捕捉灰天鹅和黑天鹅：开集监督异常检测

摘要

尽管大多数现有的异常检测研究只假设有正常的训练样本，但在许多现实世界的应用中往往有一些标记的异常例子，如随机质量检查中发现的缺陷样本，日常医疗检查中由放射科医生确认的病变图像等。这些异常例子提供了关于特定应用异常的有价值的知识，使得在最近的一些模型中对类似异常的检测有了明显的改善。然而，在训练过程中看到的那些异常往往不能说明每一种可能的异常类别，使得这些模型不能有效地归纳出未见过的异常类别。本文讨论了开放集监督的异常检测，其中我们使用异常实例学习检测模型，目的是检测已见的异常（”灰天鹅”）和未见的异常（”黑天鹅”）。我们提出了一种新的方法，学习由所见异常、伪异常和潜伏的残余异常（即在潜伏空间中与正常数据相比具有不寻常的残余的样本）所说明的异常的分解表示，最后两种异常被设计用来检测未见异常。在 9 个真实世界的异常检测数据集上进行的广泛实验表明，我们的模型在不同的设置下，在检测可见和不可见的异常方面有卓越的表现。

关键词：Open-set；Anomaly Detection; Detection Modelling; Reset

1 引言

异常检测（AD）旨在识别不符合预期模式的特殊样本 [28]。它在不同领域有广泛的应用，例如，医学图像分析中的病变检测 [29]，工业检测中的微裂纹/缺陷检测 [3, 5]，视频监控中的犯罪/事故检测 [11, 12]，以及自动驾驶中的未知物体检测 [10]。现有的大多数异常检测方法 [2, 8, 13] 是无监督的，它们假设只有正常的训练样本，即无异常的训练数据，因为很难，甚至不可能收集大规模的异常数据。然而，在许多相关的实际应用中，往往有少量（例如，一个到多个）标记的异常实例，例如在随机质量检查中发现的一些缺陷样本，日常医疗检查中由放射科医生确认的病变图像等。这些异常实例提供了关于特定应用异常的宝贵知识 [26]，但无监督检测器无法利用它们。

由于缺乏关于异常的知识，无监督模型中学习的特征没有足够的鉴别力来区分异常（尤其是一些具有挑战性的异常）和正常数据，如图 1 中的 KDAD，一个最近最先进的（SotA）

无监督方法，在两个 MVTec AD 缺陷检测数据集 [3] 上的结果说明了这一点。近年来，有一些研究 [4] 在探索监督检测范式，旨在利用那些小的、容易获得的异常数据–罕见但先前发生的特殊案例/事件，又称灰天鹅 [21]–来训练异常情况下的检测模型。这条线上目前的方法主要是使用单类度量学习来拟合这些异常例子，将异常现象作为负面样本 [7] 或单侧的以异常现象为重点的偏差损失 [9]。尽管异常数据的数量有限，但他们在检测与训练期间看到的异常例子相似的异常现象方面取得了很大的改进。然而，这些看到的异常往往不能说明每一类可能的异常，因为 i）异常本身是未知的，ii）看到的和未看到的异常类别可能在很大程度上彼此不同 [14, 15]，例如，颜色污渍的缺陷特征与皮革缺陷检测中的褶皱和切割的缺陷特征非常不同。

为了解决这个问题，本文讨论了开放集监督异常检测 [16]，在开放集环境中，检测模型是使用小的异常例子来训练的，也就是说，目标是检测看到的异常（”灰天鹅”）和未看到的异常（”黑天鹅”）。为此，我们提出了一种新的异常检测方法，称为 DRA，它可以学习异常情况的分解表征，以实现普遍的检测。特别是，我们将无界的异常情况分解为三个大类：与有限的可见异常情况相似的异常情况、与从数据增强或外部数据源创建的伪异常情况相似的异常情况，以及在一些基于潜在残差的复合特征空间中可以检测到的未见异常情况。我们进一步设计了一个多头网络，用单独的头强制学习这三种分离的异常的每一种类型。这样一来，我们的模型就可以学习多样化的异常表征，而不是只学习已知的异常，这样就可以从正常数据中分辨出看到和未看到的异常。

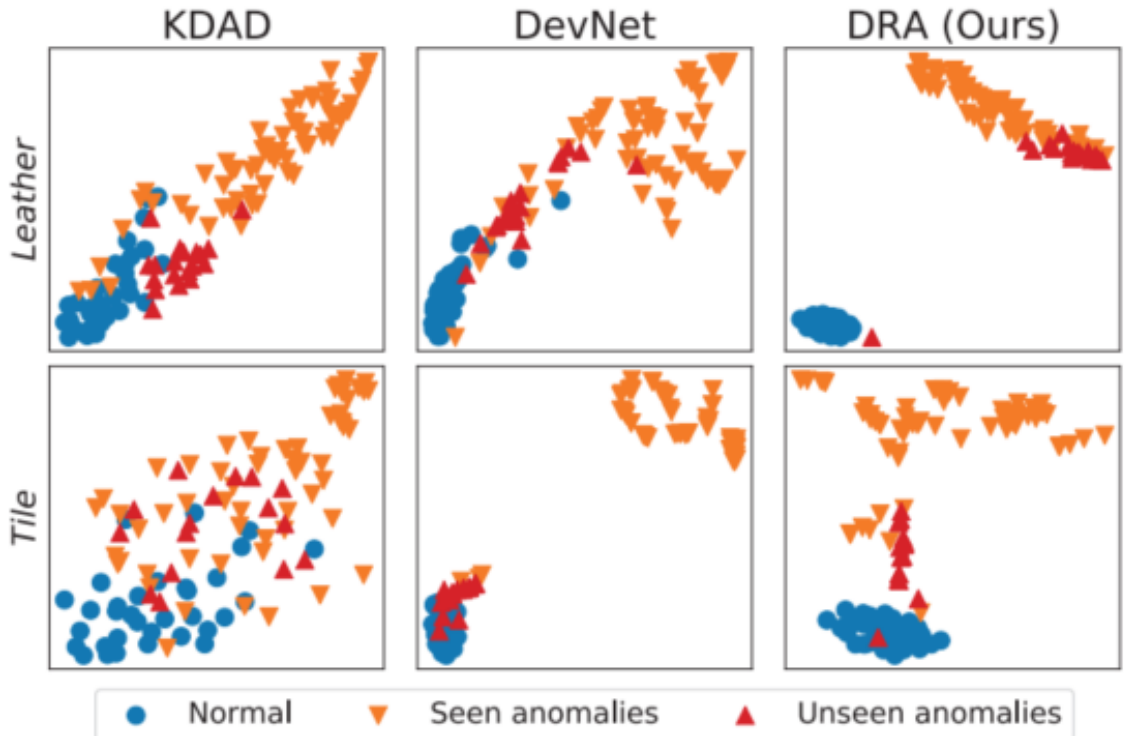

图 1. 特征的 t-SNE 可视化

2 相关工作

2.1 无监督的方法

大多数现有的异常检测方法，如基于自动编码器的方法 [13,19]、基于 GAN 的方法 [17]、自我监督的方法 [30] 和单类分类方法 [20]，假定在训练期间只能访问正常数据。尽管它们没有偏向所见异常的风险，但由于缺乏对真实异常的了解，它们很难将异常与正常样本区分开来。

2.2 监督的方法

最近出现的一个方向是监督（或半监督）异常检测，通过利用小的异常例子来学习异常信息模型，缓解了异常信息的缺乏。这是通过将异常点作为负样本的单类度量学习 [1] 或单侧异常点关注的偏差损失 [22,24] 实现。然而，这些模型在很大程度上依赖于所看到的异常情况，并可能过度拟合已知的异常情况。在 [27] 中引入了一种强化学习方法来缓解这种过拟合问题，但是它假设有大规模的未标记数据，并且在这些数据中存在未见过的异常现象。监督异常检测与不平衡分类 [6] 类似，它们都是用少数标记的例子检测罕见的类别。然而，由于异常现象的非约束性和不可知性，异常检测本质上是一个开放集任务，而不平衡分类任务通常被表述为一个封闭集问题。

2.3 学习分布内和分布外

分布外（OOD）检测 [18,19] 和开放集识别 [23,25] 是与我们相关的任务。然而，他们的目标是在检测 OOD/不确定样本的同时保证准确的多类内尔分类，而我们的任务只关注异常检测。此外，尽管使用像异常点暴露这样的伪异常现象 [18, 20] 显示了有效的性能，但这两个任务中的现有模型也被认为是无法接触到任何真正的异常样本。

3 本文方法

3.1 本文方法概述

DRA 旨在学习各种异常情况的分解表征，以有效地检测可见和不可见的异常情况。学习到的异常表征包括由有限的给定异常例子说明的看到的异常，以及由伪异常和潜在的残余异常说明的看不到的异常（即在学习到的特征空间中与正常例子相比具有不寻常残余的样本）。这样一来，DRA 减轻了对所见异常现象的偏见问题，并学习了通用的检测模型。图 2(a) 提供了提出的框架的高层次概述，它由三个主要模块组成，包括已见异常、伪异常和潜在的残

余异常学习头。前两个头在普通（非复合）特征空间中学习异常表示，如图 2(b) 所示，而最后一个头通过研究输入样本的残余特征与一些参考（即正常）图像在学习特征空间中的偏差，学习复合异常表示，如图 2(c) 所示。

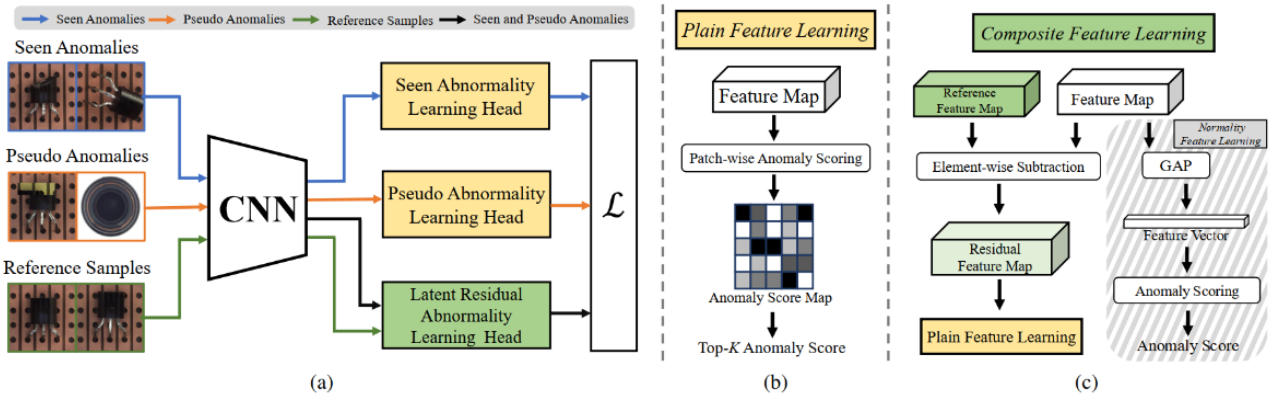

图 2. 方法框架示意图

给定一个特征提取网络 $f : x \to \mathcal{M}$，用于从训练图像 $\mathbf{x} \in X \subset \mathbb{R}^{c \times h \times w}$ 中提取中间特征图，以及一组异常学习头 $\mathcal{G} = \{g_i\}_{i=1}^{|\mathcal{G}|}$，其中每个头 $g : \mathcal{M} \mapsto \mathbb{R}$ 学习一种类型的异常得分，则可以如下给出 DRA 的总体目标:

$$\arg\min_{\Theta} \sum_{i=1}^{|\mathcal{G}|} \ell_i \left(g_i \left(f \left(\mathbf{x}; \Theta_f \right); \Theta_i \right), y_{\mathbf{x}} \right), \tag{1}$$

其中 Θ 包含所有权重参数，$y_{\mathbf{x}}$ 表示 x 的监督信息，ℓ_i 表示一个头的损失函数。特征网络 f 是由所有下游的异常学习头共同优化的，而这些头在学习具体的异常时是相互独立的。

3.2 学习分离的异常点

3.2.1 用所见的异常学习异常

DRA 利用基于 $top-K$ 多实例学习（MIL）的方法来有效地学习所看到的异常情况。如图 2(b) 所示，对于每个输入图像 x 的特征图 $\mathbf{M}_{\mathbf{x}}$，我们生成像素向量表示 $\mathcal{D} = \{\mathbf{d}_i\}_{i=1}^{h' \times w'}$，每个表示对应于输入图像的一个小斑块的特征向量。然后，这些像素向量表示被映射到异常分类器 g_s 以学习图像斑块的异常分数。由于只有选择性的图像斑块包含异常特征，我们利用 $top-K$ MIL 的优化，根据最异常的 K 个图像斑块来学习一个图像的异常得分，损失函数定义如下:

$$\ell_s \left(\mathbf{x}, y_{\mathbf{x}} \right) = \ell \left(g_s \left(\mathbf{M}_{\mathbf{x}}; \Theta_s \right), y_{\mathbf{x}} \right) \tag{2}$$

其中是 ℓ 二元分类损失函数, 如果 $y_{\mathbf{x}}$=0 则为正常样本，其中

$$g_s\left(\mathbf{M}_{\mathbf{x}};\Theta_s\right)=\max_{\Psi_K(\mathbf{M}_{\mathbf{x}})\subset\mathcal{D}}\frac{1}{K}\sum_{\mathbf{d}_i\in\Psi_K(\mathbf{M}_{\mathbf{x}})}g_s\left(\mathbf{d}_i;\Theta_s\right)\tag{3}$$

其中 $\Psi_K\left(\mathbf{M}_{\mathbf{x}}\right)$ 是一组 K 个向量，在 $\mathbf{M}_{\mathbf{x}}$ 的所有向量中具有最大的异常得分。

3.2.2 伪异常学习

设计了一个单独的头来学习与所见异常不同的异常，并模拟一些可能的未见异常的类别。有两种有效的方法来创建这种伪异常，包括基于数据增强的方法 [26,54] 和离群点暴露 [18,42]。对于基于数据增强的方法，本文改编了流行的 CutMix 方法 [67]，从正常图像 $\mathbf{x}_n$ 中生成伪异常图像 $\tilde{\mathbf{x}}$ 用于训练:

$$\tilde{\mathbf{x}}=T\circ C\left(\mathbf{R}\odot\mathbf{x}_n\right)+\left(\mathbf{1}-T(\mathbf{R})\right)\odot\mathbf{x}_n\tag{4}$$

其中，$\mathbf{R}\in\{0,1\}^{h\times w}$ 表示随机矩形的二进制掩码，1 是全一矩阵，$\odot$ 是元素相乘，$T(\cdot)$ 是随机平移变换，$C(\cdot)$ 是随机颜色抖动。如图 2(a) 所示，伪异常学习使用与所见异常学习相同的架构和异常评分方法来学习细粒度的伪异常特征:

$$\ell_p\left(\mathbf{x},y_{\mathbf{x}}\right)=\ell\left(g_p\left(\mathbf{M}_{\mathbf{x}};\Theta_p\right),y_{\mathbf{x}}\right)\tag{5}$$

其中，如果 x 是伪异常，即 x=$\tilde{\mathbf{x}}$，则 $y_{\mathbf{x}}$=1; 其中 g_p 是在一个单独的头中用与 g_s 不同的异常数据和参数来学习伪异常的训练的。

3.2.3 潜在的残差异常学习

有些异常现象，如以前未知的异常现象，与所见的异常现象没有共同的异常特征，与正常样本只有很小的差异，只用异常现象本身的特征是很难检测出来的，但只要复合特征具有较强的鉴别力，就可以在高阶复合特征空间中轻松检测出来。由于异常现象的特点是与正常数据的差异，我们利用异常现象的特征和正常特征表征之间的差异来学习这种辨别性的复合特征。更具体地说，本文提出了潜在的残差异常学习，即根据样本的特征残差与一些参考图像（正常图像）的特征相比较，在学习的特征空间中学习异常的分数。如图 2(c) 所示，为了获得潜在特征残差，首先使用从正常数据中随机抽取的一小部分图像作为参考数据，并计算其特征图的平均值以获得参考正常特征图:

$$\mathbf{M}_r=\frac{1}{N_r}\sum_{i=1}^{N_r}f\left(\mathbf{x}_{r_i};\Theta_f\right)\tag{6}$$

其中 $\mathbf{x}_{r_i}$ 是参考正常图像，N_r 是一个超参数，表示参考集的大小。对于一个给定的训练图像 x，我们在其特征图 $\mathbf{M_x}$ 和对所有训练和测试样本固定的参考正态特征图 $\mathbf{M_r}$ 之间进行逐元相减，结果是 x 的残余特征图 $\mathbf{M}_{r\ominus\mathbf{x}}$:

$$\mathbf{M}_{r\ominus\mathbf{x}} = \mathbf{M}_r \ominus \mathbf{M}_\mathbf{x} \tag{7}$$

其中 $\ominus$ 表示逐项减去。然后，对这些残差的特征进行异常分类。

$$\ell_r\left(\mathbf{x}, y_\mathbf{x}\right) = \ell\left(g_r\left(\mathbf{M}_{r\ominus\mathbf{x}}; \Theta_r\right), y_\mathbf{x}\right) \tag{8}$$

其中，如果 x 是一个已见的异常或伪异常则 $y_\mathbf{x}$=1，否则，如果 x 是一个正常的样本，$y_\mathbf{x}$=0。g_r 使用与公式 3 中 g_s 完全相同的方法来获得异常得分，但它是在一个单独的头中使用不同的训练输入，即残差特征图 $\mathbf{M}_{r\ominus\mathbf{x}}$ 来训练参数 Θ_r。

由于 g_s、g_p 和 g_r 头侧重于学习异常表征，f 中联合学习的特征图不能很好地模拟正常特征。为了解决这个问题，增加了一个单独的正常学习头，如下所示。

$$\ell_n\left(\mathbf{x}, y_\mathbf{x}\right) = \ell\left(g_n\left(\frac{1}{h'\times w'}\sum_{i=1}^{h'\times w'}\mathbf{d}_i; \Theta_n\right), y_\mathbf{x}\right) \tag{9}$$

其中 $g_n : \mathcal{D} \to \mathbb{R}$ 是一个完全连接的二进制异常分类器，可以区分正常样本和所有可见的伪异常。

4 复现细节

4.1 与已有开源代码对比

复现过程中参考了 Github 上的开源代码 (https://github.com/Choubo/DRA) 的官方 Pytorch 实现，这次的复现工作基于 Pytorch 进行了改进，包括对网络模型的改进和对损失函数的改进。

4.1.1 源代码损失函数

原开源代码自定义的损失函数，称为 DeviationLoss，用于异常检测任务；在 forward 中计算了一个基于标准化偏差的损失，其中使用了正态分布的参考值。然后计算其内点和外点的损失，返回损失的平均值作为最终的损失值。如下 DeviationLoss 的具体代码:

```
1 import torch
2 import torch.nn as nn
3 #定义了一个名为DeviationLoss的PyTorch模块
```

```
class DeviationLoss(nn.Module):
#调用父类nn.Module的初始化方法
  def __init__(self):
    super().__init__()
  #实现DeviationLoss的前向传播方法
  def forward(self, y_pred, y_true):
    #设置一个置信边界，该值设定了对异常值的容忍度
    confidence_margin = 5.
    #生成一个具有5000个元素的正态分布张量 ref，用作参考值
    ref = torch.normal(mean=0., std=torch.full([5000], 1.)).cuda()
    #计算预测值与参考值的标准化偏差
    dev = (y_pred - torch.mean(ref)) / torch.std(ref)
    #计算内点和外点损失
    inlier_loss = torch.abs(dev)
    outlier_loss = torch.abs((confidence_margin - dev).clamp_(min=0.))
    #根据真实标签，计算最终的损失 dev_loss
    dev_loss = (1 - y_true) * inlier_loss + y_true * outlier_loss
    #返回损失的平均值作为最终的损失值
    return torch.mean(dev_loss)
```

4.1.2 改进策略

基于我们的任务是对异常进行检测和数据的特性，对上述的 DeviationLoss 损失函数保留了偏差计算，做了以下改进：

1. 首先对于使用静态的正态分布参考值可能不够灵活。所以考虑使用动态的参考值，可以根据当前批次或数据动态计算参考值（从均值为 0、标准差为 1 的正态分布中抽取的随机值），从而使得计算的标准化偏差更小一些：

```
  ref = torch.normal(mean=0., std=torch.ones_like(y_pred)).cuda()
```

2. 考虑到不同数据样本的重要性可能不同，考虑引入样本权重，使得模型更关注一些重要的样本：

```
  sample_weights = torch.abs(y_true - 0.5)  # 根据样本类别赋予不同权重
dev_loss = (1 - y_true) * inlier_loss * sample_weights +
y_true * outlier_loss * sample_weights
```

3. 由于数据的特性以及模型对偏差的敏感小，对于内外点损失的计算由原来的绝对值损失变为了平方损失，而且外点损失使用了 Relu 线性修正单元:

```
    inlier_loss = dev.pow(2)
    outlier_loss = F.relu(confidence_margin - dev).pow(2)
```

DeviationLoss 损失函数改进后 DWeightedDeviationLoss 的损失函数具体代码如下:

```
import torch
import torch.nn as nn
import torch.nn.functional as F

class DWeightedDeviationLoss(nn.Module):
    def __init__(self, confidence_margin=5.0):
        super(DWeightedDeviationLoss, self).__init__()
        self.confidence_margin = confidence_margin
    def forward(self, y_pred, y_true):
        # 动态参考值
        ref = torch.normal(mean=0., std=torch.ones_like(y_pred)).cuda()
        # 偏差计算
        dev = (y_pred - torch.mean(ref)) / torch.std(ref)
        # 内点和外点损失
        inlier_loss = dev.pow(2)
        outlier_loss = F.relu(confidence_margin - dev).pow(2)
        # 样本权重
        sample_weights = torch.abs(y_true - 0.5)
        # 带权重的偏差损失计算
        dev_loss = (1 - y_true) * inlier_loss * sample_weights +
                        y_true * outlier_loss * sample_weights
        return torch.mean(dev_loss)
```

4.2 实验环境搭建

4.2.1 依赖

python 3.7

matplotlib 3.3.3

numpy 1.18.5

pandas 0.25.3

Pillow 8.4.0

scikit-learn 1.0.1

torch 1.1.0

torchvision 0.3.0

tqdm 4.54.0

opencv-python 3.4.2

4.2.2 使用说明

数据集的格式转换：python convert-AITEX.py –dataset-root=./AITEX

训练 DRA 模型：python train.py
–dataset-root=./data/SDD-anomaly-detection
–classname=SDD
–experiment-dir=./experiment

4.2.3 参数说明

data-root：需要训练的图像路径 (可自已创建新目录，默认./AITEX)；

network: 网络名称 (可选, 默认 resnet18)；

experiment-dir：表示存储试验设置和模型权重的路径；

classname: 表示数据集的子集名称;

total-heads：训练中的头数 (基于多头神经网络的模型 DRA);

nAnomaly：训练集中异常数据的数量;

criterion：损失函数的类型 (本文提供了 2 个损失函数 Deviation Loss 和 Binary Cross Entropy Loss)；

数据集使用：Kolektor 表面缺陷数据集（KolektorSDD）

4.3 创新点

相比于其它的无监督模型（全部采用无缺陷样本），现实中提供极少数的异常样本是可能的，然而少数的缺陷样本提供了关于特定应用中异常的有价值的知识，而这是那些无监督模型无法利用的；现有的利用少数异常样本的模型，存在在已见的缺陷上性能提升大（性能优于无监督模型），而在未见上的性能表现一般（性能差于无监督模型）。

本文弥补了在未见异常检测上的性能，将异常分为三个大类：与已见的异常相似的、与数据增强或外部数据相似的伪异常、在潜在的基于残差的复合空间中的可检测到的未见的异常；设计了一个多头网络，不同的头分别用于学习这三种不同的异常。模型给出异常分，目标是给已见和未见赋予高于正常样本的异常分。

其次利用 top-K 的基于多样本学习（MIL）方法来有效的学习已见异常；对于非医疗数据集利用 CutMix 方法从正常图像生成伪异常图像，针对医疗数据集采用（the outlier exposure method）生成伪异常图像；未见的异常在高阶复合特征空间很容易检测：1）利用异常和正常特征之间的特征差异来学习判别复合特征；2）根据样本的特征残差与一些参考图像的特征在一个学习过的特征空间中进行比较学习样本的异常分数。

通过异常评分函数，对于每个输入的样本给出一个异常评分，然后通过定义 L 个正常样本的异常分数的均值（用标准正态分布 N（0，1）估计）生成参考评分。最后用损失函数优化模型的异常评分，使得异常样本的得分显著偏离参考得分，正常样本的得分接近参考得分。

5 实验结果分析

5.1 性能指标

本实验用 KolektorSDD 数据集包含具有 52 张可见缺陷的图像，347 张无任何缺陷的图像，然后转换异常检测数据集的脚本，其中会将每个图像分成三个部分（垂直切分为三等份），并提取相应的标签；从而生成 847 张无缺陷的训练图像，在训练中从异常类数据随机取（1 或 10）个异常样本来学习可见的异常，但在测试时不包含训练时随机取的异常样本；其次通过正常图像生成伪异常，从而学习伪异常。实验运行会得出两个性能指标：AUC-ROC 和 AUC-PR 都是衡量 ROC 和 PR 曲线下的面积，因此它们提供了在所有可能的分类阈值下，模型正确分类正负样本的概率。AUC-ROC 的值在 0 到 1 之间，值越接近 1 表示模型性能越好；下面是原文实验的一个结果，用 9 个真实的异常检测数据集，以 KDAD 方法为基准，分别就 DevNet,FLOS,SAOE,MLEP,DRA 6 种异常检测模型进行了性能的比较：

Dataset	|e|	Baseline	One Training Anomaly Example					Ten Training Anomaly Examples				
		KDAD	DevNet	FLOS	SAOE	MLEP	DRA (Ours)	DevNet	FLOS	SAOE	MLEP	DRA (Ours)
MVTec AD	-	0.861±0.009	0.794±0.014	0.792±0.014	**0.834**±0.007	0.744±0.019	**0.883**±0.008	**0.945**±0.004	0.939±0.007	0.926±0.010	0.907±0.005	**0.959**±0.003
AITEX	12	0.576±0.002	0.598±0.070	0.538±0.073	**0.675**±0.094	0.564±0.055	**0.692**±0.124	**0.887**±0.013	0.841±0.049	0.874±0.024	0.867±0.037	**0.893**±0.017
SDD	1	0.888±0.005	**0.881**±0.009	0.840±0.043	0.781±0.009	0.811±0.045	**0.859**±0.014	**0.988**±0.006	0.967±0.018	0.955±0.020	0.983±0.013	**0.991**±0.005
ELPV	2	0.744±0.001	0.514±0.076	0.457±0.056	**0.635**±0.092	0.578±0.062	**0.675**±0.024	**0.846**±0.022	0.818±0.032	0.793±0.047	0.794±0.047	**0.845**±0.013
Optical	1	0.579±0.002	0.523±0.003	0.518±0.003	**0.815**±0.014	0.516±0.009	**0.888**±0.012	0.782±0.065	0.720±0.055	**0.941**±0.013	0.740±0.039	**0.965**±0.006
Mastcam	11	0.642±0.007	0.595±0.016	0.542±0.017	**0.662**±0.018	0.625±0.045	**0.692**±0.058	0.790±0.021	0.703±0.029	**0.810**±0.029	0.798±0.026	**0.848**±0.008
BrainMRI	1	0.733±0.016	**0.694**±0.004	0.693±0.036	0.531±0.060	0.632±0.017	**0.744**±0.004	0.958±0.012	0.955±0.011	0.900±0.041	**0.959**±0.011	**0.970**±0.003
HeadCT	1	0.793±0.017	0.742±0.076	0.698±0.092	0.597±0.022	**0.758**±0.038	**0.796**±0.105	**0.982**±0.009	0.971±0.004	0.935±0.021	**0.972**±0.014	**0.972**±0.002
Hyper-Kvasir	4	0.401±0.002	0.653±0.037	**0.668**±0.004	0.498±0.100	0.445±0.040	**0.690**±0.017	**0.829**±0.018	0.773±0.029	0.666±0.050	0.600±0.069	**0.834**±0.004

图 3. 9 个真实 AD 数据集的 AUC 结果（平均值 ± 标准差）

通过对方法模型的比较，与竞争的检测器相比，我们的模型显示出更好的样本效率，即：

i) 在减少异常例子的情况下，我们的模型的 AUC 下降得更少，即:。在 9 个数据集中，平均 AUC 下降 15.1，这比 DevNet（22.3）、FLOS（21.6）、SAOE（19.7）和 MLEP（21.6）要好得多；ii）我们用一个异常例子训练的模型可以在很大程度上超过用十个异常例子训练的强大竞争方法，如 DevNet、FLOS 和 MLEP 在光学上的表现，以及 SAOE 和 MLEP 在超 Kvasir 的表现。与无监督基线的比较。与无监督模型 KDAD 相比，我们的模型和其他监督模型在使用 10 个训练异常例子（即较少的开放集场景）时表现出持续更好的性能。

下面是我通过 KolektorSDD 数据集用 DRA 模型分用随机取 10 个异常数据样本，训练 30 次和 90 次的实验结果:

表 1. 复现的实验结果

Model: DRA				
Dataset	30 Epochs 10 Training Anomaly Example		90 Epochs 10 Training Anomaly Example	
	AUC-ROC	AUC-PR	AUC-ROC	AUC-PR
KolektorSDD	0.9918	0.9741	0.9393	0.8538

ROC 曲线是一种通过改变分类模型的阈值来绘制真正例率（True Positive Rate，又称为灵敏度）和假正例率（False Positive Rate）之间的关系的图表。横轴是 FPR，纵轴是 TPR。Precision-Recall 曲线是通过改变分类模型的阈值来绘制精确率（Precision）和召回率（Recall，又称为灵敏度）之间的关系的图表。横轴是召回率，纵轴是精确率。

下面是保存的实验参数:

```
batch_size : 48
steps_per_epoch : 20
epochs : 90
dataset : SDD
ramdn_seed : 42
no_cuda : False
savename : model.pth
dataset_root : ./data/SDD_anomaly_detection
experiment_dir : ./experiment
classname : SDD
img_size : 448
nAnomaly : 10
backbone : resnet18
criterion : deviation
```

```
top-k : 0.1
pretrain_dir : None
total_heads : 4
cuda : True
```

5.2 实验结果的可视化

实验的测试数据样本共有 44 个缺陷的图像和 286 个无缺陷的图像，以下是可视化结果，X 轴表示样本的索引，Y 轴表示模型对样本给出的异常得分和部分样本的异常得分结果：

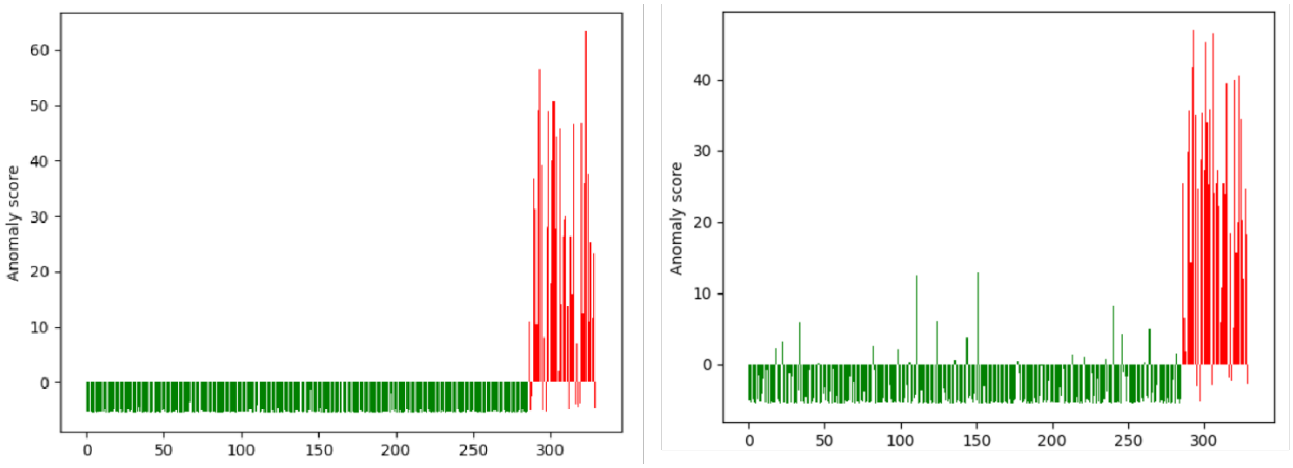

图 4. 可视化结果：左边是训练 30 次，右边是训练 90 次

表 2. 部分样本标签对应的异常得分

Label	Anomaly score	Label	Anomaly score
0.0	-5.107624087482691	1.0	12.393150687217712
0.0	-5.799913141876459	1.0	35.99082192406058
0.0	-5.875842964276671	1.0	63.34391437098384
0.0	-5.3548465222120285	1.0	37.60389802604914
0.0	-5.874393094331026	1.0	11.029805898666382
0.0	-5.893596034497023	1.0	25.288033723831177
0.0	-5.863877082243562	1.0	11.56655490398407
0.0	-5.412523992359638	1.0	23.324927926063538
0.0	-5.877066737040877	1.0	26.36124211549759
0.0	-5.662238096818328	1.0	30.00049210526049

基于改进的损失函数 DWeightedDeviationLoss，通过引入动态参考值，和给样本引入权重，将标签值平移，使得标签值为 0 的样本变为-0.5，标签值为 1 的样本变为 0.5。以便模型更关注对于模型性能更为重要的样本。在二分类问题中，将标签值平移并计算绝对值的操作可以表达出模型对于误分类的关注；最后对原文的绝对值损失计算改为平方损失使得数据样本对大偏差的敏感性较小，以下是改进后的性能指标对比结果与改进后的可视化结果（90 次训练结果和 10 个训练异常数据）：

表 3. 改进后的实验对比结果

Model: DRA		
	90 Epochs +10 Training Anomaly	
Loss	AUC-ROC	AUC-PR
DeviationLoss	0.9393	0.8538
DWeightedDeviationLoss	0.9664	0.9013

下面是在训练 90 轮和引入 10 个异常训练数据基础上，用 DeviationLoss 和 DWeighted-DeviationLoss 运行之后的效果对比图，

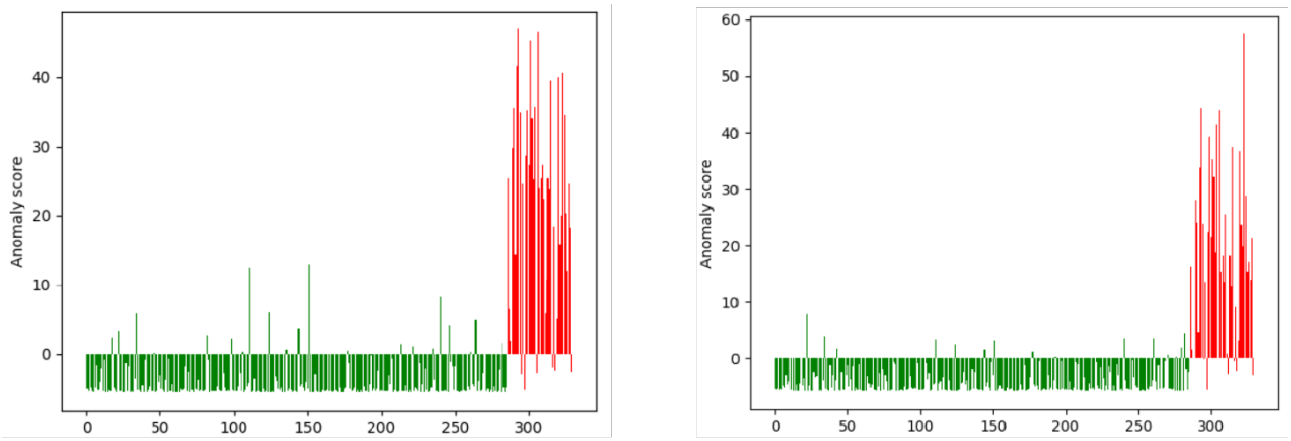

图 5. 改进后的实验对比结果图

6 总结与展望

本文提出了学习由可见异常、伪异常和基于潜在残差的异常所说明的异常的分解表示的框架，并介绍了 DRA 模型来有效地检测可见和不可见的异常。然后通过对 Deviation 损失函数的改进，我在表 3 和图 5 中的结果证明，特别是在具有挑战性的情况下，例如，只有一个训练异常例子，或检测未见过的异常。所研究的问题在很大程度上没有得到充分的探索，但它在许多相关的现实世界的应用中是非常重要的。仍有许多重大挑战需要进一步研究，例如，从较少的类中的较小的异常例子中进行归纳。

参考文献

[1] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016.

[2] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020.

[3] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019.

[4] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4183–4192, 2020.

[5] Hanna Borgli, Vajira Thambawita, Pia H Smedsrud, Steven Hicks, Debesh Jha, Sigrun L Eskeland, Kristin Ranheim Randel, Konstantin Pogorelov, Mathias Lux, Duc Tien Dang Nguyen, et al. Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. *Scientific data*, 7(1):283, 2020.

[6] Paula Branco, Luís Torgo, and Rita P Ribeiro. A survey of predictive modeling on imbalanced domains. *ACM computing surveys (CSUR)*, 49(2):1–50, 2016.

[7] Raghavendra Chalapathy, Aditya Krishna Menon, and Sanjay Chawla. Anomaly detection using one-class neural networks. *arXiv preprint arXiv:1802.06360*, 2018.

[8] Yuanhong Chen, Yu Tian, Guansong Pang, and Gustavo Carneiro. Deep one-class classification via interpolated gaussian descriptor. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 383–392, 2022.

[9] Sergiu Deitsch, Vincent Christlein, Stephan Berger, Claudia Buerhop-Lutz, Andreas Maier, Florian Gallwitz, and Christian Riess. Automatic classification of defective photovoltaic module cells in electroluminescence images. *Solar Energy*, 185:455–468, 2019.

[10] Giancarlo Di Biase, Hermann Blum, Roland Siegwart, and Cesar Cadena. Pixel-wise anomaly detection in complex driving scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16918–16927, 2021.

[11] Mariana-Iuliana Georgescu, Antonio Barbalau, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah. Anomaly detection in video via self-supervised and multi-task learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12742–12752, 2021.

[12] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. *Advances in neural information processing systems*, 31, 2018.

[13] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019.

[14] Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013.

[15] Haibo He and Edwardo A Garcia. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9):1263–1284, 2009.

[16] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[17] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

[18] Jinlei Hou, Yingying Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, and Hong Zhou. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8791–8800, 2021.

[19] Rui Huang and Yixuan Li. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8710–8719, 2021.

[20] Nima Khakzad, Faisal Khan, and Paul Amyotte. Major accidents (gray swans) likelihood modeling using accident precursors and approximate reasoning. *Risk analysis*, 35(7):1336–1347, 2015.

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[22] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9664–9674, 2021.

[23] Liu Li, Mai Xu, Xiaofei Wang, Lai Jiang, and Hanruo Liu. Attention based glaucoma detection: A large-scale database and cnn model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10571–10580, 2019.

[24] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[25] Ziqian Lin, Sreya Dutta Roy, and Yixuan Li. Mood: Multi-level out-of-distribution detection. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 15313–15323, 2021.

[26] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2537–2546, 2019.

[27] 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.

[28] Amir Markovitz, Gilad Sharir, Itamar Friedman, Lihi Zelnik-Manor, and Shai Avidan. Graph embedded pose clustering for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10539–10547, 2020.

[29] Javier Silvestre-Blanes, Teresa Albero-Albero, Ignacio Miralles, Rubén Pérez-Llorens, and Jorge Moreno. A public fabric database for defect detection methods and results. *Autex Research Journal*, 19(4):363–374, 2019.

[30] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. Unmasking the abnormal events in video. In *Proceedings of the IEEE international conference on computer vision*, pages 2895–2903, 2017.