutPaste实现细节

我们对CutPaste数据增强的实现方法基本遵循RandomErasing³[65]的做法。首先，我们通过从(0.02, 0.15)区间采样来确定补丁与完整图像的面积比，从而确定补丁尺寸。接着通过从 $(0.3, 1) \cup (1, 3.3)$ 区间采样来确定宽高比。补丁的剪切位置和粘贴位置均以整块补丁能完整出现在图像内的方式进行随机选择。对于CutPaste-Scar变体，我们直接采样像素宽度[2, 16]与像素长度[10, 25]的数值。CutPaste-Scar会随机旋转(-45, 45)度。在粘贴前，我们会应用颜色抖动⁴——按随机顺序依次进行亮度、对比度、饱和度和色相变换，最大变换强度为0.1。

A.3. 超参数消融研究

我们研究了不同优化超参数值的影响。此外，我们还探讨了CutPaste超参数（如抖动强度或补丁大小）的作用。所有实验均在CutPaste三分类设置下进行。下面列举了我们研究的超参数范围，其中加粗项是主文本及本节实验所使用的默认值。

1. 学习率 $\in \{0.1, \mathbf{0.03}, 0.01, 0.003\}$ 。
2. 训练轮数 $\in \{12, 8, 192, \mathbf{256}, 320, 384\}$ 。⁵
3. 图像块最大抖动强度 $\in \{0, \mathbf{0.1}, 0.2, 0.3\}$ 。
4. 图像块最大尺寸 $\in \{0.1, \mathbf{0.15}, 0.2, 0.3\}$ 。

我们在表7和表8中报告了检测AUC的平均值和标准误差。我们观察到，在不同学习率和训练轮数下，我们的方法都表现出较好的鲁棒性。当学习率过小 (≤ 0.003) 时会出现收敛缓慢现象，建议适当延长训练时间。训练轮数同样是半监督异常检测的重要超参数，因为早期停止策略在此场景下尤其难以把握。实验表明，在不同训练轮数下，我们的方法均能提供稳定可靠的解决方案。

对于CutPaste增强中贴片的抖动强度，我们发现这对纹理类别的检测更为重要。部分原因在于，当CutPaste增强未配合抖动增强时，由于纹理类别包含重复图案，会导致贴片与原始图像难以区分。通过在贴片上添加抖动，粘贴贴片与周围区域之间的对比度变得更加明显，这使得CutPaste预测任务稍微容易一些。同样地，贴片尺寸主要影响纹理类别的检测性能，我们观察到该方法通常更倾向于使用较小的贴片尺寸。

A.4. 使用EfficientNet进行实验

我们遵循Keras指南⁶对EfficientNet进行微调。采用EfficientNet B4架构，批处理大小为24——这是我们在单GPU训练中尝试达到的极限配置。首先冻结预训练主干网络，仅训练MLP头部10个周期，学习率设置为0.03；随后解冻所有层进行64个周期的微调，学习率调整为0.0001。需要说明的是，按照Keras指南建议，批归一化层始终保持冻结状态。其他未明确列出的超参数均与附录A.1章节保持一致。

³https://pytorch.org/docs/stable/_modules/torchvision/transforms/transforms.html#RandomErasing

⁴https://pytorch.org/docs/stable/_modules/torchvision/transforms/transforms.html#ColorJitter

⁵Note that, unlike conventional definition for an epoch, we define 256 parameter update steps as one epoch.

⁶https://keras.io/examples/vision/image_classification_efficientnet_fine_tuning/

Table 7: Detection AUCs using (1) different learning rates and (2) the number of epochs. We report the mean and standard error of AUCs for 5 runs with different random seeds.

Category	Learning rates				Number of epochs				
	0.1	0.03	0.01	0.003	128	192	256	320	384
texture	97.1 \pm 0.3	97.0 \pm 0.5	97.2 \pm 0.3	96.1 \pm 0.7	96.6 \pm 0.4	96.1 \pm 0.7	97.0 \pm 0.5	97.0 \pm 0.4	96.3 \pm 0.4
object	94.4 \pm 0.6	94.3 \pm 0.6	94.2 \pm 0.6	93.9 \pm 0.5	94.9 \pm 0.6	94.5 \pm 0.4	94.3 \pm 0.6	94.7 \pm 0.5	94.0 \pm 0.6
all	95.3 \pm 0.5	95.2 \pm 0.6	95.2 \pm 0.5	94.6 \pm 0.6	95.4 \pm 0.5	95.0 \pm 0.5	95.2 \pm 0.6	95.5 \pm 0.4	94.8 \pm 0.5

Table 8: Detection AUCs using (1) different jitter intensity and (2) the size of patch of the CutPaste augmentation. We report the mean and standard error of AUCs for 5 runs with different random seeds.

Category	Jitter intensity				Size of patch			
	0.0	0.1	0.2	0.3	0.1	0.15	0.2	0.3
texture	96.2 \pm 0.6	97.0 \pm 0.5	97.4 \pm 0.3	97.5 \pm 0.2	97.1 \pm 0.4	97.0 \pm 0.5	95.8 \pm 0.9	96.6 \pm 0.4
object	94.3 \pm 0.6	94.3 \pm 0.6	94.5 \pm 0.5	94.5 \pm 0.5	94.5 \pm 0.5	94.3 \pm 0.6	94.4 \pm 0.5	94.5 \pm 0.5
all	94.9 \pm 0.6	95.2 \pm 0.6	95.5 \pm 0.4	95.5 \pm 0.4	95.3 \pm 0.5	95.2 \pm 0.6	94.8 \pm 0.6	95.2 \pm 0.5

A.5. Details on Localization with Patch-based model

The training procedure of patch-based model with CutPaste should be straightforward from Section 2.4. Once patch-based models are trained, we densely extract representations of 32×32 patches with the stride of 4, which results in $(\frac{256-32}{4} + 1) \times (\frac{256-32}{4} + 1) \times 512 = 57 \times 57 \times 512$ dimensional tensor of embedding vectors. Then we compute the anomaly score of 512 dimensional embedding vector at each location to obtain 57×57 anomaly score map. Finally, to obtain an anomaly score map of full resolution (256×256), we apply receptive field upsampling via Gaussian smoothing following [32], which essentially applies the transposed convolution with the stride of 4, the same stride that we used for dense feature extraction, using a single convolution kernel of size 32×32 whose weights are determined by a Gaussian distribution.

Depending on the category, we find different strategies for computing score improves the performance. For those categories with “aligned objects” (bottle, cable, capsule, metal nut, pill, toothbrush, transistor, zipper), we find it useful to construct one-class classifiers at each location separately. This is particularly useful for detecting missing or dislocated components (e.g., examples in the second row of Figure 15) as global context is captured by classifiers at each location. For some object categories, such as hazelnut or screw whose objects are randomly rotated, and texture categories, we use a single one-class classifier applied to all locations.

B. More Localization Visualizations

From Figure 7 to Figure 21, we show localization visualizations of 24 examples for each of 10 object and 5 texture categories via GradCAM for image-level CutPaste models and patch heatmap for patch-based models. We not only show successful cases, but also some failure cases.

B.1. Failure Case Analysis

Here we enumerate some failure cases. Note that this is not the comprehensive list of all failure cases, but a few representative failure cases we find via visually inspecting localization visualizations.

- Cable (Figure 8): Missing components (row 2 column 6–8).
- Metal Nut (Figure 11): Flipped components (row 1 column 1–2).
- Screw (Figure 13): Speckle noise in the background (row 3 column 1, 6).
- Transistor (Figure 15): Dislocated or missing components (row 2).
- Tile (Figure 20): dyed tiles (row 2 column 7–8, row 3 column 1–2).

表7：使用(1)不同学习率和(2)训练轮次数量检测的AUC值。我们报告了5次不同随机种子运行所得AUC的平均值及标准误差。

Category	Learning rates				Number of epochs				
	0.1	0.03	0.01	0.003	128	192	256	320	384
texture	97.1±0.3	97.0±0.5	97.2±0.3	96.1±0.7	96.6±0.4	96.1±0.7	97.0±0.5	97.0±0.4	96.3±0.4
object	94.4±0.6	94.3±0.6	94.2±0.6	93.9±0.5	94.9±0.6	94.5±0.4	94.3±0.6	94.7±0.5	94.0±0.6
all	95.3±0.5	95.2±0.6	95.2±0.5	94.6±0.6	95.4±0.5	95.0±0.5	95.2±0.6	95.5±0.4	94.8±0.5

表8：使用(1)不同抖动强度和(2)CutPaste增强的补丁尺寸所获得的检测AUC值。我们报告了5次不同随机种子运行所得AUC的平均值及标准误差。

Category	Jitter intensity				Size of patch			
	0.0	0.1	0.2	0.3	0.1	0.15	0.2	0.3
texture	96.2±0.6	97.0±0.5	97.4±0.3	97.5±0.2	97.1±0.4	97.0±0.5	95.8±0.9	96.6±0.4
object	94.3±0.6	94.3±0.6	94.5±0.5	94.5±0.5	94.5±0.5	94.3±0.6	94.4±0.5	94.5±0.5
all	94.9±0.6	95.2±0.6	95.5±0.4	95.5±0.4	95.3±0.5	95.2±0.6	94.8±0.6	95.2±0.5

A.5. 基于图像块的定位模型详解

基于补丁的模型与CutPaste的训练过程应直接从第2.4节中得出。一旦基于补丁的模型训练完成，我们以4的步幅密集提取 32×32 补丁的表征，这将产生 $(\frac{256-32}{4} + 1) \times (\frac{256-32}{4} + 1) \times 512 = 57 \times 57 \times 512$ 维的嵌入向量张量。然后我们计算每个位置512维嵌入向量的异常分数，得到 57×57 的异常分数图。最后，为了获得全分辨率(256×256)的异常分数图，我们按照[32]的方法通过高斯平滑进行感受野上采样，这实质上应用了步幅为4的转置卷积（与我们用于密集特征提取的步幅相同），使用大小为 32×32 的单一卷积核，其权重由高斯分布确定。

根据类别不同，我们发现采用不同的分数计算策略能提升性能。对于存在“对齐物体”的类别（瓶子、电缆、胶囊、螺母、药片、牙刷、晶体管、拉链），我们发现分别在每个位置构建单类别分类器具有显著效果。这种方法能有效检测缺失或错位组件（例如图15第二行示例），因为每个位置的分类器能捕捉全局上下文。对于物体随机旋转的类别（如榛果、螺丝）以及纹理类类别，我们采用单一单类别分类器应用于所有位置。

B. 更多定位可视化

从图7到图21，我们通过GradCAM对图像级CutPaste模型、通过补丁热力图对基于补丁的模型，分别展示了10个物体类别和5个纹理类别各24个样本的定位可视化结果。我们不仅展示了成功案例，也呈现了部分失败案例。

B.1. 失败案例分析

这里我们列举了一些失败案例。请注意，这并非所有失败案例的完整列表，而是我们通过可视化检查定位结果时发现的几个代表性失败案例。

- 电缆（图8）：缺少组件（第2行第6-8列）。
- 金属螺母（图11）：翻转的组件（第1行第1-2列）。
- 螺钉（图13）：背景中的散斑噪声（第3行第1列，第6列）。
- 晶体管（图15）：错位或缺失的元件（第2行）。
- 瓷砖（图20）：染色瓷砖（第2行第7-8列，第3行第1-2列）。

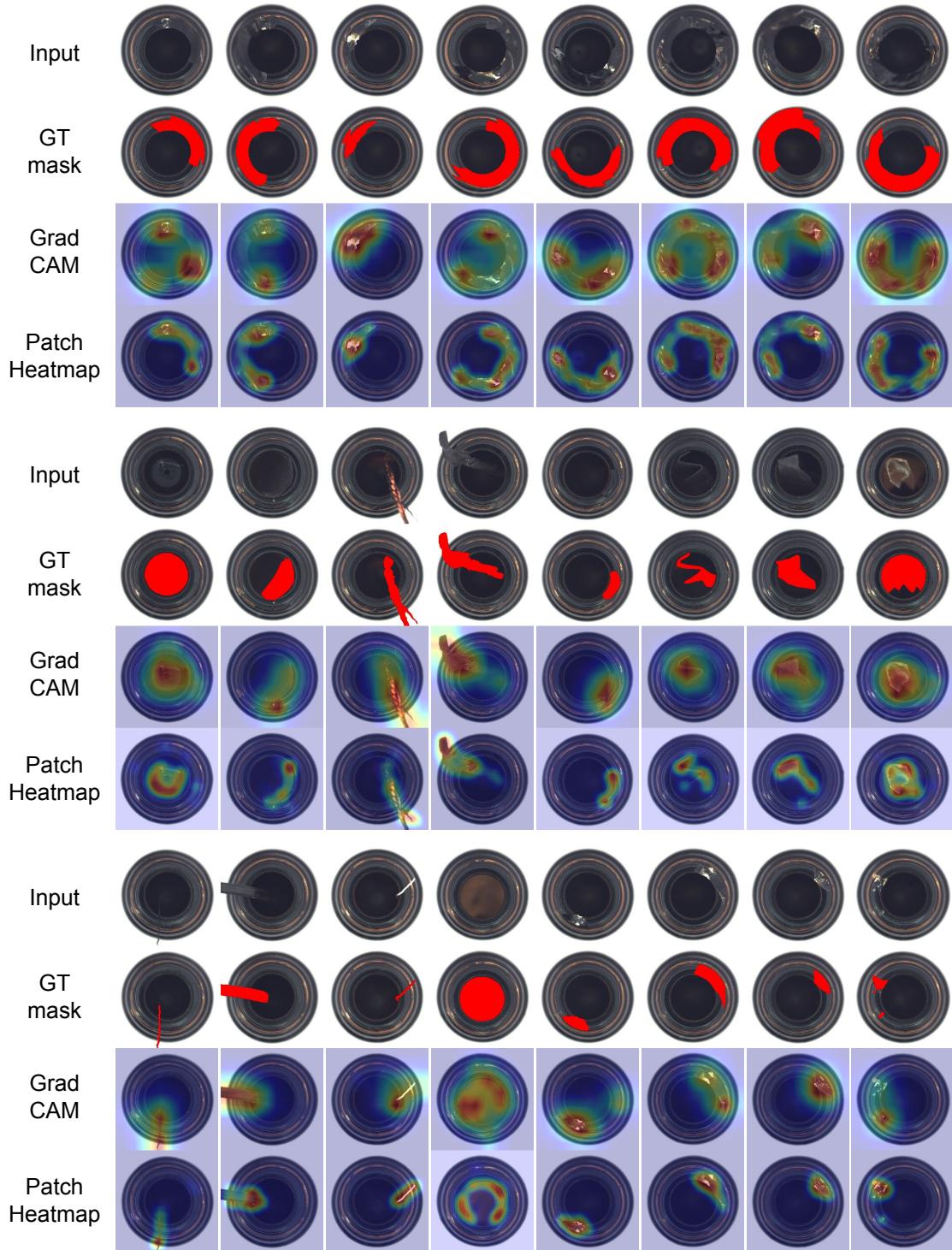


Figure 7: Defect localization on bottle class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

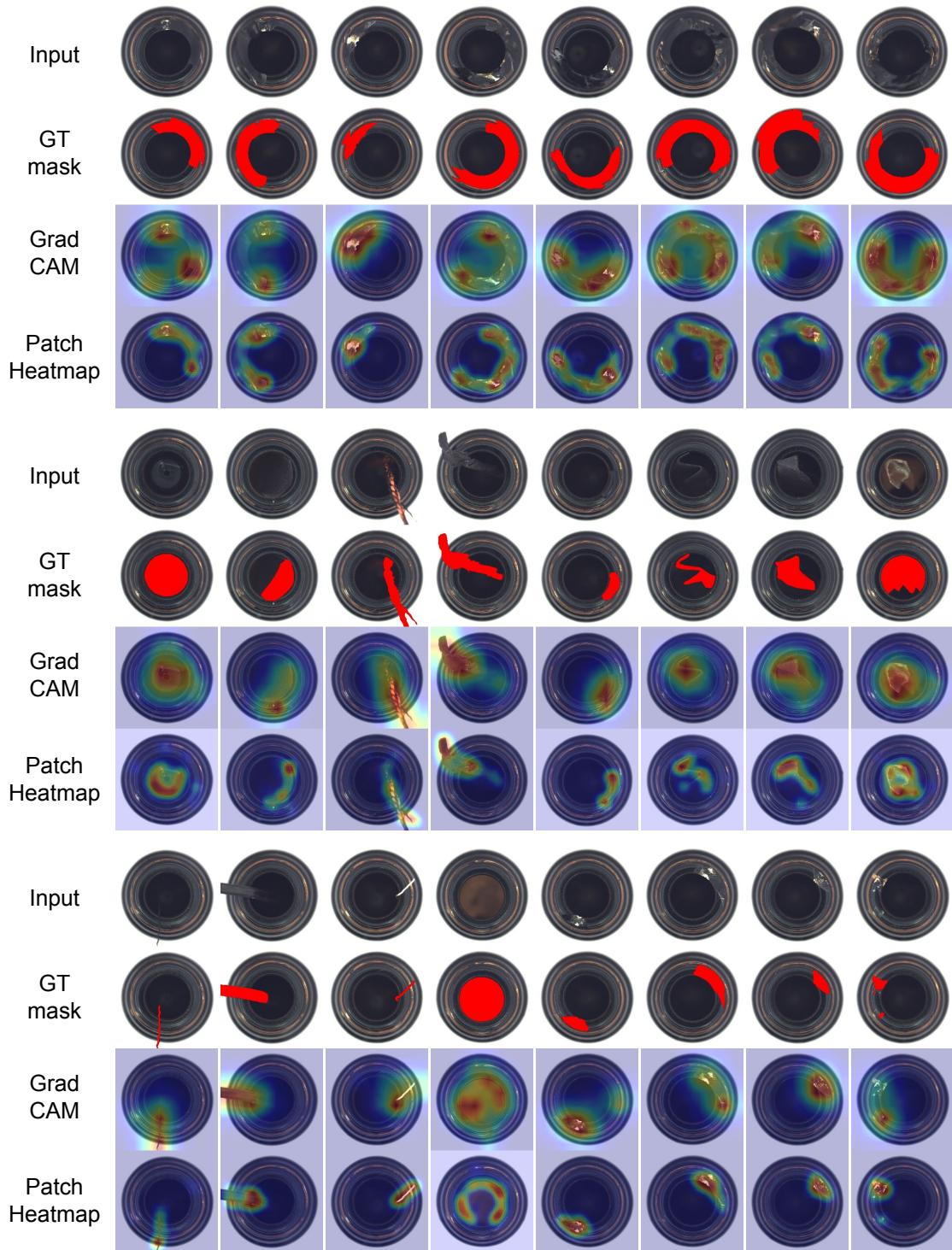


图7：MVTec数据集瓶类别的缺陷定位结果。自上而下分别为：输入图像、带有红色真实定位掩码的图像、使用图像级检测器的GradCAM结果，以及使用块级检测器的热力图。

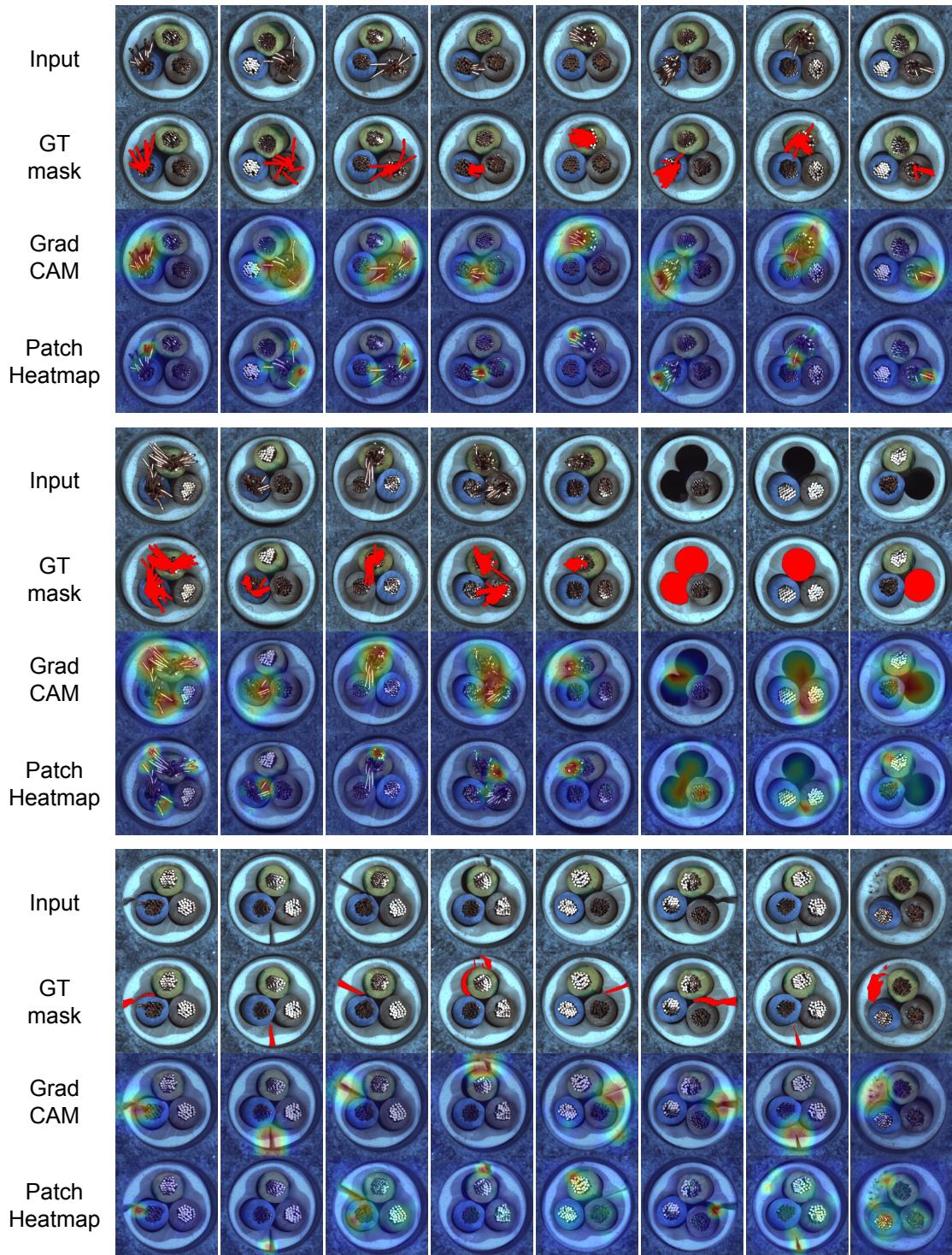


Figure 8: Defect localization on cable class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

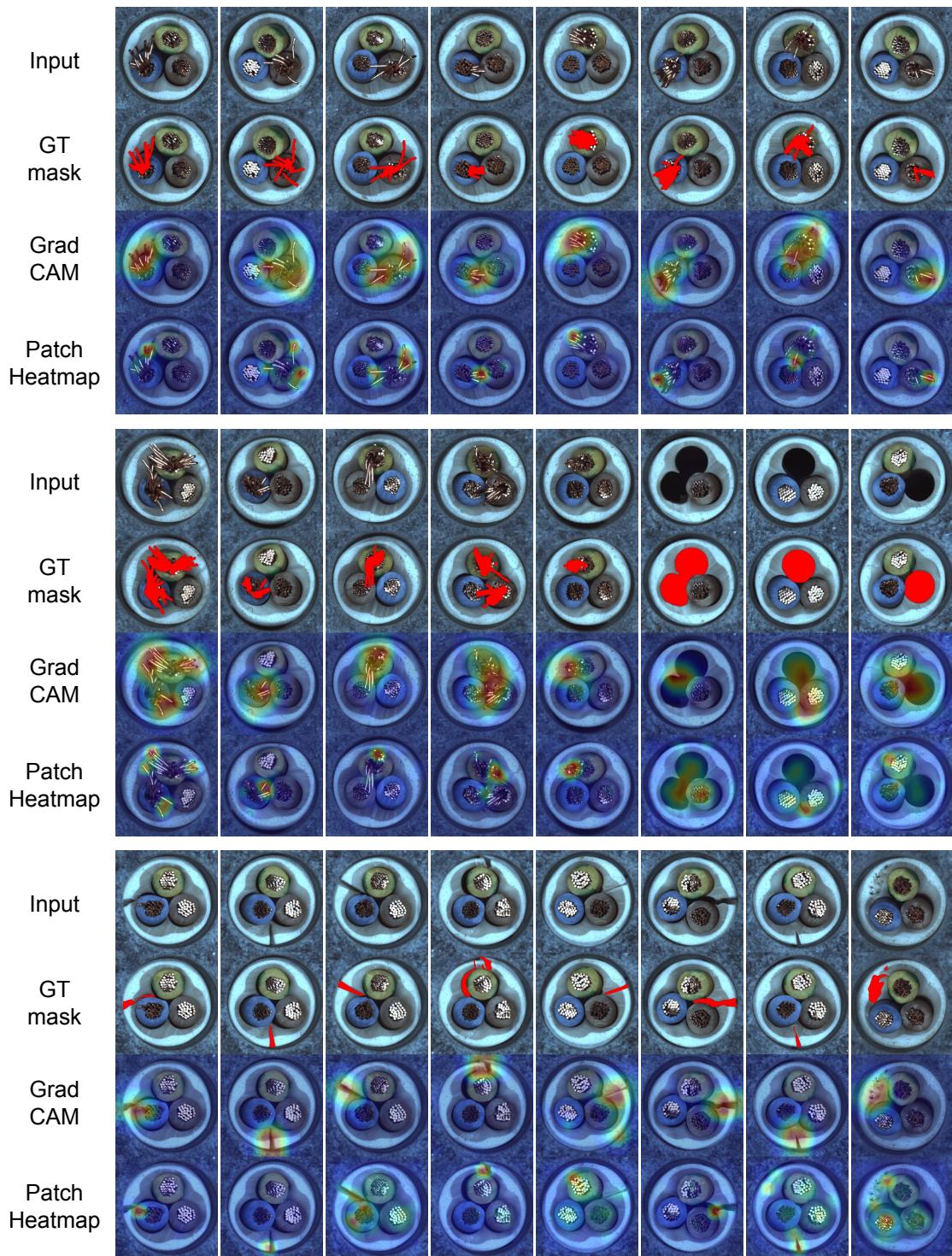


图8：MVTec数据集电缆类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

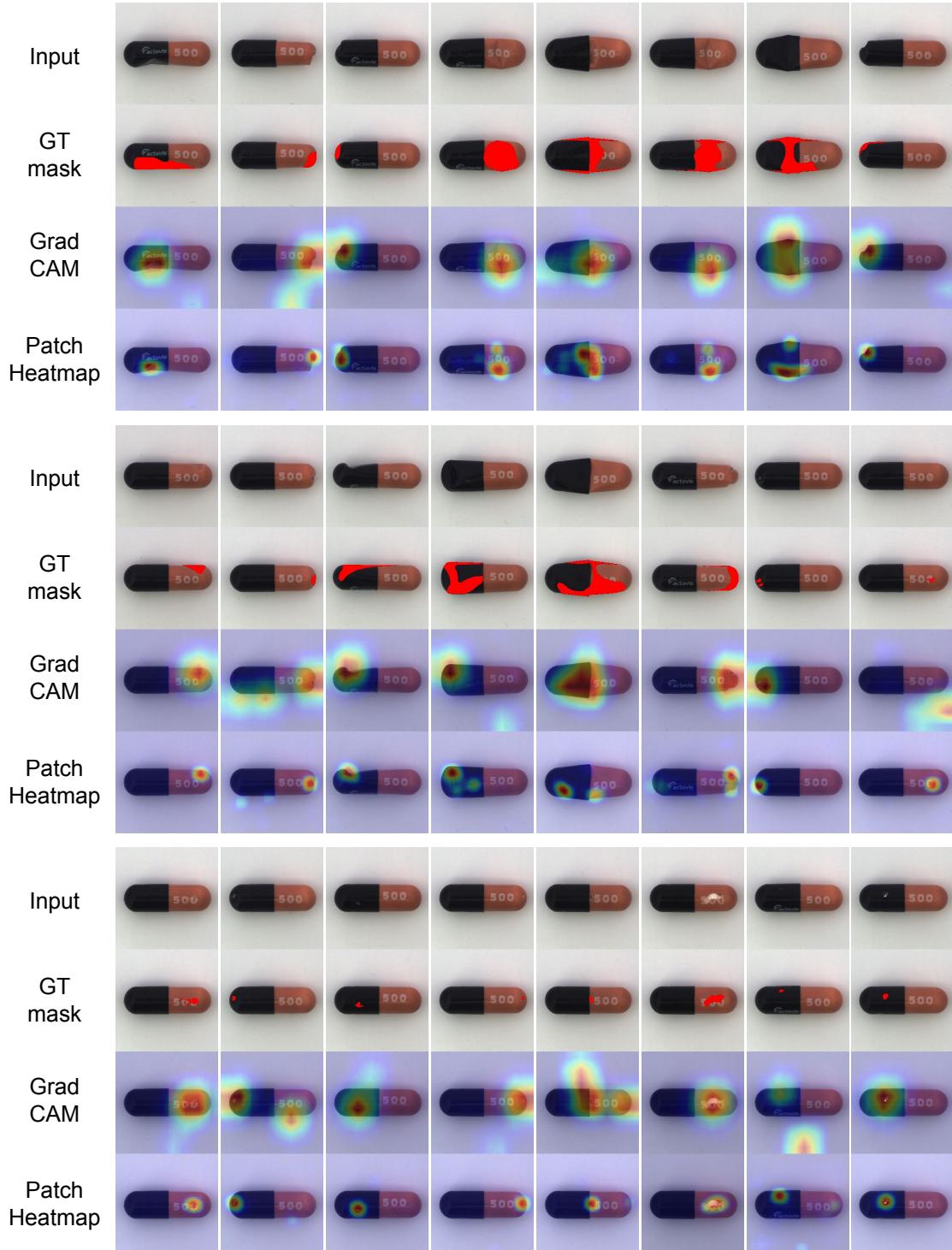


Figure 9: Defect localization on capsule class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

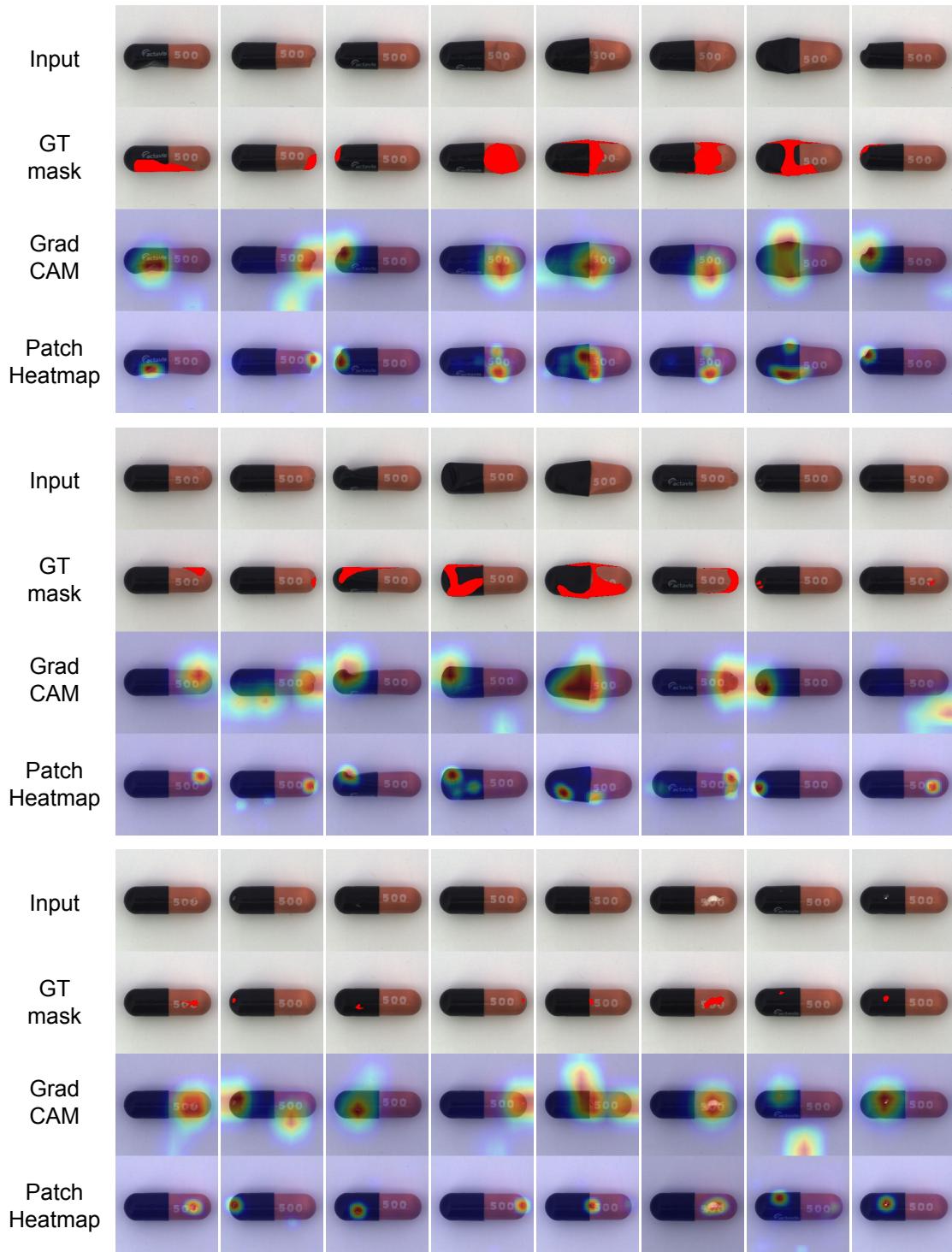


图9：MVTec数据集胶囊类别的缺陷定位结果。自上而下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

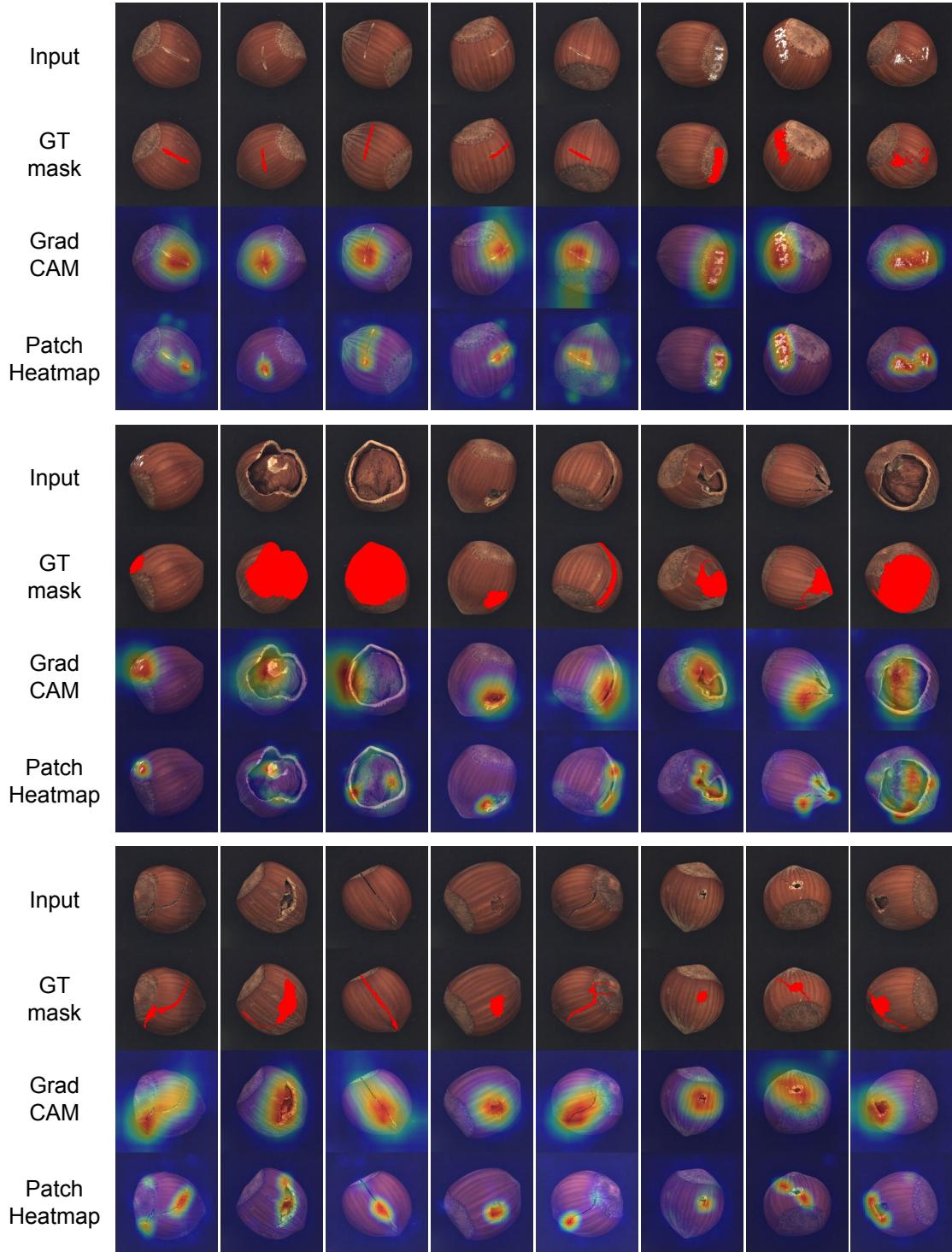


Figure 10: Defect localization on hazelnut class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

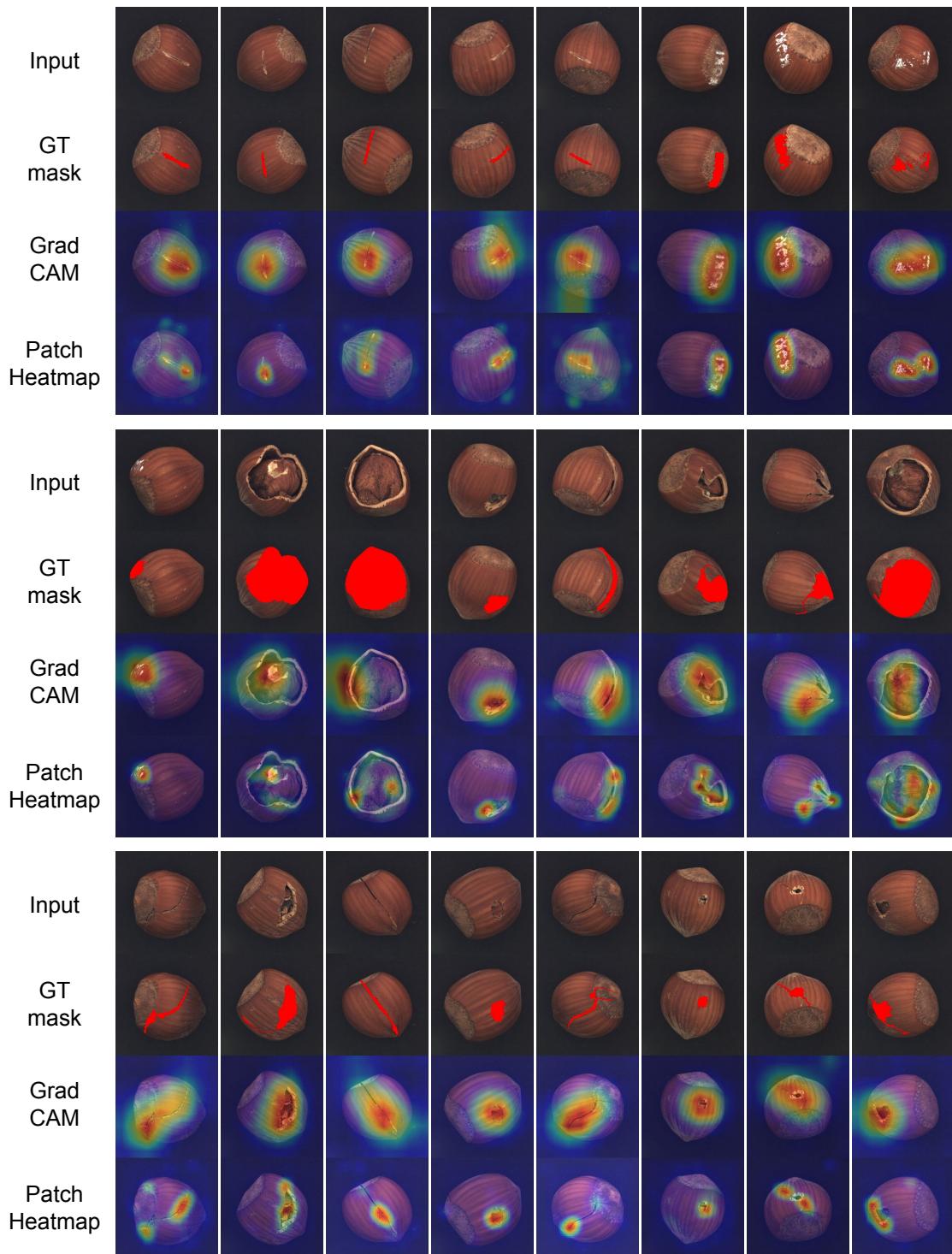


图10：MVTec数据集榛果类别的缺陷定位结果。自上而下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器生成的GradCAM热力图，以及使用区块级检测器生成的热力图。

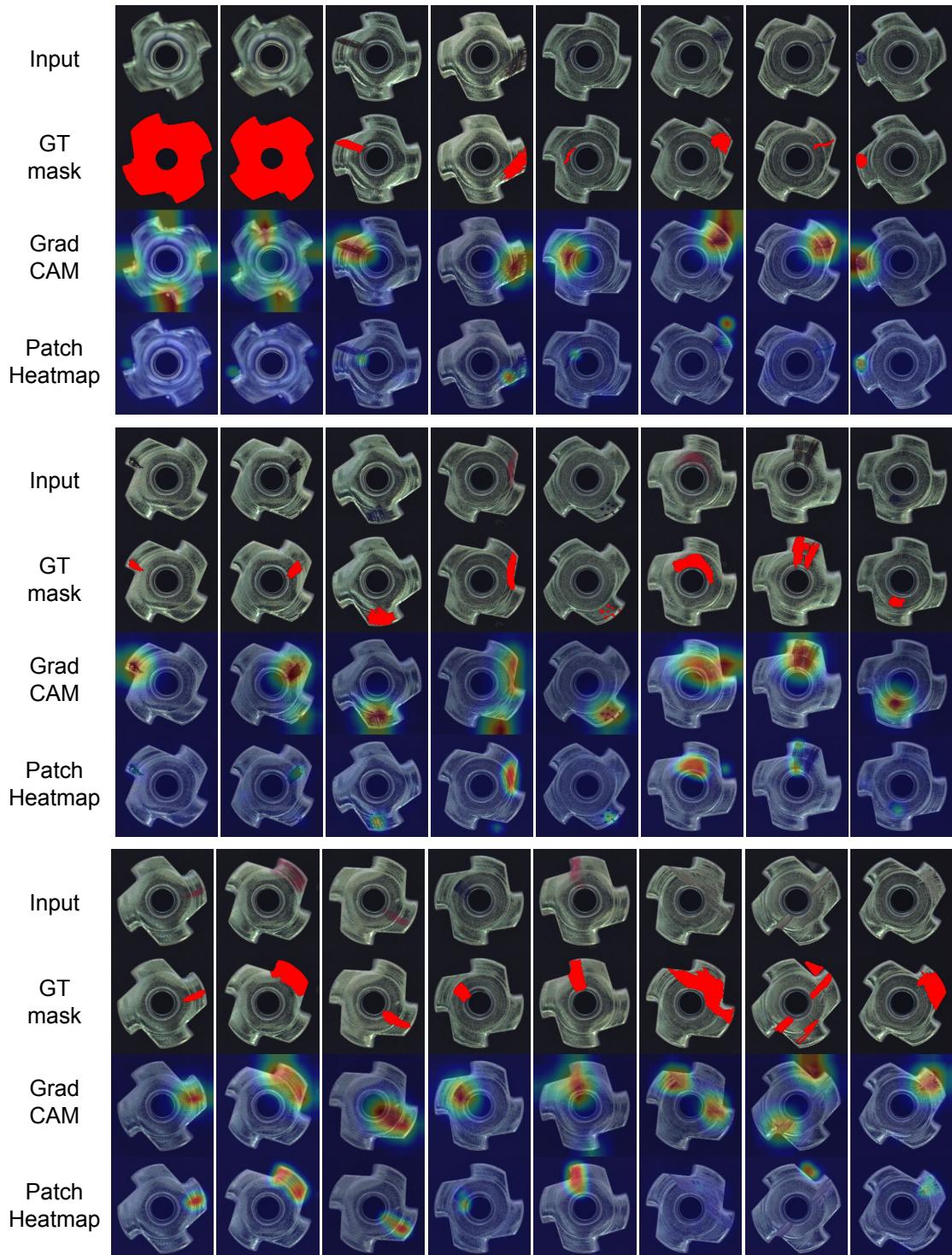


Figure 11: Defect localization on metal nut class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

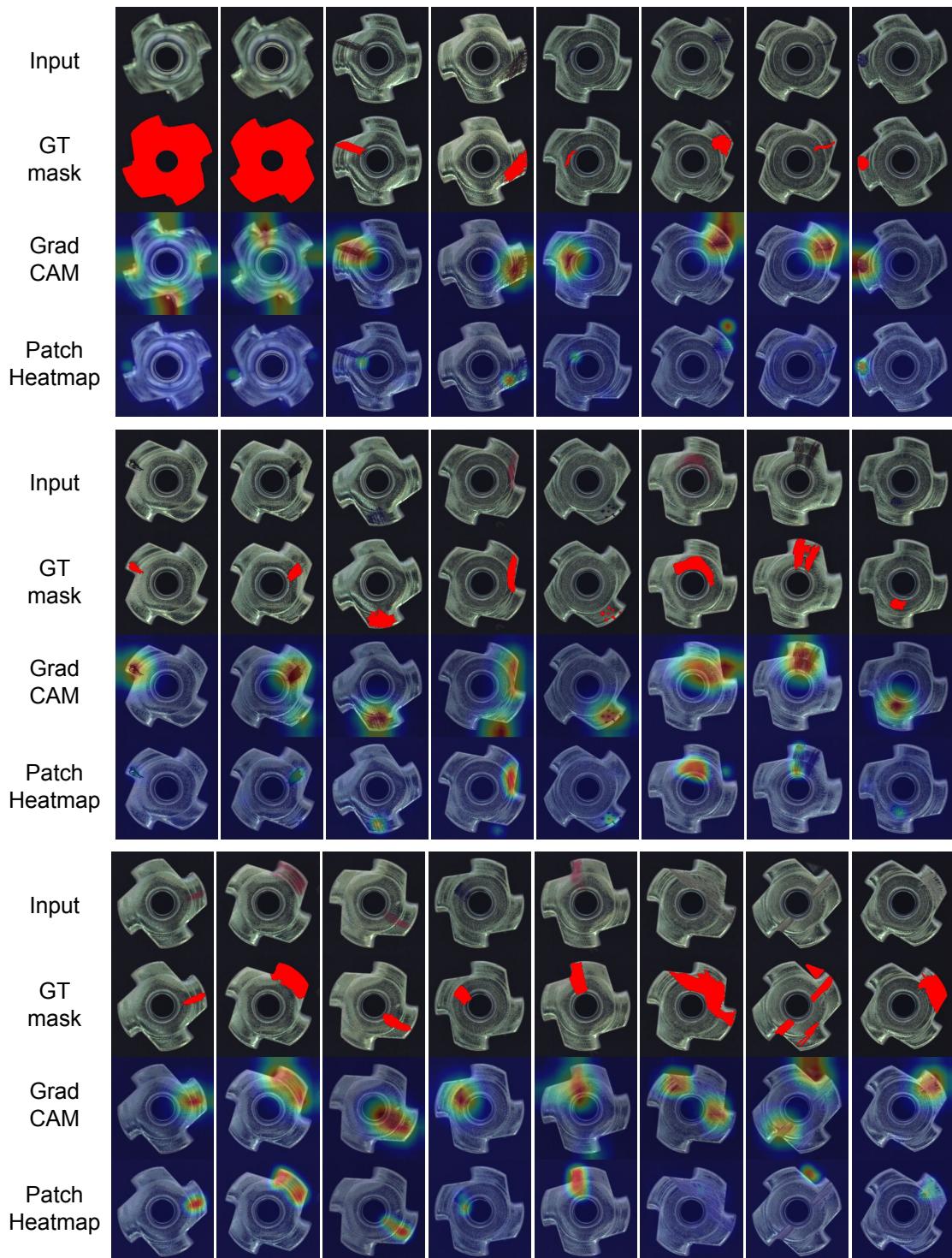


图11：MVTec数据集金属螺母类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

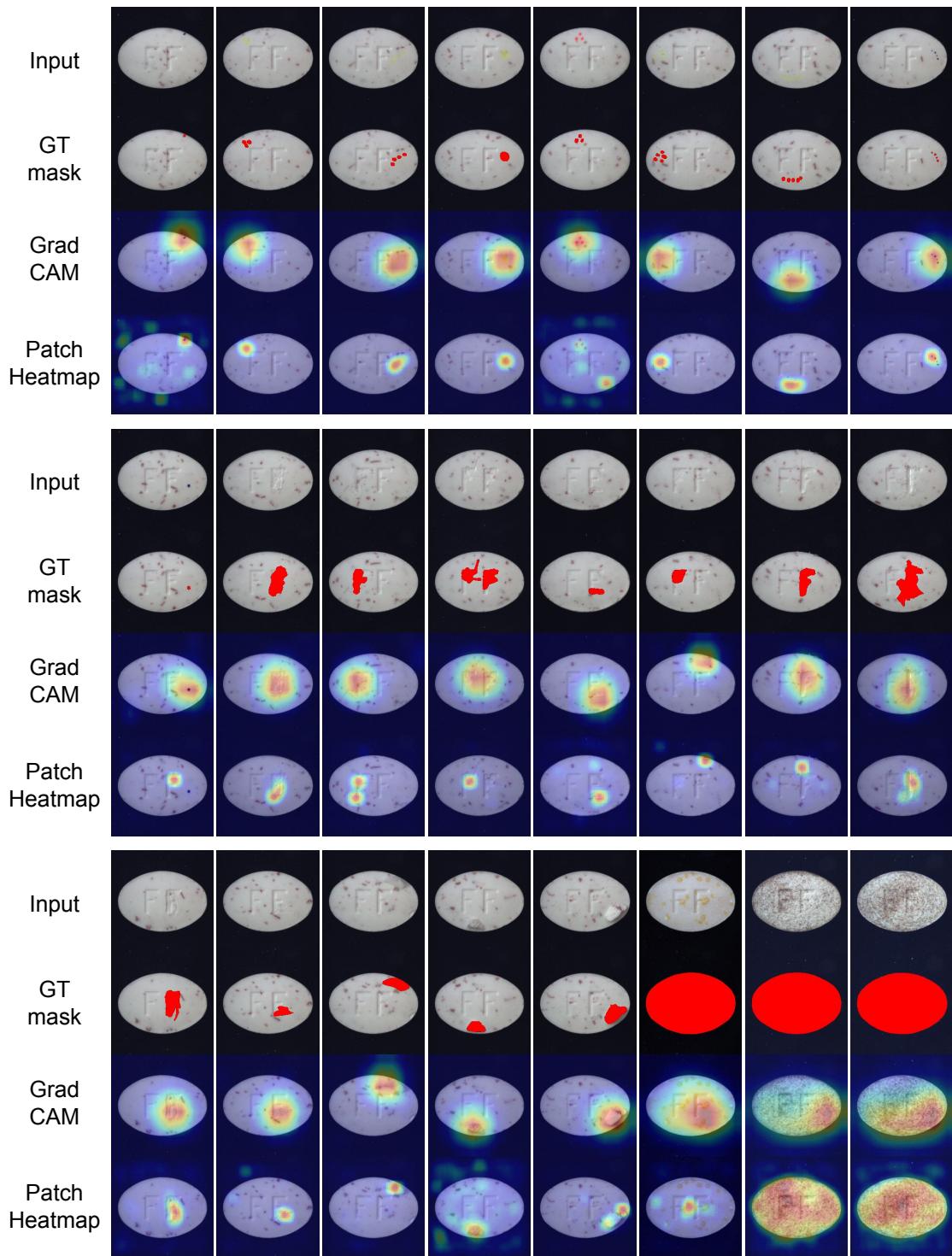


Figure 12: Defect localization on pill class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

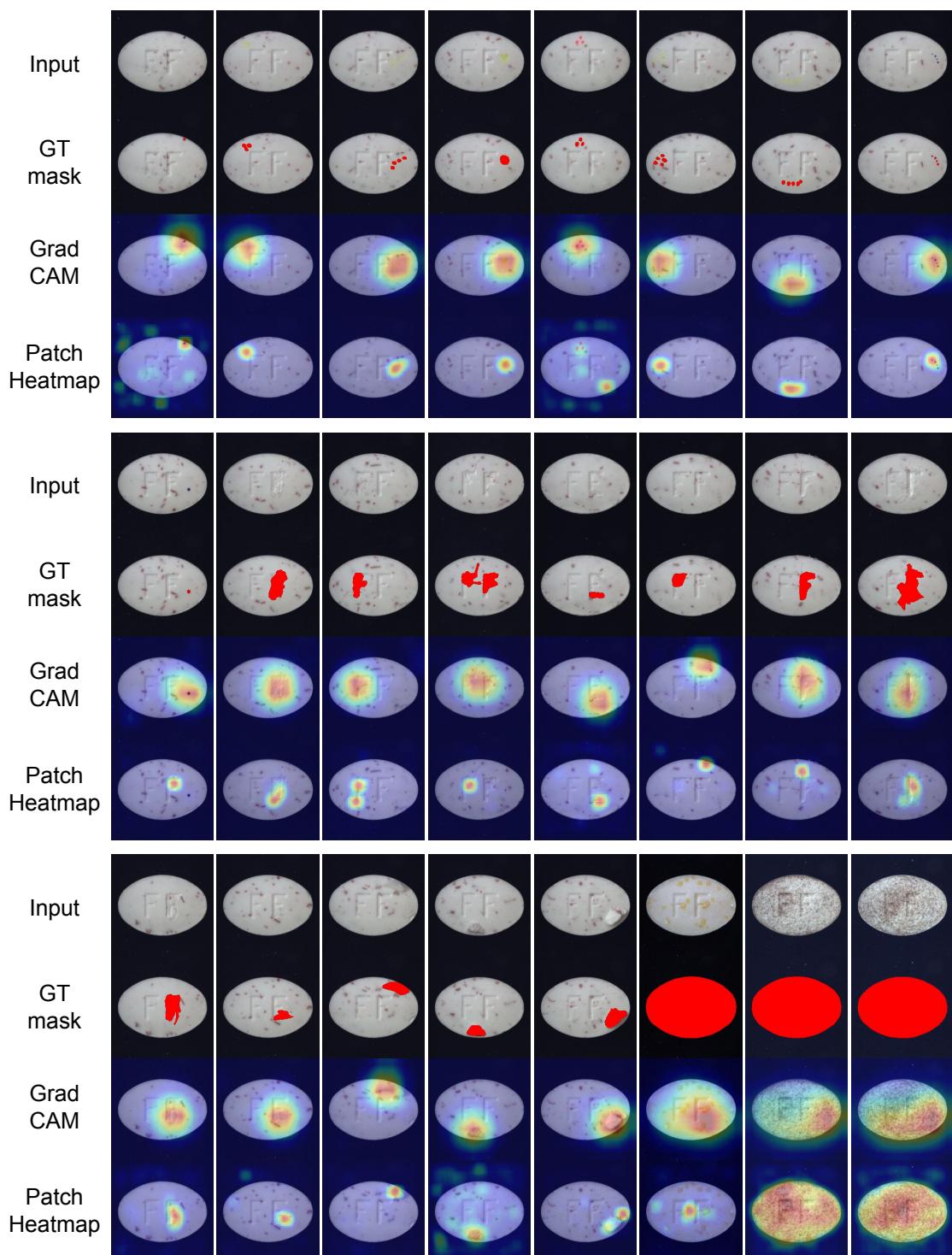


图12：MVTec数据集药片类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

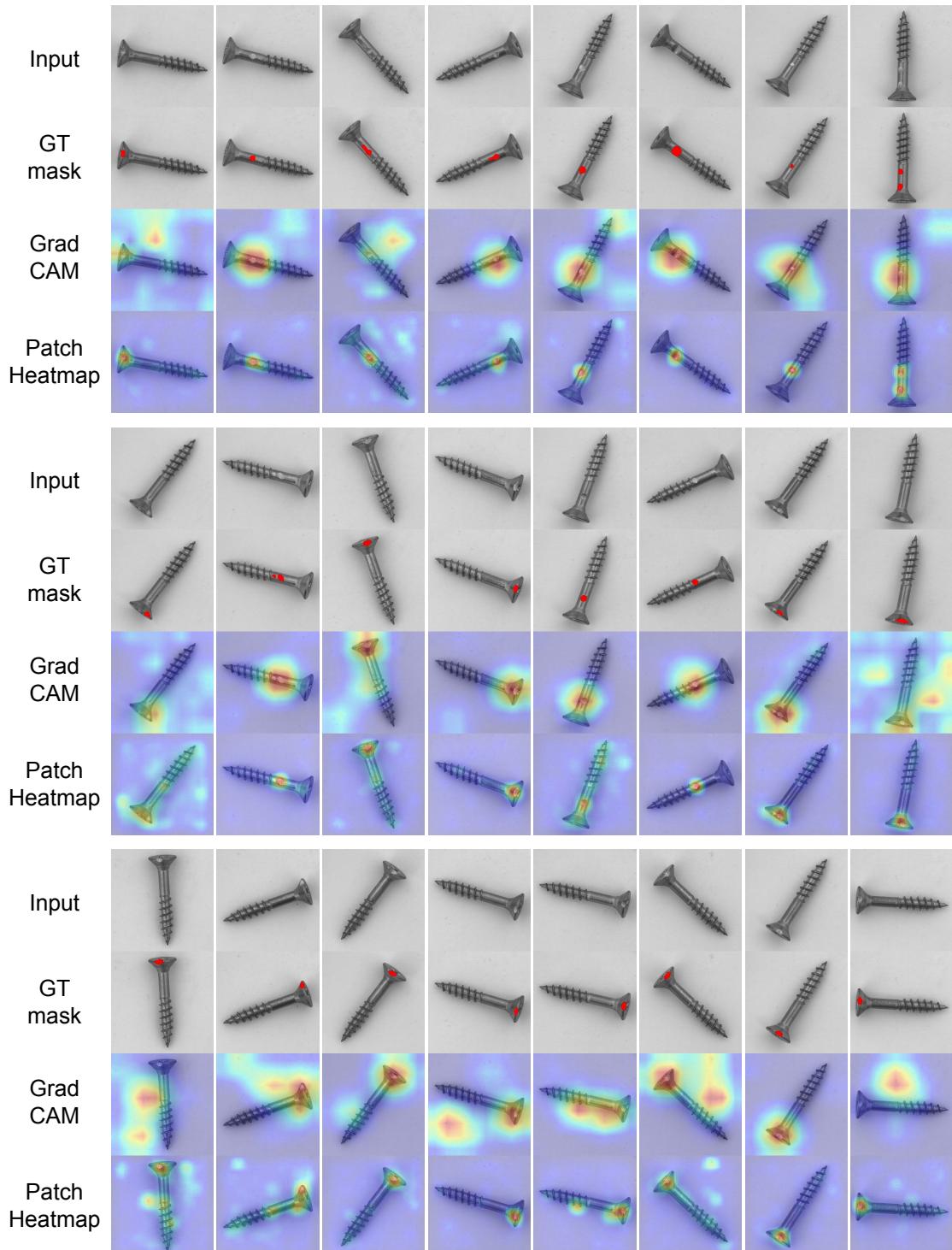


Figure 13: Defect localization on screw class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

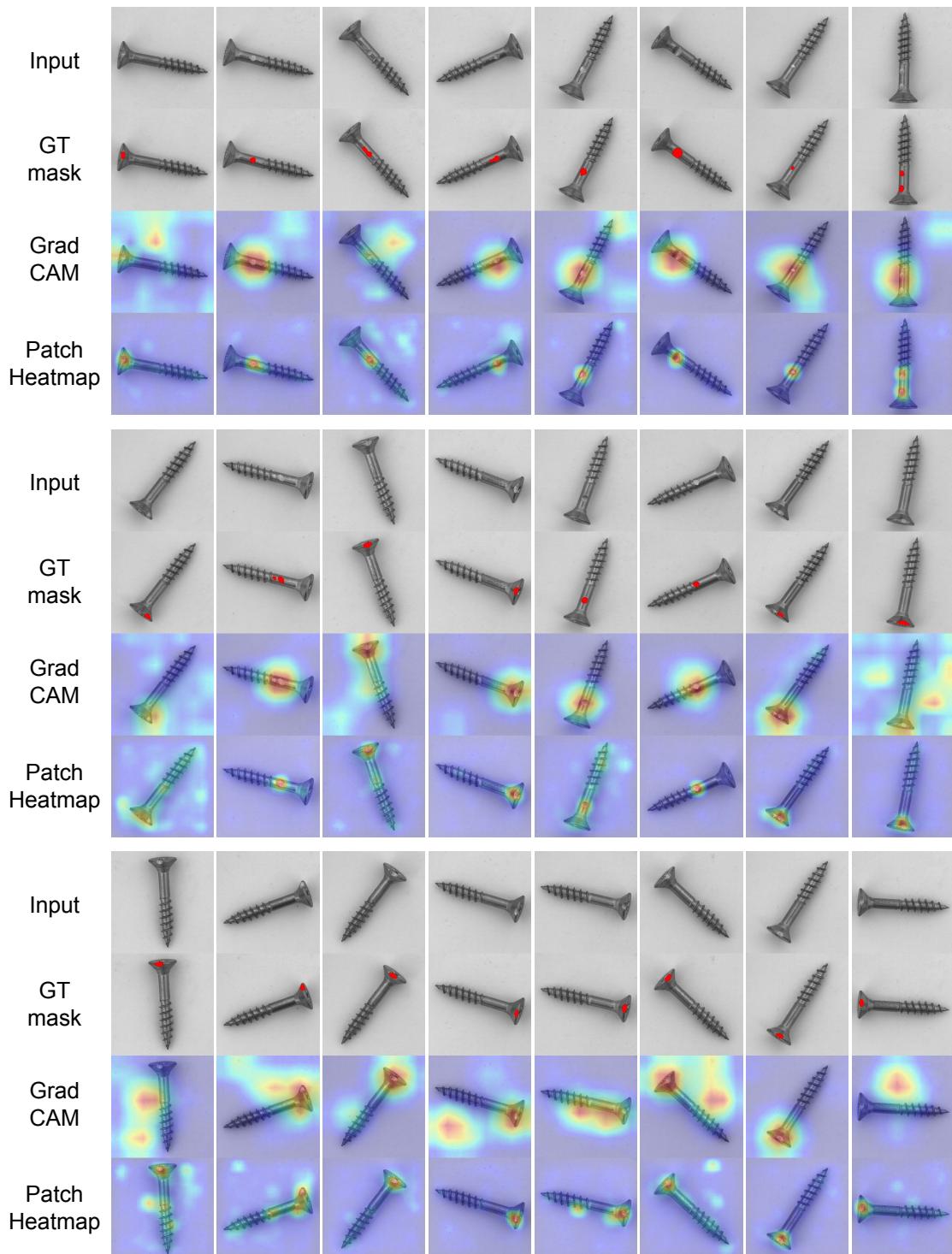


图13：MVTec数据集螺丝类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

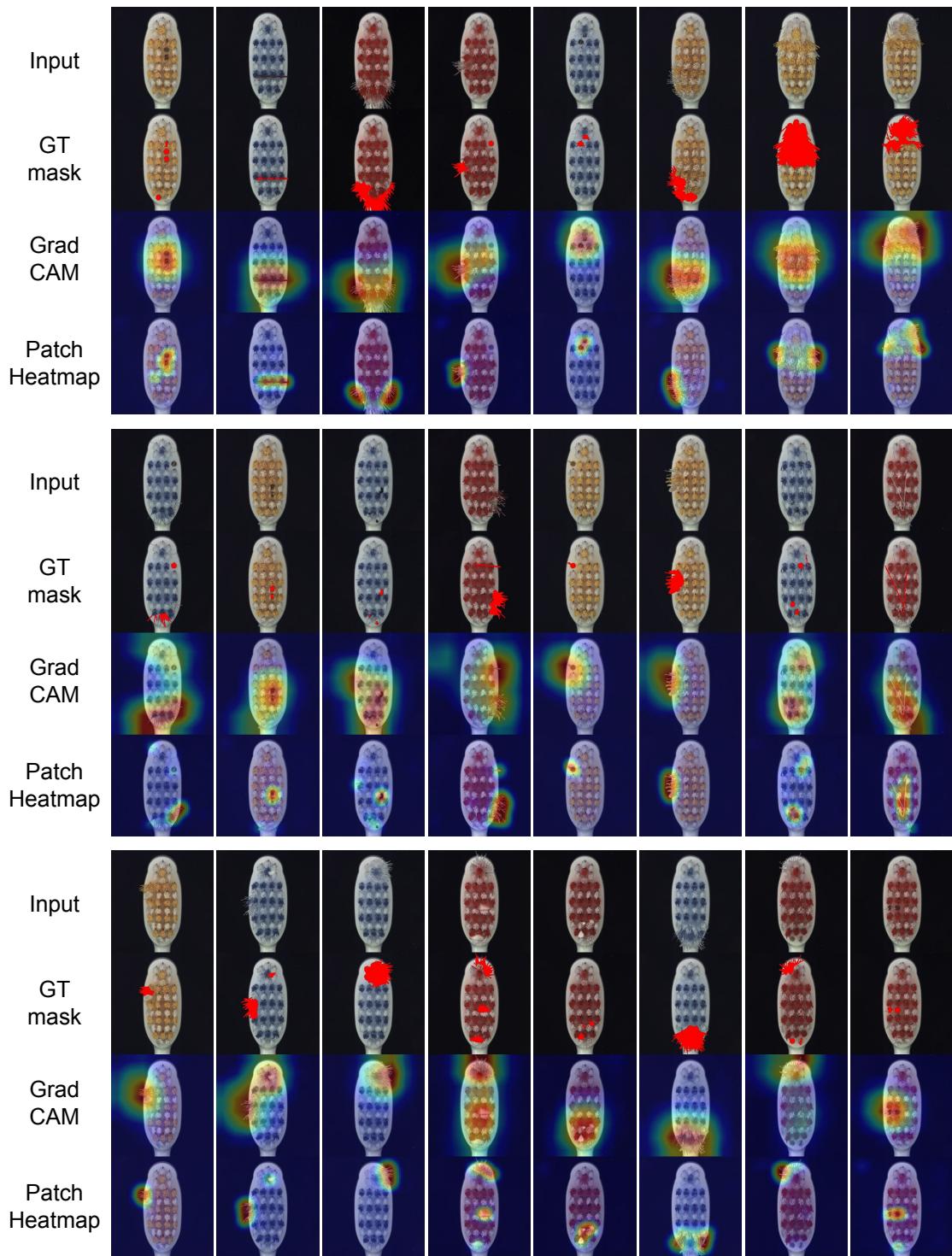


Figure 14: Defect localization on toothbrush class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

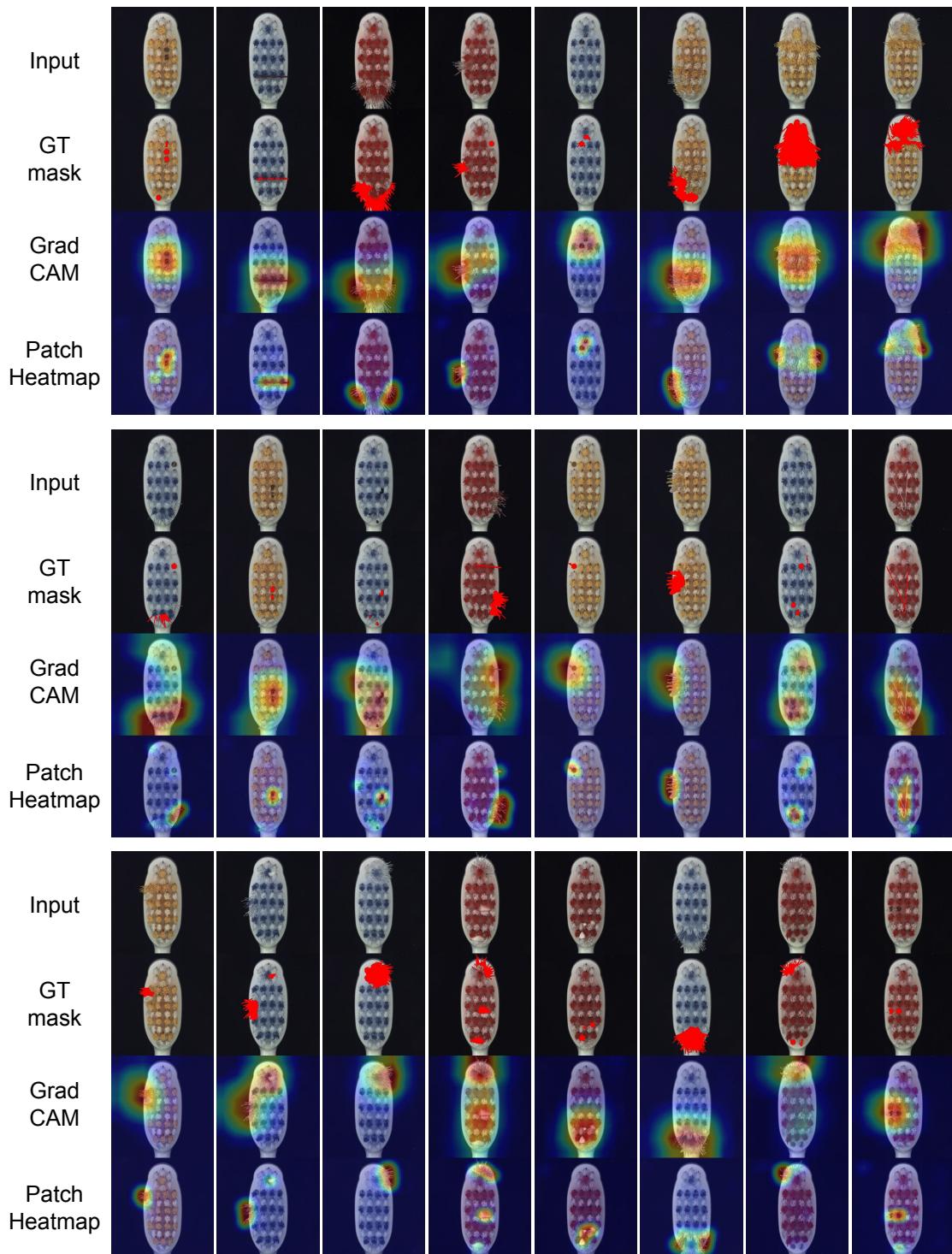


图14：MVTec数据集牙刷类别的缺陷定位。从上至下分别为：输入图像、带有红色真实定位掩码的图像、使用图像级检测器的GradCAM结果，以及使用块级检测器的热力图。

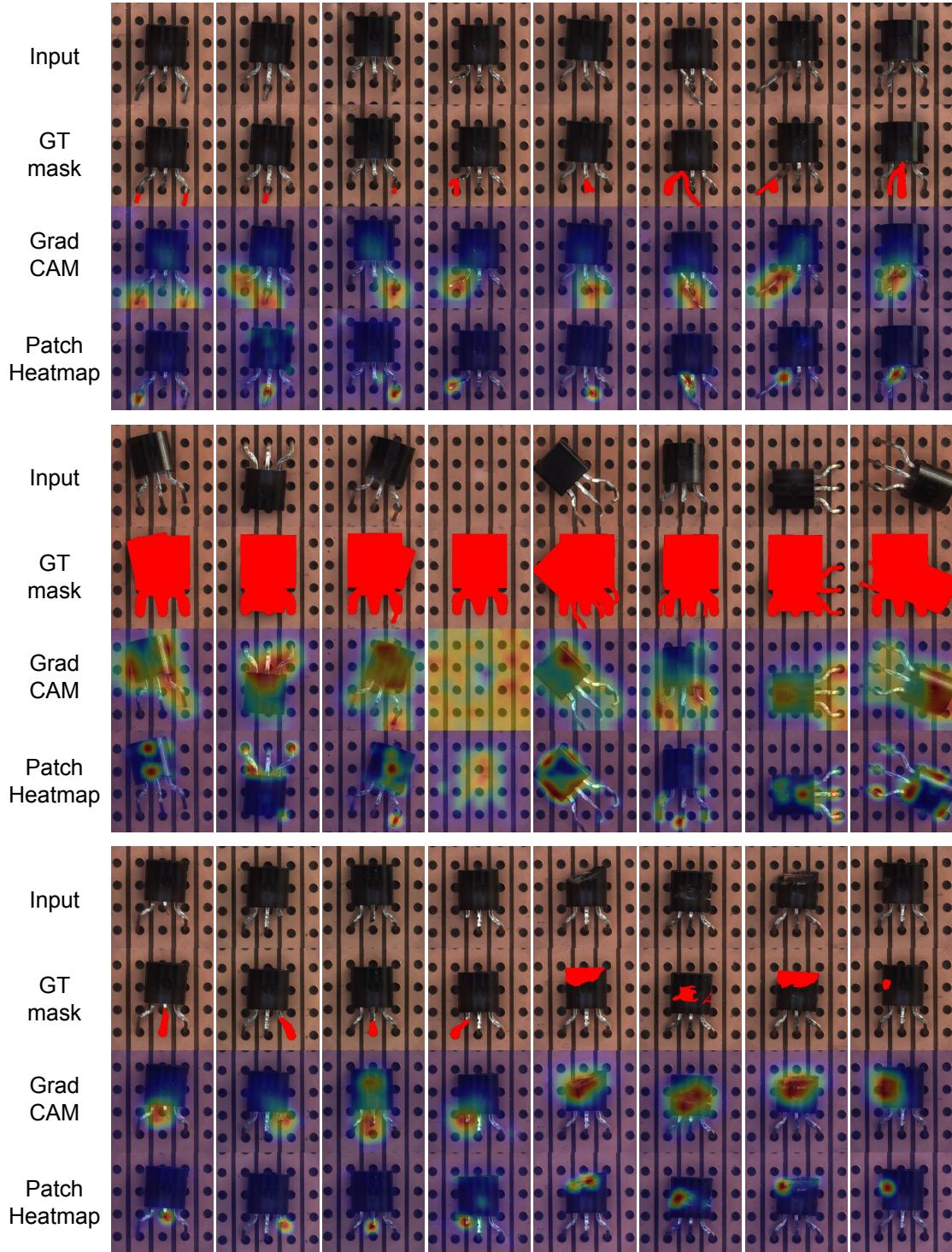


Figure 15: Defect localization on transistor class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

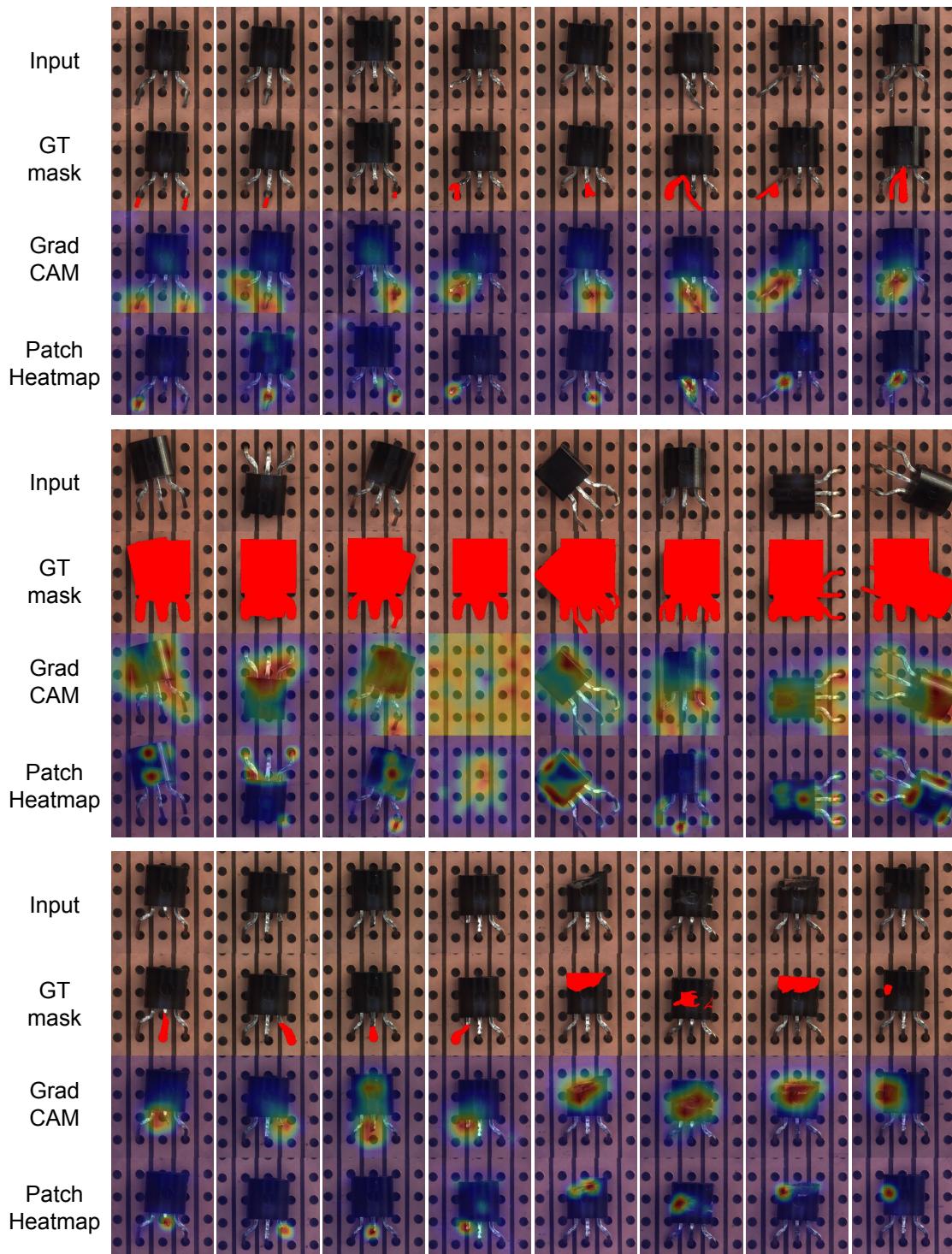


图15：MVTec数据集晶体管类别的缺陷定位结果。自上而下分别为：输入图像、带有红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

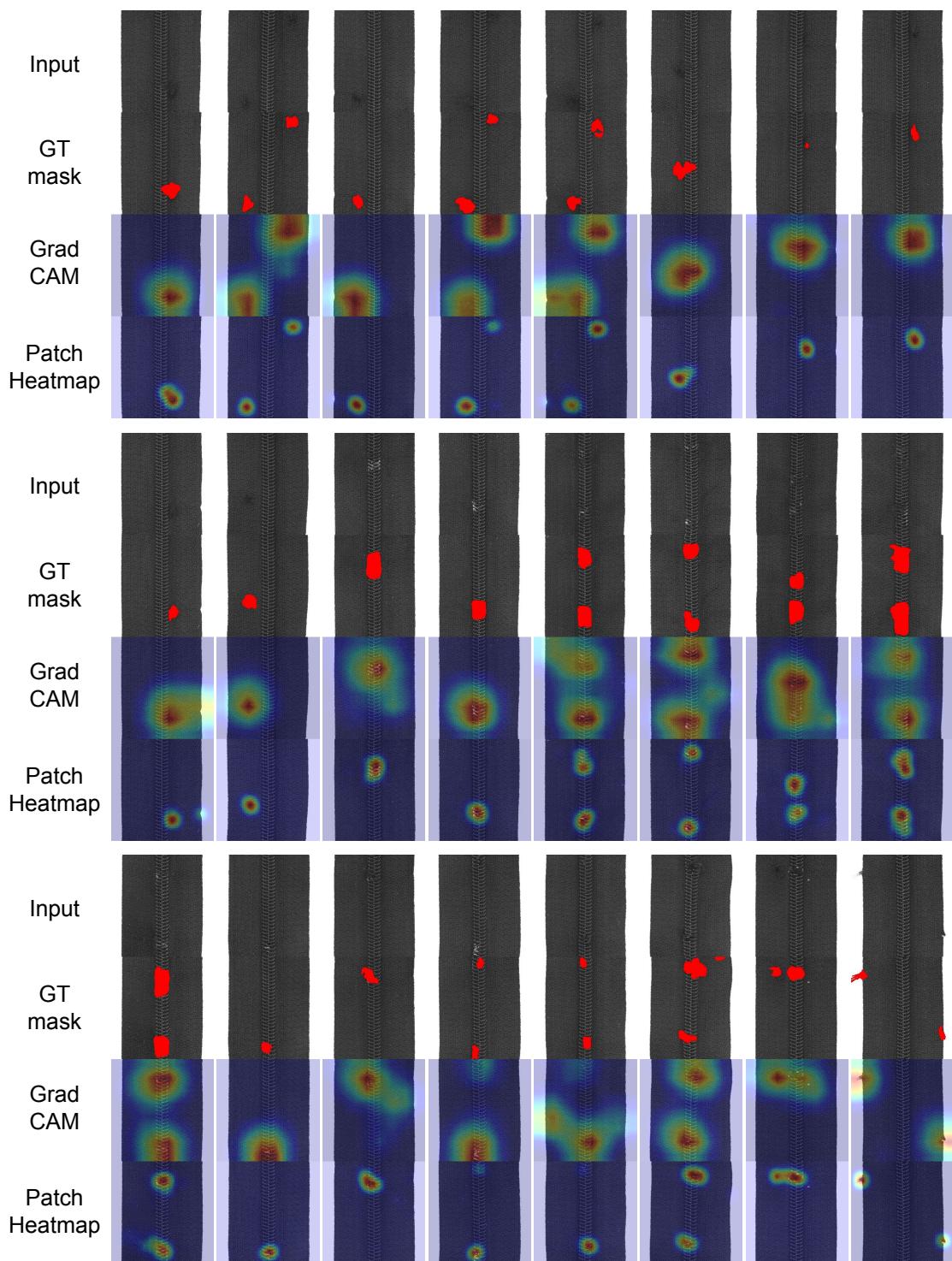


Figure 16: Defect localization on zipper class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

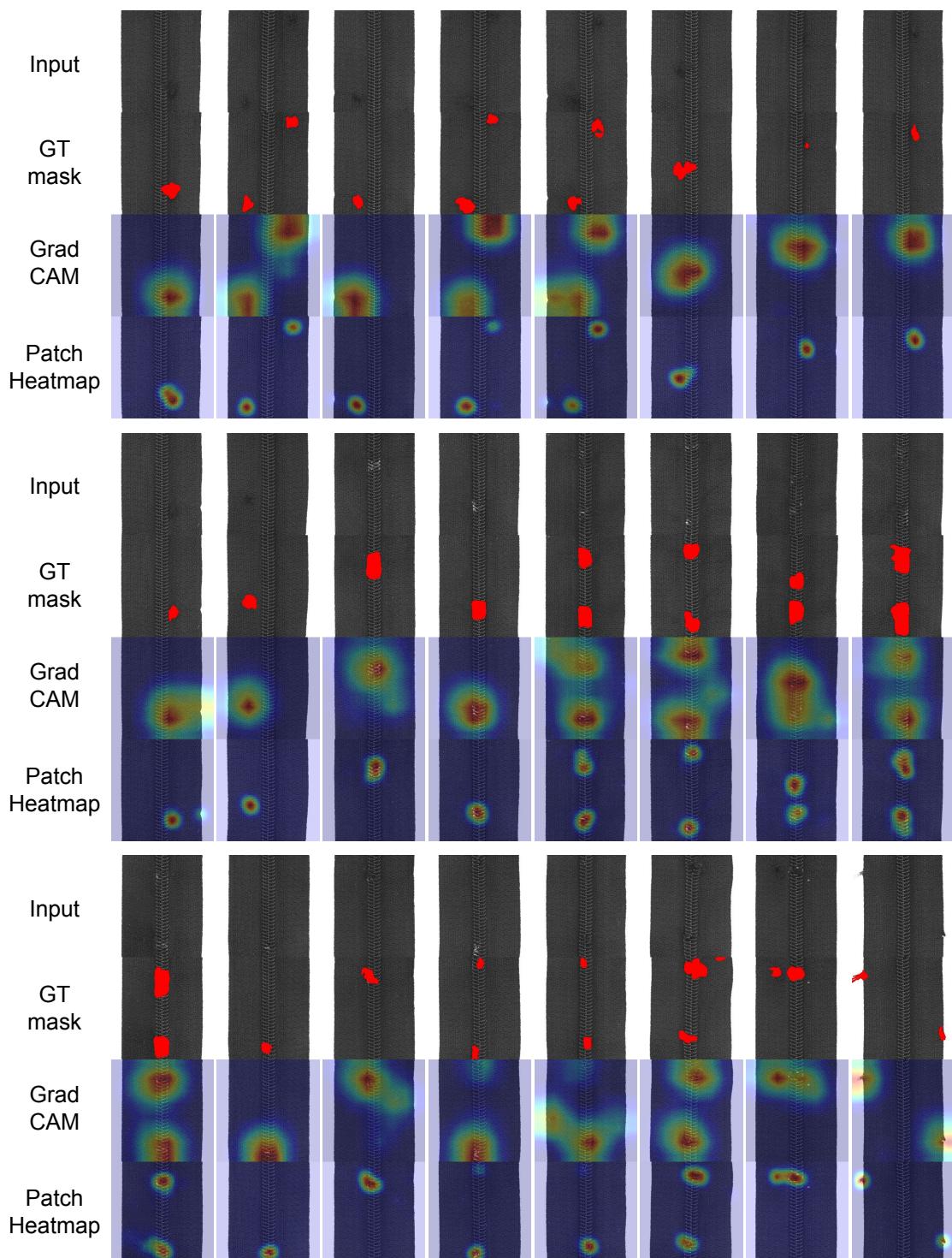


图16：MVTec数据集拉链类别的缺陷定位结果。自上而下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用区块级检测器的热力图。

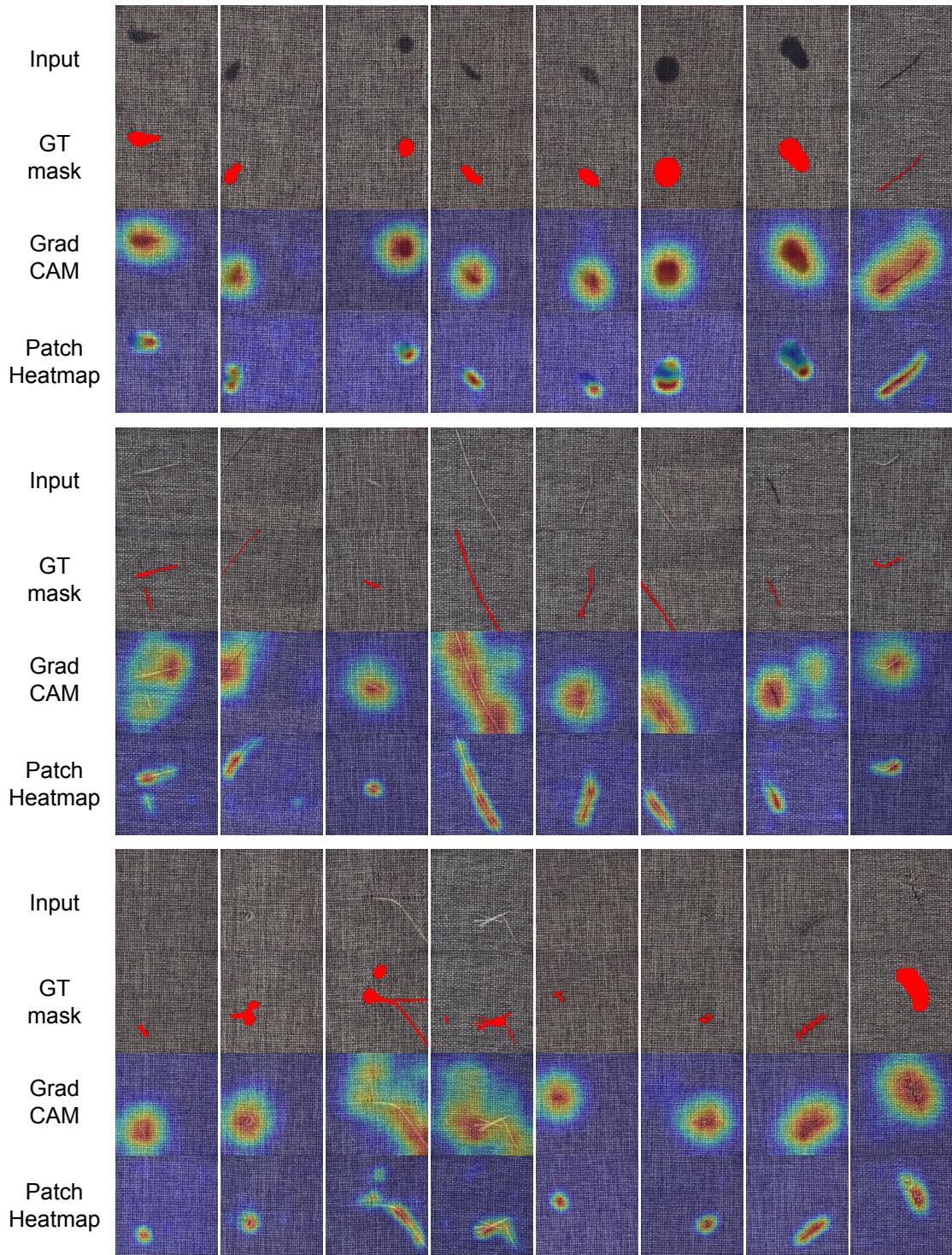


Figure 17: Defect localization on carpet class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

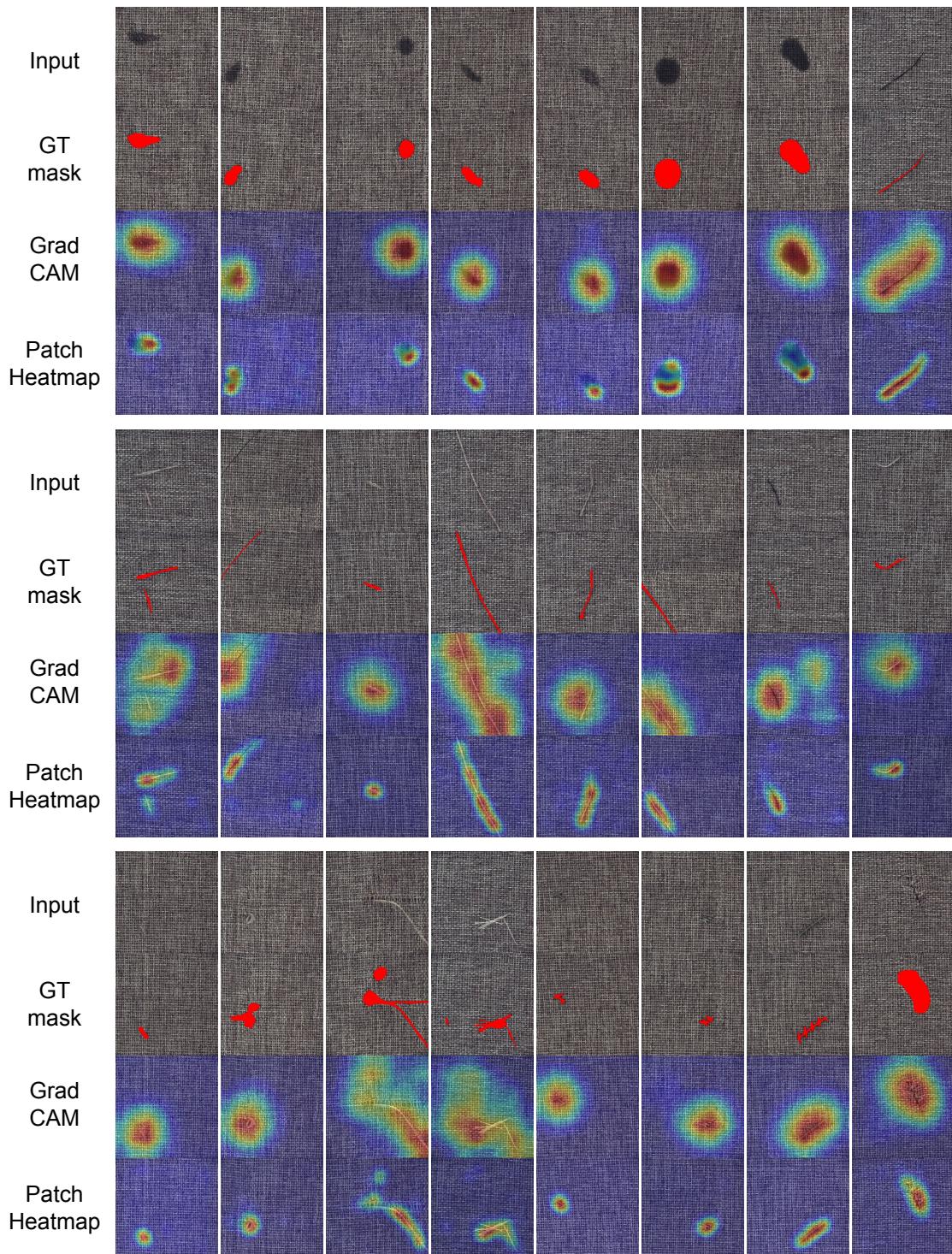


图17：MVTec数据集地毯类别的缺陷定位结果。自上而下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

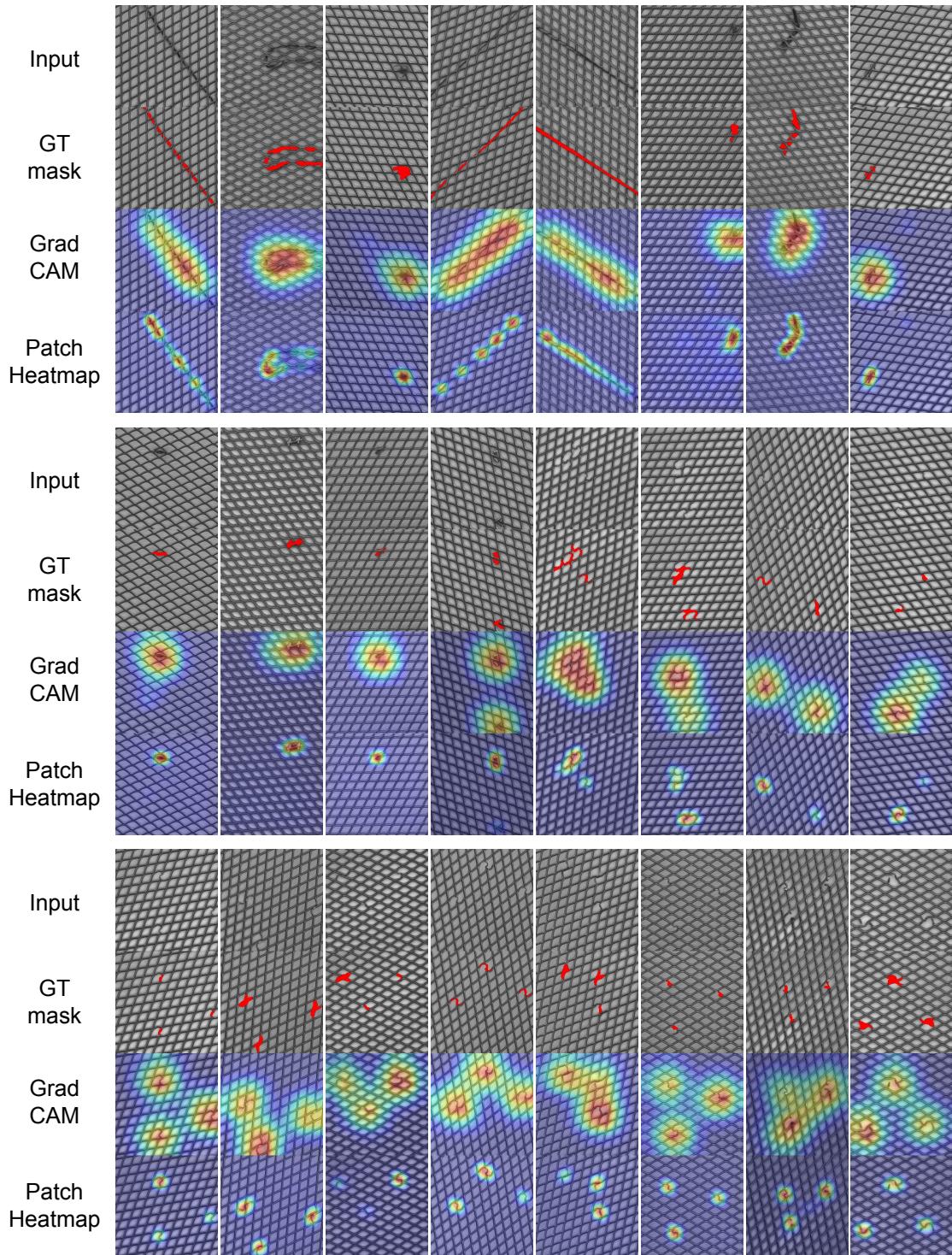


Figure 18: Defect localization on grid class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

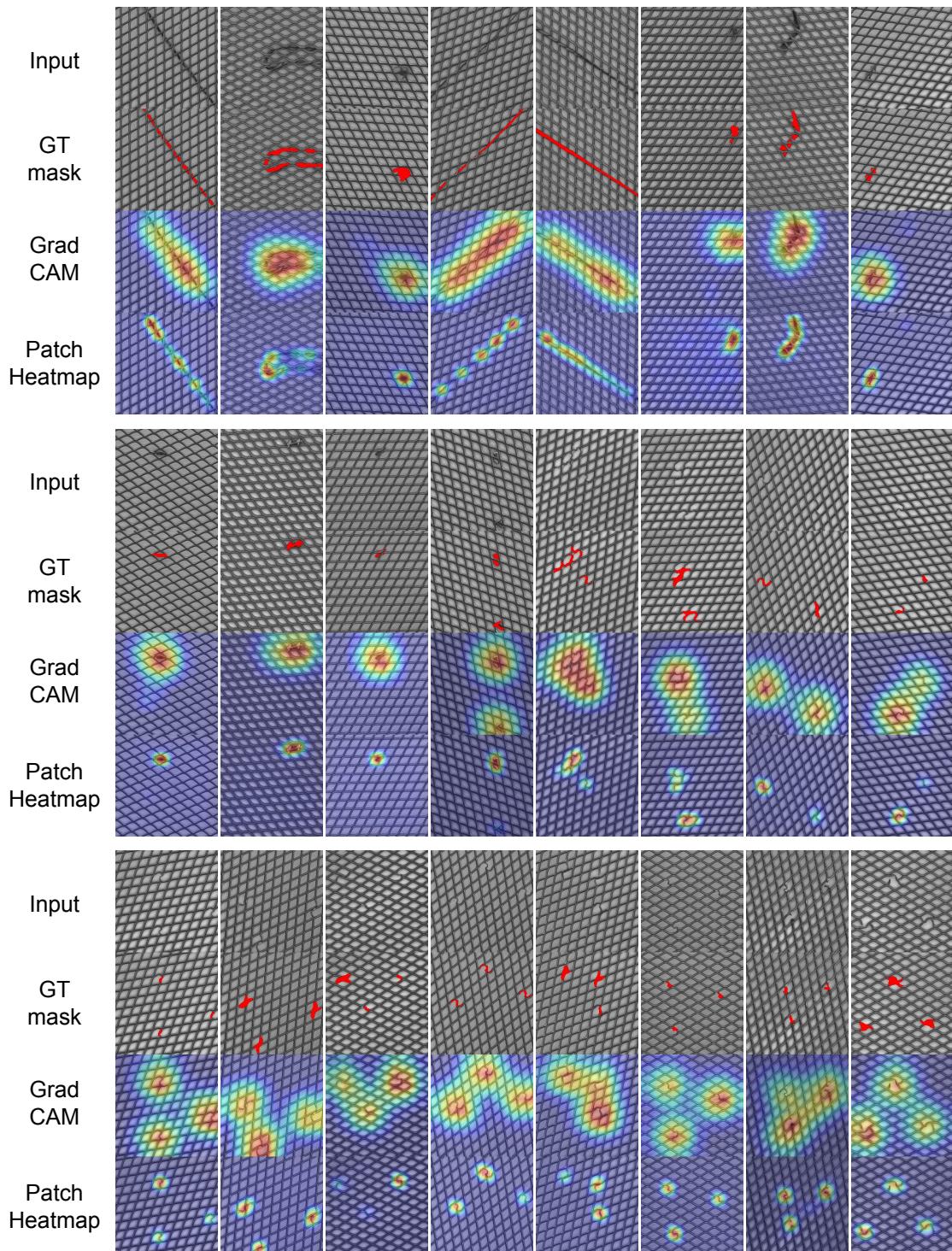


图18：MVTec数据集网格类别的缺陷定位。从上至下分别为：输入图像、带有红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

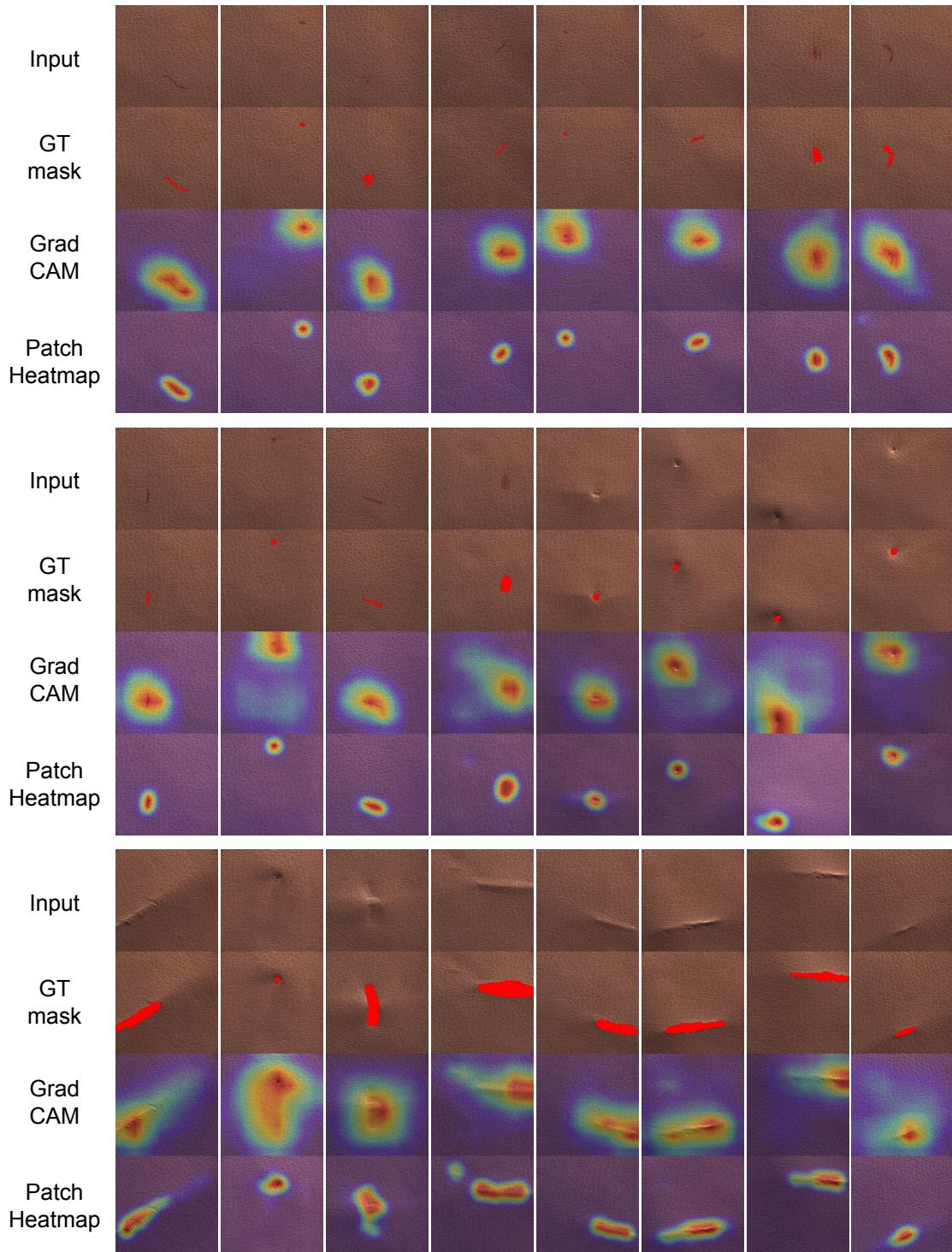


Figure 19: Defect localization on leather class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

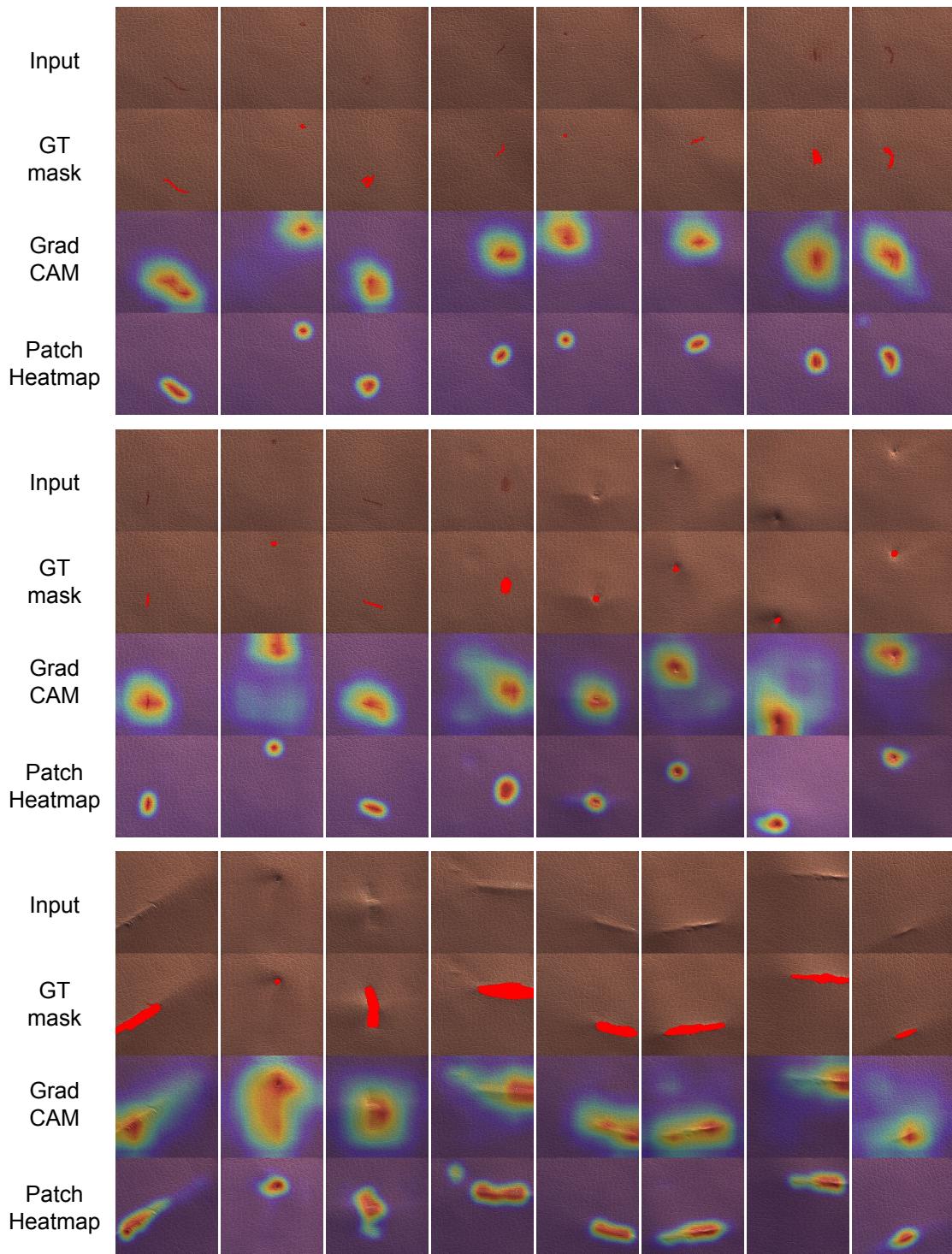


图19：MVTec数据集皮革类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

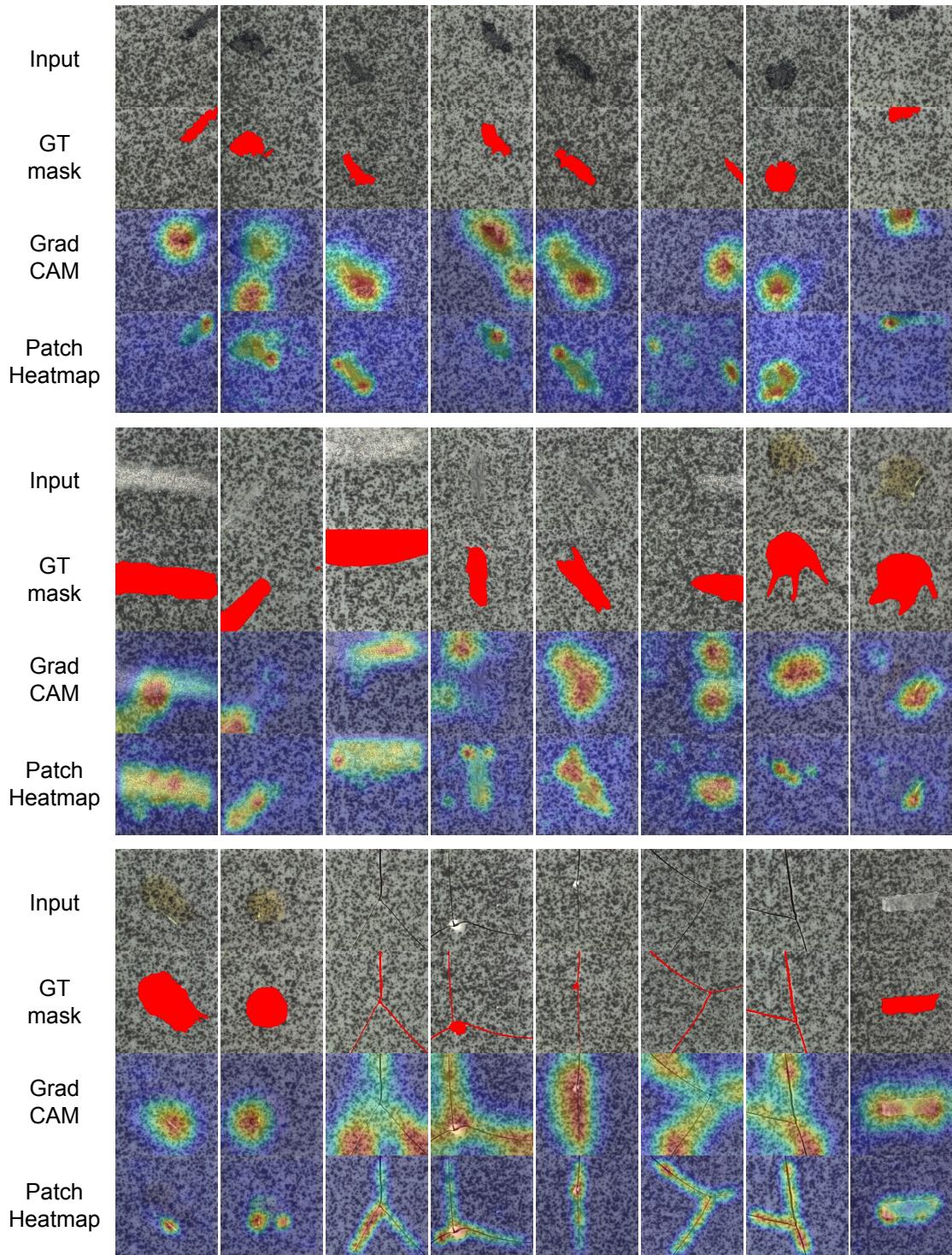


Figure 20: Defect localization on tile class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

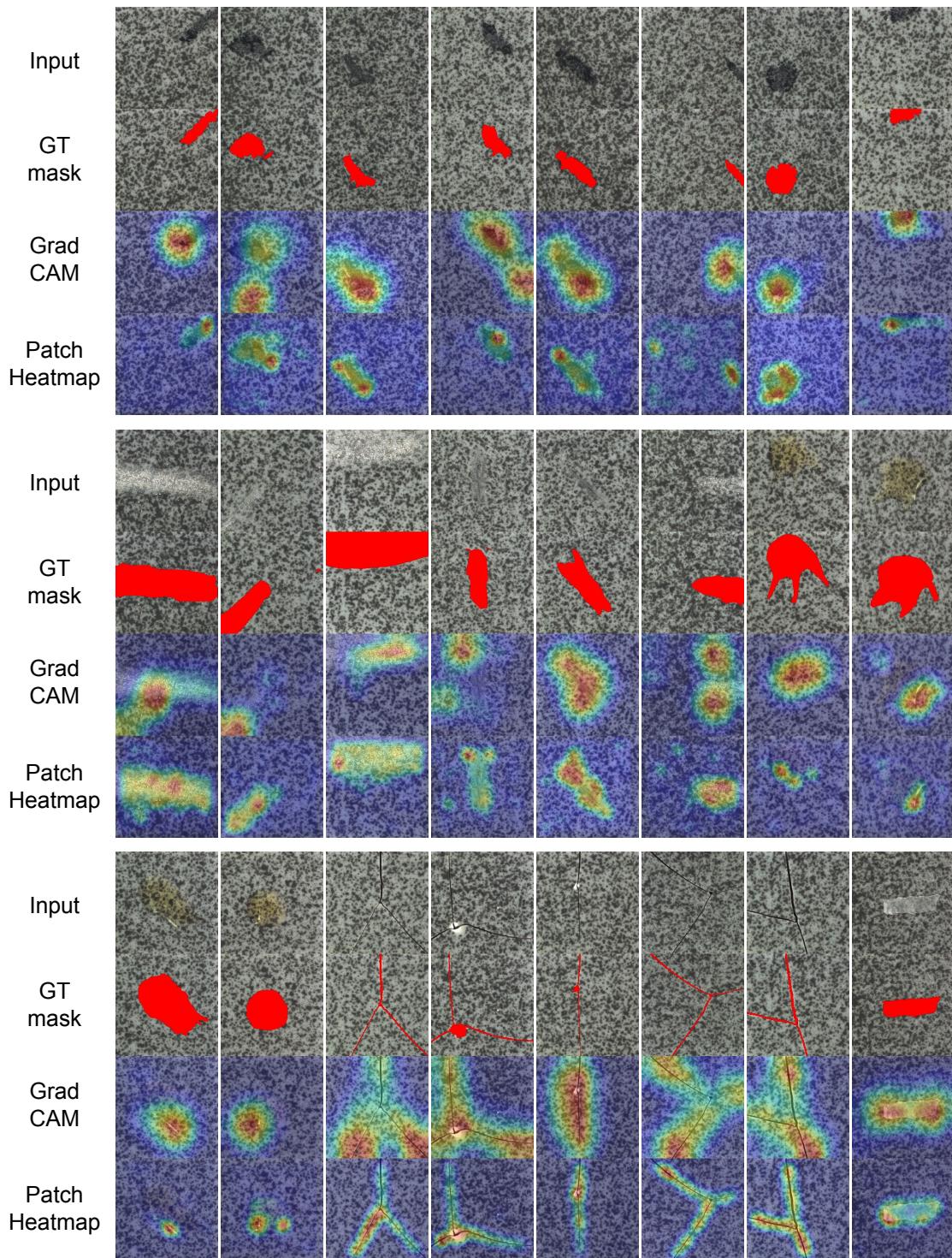


图20：MVTec数据集瓷砖类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。

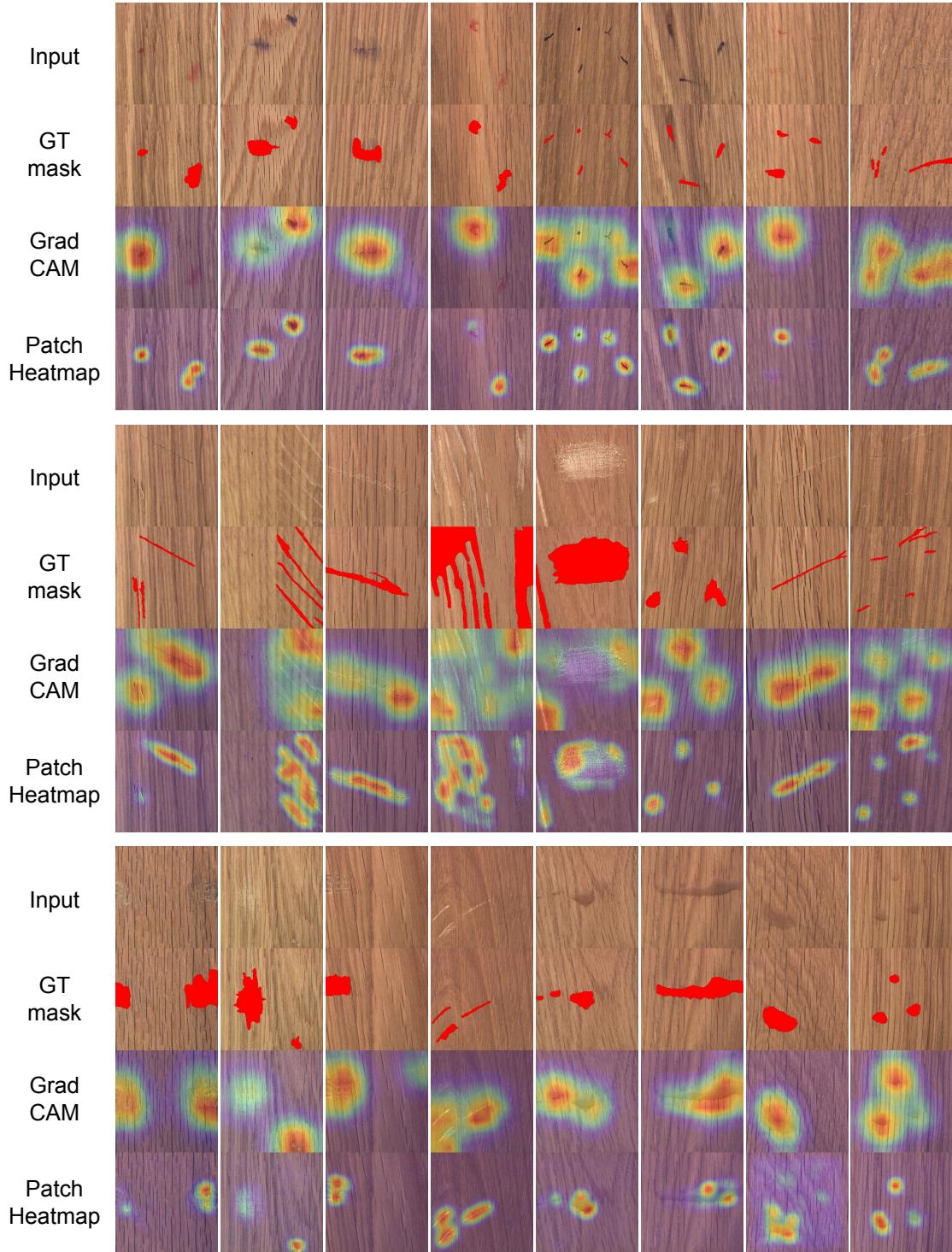


Figure 21: Defect localization on wood class of MVTec dataset. From top to bottom, input images, those with ground-truth localization mask in red, GradCAM results using image-level detector, and heatmaps using patch-level detector.

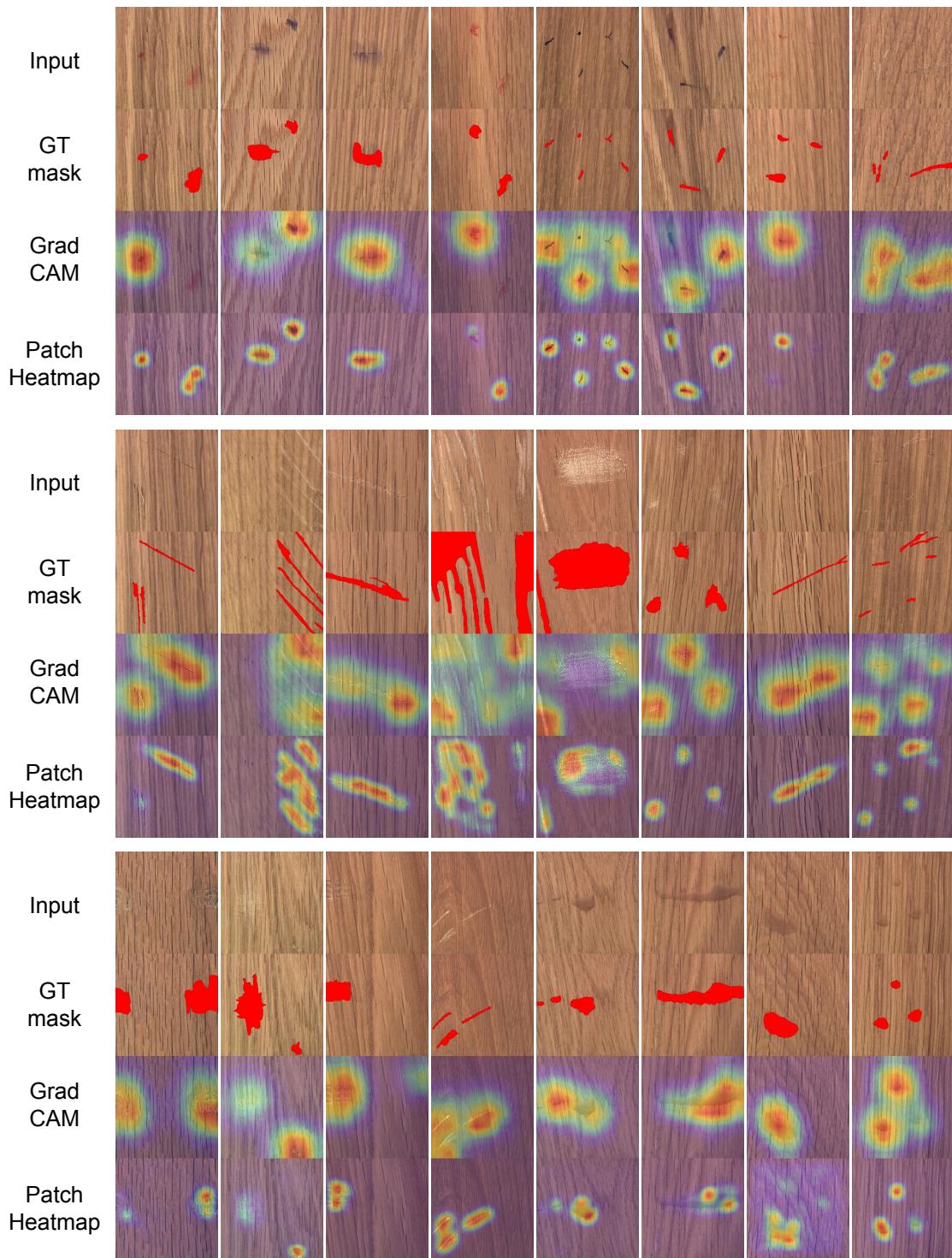


图21：MVTec数据集木材类别的缺陷定位。从上至下分别为：输入图像、带红色真实定位掩码的图像、使用图像级检测器的GradCAM结果、以及使用块级检测器的热力图。