FastFlow: Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows

Jiawei Yu[1*], **Ye Zheng**[2,3*], **Xiang Wang**[1], **Wei Li**[1], **Yushuang Wu**[4], **Rui Zhao**[1], **Liwei Wu**[1]

[1] SenseTime Research
[2]Institute of Computing Technology, Chinese Academy of Sciences
[3]University of Chinese Academy of Sciences [4]The Chinese University of Hong Kong, Shenzhen

Abstract

Unsupervised anomaly detection and localization is crucial to the practical application when collecting and labeling sufficient anomaly data is infeasible. Most existing representation-based approaches extract normal image features with a deep convolutional neural network and characterize the corresponding distribution through non-parametric distribution estimation methods. The anomaly score is calculated by measuring the distance between the feature of the test image and the estimated distribution. However, current methods can not effectively map image features to a tractable base distribution and ignore the relationship between local and global features which are important to identify anomalies. To this end, we propose FastFlow implemented with 2D normalizing flows and use it as the probability distribution estimator. Our FastFlow can be used as a plug-in module with arbitrary deep feature extractors such as ResNet and vision transformer for unsupervised anomaly detection and localization. In training phase, FastFlow learns to transform the input visual feature into a tractable distribution and obtains the likelihood to recognize anomalies in inference phase. Extensive experimental results on the MVTec AD dataset show that FastFlow surpasses previous state-of-the-art methods in terms of accuracy and inference efficiency with various backbone networks. Our approach achieves 99.4% AUC in anomaly detection with high inference efficiency.

1 Introduction

The purpose of anomaly detection and localization in computer vision field is to identify abnormal images and locate abnormal areas, which is widely used in industrial defect detection (Bergmann et al. 2019, 2020), medical image inspection (Philipp Seeböck et al. 2017), security check (Akcay, Atapour-Abarghouei, and Breckon 2018) and other fields. However, due to the low probability density of anomalies, the normal and abnormal data usually show a serious long-tail distribution, and even in some cases, no abnormal samples are available. The drawback of this reality makes it difficult to collect and annotate a large amount of abnormal data for supervised learning in practice. Unsupervised anomaly detection has been proposed to address this problem, which is also denoted as *one-class classification* or *out-of-distribution detection*. That is, we can only use normal samples during training process but need to identify and locate anomalies in testing.

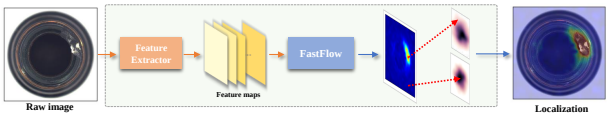

Figure 1: An example of the proposed FastFlow. FastFlow transforms features of the input image from the original distribution to the standard normal distribution. The features of the normal area in the input image fall in the center of the distribution, while the abnormal features are far away from the center of the distribution.

One promising method in unsupervised anomaly detection is using deep neural networks to obtain the features of normal images and model the distribution with some statistical methods, then detect the abnormal samples that have different distributions (Bergman and Hoshen 2020; Rippel, Mertens, and Merhof 2021; Yi and Yoon 2020; Cohen and Hoshen 2020; Defard et al. 2020). Following this methodology, there are two main components: the feature extraction module and the distribution estimation module.

To the *distribution estimation module*, previous approaches used the non-parametric method to model the distribution of features for normal images. For example, they estimated the multidimensional Gaussian distribution (Li et al. 2021; Defard et al. 2020) by calculating the mean and variance for features, or used a clustering algorithm to estimate these normal features by normal clustering (Reiss et al. 2021; Roth et al. 2021). Recently, some works (Rudolph, Wandt, and Rosenhahn 2021; Gudovskiy, Ishizaka, and Kozuka 2021) began to use normalizing flow (Kingma and Dhariwal 2018) to estimate distribution. Through a trainable process that maximizes the log-likelihood of normal image features, they embed normal image features into standard normal distribution and use the probability to identify and locate anomalies. However, original one-dimensional normalizing flow model need to flatten the two-dimensional input feature into a one-dimensional vector to estimate the distribution, which destroys the inherent spatial positional relationship of the two-dimensional image and limits the ability of flow model. In addition, these methods need to extract

*These authors contributed equally.

FastFlow：基于二维归一化流的无监督异常检测与定位

贾伟宇[1]*，叶铮[2,3]*，王翔[1]，李伟[1]，吴雨霜[4]，赵瑞[1]，吴立伟[1]

1 商汤科技研究院 [2]中国科学院计算技术研究所 [3]中国科学院大学 [4]香港中文大学（深圳）

摘要

无监督异常检测与定位在收集和标注足够异常数据不可行时，对实际应用至关重要。大多数现有基于表示的方法通过深度卷积神经网络提取正常图像特征，并通过非参数分布估计方法刻画相应分布。异常分数通过测量测试图像特征与估计分布之间的距离来计算。然而，当前方法无法有效将图像特征映射到易处理的基础分布，且忽略了局部与全局特征间的关系——这对识别异常至关重要。为此，我们提出采用二维标准化流的FastFlow，并将其用作概率分布估计器。我们的FastFlow可作为即插即用模块，与任意深度特征提取器（如ResNet和视觉Transformer）结合，用于无监督异常检测与定位。在训练阶段，FastFlow学习将输入视觉特征转换为易处理的分布，并在推理阶段通过获取似然值来识别异常。在MVTec AD数据集上的大量实验结果表明，FastFlow在不同骨干网络下，在准确率和推理效率方面均超越先前最先进方法。我们的方法在异常检测中实现了99.4%的AUC，并具备高推理效率。

1 引言

计算机视觉领域中异常检测与定位的目的是识别异常图像并定位异常区域，广泛应用于工业缺陷检测（Bergmann等人2019、2020）、医学图像检查（Philipp Seeböck等人2017）、安全检查（Akcay、Atapour-Abarghouei和Breckon 2018）等领域。然而，由于异常的低概率密度，正常与异常数据通常呈现严重的长尾分布，甚至在某些情况下完全无法获得异常样本。这一现实缺陷使得在实践中难以收集和标注大量异常数据进行监督学习。为解决此问题，无监督异常检测方法被提出，亦被记作*one-class classification*或*out-of-distribution detection*。这意味着我们仅能使用正常训练过程中采集样本，但需要在测试时识别并定位异常。

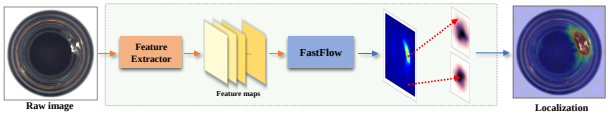

图1：提出的FastFlow示例。FastFlow将输入图像的特征从原始分布转换为标准正态分布。输入图像中正常区域的特征落在分布的中心，而异常特征则远离分布中心。

无监督异常检测中一种有前景的方法是使用深度神经网络获取正常图像的特征，并通过统计方法建模其分布，进而检测出具有不同分布的异常样本（Bergman and Hoshen 2020; Rippel, Mertens, and Merhof 2021; Yi and Yoon 2020; Cohen and Hoshen 2020; Defard et al. 2020）。遵循这一方法论，主要包含两个核心组件：特征提取模块和分布估计模块。

对于*distribution estimation module*，先前的方法使用非参数方法来建模正常图像特征的分布。例如，他们通过计算特征的均值和方差来估计多维高斯分布（Li等人，2021；Defard等人，2020），或者使用聚类算法通过正常聚类来估计这些正常特征（Reiss等人，2021；Roth等人，2021）。最近，一些工作（Rudolph, Wandt和Rosenhahn，2021；Gudovskiy, Ishizaka和Kozuka，2021）开始使用标准化流（Kingma和Dhariwal，2018）来估计分布。通过一个最大化正常图像特征对数似然的可训练过程，他们将正常图像特征嵌入到标准正态分布中，并利用概率来识别和定位异常。然而，原始的一维标准化流模型需要将二维输入特征展平为一维向量以估计分布，这破坏了二维图像固有的空间位置关系，限制了流模型的能力。此外，这些方法需要提取

*These authors contributed equally.

the features for a large number of patches in images through the sliding window method, and detect anomalies for each patch, so as to obtain anomaly location results, which leads to high complexity in inference and limits the practical value of these methods. To address above problems, we propose the FastFlow which extend the original normalizing flow to two-dimensional space. We use fully convolutional network as the subnet in our flow model and it can maintain the relative position of the space to improve the performance of anomaly detection. At the same time, it supports the end-to-end inference of the whole image and directly outputs the anomaly detection and location results at once to improve the inference efficiency.

To the *feature extraction module* in anomaly detection, besides using CNN backbone network such as ResNet (He et al. 2016) to obtain discriminant features, most of the existing work (Defard et al. 2020; Reiss et al. 2021; Rudolph, Wandt, and Rosenhahn 2021; Gudovskiy, Ishizaka, and Kozuka 2021) focuses on how to reasonably use multi-scale features to identify anomalies at different scales and semantic levels, and achieve pixel-level anomaly localization through sliding window method. The importance of the correlation between global information and local anomalies (Yan et al. 2021; Wang et al. 2021) can not be fully utilized, and the sliding window method needs to test a large number of image patches with high computational complexity. To address the problems, we use FastFlow to obtain learnable modeling of global and local feature distributions through an end-to-end testing phase, instead of designing complicated multi-scale strategy and using sliding window method. We conducted experiments on two types of backbone networks: vision transformers and CNN. Compared with CNN, vision transformers can provide a global receptive field and make better use of global and local information while maintaining semantic information in different depths. Therefore, we only use the feature of one certain layer in vision transform. Replacing CNN with vision transformer seems trivial, but we found that performing this simple replacement in other methods actually degrade the performance, but our 2D flow achieve competitive results when using CNN. Our FastFlow has stronger global and local modeling capabilities, so it can better play the effectiveness of the transformer.

As shown in Figure 1, in our approach, we first extract visual features by the feature extractor and then input them into the FastFlow to estimate the probability density. In training stage, our FastFlow is trained with normal images to transform the original distribution to a standard normal distribution in a 2D manner. In inference, we use the probability value of each location on the two-dimensional feature as the anomaly score.

To summarize, the main contributions of this paper are:

- We propose a 2D normalizing flow denoted as FastFlow for anomaly detection and localization with fully convolutional networks and two-dimensional loss function to effectively model global and local distribution.
- We design a lightweight network structure for FastFlow with the alternate stacking of large and small convolution kernels for all steps. It adopts an end-to-end inference phase and has high efficiency.
- The proposed FastFlow model can be used as a plug-in model with various different feature extractors. The experimental results in MVTec anomaly detection dataset (Bergmann et al. 2019) show that our method outperforms the previous state-of-the-art anomaly detection methods in both accuracy and reasoning efficiency.

2 Related Work

2.1 Anomaly Detection Methods

Existing anomaly detection methods can be summarized as reconstruction-based and representation-based methods. Reconstruction-based methods (Bergmann et al. 2019; Gong et al. 2019; Perera, Nallapati, and Xiang 2019) typically utilize generative models like auto-encoders or generative adversarial networks to encode and reconstruct the normal data. These methods hold the insights that the anomalies can not be reconstructed since they do not exist at the training samples. Representation-based methods extract discriminative features for normal images (Ruff et al. 2018; Bergman and Hoshen 2020; Rippel, Mertens, and Merhof 2021; Rudolph, Wandt, and Rosenhahn 2021) or normal image patches (Yi and Yoon 2020; Cohen and Hoshen 2020; Reiss et al. 2021; Gudovskiy, Ishizaka, and Kozuka 2021) with deep convolutional neural network, and establish distribution of these normal features. Then these methods obtain the anomaly score by calculating the distance between the feature of a test image and the distribution of normal features. The distribution is typically established by modeling the Gaussian distribution with mean and variance of normal features (Defard et al. 2020; Li et al. 2021), or the kNN for the entire normal image embedding (Reiss et al. 2021; Roth et al. 2021). We follow the methodology in representation-based method which extract the visual feature from vision transformer or ResNet and establish the distribution through FastFlow model.

2.2 Feature extractors for Anomaly Detection

With the development of deep learning, recent unsupervised anomaly detection methods use deep neural networks as feature extractors, and produce more promising anomaly results. Most of them (Cohen and Hoshen 2020; Defard et al. 2020; Roth et al. 2021) use ResNet (He et al. 2016) to extract distinguish visual features. Some work has also begun to introduce ViT (Dosovitskiy et al. 2020) into unsupervised anomaly detection fields, such as VT-ADL (Mishra et al. 2021) uses vision transformer as backbone in a generated-based way. ViT has a global receptive field and can learn the relationship between global and local better. DeiT (Touvron et al. 2021a) and CaiT (Touvron et al. 2021b) are two typical models for ViT. DeiT introduces a teacher-student strategy specific to transformers, which makes image transformers learn more efficiently and got a new state-of-the-art performance. CaiT proposes a simple yet effective architecture designed in the spirit of encoder/decoder architecture and demonstrates that transformer models offer a competitive alternative to the best convolutional neural networks. In

通过滑动窗口方法提取图像中大量图像块的特征，并对每个图像块进行异常检测，从而获得异常定位结果，这导致推理过程复杂度高，限制了这些方法的实用价值。为解决上述问题，我们提出了FastFlow，将原始标准化流扩展至二维空间。我们在流模型中使用全卷积网络作为子网络，该网络能保持空间相对位置关系以提升异常检测性能。同时，该方法支持对整个图像进行端到端推理，直接一次性输出异常检测与定位结果，从而显著提升推理效率。

在异常检测的*feature extraction module*中，除了使用ResNet（He等人，2016）等CNN骨干网络获取判别性特征外，现有大多数工作（Defard等人，2020；Reiss等人，2021；Rudolph、Wandt和Rosenhahn，2021；Gudovskiy、Ishizaka和Kozuka，2021）主要关注如何合理利用多尺度特征来识别不同尺度和语义层次的异常，并通过滑动窗口方法实现像素级异常定位。全局信息与局部异常之间的关联重要性（Yan等人，2021；Wang等人，2021）未能得到充分利用，且滑动窗口方法需要测试大量图像块，计算复杂度较高。为解决这些问题，我们采用FastFlow通过端到端的测试阶段实现对全局和局部特征分布的可学习建模，而非设计复杂的多尺度策略或使用滑动窗口方法。我们在两类骨干网络上进行了实验：视觉Transformer和CNN。与CNN相比，视觉Transformer能提供全局感受野，在保持不同深度语义信息的同时更好地利用全局与局部信息。因此，我们仅使用视觉Transformer中特定单层的特征。将CNN替换为视觉Transformer看似简单，但我们发现其他方法进行这种简单替换反而会降低性能，而我们的二维流模型在使用CNN时已取得具有竞争力的结果。由于FastFlow具备更强的全局与局部建模能力，因此能更好地发挥Transformer的效能。

如图1所示，在我们的方法中，首先通过特征提取器提取视觉特征，然后将其输入FastFlow以估计概率密度。在训练阶段，我们的FastFlow使用正常图像进行训练，以二维方式将原始分布转换为标准正态分布。在推理过程中，我们使用二维特征上每个位置的概率值作为异常得分。

总而言之，本文的主要贡献在于：

- 我们提出了一种称为FastFlow的二维归一化流，用于异常检测和定位，它结合了全卷积网络和二维损失函数，以有效建模全局和局部分布。
- 我们为FastFlow设计了一个轻量级网络结构，采用大小卷积交替堆叠的方式。所有步骤的内核。它采用端到端的推理阶段，并具有高效率。
- 提出的FastFlow模型可作为插件模型与多种不同的特征提取器配合使用。在MVTec异常检测数据集（Bergmann等人，2019年）上的实验结果表明，我们的方法在准确性和推理效率方面均优于先前最先进的异常检测方法。

2 相关工作

2.1 异常检测方法

现有的异常检测方法可归纳为基于重构和基于表示的方法。基于重构的方法（Bergmann等人2019；Gong等人2019；Perera、Nallapati和Xiang 2019）通常利用自编码器或生成对抗网络等生成模型对正常数据进行编码与重构。这类方法的核心理念在于：异常样本因未出现在训练数据中而无法被准确重构。基于表示的方法则通过深度卷积神经网络提取正常图像（Ruff等人2018；Bergman和Hoshen 2020；Rippel、Mertens和Merhof 2021；Rudolph、Wandt和Rosenhahn 2021）或正常图像块（Yi和Yoon 2020；Cohen和Hoshen 2020；Reiss等人2021；Gudovskiy、Ishizaka和Kozuka 2021）的判别性特征，并建立这些正常特征的分布模型。此类方法通过计算测试图像特征与正常特征分布之间的距离来获得异常分数。分布模型通常通过两种方式构建：一是基于正常特征的均值与方差建立高斯分布模型（Defard等人2020；Li等人2021），二是对全体正常图像嵌入进行k近邻建模（Reiss等人2021；Roth等人2021）。我们遵循基于表示的方法论，通过视觉Transformer或ResNet提取视觉特征，并利用FastFlow模型建立特征分布。

2.2 异常检测的特征提取器

随着深度学习的发展，近期的无监督异常检测方法采用深度神经网络作为特征提取器，并取得了更具前景的异常检测结果。其中大多数方法（Cohen与Hoshen 2020；Defard等人2020；Roth等人2021）使用ResNet（He等人2016）来提取区分性视觉特征。部分研究也开始将ViT（Dosovitskiy等人2020）引入无监督异常检测领域，例如VT-ADL（Mishra等人2021）以生成式方法将视觉Transformer作为主干网络。ViT具有全局感受野，能更好地学习全局与局部特征间的关系。DeiT（Touvron等人2021a）与CaiT（Touvron等人2021b）是ViT的两种典型模型：DeiT引入了针对Transformer设计的师生策略，使图像Transformer能更高效学习并达到新的最优性能；CaiT则基于编码器/解码器架构思想提出简洁有效的设计，证明Transformer模型能成为最佳卷积神经网络的有力竞争者。

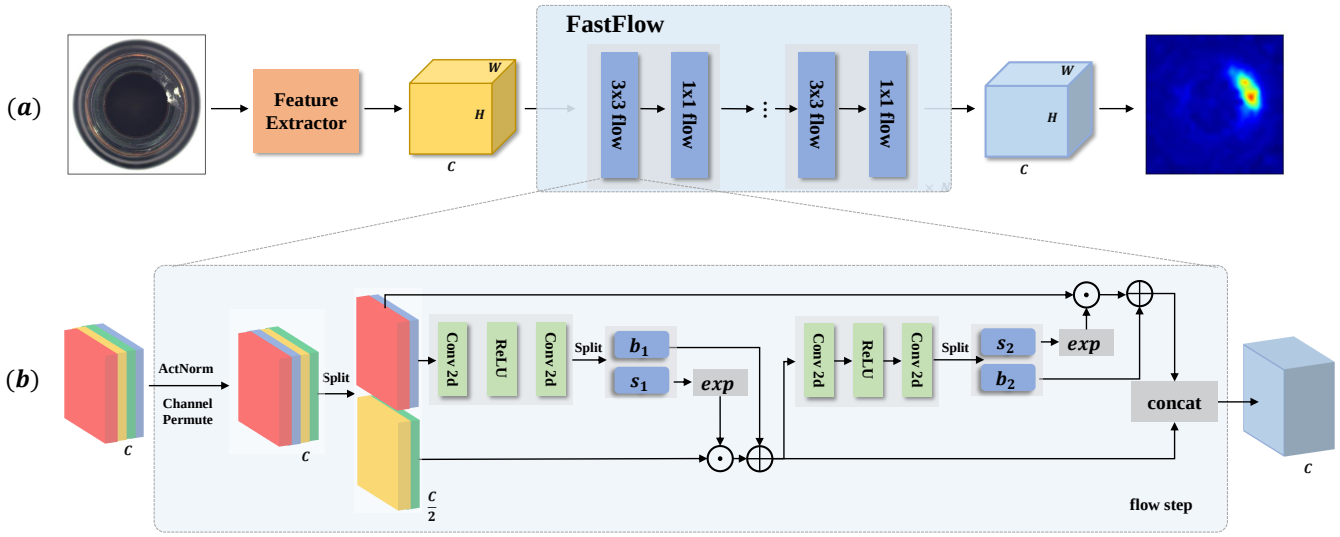

Figure 2: (a) the whole pipeline for unsupervised anomaly detection and localization in our method, which consists of a feature extractor and our FastFlow model. We can use an arbitrary network as the feature extractor such as CNN or vision transformer. FastFlow is alternatly stacked by the "3×3" and "1×1" flow. (b) one flow step for our FastFlow, the "Conv 2d" can be 3×3 or 1×1 convolution layer for 3×3 or 1×1 flow, respectively.

this paper, we use various networks belonging to CNN and ViT to prove the universality of our method.

2.3 Normalizing Flow

Normalizing Flows (NF) (Rezende and Mohamed 2015) are used to learn transformations between data distributions with special property that their transform process is bijective and the flow model can be used in both directions. Real-NVP (Dinh, Sohl-Dickstein, and Bengio 2016) and Glow (Kingma and Dhariwal 2018) are two typical methods for NF, in which both forward and reverse processes can be processed quickly. NF is generally used to generate data from variables sampled in a specific probability distribution, such as images or audios. Recently, some work (Rudolph, Wandt, and Rosenhahn 2021; Gudovskiy, Ishizaka, and Kozuka 2021) began to use it for unsupervised anomaly detection and localization. DifferNet (Rudolph, Wandt, and Rosenhahn 2021) achieved good image level anomaly detection performance by using NF to estimate the precise likelihood of test images. Unfortunately, this work failed to obtain the exact anomaly localization results since they flattened the outputs of feature extractor. CFLOW-AD (Gudovskiy, Ishizaka, and Kozuka 2021) proposes to use hard code position embedding to leverage the distribution learned by NF, which probably underperforms at more complicated datasets.

3 Methodology

In this section, we introduce the pipeline of our method and the architecture of the FastFlow, as shown in Figure 2. We first set up the problem definition of unsupervised anomaly detection, and introduce the basic methodology that uses a learnable probability density estimation model in the representation-based method. Then we describe the details of feature extractor and FastFlow models, respectively.

3.1 Problem Definition and Basic Methodology

Unsupervised anomaly detection is also denoted as one-class classification or out-of-distribution detection which requires the model to determines whether the test image is normal or abnormal. Anomaly localization requires a more fine-grained result that gives the anomalies label for each pixel. During the training stage, only normal images were observed, but the normal images and abnormal images simultaneously appear in inference. One of the mainstream methods is representation-based method which extracts discriminative feature vectors from normal images or normal image patches to construct the distribution and calculate anomaly score by the distance between the embedding of a test image and the distribution. The distribution is typically characterized by the center of an n-sphere for the normal image, the Gaussian distribution of normal images, or the normal embedding cluster stored in the memory bank obtained from KNN. After extract the features of the training dataset $D = \{x_1, x_2, \cdots, x_N\}$ where $x_i, i = 1, 2, \cdots, N$ are samples from the distribution $p_X(x)$, a representation-based anomaly detection model $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ aims to learn the parameter θ in the parameter space Θ to map all x_i from the raw distribution $p_X(x)$ into the same distribution $p_Z(z)$, with anomalous pixels or instances mapped out of the distribution. In our method, we follow this methodology and propose FastFlow P_θ to project the high-dimensional visual features of normal images extracted from typical backbone networks into the standard normal distribution.

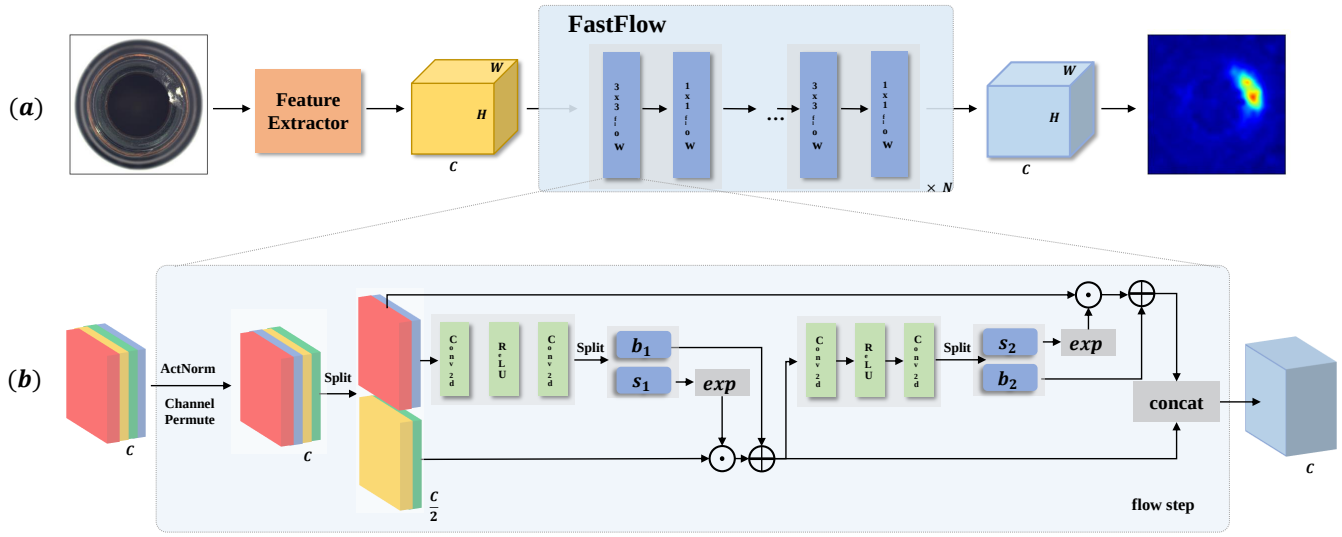

图2：(a) 我们方法中无监督异常检测与定位的整体流程，包含特征提取器和FastFlow模型。特征提取器可采用任意网络架构（如CNN或视觉Transformer）。FastFlow通过交替堆叠“3×3”与“1×1”流模块构建。(b) FastFlow的单流步骤示意图，其中“Conv 2d”可分别对应3×3卷积层（用于3×3流）或1×1卷积层（用于1×1流）。

在本文中，我们使用属于CNN和ViT的各种网络来证明我们方法的普适性。

2.3 归一化流

归一化流（NF）（Rezende and Mohamed 2015）用于学习数据分布之间的变换，其具有变换过程可逆的特殊性质，因此流模型可以双向使用。Real-NVP（Dinh, Sohl-Dickstein, and Bengio 2016）和Glow（Kingma and Dhariwal 2018）是NF的两种典型方法，其中正向和反向过程均可快速处理。NF通常用于从特定概率分布（如图像或音频）中采样的变量生成数据。近年来，一些工作（Rudolph, Wandt, and Rosenhahn 2021; Gudovskiy, Ishizaka, and Kozuka 2021）开始将其用于无监督异常检测与定位。DifferNet（Rudolph, Wandt, and Rosenhahn 2021）通过使用NF估计测试图像的精确似然，实现了良好的图像级异常检测性能。然而，由于该方法将特征提取器的输出展平，未能获得精确的异常定位结果。CFLOW-AD（Gudovskiy, Ishizaka, and Kozuka 2021）提出使用硬编码位置嵌入来利用NF学习到的分布，但在更复杂的数据集上可能表现不佳。

3 方法论

在本节中，我们将介绍我们方法的流程和FastFlow的架构，如图2所示。我们首先建立无监督异常检测的问题定义，并介绍使用可学习概率密度估计的基本方法。在基于表示的方法中的模型。接着我们分别详细介绍了特征提取器和FastFlow模型的细节。

3.1 问题定义与基本方法

无监督异常检测也被称为单类分类或分布外检测，它要求模型判断测试图像是正常还是异常。异常定位则需要更细粒度的结果，为每个像素提供异常标签。在训练阶段，仅观察到正常图像，而正常图像和异常图像在推理过程中同时出现。主流方法之一是表示学习方法，该方法从正常图像或正常图像块中提取判别性特征向量以构建分布，并通过测试图像的嵌入与分布之间的距离计算异常分数。该分布通常由正常图像的n维球体中心、正常图像的高斯分布或通过KNN从记忆库中获取的正常嵌入聚类来表征。在提取训练数据集的特征向量 $D = \{x_1, x_2, \cdots, x_N\}$后（其中$x_i, i =$、、、$, \cdots, N$为分布$p_X(x)$的样本），基于表示的异常检测模型 $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$旨在学习参数空间$\Theta$中的参数$\theta$，将所有原始分布$p_X(x)$中的样本$x_i$映射到同一分布$p_Z(z)$，并将异常像素或实例映射至该分布之外。在我们的方法中，我们遵循这一方法论，提出FastFlowP_θ，将典型骨干网络提取的正常图像高维视觉特征投影到标准正态分布中。

3.2 Feature Extractor

In the whole pipeline of our method, we first extract the representative feature from the input image through ResNet or vision transformers. As mentioned in the Sec 1, one of significant challenges in the anomaly detection task is the global relation grasped to distinguish those abnormal regions from other local parts. Therefore, when using vision transformer (ViT) (Dosovitskiy et al. 2020) as the feature extractor, we only use the feature of one certain layer because ViT has stronger ability to capture the relationship between local patches and the global feature. For ResNet, we directly use the features of the last layer in the first three blocks, and put these features into three corresponding FastFlow model.

3.3 2D Flow Model

As shown in Figure 2, our 2D flow $f : X \rightarrow Z$ is used to project the image features $x \in p_X(x)$ into the hidden variable $z \in p_Z(z)$ with a bijective invertible mapping. For this bijection function, the change of the variable formula defines the model distribution on X by:

$$p_X(x) = p_Z(z) \left| \mathbf{det}(\frac{\partial z}{\partial x}) \right| \tag{1}$$

We can estimate the log likelihoods for image features from $p_Z(z)$ by:

$$\begin{aligned} \log p_X(x) &= \log p_Z(z) + \log \left| \mathbf{det}(\frac{\partial z}{\partial x}) \right| \\ &= \log p_Z(f_\theta(x)) + \log \left| \mathbf{det}(\frac{\partial f_\theta(x)}{\partial x}) \right|, \end{aligned} \tag{2}$$

where $z \sim \mathcal{N}(o, I)$ and the $\frac{\partial \mathbf{f}_\theta(\mathbf{x})}{\partial \mathbf{x}}$ is the Jacobian of a bijective invertible flow model that $z = f_\theta(x)$ and $x = f_\theta^{-1}(z)$, θ is parameters of the 2D flow model. In inference, the features of anomalous images should be out of distribution and hence have lower likelihoods than normal images and the likelihood can be used as the anomaly score. Specifically, we sum the two-dimensional probabilities of each channel to get the final probability map and upsample it to the input image resolution using bilinear interpolation. In actual implementation, our flow model f_{2d} is constructed by stacking multiple invertible transformations blocks f_i in a sequence that:

$$X \xrightarrow{f_1} H_1 \xrightarrow{f_2} H_2 \xrightarrow{f_3} \cdots \xrightarrow{f_K} Z,$$

and

$$X \xleftarrow{f_1^{-1}} H_1 \xleftarrow{f_2^{-1}} H_2 \xleftarrow{f_3^{-1}} \cdots \xleftarrow{f_K^{-1}} Z,$$

where the 2D flow model is $f_{2d} = f_1 \circ f_2 \circ f_3 \circ \cdots \circ f_K$ with K transformation blocks. Each transformation block f_i consists of multiple steps. Following (Dinh, Krueger, and Bengio 2014), we employ affine coupling layers in each block, and each step is formulated as follow:

$$\begin{aligned} y_a, y_b &= \text{split}(y) \\ y'_a &= y_a \\ y'_b &= s(y_a) \odot y_b + b(y_a) \\ y' &= \text{concat}(y'_a, y'_b), \end{aligned} \tag{3}$$

where $s(y_a)$ and $b(y_a)$ are outputs of two neural networks. The split$(\cdot)$ and concat$(\cdot)$ functions perform splitting and concatenation operations along the channel dimension. The two subnets s$(\cdot)$ and b$(\cdot)$ are usually implemented as fully connected networks in original normalizing flow model and need to flatten and squeeze the input visual features from 2D to 1D which destroy the spatial position relationship in the feature map. To convert the original normalizing flow to 2D manner, we adopt two-dimensional convolution layer in the default subnet to reserve spatial information in the flow model and adjust the loss function accordingly. In particular, we adopt a fully convolutional network in which 3×3 convolution and 1×1 convolution appear alternately, which reserves spatial information in the flow model.

4 Experiments

4.1 Datasets and Metrics

We evaluate the proposed method on three datasets: MVTec AD (Bergmann et al. 2019), BTAD (Mishra et al. 2021) and CIFAR-10 (Krizhevsky, Hinton et al. 2009). MVTec AD and BTAD are both industrial anomaly detection datasets with pixel-level annotations, which are used for anomaly detection and localization. CIFAR-10 is built for image classification and we use it to do anomaly detection. Following the previous works, we choose one of the categories as normal, and the rest as abnormal. The anomalies in these industrial datasets are finer than those in CIFAR-10, and the anomalies in CIFAR-10 are more related to the semantic high-level information. For example, the anomalies in MVTec AD are defined as small areas, while the anomalies in CIFAR-10 dataset are defined as different object categories. Under the unsupervised setting, we train our model for each category with its respective normal images and evaluate it in test images that contain both normal and abnormal images.

The performance of the proposed method and all comparable methods is measured by the area under the receiver operating characteristic curve (AUROC) at image or pixel level. For the detection task, evaluated models are required to output single score (anomaly score) for each input test image. In the localization task, methods need to output anomaly scores for every pixel.

4.2 Complexity Analysis

We make a complexity analysis of FastFlow and other methods from aspects of inference speed, additional inference time and additional model parameters, "additional" refers to not considering the backbone itself. The hardware configuration of the machine used for testing is Intel(R) Xeon(R) CPU E5-2680 V4@2.4GHZ and NVIDIA GeForce GTX 1080Ti. SPADE and Patch Core perform KNN clustering between each test feature of each image patch and the gallery features of normal image patches, and they do not need to introduce parameters other than backbone. CFlow avoids the time-consuming k-nearest-neighbor-search process, but it still needs to perform testing phase in the form of a slice window. Our FastFlow adopts an end-to-end inference phase which has high efficiency of inference. The analysis results are shown in Table 1, we can observe that our method is up

3.2 特征提取器

在我们方法的整个流程中，我们首先通过ResNet或视觉变换器从输入图像中提取代表性特征。如第1节所述，异常检测任务中的一项重要挑战是把握全局关系，以区分异常区域与其他局部部分。因此，当使用视觉变换器（ViT）（Dosovitskiy等人，2020）作为特征提取器时，我们仅使用特定某一层的特征，因为ViT具有更强的能力来捕捉局部图像块与全局特征之间的关系。对于ResNet，我们直接使用前三个块中最后一层的特征，并将这些特征输入三个对应的FastFlow模型。

3.3 二维流动模型

如图2所示，我们的二维流 $f\colon X \rightarrow Z$ 通过双射可逆映射将图像特征 $x \in p_X(x)$ 投影到隐变量 $z \in p_Z(z)$ 中。对于该双射函数，变量公式的变化通过以下方式定义了 X 上的模型分布：

$$p_X(x) = p_Z(z) \left| \mathbf{det}(\frac{\partial z}{\partial x}) \right| \tag{1}$$

我们可以通过以下方式估计来自$p_Z(z)$的图像特征的对数似然：

$$\begin{aligned} \log p_X(x) &= \log p_Z(z) + \log \left| \mathbf{det}(\frac{\partial z}{\partial x}) \right| \\ &= \log p_Z(f_\theta(x)) + \log \left| \mathbf{det}(\frac{\partial f_\theta(x)}{\partial x}) \right|, \end{aligned} \tag{2}$$

其中$z \sim \mathcal{N}(o, I)$和$\frac{\partial \mathbf{f}_\theta(\mathbf{x})}{\partial \mathbf{x}}$是双射可逆流模型的雅可比矩阵，该模型包含$z = f_\theta(x)$和$x = f_\theta^{-1}(z)$，$\theta$是二维流模型的参数。在推理过程中，异常图像的特征应处于分布之外，因此其似然度低于正常图像，该似然度可用作异常分数。具体而言，我们对每个通道的二维概率求和以获得最终概率图，并使用双线性插值将其上采样至输入图像分辨率。在实际实现中，我们的流模型f_{2d}通过按顺序堆叠多个可逆变换块f_i构建：

$$X \xrightarrow{f_1} H_1 \xrightarrow{f_2} H_2 \xrightarrow{f_3} \cdots \xrightarrow{f_K} Z,$$

和

$$X \xleftarrow{f_1^{-1}} H_1 \xleftarrow{f_2^{-1}} H_2 \xleftarrow{f_3^{-1}} \cdots \xleftarrow{f_K^{-1}} Z,$$

其中，二维流模型为$f_{2d} = f_1 \circ f_2 \circ f_3 \circ \cdots \circ f_K$，包含$K$个变换块。每个变换块$f_i$由多个步骤组成。遵循(Dinh, Krueger, and Bengio 2014)的方法，我们在每个块中使用仿射耦合层，每个步骤的公式如下：

$$\begin{aligned} y_a, y_b &= \text{split}(y) \\ y'_a &= y_a \\ y'_b &= s(y_a) \odot y_b + b(y_a) \\ y' &= \text{concat}(y'_a, y'_b), \end{aligned} \tag{3}$$

其中$s(y_a)$和$b(y_a)$是两个神经网络的输出。split(·)和concat(·)函数沿通道维度执行分割与拼接操作。两个子网络s(·)和b(·)在原始标准化流模型中通常实现为全连接网络，需要将输入的视觉特征从二维展平压缩至一维，这会破坏特征图中的空间位置关系。为将原始标准化流转换为二维形式，我们在默认子网络中采用二维卷积层以保留流模型中的空间信息，并相应调整损失函数。具体而言，我们采用全卷积网络架构，其中3×3卷积与1×1卷积交替出现，从而在流模型中保留了空间信息。

4 实验

4.1 数据集与评估指标

我们在三个数据集上评估了所提出的方法：MVTec AD（Bergmann等人，2019年）、BTAD（Mishra等人，2021年）和CIFAR-10（Krizhevsky、Hinton等人，2009年）。MVTec AD和BTAD均为具有像素级标注的工业异常检测数据集，用于异常检测与定位。CIFAR-10最初为图像分类任务构建，我们将其用于异常检测。遵循先前研究的方法，我们选择其中一个类别作为正常样本，其余类别视为异常。这些工业数据集中的异常比CIFAR-10中的异常更为精细，而CIFAR-10中的异常更侧重于语义高层信息。例如，MVTec AD中的异常被定义为小区域缺陷，而CIFAR-10数据集中的异常则被定义为不同物体类别。在无监督设置下，我们使用每个类别的正常图像分别训练模型，并在包含正常与异常图像的测试集上进行评估。

所提方法与所有可比方法的性能通过图像或像素级别的接收者操作特征曲线下面积（AUROC）进行衡量。在检测任务中，评估模型需为每个输入测试图像输出单一分数（异常分数）。在定位任务中，方法需要为每个像素输出异常分数。

4.2 复杂度分析

我们对FastFlow及其他方法在推理速度、额外推理时间和额外模型参数方面进行了复杂度分析，“额外”指的是不考虑主干网络本身。测试所用机器的硬件配置为Intel(R) Xeon(R) CPU E5-2680 V4@2.4GHZ及NVIDIA GeForce GTX 1080Ti。SPADE与Patch Core需对每个图像块的所有测试特征与正常图像块的图库特征进行KNN聚类，且除主干网络外无需引入额外参数。CFlow避免了耗时的k近邻搜索过程，但仍需以滑动窗口形式进行测试阶段。我们的FastFlow采用端到端的推理阶段，具有高效的推理性能。分析结果如表1所示，我们可以观察到本方法最高可达到

Model	FPS	A.d. Time (ms)	A.d. Params (M)	Image-level AUC	Pixel-level AUC
CaiT-M48-distilled					
+ Patch Core	2.39	107	0	97.9	96.5
+ CFlow	2.76	42	10.5	97.7	96.2
+ FastFlow	**3.08**	**9**	14.8	**99.4**	**98.5**
DeiT-base-distilled					
+ Patch Core	15.45	39	0	96.5	97.9
+ CFlow	16.91	34	10.5	95.6	97.9
+ FastFlow	**30.14**	**8**	14.8	**98.7**	**98.1**
ResNet18					
+ SPADE	3.92	250	0	-	-
+ CFlow	20.3	44	5.5	96.8	**98.1**
+ FastFlow	**30.8**	**27**	4.9	**97.9**	97.2
Wide-ResNet50-2					
+ SPADE	0.67	1481	0	96.2	96.5
+ Patch Core	5.88	159	0	99.1	98.1
+ CFlow	14.9	56	81.6	98.3	**98.6**
+ FastFlow	**21.8**	**34**	41.3	**99.3**	98.1

Table 1: Complexity comparison in terms of inference speed (FPS), additional inference time (millisecond) and number of additional parameters (M) for various backbones. A.d. Time means the additional inference time and A.d. Parmas is the number of additional parameters compared with backbone network.

to $10\times$ faster than other methods. Compared with CFlow which also uses flow model, our method achieves $1.5\times$ speedup and $2\times$ parameter reduction. When using vision transformers (deit and cait) as the feature extractor, our FastFlow can achieve 99.4 image-level AUC for anomaly detection which is superior to CFlow and Patch Core. From the perspective of additional inference time, our method achieves up to $4\times$ reduction compared to Cflow and $10\times$ reduction compared to Patch Core. Our FastFlow can still have a competitive performance when using ResNet model as feature extractor.

4.3 Quantitative Results

MVTec AD There are 15 industrial products in MVTec AD dataset (Bergmann et al. 2019), with a total of 5,354 images, among which 10 are objects and the remaining 5 are textures. The training set is only composed of normal images, while the test set is a mixture of normal images and abnormal images. We compare our proposed method with the state-of-the-art anomaly detection works, including SPADE* (Reiss et al. 2021), PatchSVDD (Yi and Yoon 2020), DifferNet (Rudolph, Wandt, and Rosenhahn 2021), Mah.AD (Rippel, Mertens, and Merhof 2021), PaDiM (Defard et al. 2020), Cut Paste (Li et al. 2021), Patch Core (Roth et al. 2021), CFlow (Gudovskiy, Ishizaka, and Kozuka 2021) under the metrics of image-level AUC and pixel-level AUC. The detailed comparison results of all categories are shown in Table 2. We can observe that FastFlow achieves **99.4** AUC on image-level and **98.5** AUC on pixel-level, suppresses all other methods in anomaly detection task.

BTAD BeanTech Anomaly Detection dataset (Mishra et al. 2021) has 3 categories industrial products with 2540 images. The training set consists only of normal images, while the test set is a mixture of normal images and abnormal images. Under the measure of pixel-level AUC, we compare the results of our FastFlow with the results of three methods reported in VT-ADL (Mishra et al. 2021): auto encoder with mean square error, automatic encoder with SSIM loss and VT-ADL. The comparison results are shown in Table 3. We can observe that our FastFlow achieves 97.0 pixel-wise AUC and suppresses other methods as high as 7% AUC.

CIFAR-10 dataset CIFAR-10 has 10 categories with 60000 natural images. Under the setting of anomaly detection, one category is regarded as anomaly and other categories are used as normal data. And we need to train the corresponding model for each class respectively. The AUC scores of our method and other methods are reported in Table 4. Methods for comparison includes OC-SVM (Schölkopf et al. 1999), KDE (Bishop 2006), l_2-AE (Hadsell, Chopra, and LeCun 2006), VAE (An and Cho 2015), Pixel CNN (Oord et al. 2016), LSA (Abati et al. 2019), AnoGAN (Schlegl et al. 2017), DSVDD (Ruff et al. 2018) and OCGAN (Perera, Nallapati, and Xiang 2019). Our method outperforms these comparison methods. The results in three different datasets show that our method can adapt to different anomaly detection settings.

4.4 Ablation Study

To investigate the effectiveness of the proposed FastFlow structure, we design ablation experiments about the convolution kernel selection in subnet. We compare alternately using 3×3 and 1×1 convolution kernel and only using 3×3 kernel under the AUC and inference speed for the subnet with various backbone networks. The results are shown in Table 5. For the backbone network with large model capacities such as CaiT and Wide-ResNet50-2, alternate using 3×3 and 1×1 convolution layer can obtain higher per-

Model	FPS	A.d. Time (ms)	A.d. Params (M)	Image-level AUC	Pixel-level AUC
CaiT-M48-distilled					
+ Patch Core	2.39	107	0	97.9	96.5
+ CFlow	2.76	42	10.5	97.7	96.2
+ FastFlow	**3.08**	**9**	14.8	**99.4**	**98.5**
DeiT-base-distilled					
+ Patch Core	15.45	39	0	96.5	97.9
+ CFlow	16.91	34	10.5	95.6	97.9
+ FastFlow	**30.14**	**8**	14.8	**98.7**	**98.1**
ResNet18					
+ SPADE	3.92	250	0	-	-
+ CFlow	20.3	44	5.5	96.8	**98.1**
+ FastFlow	**30.8**	**27**	4.9	**97.9**	97.2
Wide-ResNet50-2					
+ SPADE	0.67	1481	0	96.2	96.5
+ Patch Core	5.88	159	0	99.1	98.1
+ CFlow	14.9	56	81.6	98.3	**98.6**
+ FastFlow	**21.8**	**34**	41.3	**99.3**	98.1

表1：不同骨干网络在推理速度（FPS）、额外推理时间（毫秒）和额外参数量（M）方面的复杂度对比。A.d. Time 指相比骨干网络增加的推理时间，A.d. Params 指相比骨干网络增加的参数量。

比其他方法快10×倍。与同样使用流模型的CFlow相比，我们的方法实现了1.5×倍的加速和2×倍的参数减少。当使用视觉变换器（deit和cait）作为特征提取器时，我们的FastFlow在异常检测中可以达到99.4的图像级AUC，优于CFlow和Patch Core。从额外推理时间的角度来看，与Cflow相比，我们的方法最多减少了4×倍，与Patch Core相比减少了10×倍。当使用ResNet模型作为特征提取器时，我们的FastFlow仍能保持有竞争力的性能。

4.3 定量结果

MVTec AD 数据集中包含15种工业产品（Bergmann等人，2019年），共计5,354张图像，其中10类为物体，其余5类为纹理。训练集仅由正常图像组成，而测试集则混合了正常图像与异常图像。我们将提出的方法与当前最先进的异常检测工作进行对比，包括SPADE*（Reiss等人，2021年）、PatchSVDD（Yi和Yoon，2020年）、DifferNet（Rudolph、Wandt和Rosenhahn，2021年）、Mah.AD（Rippel、Mertens和Merhof，2021年）、PaDiM（Defard等人，2020年）、Cut Paste（Li等人，2021年）、Patch Core（Roth等人，2021年）以及CFlow（Gudovskiy、Ishizaka和Kozuka，2021年），评估指标为图像级AUC和像素级AUC。所有类别的详细对比结果如表2所示。我们可以观察到，FastFlow在图像级AUC达到99.4，像素级AUC达到98.5，在异常检测任务中超越了所有其他方法。

BTAD BeanTech异常检测数据集（Mishra等人，2021年）包含3类工业产品，共2540张图像。训练集仅由正常图像组成，而测试集则是正常图像与异常图像的混合。正常图像。在像素级AUC的衡量标准下，我们将FastFlow的结果与VT-ADL（Mishra等人，2021年）中报告的三种方法的结果进行了比较：使用均方误差的自编码器、使用SSIM损失的自编码器以及VT-ADL。比较结果如表3所示。我们可以观察到，FastFlow实现了97.0的像素级AUC，并比其他方法高出多达7%的AUC。

CIFAR-10数据集包含10个类别，共60000张自然图像。在异常检测的设置下，其中一个类别被视为异常，其他类别则用作正常数据。我们需要分别为每个类别训练相应的模型。我们的方法与其他方法的AUC分数如表4所示。比较方法包括OC-SVM（Schölkopf等人，1999）、KDE（Bishop，2006）、l_2-AE（Hadsell、Chopra和LeCun，2006）、VAE（An和Cho，2015）、Pixel CNN（Oord等人，2016）、LSA（Abati等人，2019）、AnoGAN（Schlegl等人，2017）、DSVDD（Ruff等人，2018）和OCGAN（Perera、Nallapati和Xiang，2019）。我们的方法优于这些比较方法。在三个不同数据集上的结果表明，我们的方法能够适应不同的异常检测设置。

4.4 消融研究

为了研究提出的FastFlow结构的有效性，我们设计了关于子网络中卷积核选择的消融实验。我们比较了在AUC和推理速度下，交替使用3×3和1×1卷积核与仅使用3×3卷积核在不同骨干网络中的表现。结果如表5所示。对于具有较大模型容量的骨干网络，如CaiT和Wide-ResNet50-2，交替使用3×3和1×1卷积层可以获得更高的性-

Method	PatchSVDD	SPADE*	DifferNet	PaDiM	Cut Paste	PatchCore	CFlow	FastFlow
carpet	(92.9,92.6)	(98.6,97.5)	(84.0,-)	(-,99.1)	(**100.0**,98.3)	(98.7,98.9)	(**100.0**,99.3)	(**100.0**,**99.4**)
grid	(94.6,96.2)	(99.0,93.7)	(97.1,-)	(-,97.3)	(96.2,97.5)	(98.2,98.7)	(97.6,**99.0**)	(**99.7**,98.3)
leather	(90.9,97.4)	(99.5,97.6)	(99.4,-)	(-,99.2)	(95.4,99.5)	(**100.0**,99.3)	(97.7,**99.7**)	(**100.0**,99.5)
tile	(97.8,91.4)	(89.8,87.4)	(92.9,-)	(-,94.1)	(**100.0**,90.5)	(98.7,95.6)	(98.7,**98.0**)	(**100.0**,96.3)
wood	(96.5,90.8)	(95.8,88.5)	(99.8,-)	(-,94.9)	(99.1,95.5)	(99.2,95.0)	(99.6,96.7)	(**100.0**,**97.0**)
bottle	(98.6,98.1)	(98.1,98.4)	(99.0,-)	(-,98.3)	(99.9,97.6)	(**100.0**,98.6)	(**100.0**,**99.0**)	(**100.0**,97.7)
cable	(90.3,96.8)	(93.2,97.2)	(86.9,-)	(-,96.7)	(**100.0**,90.0)	(99.5,**98.4**)	(**100.0**,97.6)	(**100.0**,**98.4**)
capsule	(76.7 ,95.8)	(98.6,99.0)	(88.8,-)	(- ,98.5)	(98.6 ,97.4)	(98.1,98.8)	(99.3 ,99.0)	(**100.0**,**99.1**)
hazelnut	(92.0 ,97.5)	(98.9,99.1)	(99.1,-)	(- ,98.2)	(93.3 ,97.3)	(**100.0**,98.7)	(96.8 ,98.9)	(**100.0**,**99.1**)
meta nut	(94.0,98.0)	(96.9,98.1)	(95.1,-)	(-,97.2)	(86.6,93.1)	(**100.0**,98.4)	(91.9,**98.6**)	(**100.0**,98.5)
pill	(86.1,95.1)	(96.5,96.5)	(95.9,-)	(-,95.7)	(**99.8**,95.7)	(96.6,97.1)	(99.9,99.0)	(99.4,**99.2**)
screw	(81.3,95.7)	(99.5,98.9)	(99.3,-)	(-,98.5)	(90.7,96.7)	(98.1,**99.4**)	(**99.7**,98.9)	(97.8,**99.4**)
toothbrush	(**100.0**,98.1)	(98.9,97.9)	(96.1,-)	(-,98.8)	(97.5 ,98.1)	(**100.0**,98.7)	(95.2,**99.0**)	(94.4 ,98.9)
transistor	(91.5,97.0)	(81.0,94.1)	(96.3,-)	(-,97.5)	(99.8,93.0)	(**100.0**,96.3)	(99.1 ,**98.0**)	(99.8,97.3)
zipper	(97.9,95.1)	(98.8,96.5)	(98.6,-)	(-,98.5)	(**99.9**,**99.3**)	(98.8,98.8)	(98.5 ,99.1)	(99.5,98.7)
AUCROC	(92.1,95.7)	(96.2,96.5)	(94.9,-)	(97.9,97.5)	(97.1,96.0)	(99.1,98.1)	(98.3,**98.6**)	(**99.4**,98.5)

Table 2: Anomaly detection and localization performance on MVTec AD dataset with the format (image-level AUC, pixel-level AUC). We report the detailed results for all categories.

Categories	AE MSE	AE MSE+SSIM	VT-ADL	FastFlow
0	0.49	0.53	0.99	0.95
1	0.92	0.96	0.94	0.96
2	0.95	0.89	0.77	0.99
Mean	0.78	0.79	0.90	**0.97**

Table 3: Anomaly localization results on BTAD datasts. We compare our method with convolutional auto encoders trained with MSE-loss and MSE+SSIM loss, and VT-ADL.

Method	OC-SVM	KDE	l_2-AE	VAE	Pixel CNN
AUC	58.6	61.0	53.6	58.3	55.1
Method	LSA	AnoGAN	DSVDD	OCGAN	FastFlow
AUC	64.1	61.8	64.8	65.6	**66.7**

Table 4: Anomaly detection results on CIFAR-10 datast.

Method	A.d. Params (M)	Image-level AUC	Pixel-level AUC
DeiT			
3-1	14.8	98.7	98.1
3-3	26.6	98.7	**98.3**
CaiT			
3-1	14.8	**99.4**	98.5
3-3	26.6	98.9	98.5
ResNet18			
3-1	2.7	97.3	96.8
3-3	4.9	**97.9**	**97.2**
Wide-ResNet50-2			
3-1	41.3	**99.3**	**98.1**
3-3	74.4	98.2	97.6

Table 5: Results of ablation experiments with various backbone networks. 3-1 means alternately using 3×3 and 1×1 convolution layers and 3-3 is only using 3×3 convolution layer in the subnet for FastFlow. A.d. Params is the number of additional model parameters compared with backbone network.

formance while reducing the amount of parameters. For the backbone network with small model capacities such as DeiT and ResNet18, only using 3×3 convolution layer has higher performance. To achieve the balance of accuracy and inference speed, we use alternate convolution kernels of 3×3, 1×1 with DeiT, CaiT and Wide-ResNet50-2, and only use 3×3 convolution layer with ResNet18.

4.5 Feature Visualization and Generation.

Our FastFlow model is a bidirectional invertible probability distribution transformer. In the forward process, it takes the feature map from the backbone network as input and transforms its original distribution into a standard normal distribution in two-dimensional space. In the reverse process, the inverse of FastFlow can generate the visual feature from a specific probability sampling variable. To better understand this ability in view of our FastFlow, we visualize the forward (from visual features to probability map) and reverse (from probability map to visual features) processes. As shown in Figure 4, we extract the features of an input image belonging to the leather class and the abnormal area is indicated by the red arrow. We forward it through the FastFlow model to obtain the probability map. Our FastFlow successfully transformed the original distribution into the standard normal distribution. Then, we add noise interference to a certain spatial area which is indicated by the yellow arrow in this probability map, and generate a leather feature tensor from the pollution probability map by using the inverse Fastflow model. In which we visualized the feature map of one channel in this feature tensor, and we can observe that new anomaly appeared in the corresponding pollution position.

4.6 Qualitative Results

We visualize some results of anomaly detection and localization in Figure 3 with the MVTec AD dataset. The top row shows test images with ground truth masks with and without anomalies, and the anomaly localization score heatmap is shown in the bottom row. There are both normal and abnormal images and our FastFlow gives accurate anomaly localization results.

Method	PatchSVDD	SPADE*	DifferNet	PaDiM	Cut Paste	PatchCore	CFlow	FastFlow
carpet	(92.9,92.6)	(98.6,97.5)	(84.0,-)	(-,99.1)	(**100.0**,98.3)	(98.7,98.9)	(**100.0**,99.3)	(**100.0**,**99.4**)
grid	(94.6,96.2)	(99.0,93.7)	(97.1,-)	(-,97.3)	(96.2,97.5)	(98.2,98.7)	(97.6,**99.0**)	(**99.7**,98.3)
leather	(90.9,97.4)	(99.5,97.6)	(99.4,-)	(-,99.2)	(95.4,99.5)	(**100.0**,99.3)	(97.7,**99.7**)	(**100.0**,99.5)
tile	(97.8,91.4)	(89.8,87.4)	(92.9,-)	(-,94.1)	(**100.0**,90.5)	(98.7,95.6)	(98.7,**98.0**)	(**100.0**,96.3)
wood	(96.5,90.8)	(95.8,88.5)	(99.8,-)	(-,94.9)	(99.1,95.5)	(99.2,95.0)	(99.6,96.7)	(**100.0**,**97.0**)
bottle	(98.6,98.1)	(98.1,98.4)	(99.0,-)	(-,98.3)	(99.9,97.6)	(**100.0**,98.6)	(**100.0**,**99.0**)	(**100.0**,97.7)
cable	(90.3,96.8)	(93.2,97.2)	(86.9,-)	(-,96.7)	(**100.0**,90.0)	(99.5,**98.4**)	(**100.0**,97.6)	(**100.0**,**98.4**)
capsule	(76.7 ,95.8)	(98.6,99.0)	(88.8,-)	(- ,98.5)	(98.6 ,97.4)	(98.1,98.8)	(99.3 ,99.0)	(**100.0**,**99.1**)
hazelnut	(92.0 ,97.5)	(98.9,99.1)	(99.1,-)	(- ,98.2)	(93.3 ,97.3)	(**100.0**,98.7)	(96.8 ,98.9)	(**100.0**,**99.1**)
meta nut	(94.0,98.0)	(96.9,98.1)	(95.1,-)	(-,97.2)	(86.6,93.1)	(**100.0**,98.4)	(91.9,**98.6**)	(**100.0**,98.5)
pill	(86.1,95.1)	(96.5,96.5)	(95.9,-)	(-,95.7)	(**99.8**,95.7)	(96.6,97.1)	(99.9,99.0)	(99.4,**99.2**)
screw	(81.3,95.7)	(99.5,98.9)	(99.3,-)	(-,98.5)	(90.7,96.7)	(98.1,**99.4**)	(**99.7**,98.9)	(97.8,**99.4**)
toothbrush	(**100.0**,98.1)	(98.9,97.9)	(96.1,-)	(-,98.8)	(97.5 ,98.1)	(**100.0**,98.7)	(95.2,**99.0**)	(94.4 ,98.9)
transistor	(91.5,97.0)	(81.0,94.1)	(96.3,-)	(-,97.5)	(99.8,93.0)	(**100.0**,96.3)	(99.1 ,**98.0**)	(99.8,97.3)
zipper	(97.9,95.1)	(98.8,96.5)	(98.6,-)	(-,98.5)	(**99.9**,**99.3**)	(98.8,98.8)	(98.5 ,99.1)	(99.5,98.7)
AUCROC	(92.1,95.7)	(96.2,96.5)	(94.9,-)	(97.9,97.5)	(97.1,96.0)	(99.1,98.1)	(98.3,**98.6**)	(**99.4**,98.5)

表2：MVTec AD数据集上的异常检测与定位性能，格式为（图像级AUC，像素级AUC）。我们报告了所有类别的详细结果。

Categories	AE MSE	AE MSE+SSIM	VT-ADL	FastFlow
0	0.49	0.53	0.99	0.95
1	0.92	0.96	0.94	0.96
2	0.95	0.89	0.77	0.99
Mean	0.78	0.79	0.90	**0.97**

表3：BTAD数据集上的异常定位结果。我们将我们的方法与使用MSE损失和MSE+SSIM损失训练的卷积自编码器以及VT-ADL进行了比较。

Method	OC-SVM	KDE	l_2-AE	VAE	Pixel CNN
AUC	58.6	61.0	53.6	58.3	55.1
Method	LSA	AnoGAN	DSVDD	OCGAN	FastFlow
AUC	64.1	61.8	64.8	65.6	**66.7**

表4：CIFAR-10数据集上的异常检测结果。

Method	A.d. Params (M)	Image-level AUC	Pixel-level AUC
DeiT			
3-1	14.8	98.7	98.1
3-3	26.6	98.7	**98.3**
CaiT			
3-1	14.8	**99.4**	98.5
3-3	26.6	98.9	98.5
ResNet18			
3-1	2.7	97.3	96.8
3-3	4.9	**97.9**	**97.2**
Wide-ResNet50-2			
3-1	41.3	**99.3**	**98.1**
3-3	74.4	98.2	97.6

表5：使用不同骨干网络的消融实验结果。3-1表示在FastFlow的子网中交替使用3×3和1×1卷积层，3-3表示仅使用3×3卷积层。A.d. Params指相较于骨干网络额外增加的模型参数量。

在减少参数量的同时保持性能。对于模型容量较小的骨干网络，如DeiT和ResNet18，仅使用3×3卷积层即可获得更高性能。为平衡精度与推理速度，我们在DeiT、CaiT和Wide-ResNet50-2中交替使用3×3和1×1卷积核，而在ResNet18中仅使用3×3卷积层。

4.5 特征可视化与生成。

我们的FastFlow模型是一个双向可逆概率分布转换器。在前向过程中，它以主干网络提取的特征图作为输入，将其原始分布转换为二维空间中的标准正态分布。在反向过程中，FastFlow的逆变换能够从特定概率采样变量生成视觉特征。为了更好地理解FastFlow的这一能力，我们可视化展示了前向（从视觉特征到概率图）与反向（从概率图到视觉特征）过程。如图4所示，我们提取了属于皮革类别输入图像的特征，异常区域通过{v*}标示。

红色箭头。我们将其通过FastFlow模型前向传播以获得概率图。我们的FastFlow成功地将原始分布转换为标准正态分布。接着，我们在概率图中由黄色箭头指示的特定空间区域添加噪声干扰，并利用逆FastFlow模型从污染概率图生成皮革特征张量。在此过程中，我们可视化了该特征张量中一个通道的特征图，可以观察到新的异常出现在相应的污染位置。

4.6 定性结果

我们在图3中展示了使用MVTec AD数据集进行异常检测和定位的部分结果。顶行显示带有真实标注掩码的测试图像（包含异常与正常情况），底行则呈现异常定位得分的热力图。图中既包含正常图像也包含异常图像，我们的FastFlow模型能提供精确的异常定位结果。

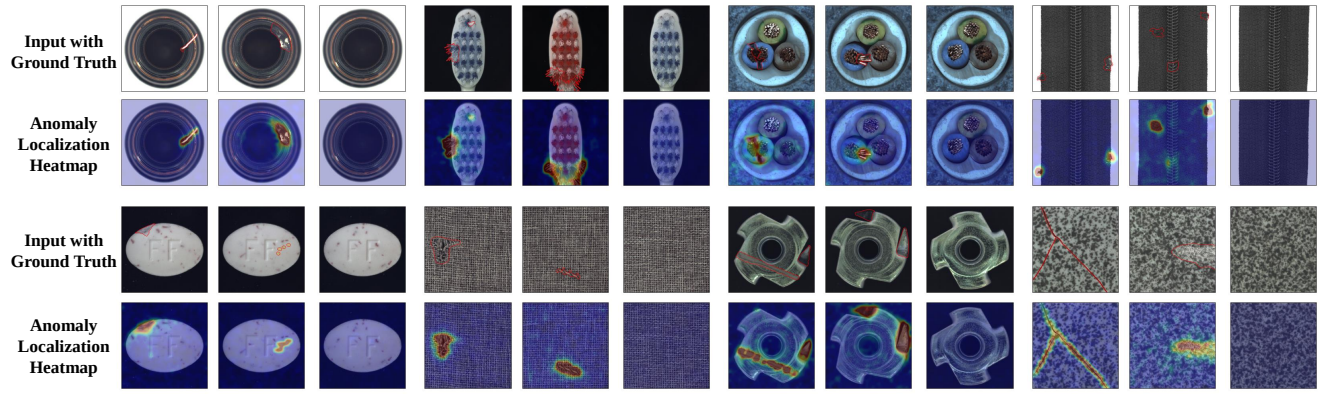

Figure 3: Anomaly localization results of MVTec AD datasets. From top to bottom, input images with ground-truth localization area labeled in red and anomaly localization heatmaps.

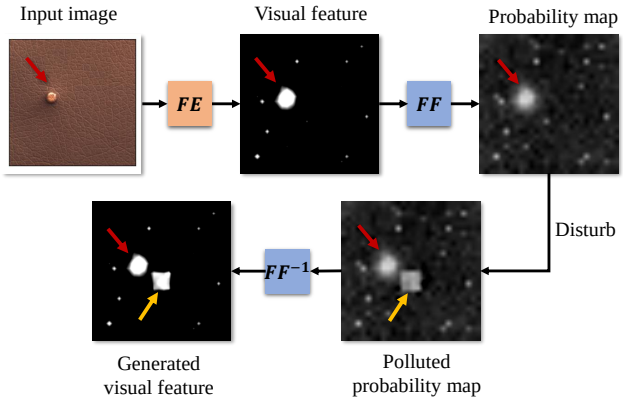

Figure 4: The bidirectional invertible process for FastFlow. "FE" is the feature extractor, "FF" is our FastFlow model, "FF^{-1}" is the reverse for FastFlow. The red and yellow arrows point to the original anomaly and the new anomaly introduced after the noise disturbance respectively.

4.7 Implementation Details

We provide the details of the structure of feature extractor, the selection of feature layer and the size of input image in Table 6. For vision transformer, our method only uses feature maps of a specific layer, and does not need to design complicated multi-scale features manually. For ResNet18 and Wide-ResNet50-2, we directly use the features of the last layer in the first three blocks, put these features into the 2D flow model to obtain their respective anomaly detection and localization results, and finally take the average value as the final result. All these backbone are initialized with the ImageNet pre-trained weights and their parameters are frozen in the following training process. For FastFlow, we use 20-step flows in CaiT and DeiT and 8-step flows for ResNet18 and Wide-ResNet50-2. We train our model using Adam optimizer with the learning rate of 1e-3 and weight decay of 1e-5. We use a 500 epoch training schedule, and the batch size is 32.

Backbone	Input Size	Block Index	Feature Size
CaiT-M48-distilled	448	40	28
DeiT-base-distilled	384	7	24
Res18	256	[1,2,3]	[64, 32, 16]
WR50	256	[1,2,3]	[64, 32, 16]

Table 6: We use four different feature extractors in all experiments. The input picture size and feature size are set according to the backbone network and the block index indicates the block from which the feature is obtained..

5 Conclusion

In this paper, we propose a novel approach named FastFlow for unsupervised anomaly detection and localization. Our key observation is that anomaly detection and localization requires comprehensive consideration of global and local information with a learnable distribution modeling method, and efficient inference process, which are ignored in the existing approaches. To this end, we present a 2D flow model denoted as FastFlow which has a lightweight structure and is used to project the feature distribution of normal images to the standard normal distribution in training, and use the probabilities as the anomaly score in testing. FastFlow can be used in typical feature extraction networks such as ResNet and ViT in the form of plug-ins. Extensive experimental results on MVTec AD dataset show FastFlow superiority over the state-of-the art methods in terms of accuracy and reasoning efficiency.

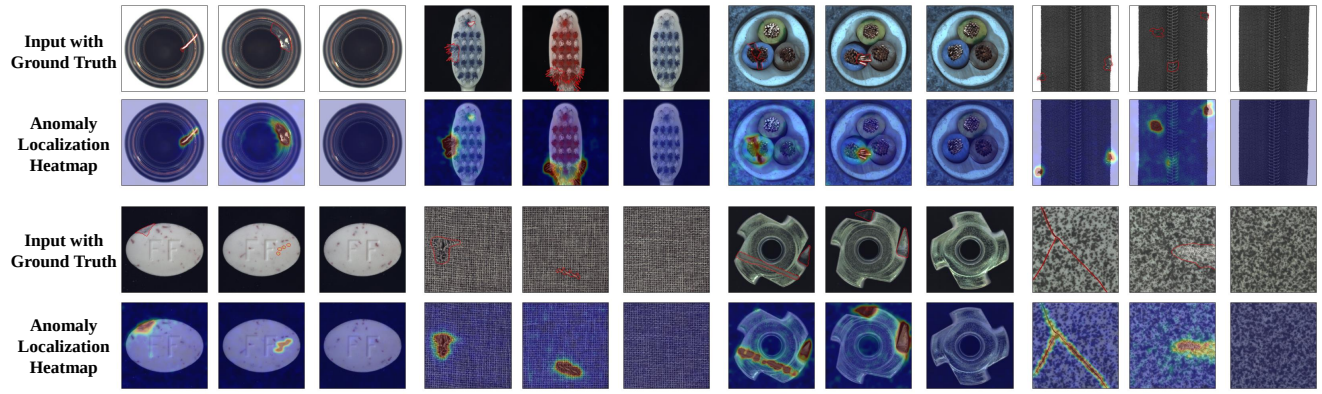

图3：MVTec AD数据集的异常定位结果。从上至下依次为：输入图像（真实异常区域以红色标出）与异常定位热力图。

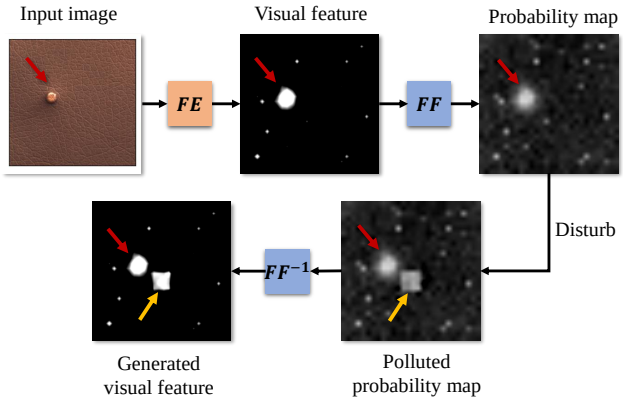

图4：FastFlow的双向可逆过程。“FE”是特征提取器，“FF”是我们的FastFlow模型，“FF{v*}”是FastFlow的反向过程。红色和黄色箭头分别指向原始异常以及噪声扰动后引入的新异常。

4.7 实现细节

我们在表6中提供了特征提取器的结构细节、特征层的选择以及输入图像的尺寸。对于视觉Transformer，我们的方法仅使用特定层的特征图，无需手动设计复杂的多尺度特征。对于ResNet18和Wide-ResNet50-2，我们直接使用前三个模块中最后一层的特征，将这些特征输入二维流模型以获得各自的异常检测与定位结果，最终取平均值作为最终结果。所有骨干网络均采用ImageNet预训练权重初始化，并在后续训练过程中冻结其参数。对于FastFlow，我们在CaiT和DeiT中使用20步流，在ResNet18和Wide-ResNet50-2中使用8步流。我们使用Adam优化器训练模型，学习率为1e-3，权重衰减为1e-5。训练周期为500轮，批次大小为32。

Backbone	Input Size	Block Index	Feature Size
CaiT-M48-distilled	448	40	28
DeiT-base-distilled	384	7	24
Res18	256	[1,2,3]	[64, 32, 16]
WR50	256	[1,2,3]	[64, 32, 16]

表6：在所有实验中，我们使用了四种不同的特征提取器。输入图片尺寸和特征尺寸根据主干网络设定，块索引表示获取特征的来源块。

5 结论

本文提出了一种名为FastFlow的无监督异常检测与定位新方法。我们的核心观点是：异常检测与定位需要结合可学习的分布建模方法，综合考虑全局与局部信息，并实现高效推理过程——这些要素在现有方法中均被忽视。为此，我们提出了一种轻量级二维流模型FastFlow，该模型在训练阶段将正常图像的特征分布映射到标准正态分布，在测试阶段则利用概率值作为异常评分指标。FastFlow能以插件形式适配ResNet、ViT等典型特征提取网络。在MVTec AD数据集上的大量实验表明，FastFlow在检测精度与推理效率方面均优于当前最先进方法。

Supplementary Material for
FastFlow: Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows

Channel Ratio	Parameters (M)	Image-level AUC	Pixel-level AUC
CaiT			
$0.16\times$	14.8	99.4	98.5
$0.33\times$	29.6	98.9	98.4
Wide-ResNet50-2			
$0.25\times$	10.9	98.9	98.0
$0.5\times$	20.7	99.1	98.1
$1.0\times$	41.3	99.3	98.1
$2.0\times$	82.6	99.4	98.1

Table 7: Ablation study results about the hidden layer channels for CNN and vision transformer in MVTec AD dataset. Channel Ratio means the ratio of the number of channels in the hidden layer to the number of channels in the input and output layers for subnet in our FastFlow.

Data Augmentation	Image-level AUC	Pixel-level AUC
CaiT		
w/o	99.3	98.4
w	99.4	98.5
Wide-ResNet50-2		
w/o	98.9	98.2
w	99.3	98.1

Table 8: The effect of data augmentation on the anomaly detection and localization performance.

6 More Ablation Studies

6.1 Channels of Hidden Layers in Flow Model

In the original flow model which has been used in DifferNet (Rudolph, Wandt, and Rosenhahn 2021) and CFLOW (Gudovskiy, Ishizaka, and Kozuka 2021), the number of channels of hidden layers in all subnet is set to $2\times$ as much the input and output layer's channel. This kind of design improves the results by increasing the complexity of the model, but it reduces the efficiency of inference. In our FastFlow, we found that using $0.16\times$ number of channels in CaiT and $1\times$ number of channels in Wide-ResNet50-2 can achieve a balance between performance and model parameters. In addition, when we use $0.25\times$ number of channels of Wide-ResNet50-2, we can further reduce the model parameters while still maintaining high accuracy. The results are shown in Table 7.

6.2 Training Data Augmentation

In order to learn a more robust FastFlow model, we apply various data augmentation methods to the MVTec AD dataset during the training phase. We use random horizontal flip, vertical flip and rotation, with probabilities of 0.5, 0.3 and 0.7, respectively. It should be noted that some categories are not suitable for violent data augmentation. For example, the transistor can not be flipped upside down and rotated. The results are shown in Table 8.

7 Bad Cases and Ambiguity Label

We visualize bad cases for our method on MVTec AD dataset in Figure 5 to Figure 7 which are summarized into three categories. We show the missing detection cases in Figure 5, false detection cases in Figure 6 and label ambiguity cases in Figure 7. In Figure 5, our method missed a few small and unobvious anomalies. In Figure 6, our method had false detection results in some background areas, such as areas with hair and dirt in the background. In Figure 7, our method found some areas belong to abnormal but not be labeled, such as the "scratch neck" for screw and the "fabric interior" for zipper.

8 Non-aligned Disturbed MVTec AD Dataset

Considering that the MVTec AD dataset has the characteristic of sample alignment which is infrequent in practical application, we perform a series of spatial perturbations on the test data to obtain an unaligned MVTec AD dataset. In detail, we apply random zoom in/out with 0.85 ratio, random rotation with ± 15 angle, random translation with 0.15 ratio to expand the original test dataset by $4\times$ to the new test dataset. We evaluate our FastFlow (with CaiT) in this new test dataset and we obtain 99.2 image-level AUC and 98.1 pixle-level AUC. There is almost no performance loss compared with the results in original aligned MVTec AD test dataset, which proves the robustness of our method. We also give some visualization results in Figure 8. We can observe that FastFlow can still have high performance on anomaly detection and location result in this non-aligned disturbed MVTec AD dataset.

References

Abati, D.; Porrello, A.; Calderara, S.; and Cucchiara, R. 2019. Latent space autoregression for novelty detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 481–490.

An, J.; and Cho, S. 2015. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1): 1–18.

Bergman, L.; and Hoshen, Y. 2020. Classification-based anomaly detection for general data. *International Conference on Learning Representations (ICLR)*.

Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2019. MVTec AD – A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9592–9600.

Bishop, C. M. 2006. Pattern recognition. *Machine learning*, 128(9).

Cohen, N.; and Hoshen, Y. 2020. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*.

Supplementary Material for

FastFlow: Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows

Channel Ratio	Parameters (M)	Image-level AUC	Pixel-level AUC
CaiT			
0.16×	14.8	99.4	98.5
0.33×	29.6	98.9	98.4
Wide-ResNet50-2			
0.25×	10.9	98.9	98.0
0.5×	20.7	99.1	98.1
1.0×	41.3	99.3	98.1
2.0×	82.6	99.4	98.1

表7：关于MVTec AD数据集中CNN和视觉Transformer隐藏层通道的消融研究结果。通道比率指我们FastFlow中子网络隐藏层通道数与输入输出层通道数的比值。

Data Augmentation	Image-level AUC	Pixel-level AUC
CaiT		
w/o	99.3	98.4
w	99.4	98.5
Wide-ResNet50-2		
w/o	98.9	98.2
w	99.3	98.1

表8：数据增强对异常检测与定位性能的影响。

6 更多消融研究

6.1 流模型中隐藏层的通道

在DifferNet（Rudolph, Wandt和Rosenhahn 2021）和CFLOW（Gudovskiy, Ishizaka和Kozuka 2021）所使用的原始流模型中，所有子网隐藏层的通道数均设置为输入和输出层通道数的2×倍。这种设计通过增加模型复杂度提升了结果，但降低了推理效率。在我们的FastFlow中，我们发现对CaiT使用0.16×通道数，对Wide-ResNet50-2使用1×通道数，可以在性能和模型参数之间取得平衡。此外，当我们将Wide-ResNet50-2的通道数设置为0.25×时，能在保持高精度的同时进一步减少模型参数。结果如表7所示。

6.2 训练数据增强

为了学习一个更稳健的FastFlow模型，我们在训练阶段对MVTec AD数据集应用了多种数据增强方法。我们分别以0.5、0.3和0.7的概率使用随机水平翻转、垂直翻转和旋转。需要注意的是，某些类别不适合进行剧烈的数据增强。例如，晶体管不能上下翻转或旋转。结果如表8所示。

7个不良案例与歧义标签

我们在图5至图7中展示了我们的方法在MVTec AD数据集上的失败案例，这些案例被归纳为三类。图5展示了漏检案例，图6展示了误检案例，图7展示了标签模糊案例。在图5中，我们的方法遗漏了一些微小且不明显的异常。在图6中，我们的方法在一些背景区域出现了误检，例如背景中含有毛发和污渍的区域。在图7中，我们的方法发现了一些属于异常但未被标注的区域，例如螺丝的“颈部划痕”和拉链的“织物内部”。

8 非对齐干扰MVTec AD数据集

考虑到MVTec AD数据集具有样本对齐的特性，这在实际应用中较为少见，我们对测试数据进行了一系列空间扰动，以获取一个非对齐的MVTec AD数据集。具体而言，我们以0.85的比例进行随机缩放、以±15角度进行随机旋转、以0.15的比例进行随机平移，将原始测试数据集扩展4×倍，形成新的测试数据集。我们在这一新测试数据集上评估了FastFlow（结合CaiT），获得了99.2的图像级AUC和98.1的像素级AUC。与原始对齐的MVTec AD测试数据集的结果相比，性能几乎没有损失，这证明了我们方法的鲁棒性。图8中我们还展示了一些可视化结果。可以观察到，FastFlow在这一非对齐且受扰动的MVTec AD数据集上，依然能在异常检测和定位结果中保持高性能。

参考文献

Abati, D.; Porrello, A.; Calderara, S.; 与 Cucchiara, R. 2019. 基于隐空间自回归的新颖性检测。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第481–490页。 An, J.; 与 Cho, S. 2015. 使用重构概率的变分自编码器异常检测。*Special Lecture on IE*，2(1): 1–18。 Bergman, L.; 与 Hoshen, Y. 2020. 面向通用数据的基于分类的异常检测。*International Conference on Learning Representations (ICLR)*。 Bergmann, P.; Fauser, M.; Sattlegger, D.; 与 Steger, C. 2019. MVTec AD——一个用于无监督异常检测的全面真实世界数据集。发表于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，第9592–9600页。 Bishop, C. M. 2006. 模式识别。*Machine learning*，128(9)。 Cohen, N.; 与 Hoshen, Y. 2020. 基于深度金字塔对应的子图像异常检测。*arXiv preprint arXiv:2005.02357*。

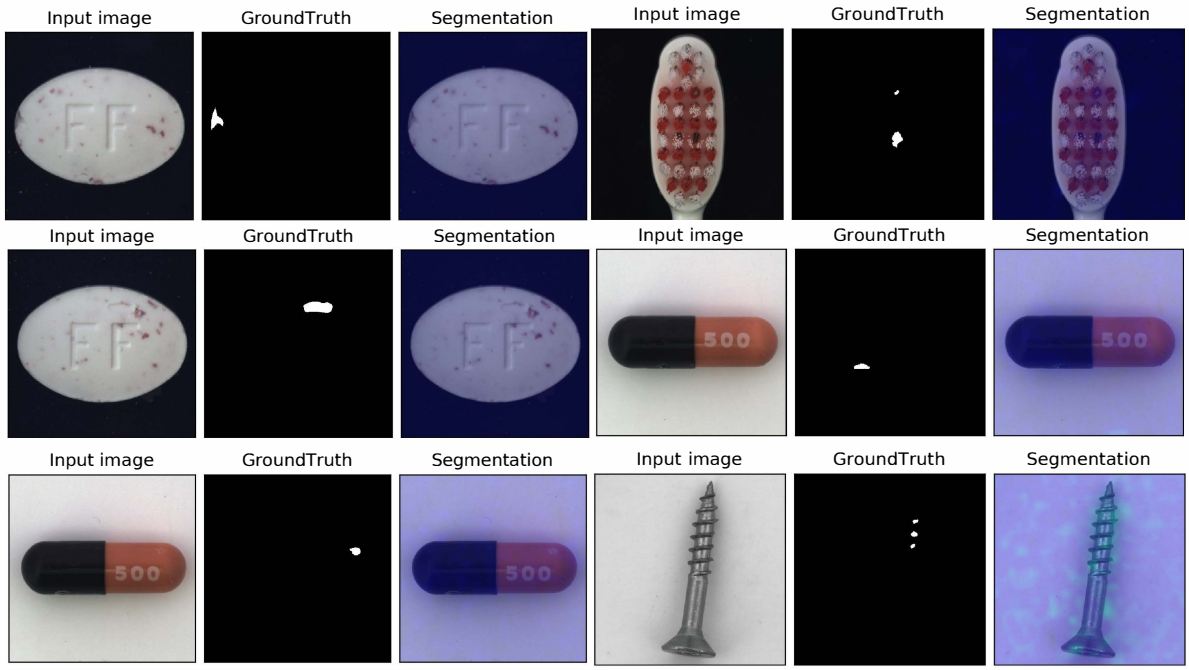

Figure 5: Bad cases of the missing detection type. All missed detection results of our method in shown in this figure.

Defard, T.; Setkov, A.; Loesch, A.; and Audigier, R. 2020. PaDiM: a Patch Distribution Modeling Framework for Anomaly Detection and Localization. *arXiv preprint arXiv:2011.08785*.

Dinh, L.; Krueger, D.; and Bengio, Y. 2014. Nice: Nonlinear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2016. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Gong, D.; Liu, L.; Le, V.; Saha, B.; Mansour, M. R.; Venkatesh, S.; and Hengel, A. v. d. 2019. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1705–1714.

Gudovskiy, D.; Ishizaka, S.; and Kozuka, K. 2021. CFLOW-AD: Real-Time Unsupervised Anomaly Detection with Localization via Conditional Normalizing Flows. *arXiv preprint arXiv:2107.12571*.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, 1735–1742. IEEE.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative flow with invertible 1x1 convolutions. *arXiv preprint arXiv:1807.03039*.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Li, C.-L.; Sohn, K.; Yoon, J.; and Pfister, T. 2021. CutPaste: Self-Supervised Learning for Anomaly Detection and Localization. *arXiv preprint arXiv:2104.04015*.

Mishra, P.; Verk, R.; Fornasier, D.; Piciarelli, C.; and Foresti, G. L. 2021. VT-ADL: A Vision Transformer Network for Image Anomaly Detection and Localization. *arXiv preprint arXiv:2104.10036*.

Oord, A. v. d.; Kalchbrenner, N.; Vinyals, O.; Espeholt, L.; Graves, A.; and Kavukcuoglu, K. 2016. Conditional image generation with pixelcnn decoders. *arXiv preprint arXiv:1606.05328*.

Perera, P.; Nallapati, R.; and Xiang, B. 2019. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2898–2906.

Reiss, T.; Cohen, N.; Bergman, L.; and Hoshen, Y. 2021. PANDA: Adapting Pretrained Features for Anomaly Detection and Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2806–2814.

Rezende, D.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *International conference on machine learning*, 1530–1538. PMLR.

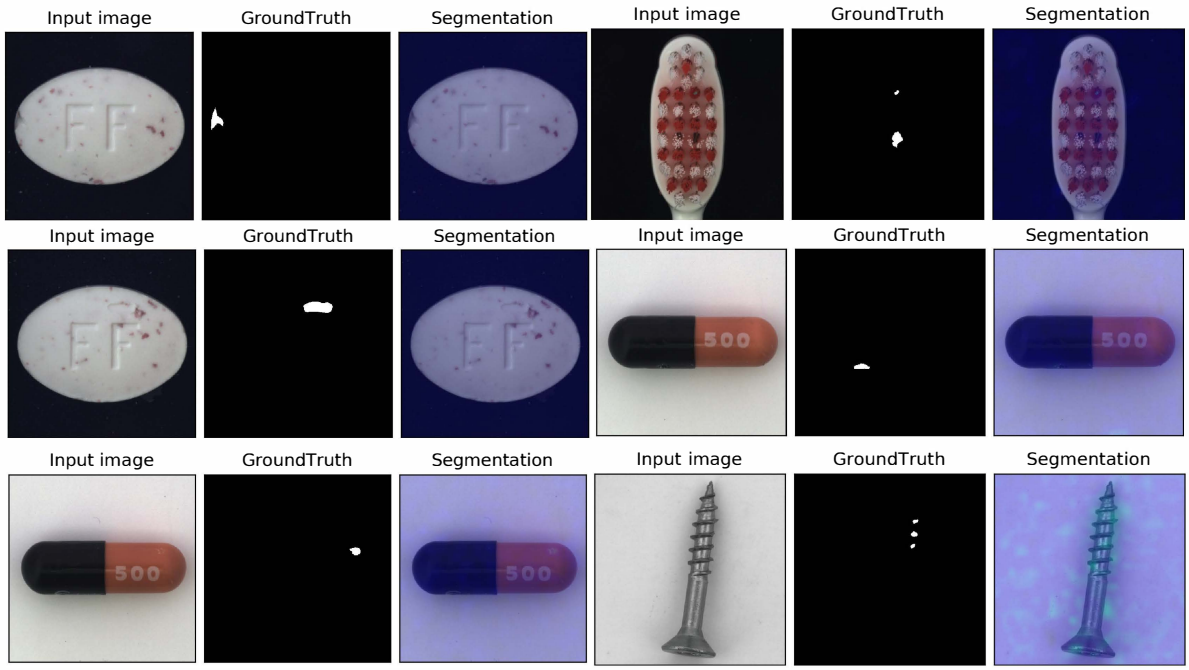

图5：漏检类型的错误案例。本图展示了我们方法的所有漏检结果。

Defard, T.; Setkov, A.; Loesch, A.; 与 Audigier, R. 2020. PaDiM：一种用于异常检测与定位的补丁分布建模框架。*arXiv preprint arXiv:2011.08785*。 Dinh, L.; Krueger, D.; 与 Bengio, Y. 2014. NICE：非线性独立分量估计。*arXiv preprint arXiv:1410.8516*。 Dinh, L.; Sohl-Dickstein, J.; 与 Bengio, S. 2016. 使用 Real NVP 进行密度估计。*arXiv preprint arXiv:1605.08803*。 Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; 等. 2020. 一张图像相当于 16x16 个词：大规模图像识别的 Transformer 模型。*arXiv preprint arXiv:2010.11929*。 Gong, D.; Liu, L.; Le, V.; Saha, B.; Mansour, M. R.; Venkatesh, S.; 与 Hengel, A. v. d. 2019. 通过记忆正常模式检测异常：用于无监督异常检测的记忆增强深度自编码器。于 *Proceedings of the IEEE/CVF International Conference on Computer Vision*，1705–1714。 Gudovskiy, D.; Ishizaka, S.; 与 Kozuka, K. 2021. CFLOW-AD：通过条件归一化流实现具有定位功能的实时无监督异常检测。*arXiv preprint arXiv:2107.12571*。 Hadsell, R.; Chopra, S.; 与 LeCun, Y. 2006. 通过学习不变映射进行降维。于 *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*，卷 2，1735–1742。IEEE。 He, K.; Zhang, X.; Ren, S.; 与 Sun, J. 2016. 用于图像识别的深度残差学习。于 *Proceedings of the IEEE conference on computer vision and pattern recognition*，770–778。Kingma, D. P.; 与 Dhariwal, P. 2018. Glow：基于可逆1x1卷积的生成流。*arXiv preprint arXiv:1807.03039*。Krizhevsky, A.; Hinton, G.; 等。2009. 从微小图像中学习多层特征。Li, C.-L.; Sohn, K.; Yoon, J.; 与 Pfister, T. 2021. CutPaste：用于异常检测与定位的自监督学习。*arXiv preprint arXiv:2104.04015*。Mishra, P.; Verk, R.; Fornasier, D.; Piciarelli, C.; 与 Foresti, G. L. 2021. VT-ADL：一种用于图像异常检测与定位的视觉Transformer网络。*arXiv preprint arXiv:2104.10036*。Oord, A. v. d.; Kalchbrenner, N.; Vinyals, O.; Espeholt, L.; Graves, A.; 与 Kavukcuoglu, K. 2016. 使用PixelCNN解码器的条件图像生成。*arXiv preprint arXiv:1606.05328*。Perera, P.; Nallapati, R.; 与 Xiang, B. 2019. OCGAN：使用具有约束潜在表示的GAN进行单类新颖性检测。于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，2898–2906。Reiss, T.; Cohen, N.; Bergman, L.; 与 Hoshen, Y. 2021. PANDA：为异常检测与分割适配预训练特征。于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*，2806–2814。Rezende, D.; 与 Mohamed, S. 2015. 使用标准化流的变分推断。于 *International conference on ma- chine learning*，1530–1538。PMLR。

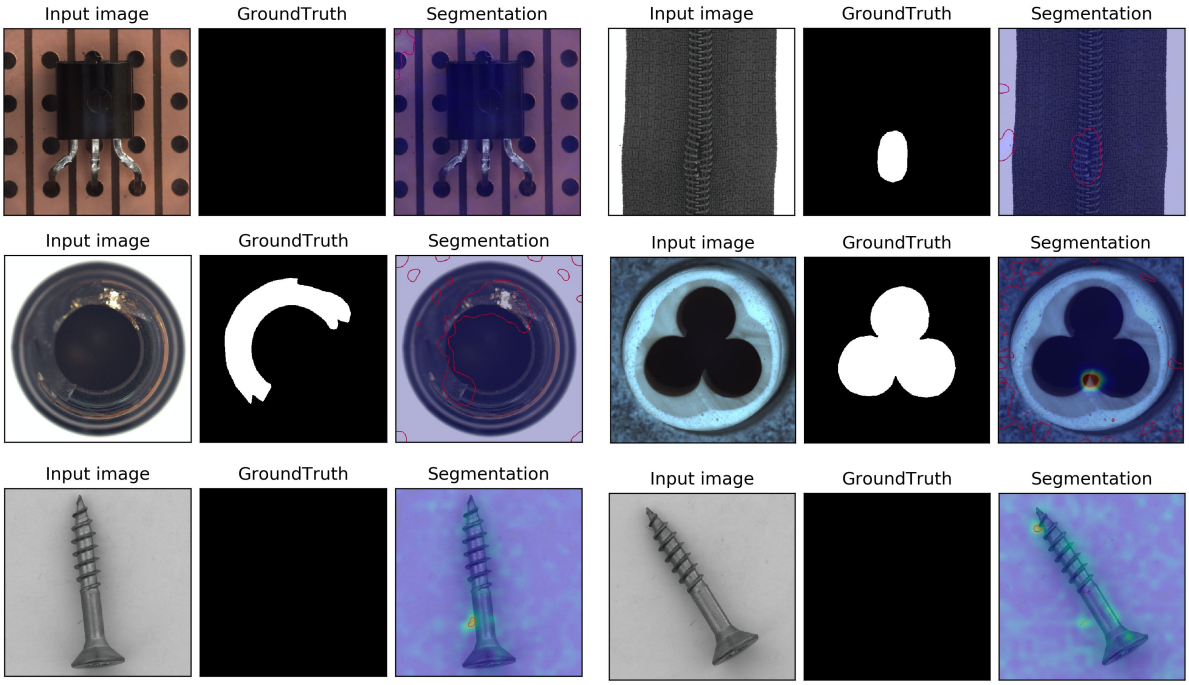

Figure 6: Bad cases of false detection type. We give the typical results of our method in this figure.

Rippel, O.; Mertens, P.; and Merhof, D. 2021. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 6726–6733. IEEE.

Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; and Gehler, P. 2021. Towards Total Recall in Industrial Anomaly Detection. *arXiv preprint arXiv:2106.08265*.

Rudolph, M.; Wandt, B.; and Rosenhahn, B. 2021. Same same but differnet: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1907–1916.

Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Siddiqui, S. A.; Binder, A.; Müller, E.; and Kloft, M. 2018. Deep one-class classification. In *International conference on machine learning*, 4393–4402. PMLR.

Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Schmidt-Erfurth, U.; and Langs, G. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, 146–157. Springer.

Schölkopf, B.; Williamson, R. C.; Smola, A. J.; Shawe-Taylor, J.; Platt, J. C.; et al. 1999. Support vector method for novelty detection. In *NIPS*, volume 12, 582–588. Citeseer.

Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021a. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, 10347–10357. PMLR.

Touvron, H.; Cord, M.; Sablayrolles, A.; Synnaeve, G.; and Jégou, H. 2021b. Going deeper with image transformers. *arXiv preprint arXiv:2103.17239*.

Wang, S.; Wu, L.; Cui, L.; and Shen, Y. 2021. Glancing at the Patch: Anomaly Localization With Global and Local Feature Comparison. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 254–263.

Yan, X.; Zhang, H.; Xu, X.; Hu, X.; and Heng, P.-A. 2021. Learning Semantic Context from Normal Samples for Unsupervised Anomaly Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 3110–3118.

Yi, J.; and Yoon, S. 2020. Patch SVDD: Patch-level SVDD for Anomaly Detection and Segmentation. In *Proceedings of the Asian Conference on Computer Vision*.

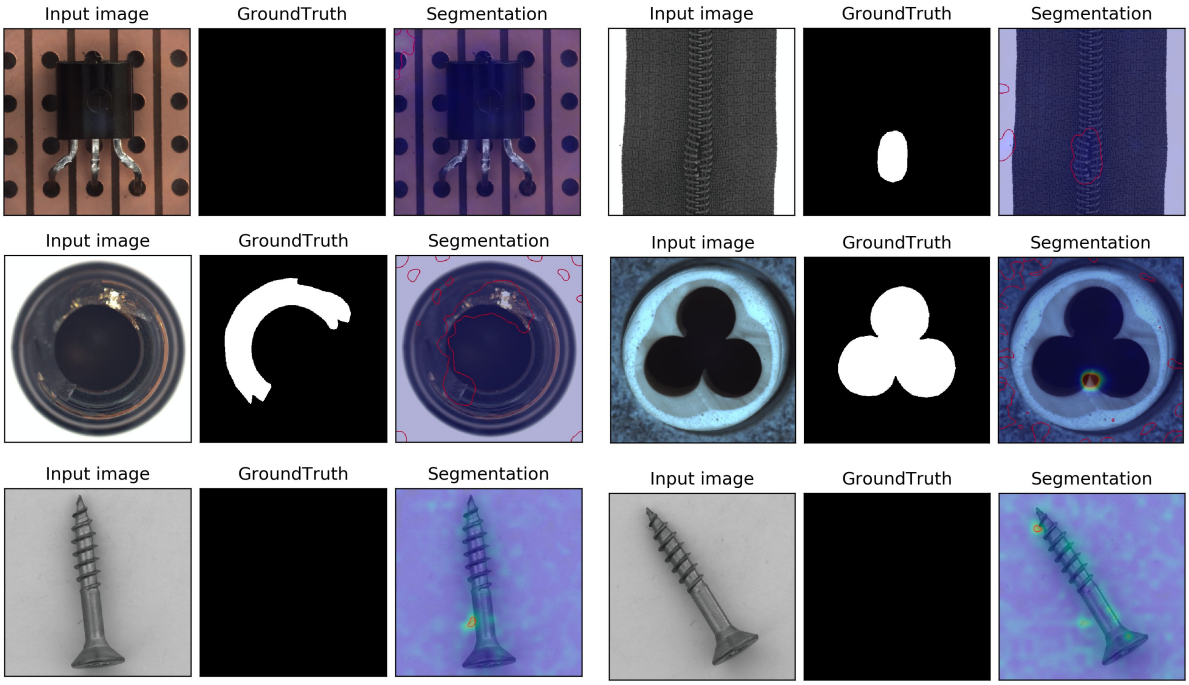

图6：失败案例　　误检测类型。我们给出我们方法的典型结果　　图中的od。

Rippel, O.; Mertens, P.; and Merhof, D. 2021. 利用预训练深度特征中的正常数据分布建模进行异常检测。发表于 *2020 25th International Conference on Pattern Recognition (ICPR)*，第6726–6733页。IEEE。Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; and Gehler, P. 2021. 迈向工业异常检测的完全召回。*arXiv preprint arXiv:2106.08265*。Rudolph, M.; Wandt, B.; and Rosenhahn, B. 2021. 相似但不同：基于归一化流的半监督缺陷检测。发表于 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*，第1907–1916页。Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Siddiqui, S. A.; Binder, A.; Müller, E.; and Kloft, M. 2018. 深度单类分类。发表于 *International conference on machine learning*，第4393–4402页。PMLR。Schlegl, T.; Seeböck, P.; Waldstein, S. M.; Schmidt-Erfurth, U.; and Langs, G. 2017. 利用生成对抗网络进行无监督异常检测以指导标记发现。发表于 *International conference on information processing in medical imaging*，第146–157页。Springer。Schölkopf, B.; Williamson, R. C.; Smola, A. J.; Shawe-Taylor, J.; Platt, J. C.; 等。1999. 用于新颖性检测的支持向量方法。发表于 *NIPS*，第12卷，第582–588页。Citeseer。Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; and Jégou, H. 2021a. 通过注意力机制训练数据高效的图像变换器及蒸馏。发表于 *International Conference on Machine Learning*，第10347–10357页。PMLR。Touvron, H.; Cord, M.; Sablayrolles, A.; Synnaeve, G.; 及 Jégou, H. 2021b. 使用图像变换器深入探索。*arXiv preprint arXiv:2103.17239*。

王, S.; 吴, L.; 崔, L.; 沈, Y. 2021. 一瞥补丁：通过全局与局部特征比较进行异常定位。于 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 254–263。

严, 徐; 张, 胡; 徐, 胡; 胡, 徐; 与 Heng, P.-A. 2021. 从正常样本中学习语义上下文用于无监督异常检测。于 *Proceedings of the AAAI Conference on Artificial Intelligence*, 第35卷, 3110–3118页。　　易.; 与 Yoon, S. 2020. 补丁SVDD：用于异常检测与分割的补丁级SVDD。于 *Proceedings of the Asian Conference on Computer Vision*。

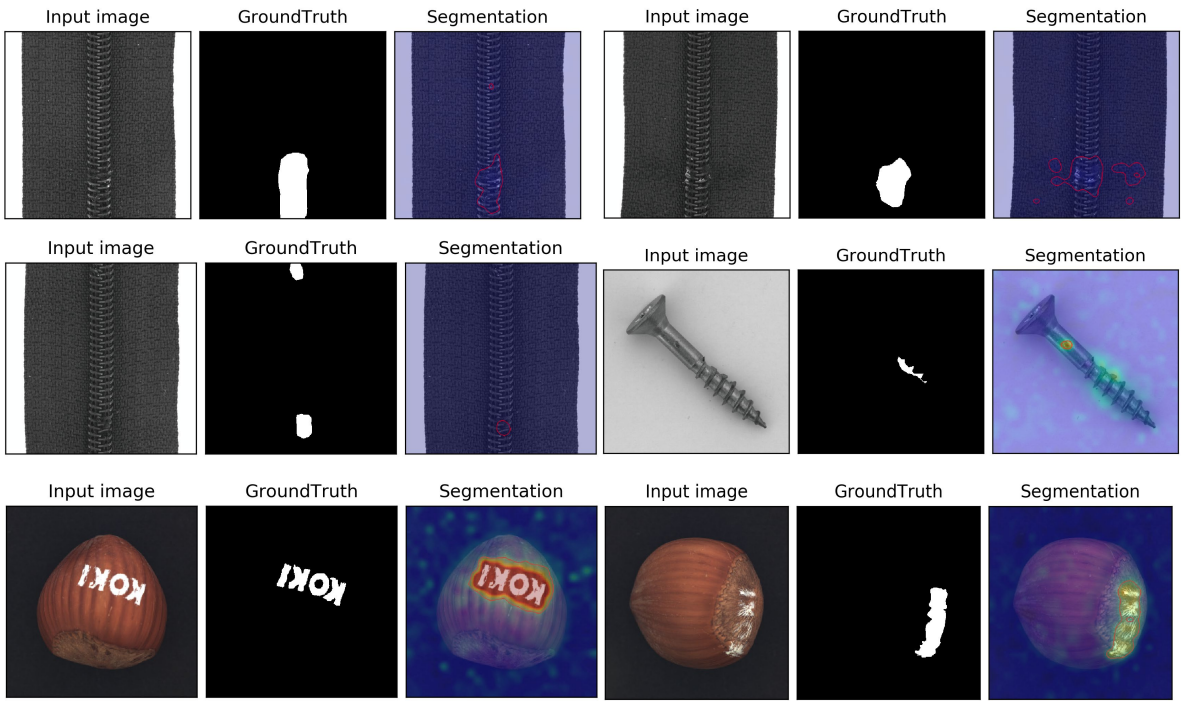

Figure 7: Bad cases caused by label ambiguity. In the first two rows, there are abnormal areas localized by our method while not labeled. In the last row of hazelnut, we show the label ambiguity of the "print" subclass, in which one hazelnut print is labeled finely, while the other is labeled with a rough area.

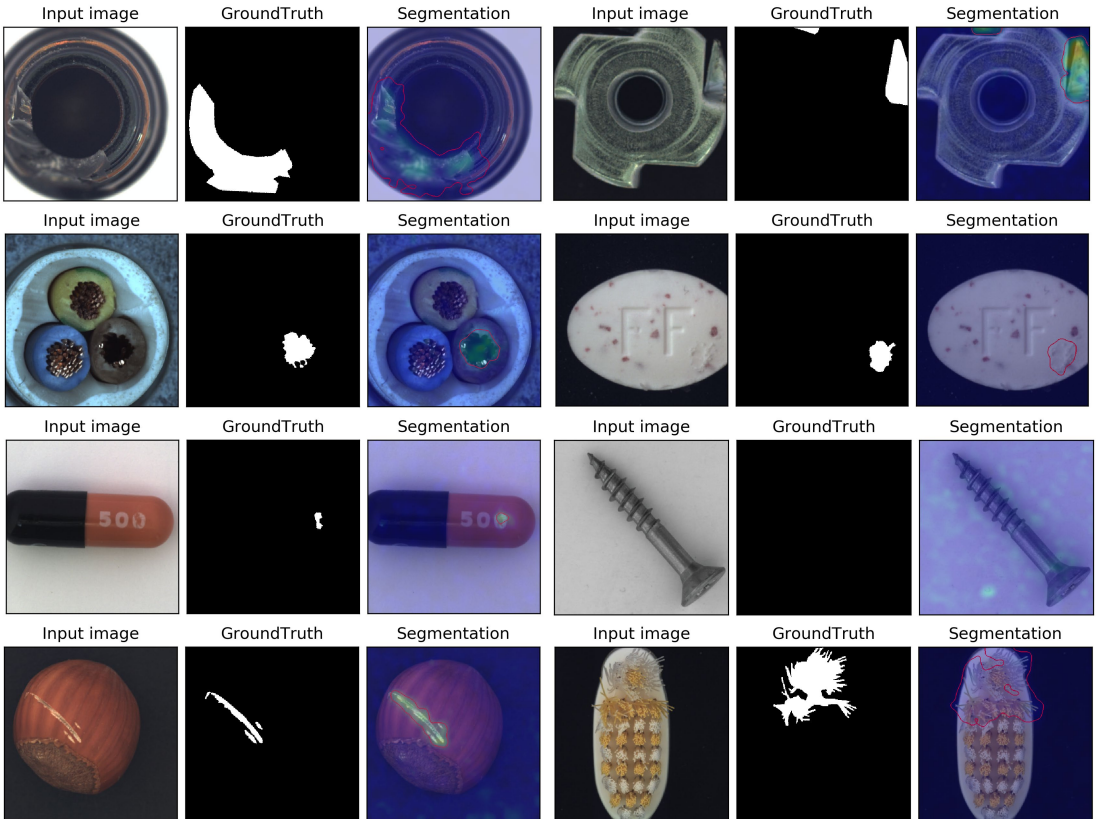

Figure 8: Anomaly localization results of the non-aligned disturbed MVTec AD datasets.

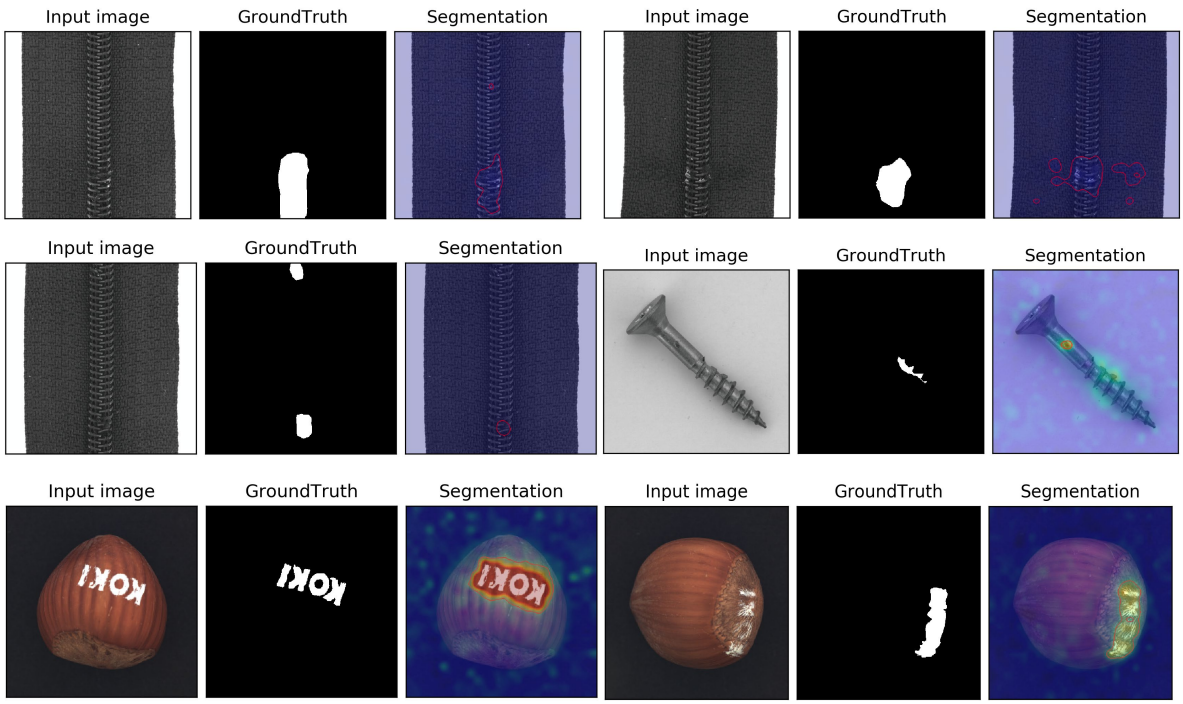

图7：由标签歧义导致的错误案例。前两行中，我们的方法定位到了异常区域但未被标注。在榛子类别的最后一行，我们展示了“印刷”子类的标签歧义问题——其中一个榛子印刷图案被精细标注，而另一个仅用粗略区域进行标注。

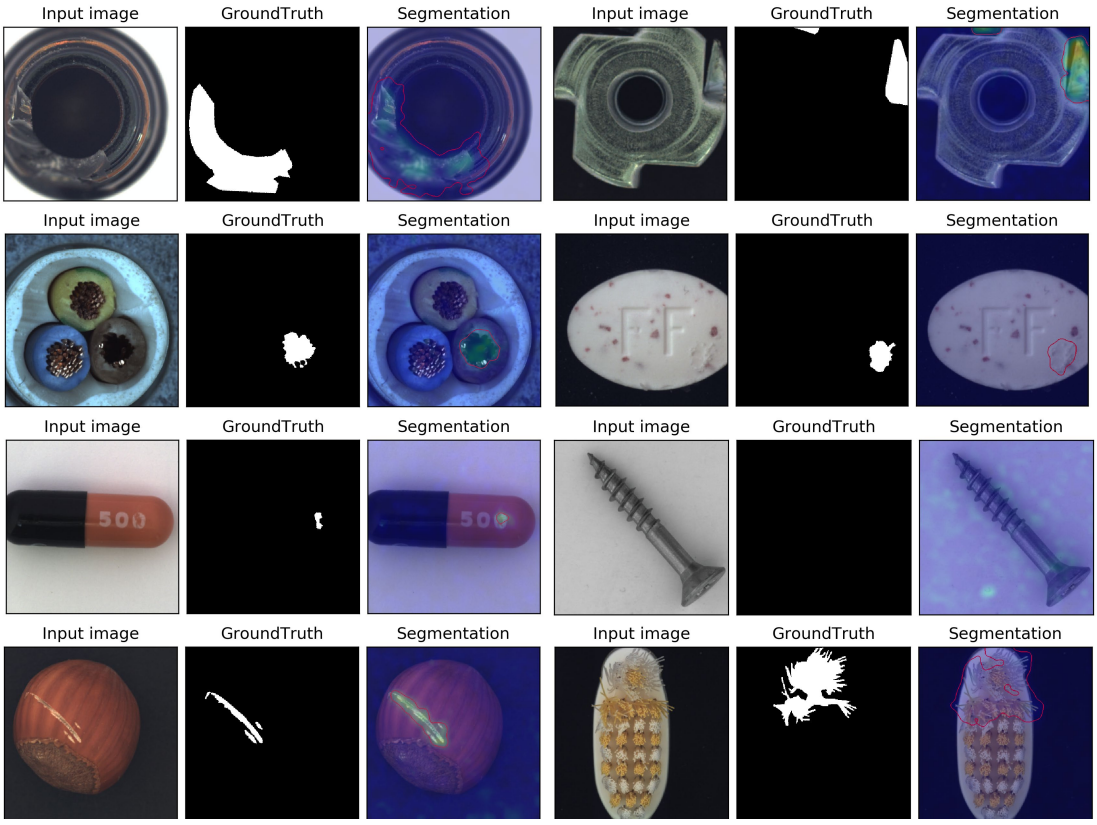

图8：未对齐扰动MVTec AD数据集的异常定位结果。