

基于背景-前景组成式建模的电路板异常检测

傅冰飞¹ 陈同林¹ 许枫² 朱麟² 李斌¹ 薛向阳¹

¹(复旦大学计算机科学技术学院 上海 200438)

²(联想研究院人工智能实验室 北京 100094)

(18110240021@fudan.edu.cn)

Circuit Boards Anomaly Detection Based on Background-Foreground Compositional Modeling

Fu Bingfei¹, Chen Tonglin¹, Xu Feng², Zhu Lin², Li Bin¹, and Xue Xiangyang¹

¹(School of Computer Science, Fudan University, Shanghai 200438)

²(Artificial Intelligence Lab, Lenovo Research, Beijing 100094)

Abstract Anomaly detection aims to detect abnormal samples among many normal samples. In the era of big data, how to apply anomaly detection to real-world scenarios has become one of the most critical problems to consider. Currently, the existing models can hardly cope with dynamic interference such as occlusion, lighting, and color difference in real-world scenarios and cannot quickly migrate application scenarios. We propose a deep learning model based on background-foreground modeling for anomaly detection tasks. Our model first reconstructs the input image into a clean background image without abnormal objects through the feature extraction network and preserves the possible dynamic interference of the image through skip-connection. After obtaining the reconstructed background, this model extracts the position information of abnormal objects through the spatial transformation network, uses an autoencoder to extract latent space representations of the appearance, shape, and presence information of abnormal objects, and reconstructs them. Finally, this model combines the reconstructed abnormal objects and the background image to obtain an overall reconstructed image and realizes anomaly detection by setting a threshold for the presence information of abnormal objects. To validate the effectiveness of the method, we collect data from a real circuit board assembly environment and simulate a scenario with limited annotations in actual production, resulting in the creation of a foreign object in circuit board (FO-CB) dataset for analysis. Additionally, we also conduct experimental validation on the foreign object debris in airport (FOD-A) dataset. The experimental results show that our proposed method performs well on the synthetic dataset and detects all anomalous objects in 9 actual scene data, with a miss rate of down to 0%, and can be applied to real-world circuit board assembly scenarios.

Key words anomaly detection; synthetic dataset; compositional modeling; generative model; multi-stage training

摘要 异常检测的目标是检测众多正常样本中的异常样本。在大数据时代,如何将异常检测应用于现实场景成为当下需要着重思考的问题之一。目前已有模型存在难以处理实际场景中遮挡、光照、色差等动态干扰,无法快速迁移应用场景等问题。基于此,提出了一种基于背景-前景组成式建模的深度学习模型,用于检测电路板场景中的异常物体。首先通过特征提取网络将输入图像重构为不包含异常物体的干净背景

收稿日期: 2023-07-03; 修回日期: 2024-05-29

基金项目: 上海市科学技术委员会项目(22511105000); 联想科学家计划项目

This work was supported by the Science and Technology Commission of Shanghai Municipality Program (22511105000) and Lenovo Scientists Program.

通信作者: 薛向阳(xyue@fudan.edu.cn)

图像,并通过跳层连接保留图像可能存在的动态干扰.得到重构背景后,通过空间变换网络提取到异常物体的位置信息,利用自编码器提取到异常物体外观、形状和存在的隐空间表示并重构出每个异常物体.将重构的异常物体和背景图像组合得到完整图像并通过对异常物体的存在表示给定阈值来实现异常检测.为了验证方法的有效性,从真实的电路板组装环境中收集数据,并模拟实际生产中标注有限的情景,从而创建用于分析的电路板异物数据集.此外,还在航道异物碎片数据集上进行实验验证.结果表明,提出的方法在该数据集上表现良好,能够检测出9个真实场景数据中的所有异常目标,漏检率低至0%,可以应用于现实世界的电路板组装场景.

关键词 异常检测;合成数据集;组成式建模;生成式模型;多阶段训练

中图法分类号 TP391

DOI: 10.7544/issn1000-1239.202330565

CSTR: 32373.14.issn1000-1239.202330565

异常检测作为计算机领域中的一项关键任务,在现实世界中有着广泛的应用(时序异常^[1-2]、网络异常^[3-4]、航道异物^[5-6]和安全监控^[7]等).视觉领域中异常检测的目标是识别出正常样本中的异常样本,又分为对图像整体进行异常检测以及对图像中的物体进行异常检测和定位.在现实场景中,由于视觉数据固有的复杂性(光照变化和像素级噪点等)和潜在干扰的多样性(遮挡、形变、异物干扰等),异常检测模型需要同时兼顾良好的泛化性和较高的准确率,因此这项研究具有较高的挑战性.良好的异常检测模型可以应用于工业生产^[8-12]、医疗诊断^[13-15]和视频监控^[16]等诸多领域,替代人工进行异常情况的检测和判断,极大提高工作效率,具有重要的现实意义.

传统异常检测研究可以分为基于分类、聚类、最近邻、统计信息和信息熵方法^[17-18]等,而最近的基于深度学习的异常检测方法大多是在基于分类的方法上进行拓展,同时融合聚类、最近邻等机器学习方法,大致可以分为基于预训练模型和基于深度生成模型这2种思路.基于预训练模型的方法^[19-24]在大数据集(如 ImageNet)^[25]上训练出较好的特征提取器,再对异常样本和正常样本的特征进行处理,进一步扩大异常样本和正常样本的特征差异,从而实现异常检测;基于深度生成模型的研究使用自编码器(auto-encoder, AE)或者生成对抗网络(generate adversarial network, GAN),经过编解码器将异常区域重构为正常区域^[26-32],通过分割结果进行异常检测.

上述2类方法均是通过学习正常样本和异常样本的特征差异实现异常检测.然而现实场景中图像的异常区域往往较小,导致正常样本和异常样本的特征差异较小.同时由于场景较为复杂,通过模型提取到的图像特征存在较大分布差异.因此无论是基于预训练模型还是基于深度生成模型的方法,在面

对复杂多变的真实场景时,均难以保持较高的异常检测性能.

基于此,本文受 Yuan 等人^[33]提出的合成场景中无限制遮挡物体的生成建模(generative modeling of infinite occluded objects, GMIOO)方法的启发,提出如下假设:图像是由正常的背景区域和可能存在异常的前景区域2部分组成的.基于上述假设,本文提出了基于背景-前景组成式建模的异常检测方法(anomaly detection based on background-foreground compositional modeling, AD-BFCM),将图像划分为背景区域和前景区域,分别进行背景建模和前景建模.在背景建模阶段,模型对输入图像进行编解码,将图片可能存在异常的区域重构为正常区域,得到干净的重构背景图像.针对图片中可能存在的各种干扰,模型通过跳层连接和对异常区域的额外约束,让模型在重构背景的同时保留干扰信息,降低模型的异常检测误识别;在前景建模阶段,模型针对图像的异常区域进行编解码,通过以物体为中心的方法^[34]学习到多个异常物体的外观、形状、位置和存在表示,再通过组成建模的方式重构出整张图像,并对异常物体的存在表示设定阈值来实现对图像中异常物体的检测和定位.

在数据集方面,现有的工业异常检测数据集如 MVTec^[8], VisA^[35]等主要关注物体表面的异常纹理,而实际工业生产往往关注场景中是否存在异常物体,大部分公开数据集难以满足这一需求.同时,由于生产工序和成本等问题,工业场景中的图片需要集中采集,数据本身存在一定的同质化.基于此,本文模拟工业生产中采集数据同质化情况,从来自工业生产的9个电路板场景图片中抽取少量数据,配合一种简单的数据合成方法构建出相当数量的合成数据集用以训练.同时剩余图片作为测试集来验证模型是否在全局真实场景中拥有较高的异常检测性能.

此外,本文还在航道异物碎片(foreign object debris in airports, FOD-A)数据集^[36]上进行方法验证.该数据集包含机场跑道上可能存在的异物碎片图片.实验结果表明本文在FOD-A数据集上的漏检率仅为2.89%,在自建的电路板异物(foreign object in circuit board, FO-CB)数据集上漏检率达到0%,检测出9个电路板场景中的全部异常物体.

本文的主要贡献包括3个方面:

1) 针对实际工业生产中异常检测的需求,利用工业生产中真实电路板图片,构建了FO-CB数据集.该数据集模拟工业生产中采集数据同质化的情况,利用少量数据配合数据合成方法得到训练集,剩余真实数据作为测试集.

2) 假设图像可划分为背景区域和前景区域,提出了一种基于背景-前景组成式建模的异常检测方法,用以检测复杂真实场景中的异常物体.模型分为背景建模阶段和前景建模阶段.在背景建模阶段,模型通过特征提取器重构出不包含异常物体的干净背景;在前景建模阶段,模型通过自编码器学习异常物体的表示,实现异常物体检测.

3) 在背景建模阶段,模型增加了跳层连接的设计和约束重构区域的损失函数,实现重构背景的同时保留干扰信息.在前景建模阶段,模型设计了针对局部前景的重构损失函数,来更好地学习异常前景表示.

1 相关工作

1.1 异常检测研究

传统的异常检测方法大多基于统计学的方法,利用距离、分布等统计信息,对正常样本和异常样本进行聚类或最近邻分析,来实现对异常样本或视频中异常物体的识别^[11-12].例如文献[37-38]基于已训练的分类器,通过计算正常样本和异常样本的分布差异,实现对图像和视频的异常检测.

基于深度学习方法的异常检测大致可以分为2个范式,即基于深度学习预训练模型的方法和基于生成模型的方法.基于深度学习预训练模型的方法^[19-24]大多通过大规模数据集训练得到特征提取器,再对异常样本特征和正常样本特征进行处理,实现异常检测. Defard 等人^[20]利用预训练的卷积神经网络(convolutional neural network, CNN)进行Patch嵌入,并通过多变量高斯分布得到正常类的概率表示,实现单类环境下异常检测和定位. Roth 等人^[21]构建了一个补丁级特征库,通过判断图像中是否存在异常

补丁来实现像素级异常检测. Gudovskiy 等人^[23]通过多尺度金字塔结构获得全局和局部语义信息,通过条件归一化流网络进行异常检测.除了对特征进行后处理,文献[39-41]对预训练模型进行知识蒸馏,通过比较教师网络和学生网络之间的异常区域特征的差异来估计异常.此外, Li 等人^[24]利用数据增强策略,将图像补丁切割并粘贴在图像随机位置,通过自监督方式学习异常表征.

基于生成模型的方法通常使用编码器将输入图像压缩成低维表示,再通过采样方式得到数据的分布,经过解码器重构输入图像,以此学习到正常数据和异常数据的特征分布差异. Dehaene 等人^[42]基于变分自编码器(variational autoencoder, VAE)模型,使用梯度下降法对自动编码器损失的能量函数进行投影,将样本投影到自动编码器正在学习的数据流上,迭代重构出正常样本. Baur 等人^[13]使用自编码器对正常的大脑MR图像进行建模,并通过比较与正常分布的偏差进行异常检测. Dehaene 等人^[28]利用变分自编码器对正常样本进行不同特征维度的重建,通过和正常样本的偏差检测异常. Hou 等人^[30]同样通过变分自编码器对正常样本进行建模,同时构建了多尺度记忆模块来储存不同分辨率的特征,并引入对抗学习来提升异常检测性能.此外,还有一些基于GAN的生成模型工作^[26-27,43],这些工作将输入图像与生成网络中生成的正常图像进行比较,通过判别网络来评估是否存在异常.

基于深度学习预训练模型的方法更加依赖于训练集的数据分布,在测试数据分布和训练数据分布差别较大时,模型难以维持稳定的高性能.基于生成模型的方法通过学习正常样本和异常样本的重构差异来进行异常检测,面对开放场景中的未知干扰时,往往会重构错误,从而导致误检测的情况.

1.2 以物体为中心的特征学习

以物体为中心表征学习方法的核心思想是:视觉场景图像可以被建模为多个物体的组成,通过学习视觉场景中出现的每个物体的表征,可以得到整个视觉场景的表征.现有的以物体为中心的特征学习方法大多使用空间混合模型或加权求和方法对视觉场景建模^[42-46].文献[44-45]使用注意力机制依次提取图像中每个物体的表征.文献[33]首先学习场景图像的表征,然后依次学习每个物体的表征,并进行迭代更新.文献[46-47]利用卷积神经网络生成候选物体,在包含大量物体的场景中有较好表现.文献[34, 48]初始化所有物体的表征,然后通过竞争机制进行

表征的迭代更新. 文献[49–50]将视觉场景中的结构信息结合到深度生成模型, 以便生成更多的连续性样本. 文献[51]首先从单个对象的场景图像中获取知识, 然后在新的多对象场景图像中学习物体表征. 这些方法通过不同的方法分离出物体, 通过以物体为中心的表征学习机制, 学习图像中的物体表征.

上述相关工作^[33–34,42–48]可以看出, 以物体为中心的表征学习方法将图像视为多个物体的组成, 而工业生产中的异常检测场景亦可视作正常背景和可能存在的异常物体的组成, 与以物体为中心的表征学习方法存在共通性. 因此, 本文方法引入以物体为中心的表征学习方法, 对图片分别进行背景和异常前景的建模.

2 模拟工业生产环境的合成数据集

本节主要描述工业生产中异常检测面临的数据问题、构成合成数据集的方法和期望解决的问题.

2.1 工业生产中的异常检测问题

工业生产中存在大量流水线装配场景, 在自动化装配过程会出现零件掉落的情况, 因此需要及时清理以免造成更大损失. 然而现有异常检测数据集大多聚焦于工业生产中单个零件或场景的缺陷问题, 缺乏针对装配场景的异常物体检测数据集.

除此之外, 考虑到生产线上的流水化运作和高昂的停机成本等问题, 工业生产中的数据采集往往会集中在同一时间段内, 这导致数据本身(训练集)存在一定的同质化. 而实际工业场景(测试集)囊括

了光照变化、遮挡形变和动态干扰等诸多因素, 相较于采集的数据(训练集)更加开放.

因此, 本文基于工业生产中的真实电路板数据, 从9个场景共5 507张图片中抽取少量不包含异常物体的正常背景图片, 配合17个异常物体图片, 通过数据合成的方式得到存在数据同质化的训练集. 剩余真实电路板数据归为测试集.

2.2 数据集合成

图1详细展示了数据集合成的过程, 具体可以分为2个步骤: 步骤1从真实数据集中随机抽取图片; 步骤2基于步骤1中抽取的图片, 将17个异常物体和抽取的图片按照规则进行图片合成. 图1中最左侧的代表图像展示9种不同真实电路板场景: 序号为1, 2, 3, 4的场景包含常见的电子元件和敷铜板, 同时囊括了电路板场景中常见的反光问题; 序号5, 6的场景包含长宽比较大的电子元件; 序号7, 8的场景包含长宽较小的电子元件, 同时囊括了墨点等干扰问题; 序号9的场景包含多种电路板原件, 反映了真实场景的复杂性.

首先, 本文提出真实场景图片抽取算法, 如算法1所示. 算法1读取9个真实场景共5 507张图片, 依据类别信息从每个场景中随机选取6~10张不包含异常前景物体的图片.

算法1. 真实场景图片抽取算法.

输入: 9类场景中不包含异常物体的真实图片集合 *NormalImages* 和每类场景随机抽取图片数量集合 *Num*;

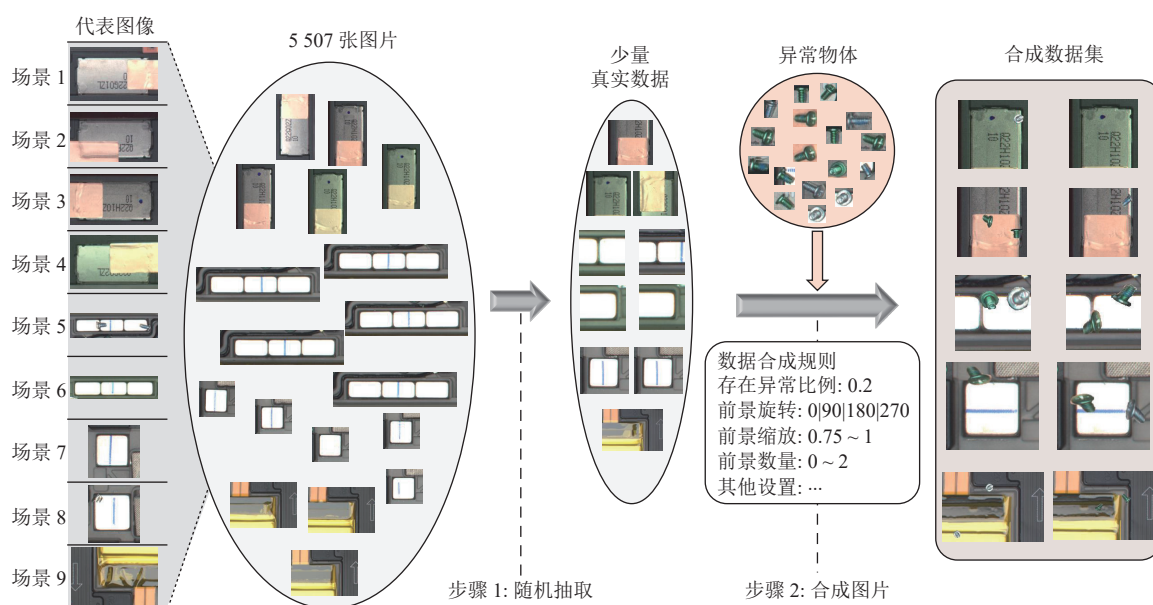


Fig. 1 Illustration of dataset synthesizing

图1 数据集合成示意图

输出: 随机抽取场景图片的集合 *SampleImages*.

Fuction *RandomSample*

- ① $\text{dict}_{\text{nimage}} \leftarrow \text{NormalImages}$;
- ② $\text{dict}_{\text{number}} \leftarrow \text{Num}$;
- ③ $\text{dict}_{\text{sample}} \leftarrow \{\}$;
- ④ for $\text{class}, \text{images} \in \text{enumerate}(\text{dict}_{\text{nimage}})$ do
- ⑤ $n = \text{dict}_{\text{number}}[\text{class}]$;
- ⑥ $\text{samples} = f_{\text{sample}}(\text{images}, n)$;
- ⑦ $\text{dict}_{\text{sample}}[\text{class}] \leftarrow \text{samples}$;
- ⑧ end for
- ⑨ $\text{SampleImages} \leftarrow \text{dict}_{\text{sample}}$;
- ⑩ return *SampleImages*.

基于算法 1 返回的结果, 结合已标注的 17 个异常前景物体, 本文提出电路板数据集合成算法, 如算法 2 所示. 算法 2 通过合成数据的方式将算法 1 中随机抽取的图片和已标注的异常前景物体随机组合, 生成训练集和验证集. 未被算法 1 抽取的剩余图片全部作为测试集图片. 具体的合成规则有 5 条.

1) 存在异常比例. 工业生产中正常样本数量通常远远大于异常样本数量. 基于此, 在构建合成数据集的过程中, 将图片中可能存在异常的比例调整为 20%, 即平均每 10 张图片里有 2 张存在异常.

2) 前景旋转. 由于真实场景中提供的标记数据数量有限, 为了更好地让模型学习异常物体的表征信息, 本文方法对异常前景物体进行了数据增强, 将 17 个异常前景样本随机旋转 $0^\circ, 90^\circ, 180^\circ, 270^\circ$, 然后进行数据合成.

3) 前景缩放. 不同场景中的异常物体存在尺度变化, 本文方法通过对异常前景物体进行尺度缩放来模拟这一变化, 将尺度缩放范围设置为 0.75~1, 随机缩放异常前景物体.

4) 前景数量. 实际场景中, 异常物体可能存在复数情况, 本文数据集考虑真实场景中异常物体的实际数量, 将前景数量设置为 0~2, 以此模拟真实环境中的电路板装配.

5) 其他设置. 本文方法将训练集和验证集设置为同等数量, 让模型更好地学习物体表征; 同时本文方法将合成数据集扩充倍数设置为 500, 以产生足量数据训练模型.

基于上述合成规则, 本文利用算法 2 合成电路板数据集, 具体如下所示.

算法 2. 电路板数据集合成算法.

输入: 包含 9 类场景的真实图片集合 *Images*, 随机抽取的电路板场景图片集合 *SampleImages*, 异常前

景物体集合 *ObjectImages*, 对应的异常前景图片掩码集合 *MaskImages*, 前景旋转集合 $\text{Rot} = \{\pi/2, \pi, 3\pi/2, 2\pi\}$, 前景缩放集合 $\text{Scal} = \{0.75, 1\}$, 前景数量上限 $\text{Obj}_{\text{max}} = 2$, 训练集验证集划分比例 $\text{line} = 0.5$, 合成数据扩充比例 $\text{expand} = 500$;

输出: 训练集 *Train*, 验证集 *Valid* 和测试集 *Test*.

函数: *RotObj* 实现对异常前景物体随机旋转; *AddObj* 将异常前景物体合成到图像中的随机位置.

Fuction *CreateDataset*

- ① $\text{dict}_{\text{sample}} \leftarrow \text{SampleImages}$;
- ② $\text{dict}_o \leftarrow \text{ObjectImages}$;
- ③ $\text{dict}_m \leftarrow \text{MaskImages}$;
- ④ $\text{Train}, \text{Valid}, \text{Test} \leftarrow \{\}$;
- ⑤ for $\text{id}, \text{image} \in \text{enumerate}(\text{dict}_{\text{sample}})$ do
- ⑥ $\text{iftrain} = \text{Random}(\text{line})$;
- ⑦ for $i \in \text{range}(\text{expand})$ do
- ⑧ $\text{objnum} = \text{Random}(0, \text{Obj}_{\text{max}})$;
- ⑨ $\text{im} = \text{dict}_{\text{sample}}[\text{id}]$;
- ⑩ $\text{im}_m = \text{Initial}(\text{image})$;
- ⑪ for $j \in \text{range}(\text{objnum})$ do
- ⑫ $\text{objid}, \text{obj} = f_{\text{sample}}(\text{dict}_o)$;
- ⑬ $\text{obj}_m = \text{dict}_m[\text{objid}]$;
- ⑭ $\text{obj}, \text{obj}_m = \text{RotObj}(\text{obj}, \text{obj}_m, \text{Rot})$;
- ⑮ $\text{im}, \text{im}_m = \text{AddObj}(\text{im}, \text{im}_m, \text{obj}, \text{obj}_m, \text{Scal})$;
- ⑯ end for
- ⑰ if $\text{iftrain} > \text{line}$ do
- ⑱ $\text{Train}[\text{id}, i] \leftarrow \{\text{im}, \text{im}_m\}$;
- ⑲ else do
- ⑳ $\text{Valid}[\text{id}, i] \leftarrow \{\text{im}, \text{im}_m\}$;
- ㉑ end if
- ㉒ end for
- ㉓ end for
- ㉔ $\text{Test} \leftarrow \text{Images} \setminus (\text{Images} \cap \text{SampleImages})$;
- ㉕ return *Train*, *Valid*, *Test*.

值得注意的是, 在现实场景中, 测试集作为开放集, 存在大量模型从未见过的场景图片, 对模型的泛化性有较高要求. 因此, 本文将训练集和验证集中的场景图片设置为互斥, 让模型在训练阶段尽可能基于输入图像本身进行重构, 从而在面对未知图像时仍能够重构出图像中的细节信息.

上述合成规则存在可拓展性, 当数据集或对应任务发生变化时, 可以通过调整合成规则来快速适应新的任务或模拟新的实际场景, 辅助更多研究应用和落地.

3 基于组成式场景建模的异常检测方法

本节首先简介 GIMOO^[33] 方法, 然后详细描述本文所提方法的算法细节, 具体分为背景建模阶段和前景建模阶段, 同时简述整体算法的代码流程和生成过程。

3.1 GMIOO 模型

GMIOO 模型作为深度生成模型, 在无监督场景下学习输入图像中多个物体的潜在表示, 解耦物体的位置、形状和外观等属性, 并实现对多个物体的重组和对整张图片的组成式建模。GMIOO 模型首先将

图像输入背景编解码器, 得到背景的隐空间表示; 然后将输入图像和背景隐空间表示输入前景编码器, 获得前景对象的隐空间表示, 并以此推断多个前景物体存在的重叠遮挡; 最后通过前景解码器重构出多个前景物体并和背景进行组合, 实现无监督下对整张图片的组成式重构。

3.2 基于背景-前景组成式建模的异常检测方法

本文受到 GMIOO 的启发, 提出了如下假设: 异常检测场景的图片由正常的背景区域和可能存在异常的前景区域组成。基于上述假设, 本文提出了一种基于背景-前景组成式建模的深度学习框架, 如图 2 所示。该框架包含背景建模阶段和前景建模阶段。

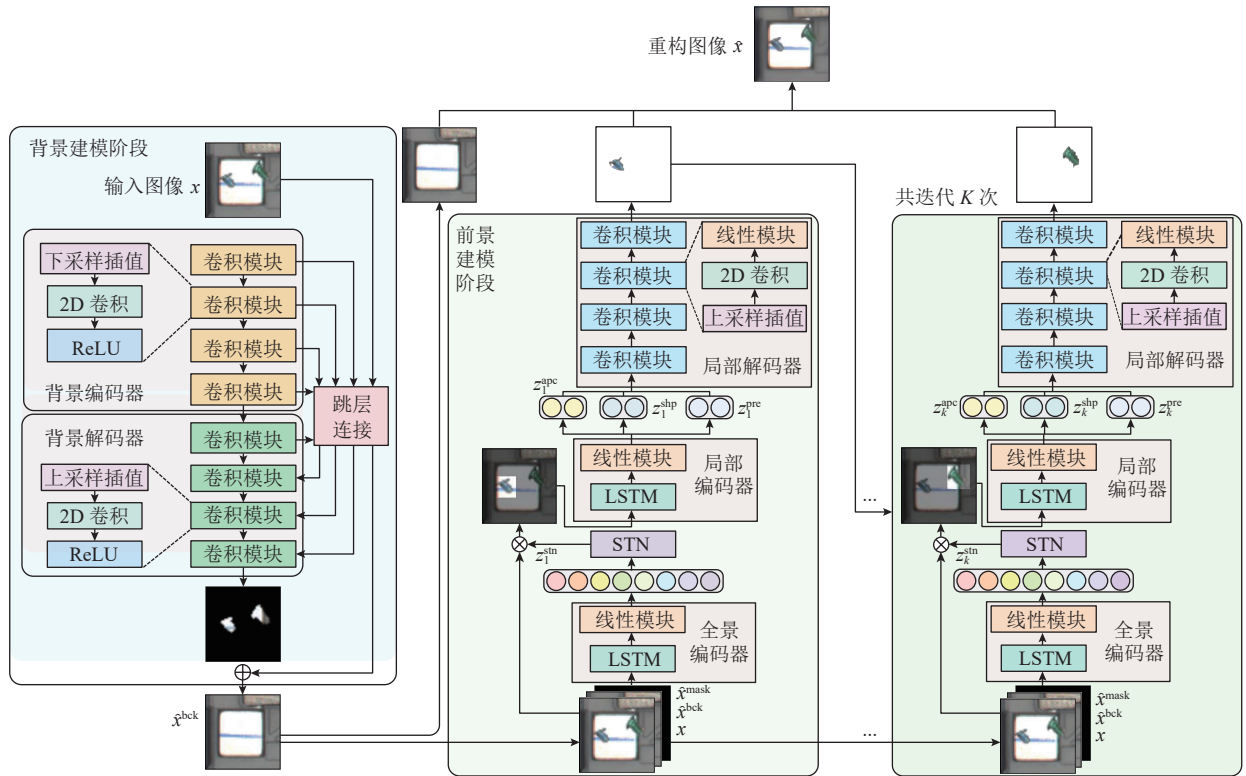


Fig. 2 Illustration of network structure

图 2 网络结构示意图

在背景建模阶段, 模型将可能的异常前景视为“噪声”, 通过多层卷积网络得到图像的低维表示, 并通过上采样插值和全连接层重构出不含“噪声”的干净背景图片。在该阶段, 模型将原图-背景对作为输入, 利用跳层连接等操作尽可能保留原图中背景的复杂纹理特征, 让模型具备重构真实背景的能力。

在前景建模阶段, 模型将重构背景和原始图像作为输入, 利用编码器将图像编码为潜在变量表示, 通过对潜在变量进行采样来学习图像中异常物体的外观、形状和存在表示, 并结合重构背景, 通过组

成式建模得到完整重构图像。该阶段模型借鉴了生成模型的思路, 在缺乏监督的情况下使用自编码器学习异常物体的表示, 这让模型能够更有效地应对真实场景中多变的异常物体, 具备良好的异常检测能力。

3.3 背景建模阶段

在该阶段, 模型由背景编码器(background encoder, BGE)、背景解码器(background decoder, BGD)和跳层连接构成。输入图片 x 经过背景编解码器, 得到不包含异常前景的干净背景 \hat{x}^{back} 。

3.3.1 背景编码器

为了对输入图像 x 进行编码, 背景编码器利用多层卷积对样本进行变换, 其中包括 2D 卷积层(convolution layer, Conv2d)、下采样插值(down interpolate)和非线性激活层 ReLU(rectified layer units)组成, 具体如下:

$$z^{\text{bck}} = f_{\text{enc}}^{\text{bck}}(x), \quad (1)$$

其中 x 为输入样本, z^{bck} 为经过背景编码器得到背景潜在表示, $f_{\text{enc}}^{\text{bck}}(\cdot)$ 为采用多层卷积的背景编码器.

3.3.2 背景解码器

得到背景潜在表示 z^{bck} 后, 通过背景解码器可以重构出不包含异常物体的背景图片 \hat{x}^{bck} . 背景解码器采用多层卷积、上采样插值、ReLU 和跳层连接, 将背景隐变量 z^{bck} 恢复到输入图像尺寸, 输出重构图像 \tilde{x} 和局部重构区域 \tilde{x}^{mask} , 具体如下所示:

$$\tilde{x}, \tilde{x}^{\text{mask}} = f_{\text{dec}}^{\text{bck}}(z^{\text{bck}}), \quad (2)$$

其中 $f_{\text{dec}}^{\text{bck}}(\cdot)$ 为包含卷积层、上采样插值、ReLU 和跳层连接的背景解码器. 值得注意的是, 由于模型面向真实场景任务, 单纯对背景隐变量解码难以重构出精确的背景图像. 受到 U-net^[52] 的启发, 本文在背景编码器和背景解码器之间增加了跳层连接, 来传递图像不同尺度下的纹理特征, 让模型能够应对真实场景中的背景建模任务.

获得重构表示 \tilde{x} 和重构区域 \tilde{x}^{mask} 后, 通过线性计算可得到不包含异常前景的干净重构背景 \hat{x}^{bck} , 具体为:

$$\hat{x}^{\text{bck}} = \tilde{x} \times \tilde{x}^{\text{mask}} + x \times (1 - \tilde{x}^{\text{mask}}). \quad (3)$$

3.3.3 背景建模损失函数

模型通过重构方法得到干净背景 \hat{x}^{bck} , 而真实场景中图像普遍存在遮挡、噪声、反光和形变等动态干扰, 这会导致模型重构的区域 \tilde{x}^{mask} 逐渐扩大, 而较大的重构区域会导致模型在前景建模阶段更加难以学习异常前景表示. 基于此, 本文在背景重构的基础上, 增加了对重构区域的约束, 让模型在有限区域内完成重构. 具体的目标损失函数为:

$$L_{\text{BGD}} = L_{\text{bck}} + \alpha_{\text{bck}} L_{\text{area}}, \quad (4)$$

$$L_{\text{bck}} = \|x - \hat{x}^{\text{bck}}\|_2^2, \quad (5)$$

$$L_{\text{area}} = \arg \min (S^{\text{bck}}), \quad (6)$$

其中 S^{bck} 为重构区域 \tilde{x}^{mask} 的激活区域, α_{bck} 为给定的超参数. 可以看出, L_{area} 通过约束重构区域 \tilde{x}^{mask} 的激活面积, 让模型集中重构可能存在异常背景区域. L_{bck} 通过计算输入图像 x 和重构背景 \hat{x}^{bck} 的均方误差实现对整体图像的重构约束. L_{area} 和 L_{bck} 的组合让模型在拥有较好重构效果的同时, 对存在干扰的真实

场景有良好的泛化性能.

3.4 前景建模阶段

在前景建模阶段, 模型借鉴 GMIOO 模型中的前景编解码器, 由 3 部分组成: 全景图像编码器(full image encoder, FIE)、空间变换网络(spatial transformer network, STN)和局部自编码器(local autoencoder, LAE). 输入图像 x 和重构背景 \hat{x}^{bck} 首先经过全景图像编解码器和空间变换网络, 得到异常物体的位置信息, 然后输入局部自编码器得到异常物体的外观、形状和存在表示. 上述操作将迭代 K 次(K 为超参数, 代表图像中可能存在异常物体的最大数目), 每次得到的异常物体表示将在下一次迭代中输入全景图像编码器. 在经过 K 次迭代后, 模型通过组成式建模的方式得到最终的重构图像 \hat{x} .

3.4.1 全景图像编码器

全景图像编码器由多层卷积网络和长短期记忆网络组成. 首先将输入图像 x 、重构背景 \hat{x}^{bck} 、初始掩码 \tilde{x}^{mask} 和先前得到的异常物体表示进行 concatenate 操作, 然后输入 FIE 得到第 k 次迭代的异常前景空间变换表示 z_k^{stn} , 具体如下:

$$z_k^{\text{stn}} = f_{\text{FIE}}(x, \hat{x}^{\text{bck}}, \tilde{x}^{\text{mask}}, \hat{x}_{1:k-1}^{\text{obj}}). \quad (7)$$

当 $k=1$ 时, FIE 的输入仅包括图像 x 、重构背景 \hat{x}^{bck} 和初始掩码 \tilde{x}^{mask} .

3.4.2 空间变换网络

得到 z_k^{stn} 后, 输入 STN 网络可以得到可能存在的异常前景物体的空间变换和对应的局部图像, 具体为:

$$x_k^{\text{crop}} = f_{\text{STN}}(x, z_k^{\text{stn}}), \quad (8)$$

其中 k 表示异常物体的序号, x_k^{crop} 表示经过空间变换的局部图像.

3.4.3 局部自编码器

局部自编码器包含局部编码器(local object encoder, LOE)和局部解码器(local object decoder, LOD)这 2 部分. LOE 由多层卷积和长短期记忆网络组成, 经过 STN 的局部图像 x_k^{crop} 经过编码器后, 得到异常物体外观、形状和存在的隐空间表示:

$$z_k^{\text{apc}}, z_k^{\text{shp}}, z_k^{\text{pre}} = f_{\text{LOE}}(x_k^{\text{crop}}). \quad (9)$$

LOD 由多层反卷积网络和长短期记忆网络组成. 将异常物体的隐空间表示输入解码器, 可以得到异常物体的外观、形状和存在的重构表示:

$$\hat{x}_k^{\text{apc}}, \hat{x}_k^{\text{shp}}, \hat{x}_k^{\text{pre}} = f_{\text{LOD}}(z_k^{\text{apc}}, z_k^{\text{shp}}, z_k^{\text{pre}}, z_k^{\text{stn}}), \quad (10)$$

其中 $z_{1:k}^{\text{stn}}$ 为 z_k^{stn} 的逆变换. 经过计算可以得到序号为 k 的重构异常物体:

$$\hat{x}_k^{\text{obj}} = \hat{x}_k^{\text{apc}} \times \hat{x}_k^{\text{shp}} \times \hat{x}_k^{\text{pre}}. \quad (11)$$

上述迭代过程持续 K 次后,可以得到所有异常物体的重构表示,结合背景建模阶段的重构背景,可以得到最终的重构图像 \hat{x} , 具体为:

$$\hat{x} = \hat{x}^{\text{bck}} \times \left(1 - \sum_{k=1}^K \hat{x}_k^{\text{pre}} \hat{x}_k^{\text{shp}} \right) + \sum_{k=1}^K \hat{x}_k^{\text{obj}}. \quad (12)$$

3.4.4 前景建模损失函数

通常来说,基于生成方法的损失函数只计算整张图片的重构损失,而本文模型采取背景-前景组成式建模的方式,单独对背景进行建模,在前景建模阶段已经得到较好的重构背景表示.为了让模型更好地学习前景物体的表示,本文在整体图像损失的基础上增加了针对异常物体的局部损失函数,具体为:

$$L_{\text{recon}} = L_{\text{image}} + \alpha_{\text{local}} L_{\text{local}}, \quad (13)$$

$$L_{\text{image}} = \|x, \hat{x}\|_2^2, \quad (14)$$

$$L_{\text{local}} = \sum_{k=1}^K \|\hat{x}_{1:k}^{\text{crop}}, \hat{x}_{1:k}^{\text{obj}}\|_2^2, \quad (15)$$

其中 α_{local} 为给定的超参数, L_{recon} 由 L_{image} 和 L_{local} 组成. L_{local} 直接对异常物体进行约束,让模型对可能存在的异常物体重构更加准确. L_{image} 聚焦整体图像的重构,保证多个异常物体和重构背景组合而成的完整图像更加贴近输入图像.改进后的损失函数 L_{recon} 使得模型前景物体和背景图像均有较好的重构效果.

3.5 算法流程

本节详细展示了本文模型的算法流程,具体分为背景建模和前景建模 2 个阶段.算法 3 展示了 2 个阶段的伪代码,并详细展示了前景建模阶段中针对前景物体的 K 轮迭代过程.

算法 3. 基于背景-前景组成式建模的异常检测算法.

输入: 输入图像 x ;

输出: 重构图像 \hat{x} 和异常检测结果 $x_{1:K}^{\text{pre}}$.

- ① for $i \in \text{range}(\text{epoch})$ do /*背景建模阶段训练,
 epoch 为背景建模阶段的训练轮数*/
- ② $z^{\text{bck}} = f_{\text{BGE}}(x)$;
- ③ $\hat{x}^{\text{bck}} = f_{\text{BGD}}(z^{\text{bck}})$;
- ④ end for
- ⑤ for $i \in \text{range}(\text{epoch})$ do /*前景建模阶段训练,
 epoch 为前景建模阶段的训练轮数*/
- ⑥ for $k \in \text{range}(K)$ do /* K 次迭代过程*/
- ⑦ if $k == 0$
- ⑧ $z^{\text{image}} = f_{\text{FIE}}(x, \hat{x}^{\text{bck}}, \hat{x}^{\text{mask}}, \hat{x}^{\text{obj}})$;

- ⑨ else
- ⑩ $z^{\text{image}} = f_{\text{FIE}}(x, \hat{x}^{\text{bck}}, \hat{x}^{\text{mask}})$;
- ⑪ end if
- ⑫ $x_k^{\text{crop}}, z_k^{\text{stn}} = f_{\text{STN}}(x, z^{\text{image}})$;
- ⑬ $z_k^{\text{apc}}, z_k^{\text{shp}}, z_k^{\text{pre}} = f_{\text{LOE}}(x_k^{\text{crop}})$;
- ⑭ $x_k^{\text{apc}}, x_k^{\text{shp}}, x_k^{\text{pre}} = f_{\text{LOD}}(z_k^{\text{apc}}, z_k^{\text{shp}}, z_k^{\text{pre}})$;
- ⑮ end for
- ⑯ $\hat{x} = \hat{x}^{\text{bck}} \times (1 - \text{sum}(x_k^{\text{shp}})) + \text{sum}(x_k^{\text{apc}} \times x_k^{\text{shp}} \times x_k^{\text{pre}})$;
- ⑰ end for
- ⑱ return $\hat{x}, x_{1:K}^{\text{pre}}$.

4 实验与评估

4.1 实验环境和数据集

1) 实验环境. 本实验基于 Ubuntu 环境, 使用 Python 语言, 在 Linux 环境下进行模型搭建. 平台主要硬件参数为: CPU 使用 Intel Xeon E5-2603, 16 GB 内存, GPU 使用 NVIDIA GTX 1080 Ti, 11 GB 显存.

2) 数据集. 本文分别在公开数据集 FOD-A 和自建数据集 FO-CB 上进行实验. 公开数据集 FOD-A 由机场跑道常见的异物碎片图片组成, 共计 31 个类别, 超过 30 000 张标注图片, 囊括了不同的光照和天气类别. 自建数据集 FO-CB 由工业生产中的真实电路板装配图片组成, 共计 9 个场景 5 507 张图片, 囊括了电路板装配厂景中各种电子元件, 如表 1 所示. 其中训练集和验证集(共 186 张)由少量真实场景和异常物体(螺丝钉)通过简单数据合成方式得到, 测试集由剩余电路板装配图片(共 5 321 张)构成.

Table 1 FO-CB Dataset Information

表 1 FO-CB 数据集信息

| 数据集 | 场景数量 | 真实图像 | (合成) 图像总数 | 异常数据比例 |
|-----|------|-------|-----------|--------|
| 训练集 | 9 | 92 | 46 000 | 0.20 |
| 验证集 | 9 | 94 | 47 000 | 0.20 |
| 测试集 | 9 | 5 321 | 5 321 | 0.02 |

4.2 评估方法

异常检测通常采用深度学习领域通用的混淆矩阵来衡量模型的性能. 该矩阵描述样本的实际类别并预测类别的混合, 分为真阳性(true positive, TP)、假阳性(false positive, FP)、假阴性(false negative, FN)和真阴性(true negative, TN). 具体如表 2 所示.

本文采用实际生产中更为关注的异常元件漏检率(miss rate, MR)和异常原件误检率(incorrect rate, IR).

1) 漏检率. 表示检测为正常样本的异常样本在

Table 2 Confusion Matrix

表 2 混淆矩阵

| 实际类别 | 检测正常 | 检测异常 |
|------|------|------|
| 正常类 | TP | FN |
| 异常类 | FP | TN |

实际异常样本中所占的比例. 计算方法为:

$$MR = \frac{FP}{FP + TN} \times 100\%. \quad (16)$$

2) 误检率. 表示错误归类的样本在总体样本中的比例. 计算方法为:

$$IR = \frac{FP + FN}{TP + FP + TN + FN} \times 100\%. \quad (17)$$

4.3 实验参数

模型训练阶段分为背景建模阶段和前景建模阶段, 各阶段对应的训练参数如表 3 所示. 其中迭代轮数表示模型在不同阶段训练的轮数; 批数量表示每一次迭代输入网络的图片数量; K 表示图像中可能存在的异常物体最大数量的超参数; α_{bck} 和 α_{local} 分别为 L_{area} 和 L_{local} 的超参数.

Table 3 Training Parameters

表 3 训练参数

| 训练参数 | 背景建模阶段 | 前景建模阶段 |
|------------------|--------|--------|
| 迭代轮数 | 10 000 | 12 000 |
| 批数量 | 128 | 128 |
| K | 2 | 2 |
| α_{bck} | 5 | |
| α_{local} | | 5 |

表 4 展示了模型在背景建模阶段的网络细节. 背景建模阶段包括 BGE 和 BGD 模块.

Table 4 Network in Background Modeling Stage

表 4 背景建模阶段网络

| 模块名称 | 结构细节 |
|------|----------------------------------|
| BGE | 2D 卷积+向下插值+激活函数 (4 层) |
| BGD | 2D 卷积+向上插值+激活函数 (4 层) 4 层跳层连接 |

表 5 展示了模型前景建模阶段的网络细节. 前景建模阶段包括 FIE, STN, LAE 模块, 其中 STN 直接调用 torch 的仿射变换, 故而表 5 仅展示 FIE 和 LAE 这 2 个模块.

4.4 对比方法

为了验证本文方法在不同阶段的有效性, 本文在背景和前景建模阶段分别与不同的方法进行对比. 在背景建模阶段, 本文方法选取 U-net^[52] 和 GMIOO^[26]

Table 5 Foreground Modeling Stage Network

表 5 前景建模阶段网络

| 模块名称 | 结构细节 |
|-----------|-------------------------------------|
| FIE | 长短期记忆网络+线性层+层归一化 |
| LAE (LOE) | 长短期记忆网络+线性层+层归一化 |
| LAE (LOD) | 2D 卷积+向上插值 (4 层) 线性模块+层归一化 (4 层) |

中的背景自编码器 (GMIOO-AE) 方法进行比较. 在前景建模阶段, 本文方法选取 2 种基于深度学习的方法进行比较: 1) DFKDE (deep feature kernel density estimation)^[53]; 2) PaDiM (patch distribution model)^[14].

1) U-net. U-net 是一种用于图像分割的卷积神经网络模型. 该模型为编码-解码结构, 在编码阶段逐层提取图像的特征, 在解码阶段通过对称的结构来还原图像中的细节信息. U-net 模型在训练样本较少的情况下同样能够准确进行图像分割.

2) GMIOO-AE. GMIOO 是一种以物体为中心的组成式建模架构. 该架构包含背景建模和前景建模. GMIOO-AE 表示该架构中进行背景建模的网络结构, 由编码器和解码器组成. GMIOO-AE 能够在无监督情况下完成对图像的背景重建.

3) DFKDE. DFKDE 是一种快速异常分类算法, 分为深度特征提取阶段以及由主成分分析 (principal component analysis, PCA) 和高斯核密度估计组成的异常分类阶段. 其中特征提取阶段由在 ImageNet 上预训练的 ResNet-18^[54] 网络组成; 异常分类阶段先将特征压缩为 16 个主成分, 然后通过高斯核密度来评估图像是否存在异常.

4) PaDiM. PaDiM 是一种基于深度神经网络的异常检测方法. 该方法利用预训练的卷积神经网络进行 patch 嵌入, 通过多元高斯分布得到图像的概率表示. PaDiM 适用于单类检测问题, 并利用不同层次语义的相关性更好地定位图像中的异常.

本文将在实验部分与上述方法分别比较局部背景的重构效果和异常前景的检测结果, 以此验证本文方法在不同阶段的实验效果.

4.5 实验结果

4.5.1 背景阶段重构结果

图 3 展示了背景建模阶段中模型在 FO-CB 测试集上的重构效果. 图 3 中输入图像表示原始图像, 局部重构表示模型输出的重构结果. 通过重构结果可以看出, 本文模型能够在每个场景仅使用少量的真实图像作为合成数据训练的情况下, 准确建模真实场景中的图像背景. 这些结果同样证明了本文模型

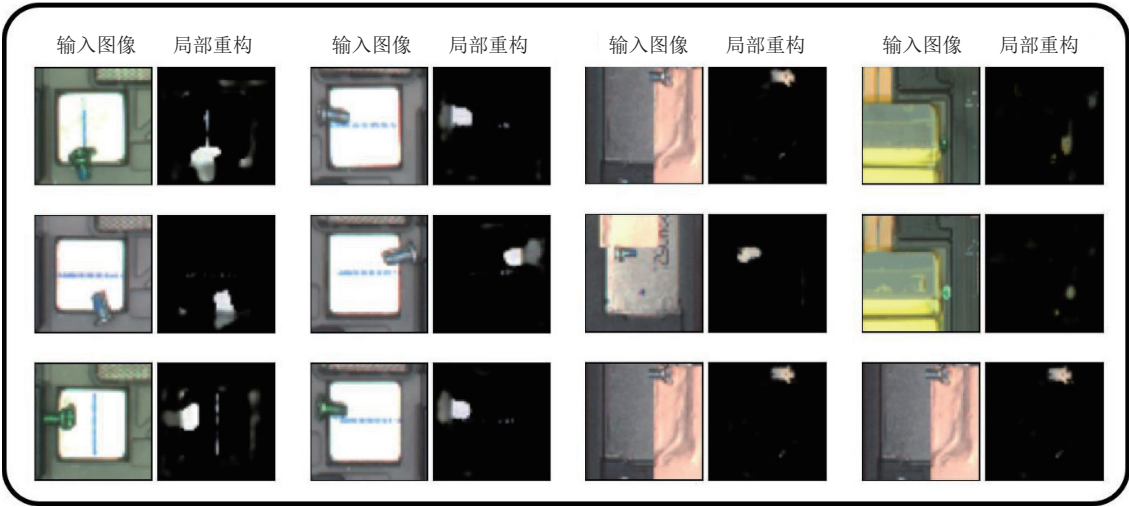


Fig. 3 Reconstruction results in the background modeling stage
图3 背景建模阶段重构结果

在背景建模阶段具备良好的鲁棒性和泛化性,对可能存在的未见过的图片仍然能够准确重构出不包含异常物体的背景图像.

鉴于现实世界场景中普遍存在照明和遮挡等干扰,模型在训练过程中存在通过扩大局部重构区域来应对潜在的异常物体的现象.这会导致模型泛化性下降,面对从未见过的图像难以准确重构出干净背景.为此,本文设计了 L_{area} 约束来调节局部重建区域的面积,迫使模型在尽可能小的区域内完成针对异常物体的重构.这一改进不仅提升了模型的泛化性能,同时确保模型在面对从未见过的图像时,能够有效还原图像中可能存在的各种干扰,让重构的干净背景图像更加贴近原有图像.表6表明在背景建模阶段添加 L_{area} 后,模型在验证集上的重构损失从13.97降低至5.66.这说明 L_{area} 能够更好地辅助模型实现对于从未见过场景的背景重构.

Table 6 Experiment Analysis of L_{area}
表6 L_{area} 实验分析

| 模块 | L_{area} | 均方误差 |
|-----|------------|-------|
| BGM | 无 | 13.97 |
| | 有 | 5.66 |

为了验证本文方法的有效性,本文方法与 U-net 和 GMIOO-AE 这2种方法进行比较.图4展示了3种方法的局部重构效果,可以看出 U-net 方法倾向于重构整张图像,局部重构较为模糊;GMIOO-AE 方法由于缺乏损失函数的约束,重构区域相较于其他2种方法更加零散,同时重构效果较差;本文方法的局部重构图像较为清晰,同时重构区域包含异常物体(螺

丝)的完整形状.由此可见,本文方法在背景建模阶段中局部重构和约束重构区域的设计,使得模型在背景重构阶段取得了较好的效果.

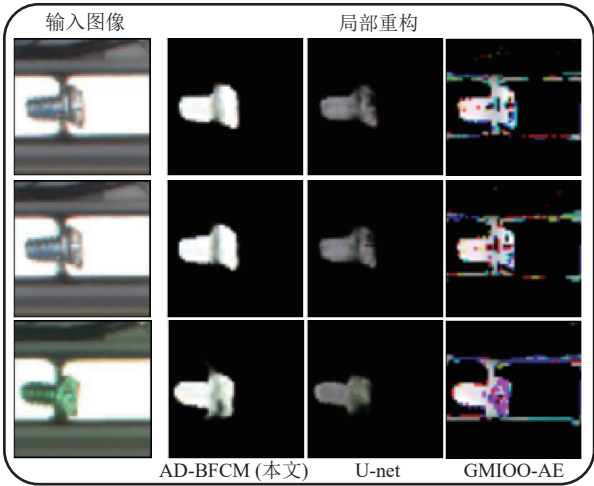
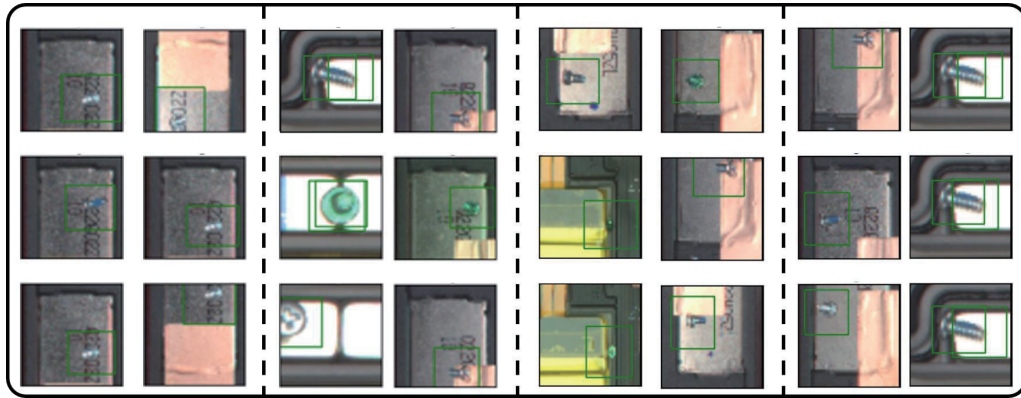


Fig. 4 Comparison of local background reconstruction results
图4 局部背景重构结果对比

4.5.2 前景阶段重构结果

图5展示了模型在 FO-CB 测试集上的异常检测结果,图5中框表示模型检测到存在异常物体的区域(有框表示结果正确).可以看出,模型在面对测试集中从未见过的真实场景,能够准确检测出图像是否存在异常并对异常物体进行定位.当螺丝钉位于图像边缘或者复杂环境(数字区域或颜色相近的区域)时,模型仍然能够给出正确的定位.这也说明模型具有较好的泛化性,能够在真实场景中完成异常物体的检测任务.

表7显示了 AD-BFCM 模型在9个不同场景下的异常检测结果.可以看出,9个场景中的漏检率为



注：图中矩形框表示检测正确。

Fig. 5 Anomaly detection result in the foreground modeling phase

图5 前景建模阶段异常检测结果

0%, 整体误检率为 4.55%。这说明模型在没有监督信息的情况下识别出了 9 个场景中所有的异常物体。虽然模型整体误检率在 4.86%, 但对于工业生产环境中的异常检测任务, 异常样本的漏检相对于正常样本的误检显得更为重要。在保证漏检率为 0% 的情况下, 4.86% 的误检率是可以接受的。此外, 该模型在其中 6 个场景(具体序号为: 3, 4, 5, 6, 7, 8)中的误检率接近或低于整体误检率, 说明模型能够在大多数场景中保持较低的误检率, 仅在少部分场景具备较高误检率。实验结果表明, 该模型在不同的电路板装配场景中, 能够检测出全部的异常物体, 并在大多数场景中保持相对较低的误检率。

Table 7 Experiment Results of Anomaly Detection

| 表 7 异常检测实验结果 | | | | | | | | | | % | |
|--------------|------|------|------|------|------|------|------|------|------|------|--|
| 评估指标 | 场景 1 | 场景 2 | 场景 3 | 场景 4 | 场景 5 | 场景 6 | 场景 7 | 场景 8 | 场景 9 | 合计 | |
| 漏检率 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 误检率 | 9.6 | 8.5 | 4.7 | 2.5 | 1.5 | 3.2 | 3.1 | 0 | 6.9 | 4.86 | |

为了验证本文方法在异常检测的效果, 与 DFKDE 和 PaDiM 这 2 种方法分别在公开数据集 FOD-A 和自建数据集 FO-CB 上进行比较。由于大多数异常检测方法在训练过程中需要较为充足的正常样本以学习正常样本和异常样本的差异, 本文在训练 DFKDE 和 PaDiM 的过程中使用了 80% 的真实数据(5 507 张真实场景图片), 并将剩余 20% 数据作为测试集。

图 6 展示了 3 种方法在 FO-CB 数据集的异常检测结果, 其中 DFKDE 和 PaDiM 展示图像级异常检测结果, 本文方法展示图像级异常结果和定位结果。由图 6 可以看出 DFKDE 和 PaDiM 方法对于图像中的异常物体都存在漏检情况, 同时异常检测阈值接近

50%。本文方法能够检测出所有包含异常物体的真实图片, 同时实现对异常物体较为准确的定位。

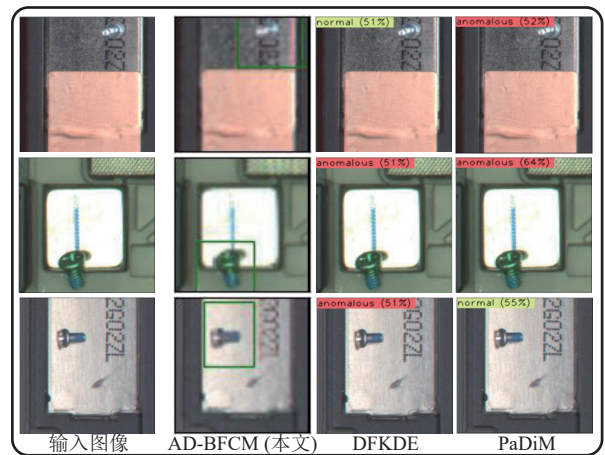


Fig. 6 Comparison of anomaly detection results

图6 异常检测结果对比

表 8 展示了 AD-BFCM 与 DFKDE 和 PaDiM 方法在数据集 FOD-A 和 FO-CB 上的异常检测性能。实验结果表明, AD-BFCM 模型不仅在自建数据集 FO-CB 漏检率达到 0%, 在公开数据集 FOD-A 上漏检率低至 2.89%, 优于其他 2 种对比方法。

Table 8 Comparison of MR in FOD-A and FO-CB Datasets

表 8 FOD-A, FO-CB 数据集漏检率对比

| 模型 | FO-CB 数据集 | FOD-A 数据集 |
|---------|-----------|-----------|
| PaDiM | 10.53 | 12.24 |
| DFKDE | 2.63 | 3.10 |
| AD-BFCM | 0.00 | 2.89 |

4.5.3 消融实验

为了验证本文方法中各个模块的有效性, 本节从 AD-BFCM 模型中删除了部分功能, 得到 5 种不同模

型, 并进行消融实验. 模型 1~5 的具体设置如下所示:

1) 模型 1. 未对图像的建模阶段进行区分, 将模型的背景阶段模块和前景阶段模块合并训练, 未使用图像-干净背景对进行训练.

2) 模型 2. 对图像的建模阶段进行划分, 分别训练背景建模阶段和前景建模阶段, 未使用图像-干净背景对进行训练背景建模模块.

3) 模型 3. 在模型 2 的基础上, 使用图像-干净背景对训练背景建模模块, 未使用 L_{area} 和 L_{local} 对模型约束.

4) 模型 4. 在模型 3 的基础上增加 L_{area} 对背景建模进行约束, 未使用 L_{local} 对前景建模进行约束.

5) 模型 5. 代表最终模型, 同时包含 L_{area} 和 L_{local} , 对背景和前景同时进行约束.

实验结果如表 9 所示. 从结果可以看出, 模型 2 相较于模型 1 漏检率和误检率分别下降 3.71 个百分点和 12.38 个百分点, 说明分阶段图像建模能够小幅提升模型性能. 模型 3 在背景建模阶段增加图像-背景对进行训练, 相较于模型 2 性能获得明显提升, 漏检率和误检率分别下降 8.74 个百分点和 21.07 个百分点. 这说明真实场景中图像背景往往较为复杂, 需要通过额外信息辅助模型重构背景. 模型 3 的误检率达到 2.17%, 优于其余模型, 这是测试集大多为正常样本导致的, 其漏检率高达 9.09% 也验证了这一点. 同时在工业生产中, 相较于正常样本的误检, 如何降低异常样本的漏检是更为重要的问题. 因此模型 4 增加了 L_{area} 来对背景重构区域进行额外约束, 使得漏检率再度下降 6.37 个百分点, 但同时误检率上升 11.33 个百分点, 这说明背景建模的额外约束提升了模型对异常物体检测的性能, 但降低了模型的泛化性能, 导致误检率上升. 最终模型 5 在模型 4 的基础上增加 L_{local} , 进一步强化模型对于异常物体表示的学习, 让模型的漏检率下降 8.64 个百分点, 同时漏检率达到 0%, 让最终模型能够在较低误检的前提下, 检测

Table 9 Ablation Experiment

表 9 消融实验

| AD-BFCM 模型 | 阶段建模 | 干净背景 | L_{area} | L_{local} | 漏检率/% | 误检率/% |
|------------|------|------|------------|-------------|-------|-------|
| 模型 1 | - | - | - | - | 21.54 | 35.62 |
| 模型 2 | + | - | - | - | 17.83 | 23.24 |
| 模型 3 | + | + | - | - | 9.09 | 2.17 |
| 模型 4 | + | + | + | - | 2.72 | 13.50 |
| 模型 5 | + | + | + | + | 0 | 4.86 |

注: “+”代表使用; “-”代表不使用.

出全部异常物体.

本节的实验结果表明, AD-BFCM 中的各个模块对模型都有正面的性能提升. 其中分阶段场景建模和图像-背景对的引入让模型能够重构真实场景图片. L_{local} 和 L_{area} 能够帮助模型更好地学习异常物体的表示, 让模型获得更好的异常检测效果.

4.5.4 超参数实验

本节实验对背景建模阶段中 L_{area} 和前景建模阶段中 L_{local} 的权重设置不同取值并进行实验以验证参数对模型的影响, 如表 10 和表 11 所示. 经过测试后可以发现, L_{area} 和 L_{local} 参数的比值为 5 : 1 时, 模型效果达到最好, 这说明模型 L_{local} 参数更加敏感. 在 L_{local} 参数取值较小时, 模型误检率会上升; 在 L_{area} 参数取值较小时, 模型漏检率会上升. 由此可以看出, L_{area} 对漏检率有较大影响, 而 L_{local} 对误检率有较大影响, 这说明背景重构性能对漏检率有较大影响, 而前景重构性能对误检率有较大影响.

Table 10 Hyperparameter Experiment Results in Background Modeling Stage

表 10 背景建模阶段的超参数实验结果

| 模型 | L_{area} 权重 | L_{local} 权重 | 漏检率/% | 误检率/% |
|---------|---------------|----------------|-------|-------|
| AD-BFCM | 0.2 | 1 | 5.83 | 6.47 |
| AD-BFCM | 1.0 | 1 | 2.72 | 5.63 |
| AD-BFCM | 5.0 | 1 | 0 | 4.86 |

Table 11 Hyperparameter Experiment Results in Foreground Modeling Stage

表 11 前景建模阶段的超参数实验结果

| 模型 | L_{area} 权重 | L_{local} 权重 | 漏检率/% | 误检率/% |
|---------|---------------|----------------|-------|-------|
| AD-BFCM | 5 | 5.0 | 1.81 | 5.09 |
| AD-BFCM | 5 | 0.2 | 0 | 7.28 |
| AD-BFCM | 5 | 1 | 0 | 4.86 |

4.5.5 数据扩充对实验效果的影响

本节对合成规则进行实验以验证不同规则对模型性能的影响. 值得注意的是, 合成规则中的前景比例和前景缩放是基于数据集驱动的, 故而本节仅探讨存在异常比例、前景旋转和背景旋转对模型性能的影响.

表 12 展示了数据集规则中前景旋转和背景旋转对于模型性能的影响. 实验结果表明, 当添加前景旋转后, 模型漏检率和误检率均出现 0.9 个百分点以上的下降, 说明前景旋转让模型更好地学习到了异常物体的表示, 同时增强了泛化性能. 而当添加背景旋

转后,模型漏检率和误检率出现大幅上升,这说模型难以学习到剧烈变化下背景的表达。

Table 12 Impact of Data Augmentation on Model Based on Rotation Rules

表 12 基于旋转规则的数据扩充对模型的影响

| 前景旋转 | 背景旋转 | 漏检率/% | 漏检率/% |
|------|------|-------|-------|
| - | - | 0.90 | 6.80 |
| + | - | 0 | 4.86 |
| + | + | 4.54 | 8.79 |

注:“+”表示使用该项数据扩充,“-”表示不使用该项数据扩充。

表 13 展示了数据集规则中存在异常比例对于模型性能的影响。通过实验可以看出,当存在异常比例较高时,模型的误检率相对较高,而漏检率却会下降。这说明更多的异常前景会让模型学到更好的表示,但同时也会增加模型的误检。故在真实场景中应用合成数据集时,需对存在异常比例进行额外调整。

Table 13 Impact of Data Augmentation on Model Based on Abnormal Proportion Rules

表 13 基于存在异常比例规则的数据扩充对模型的影响

| 存在异常比例 | 漏检率/% | 误检率/% |
|--------|-------|-------|
| 0.1 | 0.9 | 4.53 |
| 0.2 | 0 | 4.86 |
| 0.3 | 0 | 5.03 |
| 0.4 | 0 | 5.53 |
| 0.5 | 0 | 6.13 |

综合上述,AD-BFCM 在真实场景中有着良好的性能表现:异常检测实验体现了模型的有效性;消融实验表明模型的各个模块对于模型性能有正向提升;参数敏感实验表明适合的超参数能辅助模型更好地进行异常检测;对数据扩充的实验表明模型在面对不同场景时,通过分析真实数据修改合成规则参数可以辅助模型更好地区分异常样本和正常样本的表征。

5 结 论

基于深度学习的异常检测是当前工业界着重关注的应用研究之一。工业生产环境中的异常检测问题,往往存在数据标注少、场景多样等问题。然而,当前的诸多研究很少有涉及数据标注、多场景迁移等问题的讨论,使得检测效果在实际应用场景下往往表现不尽如人意。基于此,本文提出了基于背景-前景组成式建模的异常检测模型,重点研究多种电路板

装配场景中的异常检测问题。本文将电路板场景的背景和前景分别进行建模:首先采用类似 U-net 结构的编解码器,配合约束局部区域面积的损失函数完成对背景的建模;然后采用基于生成的组成式建模方法,配合局部前景的重构损失函数,通过多次迭代完成对前景的建模;最后模型通过前景建模过程中的存在表示来完成异常检测。

同时,本文模拟工业生产环境中标注数量少、标注不及时的情况,借助现实装配环境构建了一个包含 9 个场景的电路板数据集,并提出一种仅需少量标注的数据合成方法。本文模型在合成数据集上进行训练,并在电路板数据集上进行测试,实验结果表明本文方法能够检测出电路板数据集中的异常物体,具备在工业生产中应用的可行性。今后,本文将增加更多不同的应用场景,如缺陷检测场景、安全监控场景等,将背景和前景异常物体进一步解耦,构建一个能够动态适应各种复杂应用场景的异常物体检测方法。

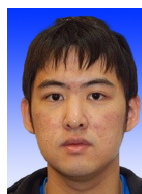
作者贡献声明:傅冰飞提出了算法思路、完成主体实验并撰写论文;陈同林负责修改论文;许枫负责提供数据集;朱麟、李斌和薛向阳提出指导意见并参与修改论文。

参 考 文 献

- [1] Han Dongming, Guo Fangzhou, Pan Jiacheng, et al. Visual analysis for anomaly detection in time-series: A survey[J]. *Journal of Computer Research and Development*, 2018, 55(9): 1843-1852 (in Chinese)
(韩东明,郭方舟,潘嘉诚,等.面向时序数据异常检测的可视分析综述[J].*计算机研究与发展*,2018,55(9):1843-1852)
- [2] Zhou Xiaohui, Wang Yijie, Xu Hongzuo, et al. Fusion learning based unsupervised anomaly detection for multi-dimensional time series[J]. *Journal of Computer Research and Development*, 2023, 60(3): 496-508 (in Chinese)
(周小晖,王意洁,徐鸿祚,等.基于融合学习的无监督多维时间序列异常检测[J].*计算机研究与发展*,2023,60(3):496-508)
- [3] Yang Yahui, Du Keming. Identification of anomalous traffic clusters for network-wide anomaly analysis[J]. *Journal of Computer Research and Development*, 2009, 46(11): 1847-1853 (in Chinese)
(杨雅辉,杜克明.全网异常流量簇的检测与确定机制[J].*计算机研究与发展*,2009,46(11):1847-1853)
- [4] Li Xiaocan, Xie Kun, Zhang Dafang, et al. Survey of network anomaly detection based on low-rank decomposition[J]. *Journal of Computer Research and Development*, 2022, 59(7): 1589-1609 (in Chinese)
(李晓灿,谢鲲,张大方,等.基于低秩分解的网络异常检测综述[J].

- 计算机研究与发展, 2022, 59(7): 1589–1609)
- [5] Noroozi M, Shah A. Towards optimal foreign object debris detection in an airport environment[J]. *Expert Systems with Applications*, 2023, 213(Part A): 118829
 - [6] Jing Ying, Zheng Hong, Zheng Wentao, et al. A pixel-wise foreign object debris detection method based on multi-scale feature inpainting[J]. *Aerospace*, 2022, 9(9): 480
 - [7] Yang Fan, Xiao Bin, Yu Zhiwen. Anomaly detection and modeling of surveillance video[J]. *Journal of Computer Research and Development*, 2021, 58(12): 2708–2723 (in Chinese)
(杨帆, 肖斌, 於志文. 监控视频的异常检测与建模综述[J]. *计算机研究与发展*, 2021, 58(12): 2708–2723)
 - [8] Bergmann P, Batzner K, Fauser M, et al. The MVTec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection[J]. *International Journal of Computer Vision*, 2021, 129(4): 1038–1059
 - [9] Munyer T, Brinkman D, Zhong Xin, et al. Foreign object debris detection for airport pavement images based on self-supervised localization and vision transformer[C]//Proc of the 9th Int Conf on Computational Science and Computational Intelligence. Piscataway, NJ: IEEE, 2022: 1388–1394
 - [10] Liu Jialing, Lu Yifeng. A lightweight foreign object debris detection algorithm for airport runway[C]//Proc of the 5th Int Conf on Computer Science and Software Engineering. New York: ACM, 2022: 451–455
 - [11] Duygu K. Examining the effect of different networks on foreign object debris detection[J]. *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 2023, 12(1): 151–157
 - [12] Martí L, Sanchez-Pi N, Molina J M, et al. Anomaly detection based on sensor data in petroleum industry applications[J]. *Sensors*, 2015, 15(2): 2774–2797
 - [13] Baur C, Wiestler B, Albarqouni S, et al. Deep autoencoding models for unsupervised anomaly segmentation in brain MR images[C]//Proc of the 4th Int Workshop on Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Berlin: Springer, 2018: 161–169
 - [14] Zhao Chunhui, Yao Xifeng. Progressive line processing of global and local real-time anomaly detection in hyperspectral images[J]. *Journal of Real-Time Image Processing*, 2019, 16(6): 2289–2303
 - [15] Wyatt J, Leach A, Schmon S M, et al. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise[C]//Proc of the 39th IEEE/CVF Conf on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2022: 650–656
 - [16] Nawaratne R, Alahakoon D, De Silva D, et al. Spatiotemporal anomaly detection using deep learning for real-time video surveillance[J]. *IEEE Transactions on Industrial Informatics*, 2019, 16(1): 393–402
 - [17] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey[J]. *ACM Computing Surveys*, 2009, 41(3): 1–58
 - [18] Yang Jingkang, Zhou Kaiyang, Li Yixuan, et al. Generalized out-of-distribution detection: A survey[J]. *arXiv preprint, arXiv: 2110.11334*, 2021
 - [19] Cohen N, Hoshen Y. Sub-image anomaly detection with deep pyramid correspondences[J]. *arXiv preprint, arXiv: 2005.02357*, 2020
 - [20] Defard T, Setkov A, Loesch A, et al. Padim: A patch distribution modeling framework for anomaly detection and localization[C]//Proc of the 26th Int Conf on Pattern Recognition. Berlin: Springer, 2021: 475–489
 - [21] Roth K, Pemula L, Zepeda J, et al. Towards total recall in industrial anomaly detection[C]//Proc of the 39th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 14318–14328
 - [22] Deng Hanqiu, Li Xingyu. Anomaly detection via reverse distillation from one-class embedding[C]//Proc of the 39th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 9737–9746
 - [23] Gudovskiy D, Ishizaka S, Kozuka K. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows[C]//Proc of the 22nd IEEE/CVF Winter Conf on Applications of Computer Vision. Piscataway, NJ: IEEE, 2022: 98–107
 - [24] Li Chunliang, Sohn K, Yoon J, et al. Cutpaste: Self-supervised learning for anomaly detection and localization[C]//Proc of the 38th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 9664–9674
 - [25] Deng Jia, Dong Wei, Socher R, et al. ImageNet: A large-scale hierarchical image database[C]//Proc of the 26th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2009: 248–255
 - [26] Akcay S, Atapour-Abarghouei A, Breckon T P. Ganomaly: Semi-supervised anomaly detection via adversarial training[C]//Proc of the 14th Asian Conf on Computer Vision. Berlin: Springer, 2019: 622–637
 - [27] Gong Dong, Liu Lingqiao, Le V, et al. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection[C]//Proc of the 17th IEEE/CVF Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 1705–1714
 - [28] Dehaene D, Eline P. Anomaly localization by modeling perceptual features[J]. *arXiv preprint, arXiv: 2008.05369*, 2020
 - [29] Song J W, Kong K, Park Y I, et al. AnoSeg: Anomaly segmentation network using self-supervised learning[J]. *arXiv preprint, arXiv: 2110.03396*, 2021
 - [30] Hou Jinlei, Zhang Yingying, Zhong Qiaoyong, et al. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection[C]//Proc of the 18th IEEE/CVF Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2021: 8791–8800
 - [31] Ristea N C, Madan N, Ionescu R T, et al. Self-supervised predictive convolutional attentive block for anomaly detection[C]//Proc of the 39th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 13576–13586
 - [32] Zavrtnik V, Kristan M, Skočaj D. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection[C]//Proc of the 18th IEEE/CVF Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2021: 8330–8339
 - [33] Yuan Jingyang, Li Bin, Xue Xiangyang. Generative modeling of infinite occluded objects for compositional scene representation[C]//Proc of the 36th Int Conf on Machine Learning. New York: ACM,

- 2019: 7222–7231
- [34] Locatello F, Weissenborn D, Unterthiner T, et al. Object-centric learning with slot attention[J]. *Advances in Neural Information Processing Systems*, 2020, 33: 11525–11538
- [35] Zou Y, Jeong J, Pemula L, et al. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation[C]//*Proc of the 17th European Conf on Computer Vision*. Berlin: Springer, 2022: 392–408
- [36] Munyer T, Huang P C, Huang Chenyu, et al. FOD-A: A dataset for foreign object debris in airports[J]. *arXiv preprint*, arXiv: 2110.03072, 2021
- [37] Idrees H, Shah M, Surette R. Enhancing camera surveillance using computer vision: A research note[J]. *Policing: An International Journal*, 2018, 41(2): 292–307
- [38] Diehl C P, Hampshire J B. Real-time object classification and novelty detection for collaborative video surveillance[C]//*Proc of the 12th Int Joint Conf on Neural Networks*. Piscataway, NJ: IEEE, 2002, 3: 2620–2625
- [39] Bergmann P, Fauser M, Sattlegger D, et al. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings[C]//*Proc of the 37th IEEE/CVF Conf on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2020: 4183–4192
- [40] Salehi M, Sadjadi N, Baselizadeh S, et al. Multiresolution knowledge distillation for anomaly detection[C]//*Proc of the 38th IEEE/CVF Conf on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2021: 14902–14912
- [41] Wang Guodong, Han Shumin, Ding E, et al. Student-teacher feature pyramid matching for anomaly detection[J]. *arXiv preprint*, arXiv: 2103.04257, 2021
- [42] Dehaene D, Frigo O, Combexelle S, et al. Iterative energy-based projection on a normal data manifold for anomaly localization[J]. *arXiv preprint*, arXiv: 2002.03734, 2020
- [43] Schlegl T, Seeböck P, Waldstein S M, et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery[C]//*Proc of the 25th Int Conf on Information Processing in Medical Imaging*. Berlin: Springer, 2017: 146–157
- [44] Huang J, Murphy K. Efficient inference in occlusion-aware generative models of images[J]. *arXiv preprint*, arXiv: 1511.06362, 2015
- [45] Eslami S M, Heess N, Weber T, et al. Attend, infer, repeat: Fast scene understanding with generative models[J]. *Advances in Neural Information Processing Systems*, 2016, 29: 3225–3233
- [46] Crawford E, Pineau J. Spatially invariant unsupervised object detection with convolutional neural networks[C]//*Proc of the 33rd AAAI Conf on Artificial Intelligence*. Palo Alto, CA: AAAI, 2019: 3412–3420
- [47] Lin Zhixuan, Wu Y F, Peri S V, et al. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition[J]. *arXiv preprint*, arXiv: 2001.02407, 2020
- [48] Greff K, Kaufman R L, Kabra R, et al. Multi-object representation learning with iterative variational inference[C]//*Proc of the 36th Int Conf on Machine Learning*. New York: ACM, 2019: 2424–2433
- [49] Engelcke M, Kosiorek A R, Jones O P, et al. Genesis: Generative scene inference and sampling with object-centric latent representations[J]. *arXiv preprint*, arXiv: 1907.13052, 2019
- [50] Jiang Jindong, Ahn S. Generative neurosymbolic machines[J]. *Advances in Neural Information Processing Systems*, 2020, 33: 12572–12582
- [51] Yuan Jingyang, Li Bin, Xue Xiangyang. Knowledge-guided object discovery with acquired deep impressions[C]//*Proc of the 35th AAAI Conf on Artificial Intelligence*. Palo Alto, CA: AAAI, 2021: 10798–10806
- [52] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//*Proc of the 18th Int Conf Medical Image on Medical Image Computing and Computer-Assisted Intervention*. Berlin: Springer, 2015: 234–241
- [53] Hofmeyr D P. Fast exact univariate kernel density estimation[J]. *arXiv preprint*, arXiv: 1806.00690, 2018
- [54] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition[C]//*Proc of the 33rd IEEE Conf on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2016: 770–778



Fu Bingfei, born in 1993. PhD candidate. His main research interests include video, image information processing and retrieval, and computer vision.
傅冰飞, 1993年生. 博士研究生. 主要研究方向为视频、图像信息处理与检索、计算机视觉。



Chen Tonglin, born in 1991. PhD candidate. His main research interests include machine learning, deep generative models, and object-centric representation learning.
陈同林, 1991年生. 博士研究生. 主要研究方向为机器学习、深度生成模型和以对象为中心的表示学习。



Xu Feng, born in 1984. Master, advisory researcher. His main research interests include pattern recognition and intelligent system, and software engineering.
许枫, 1984年生. 硕士, 主管研究员. 主要研究方向为模式识别与智能系统、软件工程。



Zhu Lin, born in 1980. PhD, Senior R&D Manager in Artificial Intelligence Lab, Lenovo Research Institute. His main research interests include computer vision and machine learning.
朱麟, 1980年生. 博士, 联想研究院人工智能实验室高级研发经理. 主要研究方向为计算机视觉、机器学习。



Li Bin, born in 1982. PhD, associate professor, PhD supervisor. His main research interests include machine learning and visual intelligence, particularly in compositional scene representation, modeling, and inference.

李 斌, 1982 年生. 博士, 副教授, 博士生导师. 主要研究方向为机器学习和视觉智能, 尤其是合成场景表示、建模和推理方面.



Xue Xiangyang, born in 1968. PhD, professor, PhD supervisor. His main research interests include multimedia information processing and machine learning.

薛向阳, 1968 年生. 博士, 教授, 博士生导师. 主要研究方向为多媒体信息处理、机器学习.