

面向工业品缺陷检测的对比表示学习

罗航宇, 王小平, 梅萌, 赵文豪, 刘思纯

引用本文

罗航宇, 王小平, 梅萌, 赵文豪, 刘思纯. [面向工业品缺陷检测的对比表示学习](#) [J]. 计算机科学, 2025, 52(1): 210-220.

LUO Hangyu, WANG Xiaoping, MEI Meng, ZHAO Wenhao, LIU Sichun. [Contrastive Representation Learning for Industrial Defect Detection](#) [J]. Computer Science, 2025, 52(1): 210-220.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[一种基于对偶学习的场景分割模型](#)

Scene Segmentation Model Based on Dual Learning

计算机科学, 2024, 51(8): 133-142. <https://doi.org/10.11896/jsjcx.230700207>

[一种基于属性相似性和分布结构连通性的聚类算法](#)

Clustering Algorithm Based on Attribute Similarity and Distributed Structure Connectivity

计算机科学, 2024, 51(7): 124-132. <https://doi.org/10.11896/jsjcx.231000125>

[基于视觉的神经网络三维动态手势识别方法综述](#)

Review of Vision-based Neural Network 3D Dynamic Gesture Recognition Methods

计算机科学, 2024, 51(4): 193-208. <https://doi.org/10.11896/jsjcx.230200205>

[知识型视觉问答研究综述](#)

Knowledge-based Visual Question Answering: A Survey

计算机科学, 2023, 50(1): 166-175. <https://doi.org/10.11896/jsjcx.211100237>

[基于SVM的类别增量人体活动识别方法](#)

Human Activity Recognition Method Based on Class Increment SVM

计算机科学, 2022, 49(5): 78-83. <https://doi.org/10.11896/jsjcx.210400024>

面向工业品缺陷检测的对比表示学习

罗航宇 王小平 梅萌 赵文豪 刘思纯

同济大学电子与信息工程学院 上海 200092

(lhy123@tongji.edu.cn)

摘要 在大规模制造业中,缺陷检测旨在发现有缺陷的零部件,如损坏、错位的和存在印刷错误的部件等。由于缺陷类型未知以及缺陷样本短缺,工业品缺陷检测面临着极大的挑战。为克服上述困难,一些方法利用来自自然图像数据集的通用视觉表示,提取广义特征来进行缺陷检测。然而,提取到的预训练特征与目标数据之间存在分布差异,直接使用该特征会导致检测性能不佳。因此,提出了一种基于对比表示学习的方法 ConPatch。该方法采用对比表示学习来收集相似特征或者分离不相似特征,从而学习面向目标的特征表示。为了解决缺乏缺陷标注的问题,将数据表示之间的两种相似性度量即成对相似度和全局相似性作为伪标签。此外,采用了轻量化的内存库,仅将全部正常样本即全部无缺陷样本的特征中心存储到内存库中,从而减小了空间复杂度和内存库的尺寸。最后,将正常特征拉近至一个超球面内,而缺陷特征则分布在超球面外,以此来聚集正常特征。实验结果显示,在工业品缺陷检测数据集 MVTec AD 中,基于 Wide-ResNet50 的 ConPatch 模型的 I-AUROC 和 P-AUROC 分别达到 99.35% 和 98.26%。在 VisA 数据集中,ConPatch 模型的 I-AUROC 和 P-AUROC 分别达到 95.50% 和 98.21%。上述结果验证了模型的有效性。

关键词: 工业品缺陷检测;对比表示学习;相似性度量;内存库;超球面

中图分类号 TP391

Contrastive Representation Learning for Industrial Defect Detection

LUO Hangyu, WANG Xiaoping, MEI Meng, ZHAO Wenhao and LIU Sichun

School of Electronics and Information Engineering, Tongji University, Shanghai 200092, China

Abstract Defect detection in large-scale manufacturing aims to find defective components, such as damaged, misaligned components, and components with printing errors. Due to unknown defect types and shortage of defect samples, industrial defect detection faces great challenges. To overcome the above difficulties, some methods utilize common visual representations from natural image datasets to extract generalized features for defect detection. However, there are distribution differences between the extracted pre-trained features and the target data. Using this feature directly will lead to poor detection performance. Therefore, ConPatch, a method based on contrastive representation learning is proposed. This method employs contrastive representation learning to collect similar features or separate dissimilar features, resulting in goal-oriented representations of features. In order to solve the problem of lack of defect annotation, two similarity measures in data representations, pairwise similarity and global similarity, are used as pseudo labels. In addition, the method uses a lightweight memory bank and only stores the feature centers of all normal sample which are all defect-free sample in the memory bank, reducing the space complexity and the size of the memory bank. Finally, the normal features are brought closer to a hypersphere and the defect features are distributed outside the hypersphere to gather the normal features. Experimental results show that the I-AUROC and P-AUROC of the ConPatch model based on Wide-ResNet50 reaches 99.35% and 98.26% respectively in the industrial defect detection dataset MVTec AD. In the VisA dataset, I-AUROC and P-AUROC reaches 95.50% and 98.21%, respectively. The above results verify the effectiveness of the proposed model.

Keywords Industrial product defect detection, Contrastive representation learning, Similarity measure, Memory bank, Hypersphere

到稿日期:2024-01-29 返修日期:2024-06-25

基金项目:国家重点研发计划(2022YFB4300504-4)

This work was supported by the National Key Research and Development Program of China(2022YFB4300504-4).

通信作者:王小平(xpwang6510@tongji.edu.cn)

1 引言

在工业制造中,基于视觉的缺陷检测旨在发现各种工业产品的外观缺陷,是识别产品缺陷和保证产品质量的重要手段。相较于以往高成本和低效率的人工质检,基于视觉的缺陷检测方法具有更高的准确率与效率,因此被广泛应用于无人质检、智能巡检和质量控制等各种生产与运维场景中。

相较于一般的目标检测,工业品缺陷检测技术面临着诸多挑战。首先,缺陷的种类较多,缺陷通常是不规则的,同一工业产品可能存在不同类型的缺陷,例如划痕、异色和错位等。其次,缺陷样本匮乏,由于在实际工业品缺陷检测中很少观察到缺陷,且对缺陷进行精确的标注成本高昂,因此缺陷检测中大量样本被标记为正常样本,而缺陷样本较少。除此之外,缺陷检测要求更高的检测精度,缺陷工业品投入使用可能会给企业带来经济损失,甚至是安全事故。

由于工业品缺陷检测缺少缺陷样本,因此现有工作通常采用无监督方法。基于图像重建的方法仅在正常样本上训练,使模型能重建正常样本的分布特征。此类方法假设模型能准确地重建正常样本,而缺陷区域在重建时会产生较大的误差^[1]。基于该假设,通过原始输入与重建输出的误差来计算缺陷得分^[2-7],然而该假设并不可靠。深度单类分类方法^[8-10]利用深度神经网络构造高质量的特征空间,然后划分出分类界面来区分样本。此类方法的关键是训练模型以学习数据之间的距离度量,并检测与正常样本距离较大的异常样本。然而,该方法较难寻找合适的子空间与分界面。此外,基于内存库的方法^[11-13]利用来自自然图像数据集的预训练模型得到通用视觉表示进行缺陷检测,而无需在目标数据集上训练。然而,此类方法获得的数据表示与目标数据存在较大的分布差异,直接使用该有偏表示不足以检测出工业图像中的细微缺陷。

因此,本研究提出了基于对比表示学习的 ConPatch 方法,旨在训练对称的对比表示网络,学习到面向目标数据的特征表示,聚合高相似度的特征,同时排斥低相似度的特征。为了解决缺陷检测中缺乏标记对的问题,将特征之间的相似度即成对相似度和全局相似度作为伪标签。伪标签可以提供二元标签之外的丰富信息。

现有方法^[11-13]将目标数据集提取的正常特征直接或采样后存储到内存库中,然后学习正常特征的分布,所使用内存库的尺寸与目标数据集大小成正比。然而,如果内存库中存在大量的低质量特征,可能会导致错误地将缺陷特征划分为正常特征(假阴性),同时内存库尺寸增大也增加了空间开销和测试推理时间。因此,本研究采用一种轻量化的内存库,使用压缩方法将全部正常样本的特征中心存储到内存库中。

除此之外,为了更好地区分正常特征与缺陷特征,以训练面向目标数据的表示,本研究采用最近邻搜索算法,从内存库搜索正常特征的 k 近邻特征,以 k 近邻特征为中心构成超球面来拉近正常特征,最终将正常特征聚集在超球面内,而缺陷特征分布在超球面外。

通过上述方法,提出模型能够学习面向目标的特征表示,在无需数据增强的情况下实现良好的缺陷检测性能。

本文的主要贡献如下:

1)提出了对比表示网络来聚集或者分离特征,使模型能够更好地学习工业品特征。对比表示网络使用两个对称的卷积层和线性投影层。对称网络使用特征表示之间的成对相似度和全局相似度作为伪标签,解决缺乏标记对的问题。

2)采用一种轻量化的内存库,通过压缩算法存储高质量特征,同时降低建模复杂度,然后将对比表示网络与轻量化内存库和超球面映射策略结合起来共同训练特征表示。

3)验证了所提模型的有效性。在实际工业品缺陷检测数据集 MVTEC AD 中,所提模型的 I-AUROC 和 P-AUROC 分别达到 99.35% 和 98.26%;在 VisA 数据集中,其 I-AUROC 和 P-AUROC 分别达到 95.50% 和 98.21%。

2 相关工作

2.1 工业品缺陷检测

深度学习避免了人工质检的复杂流程,减少了人工成本的开销,同时能够获得更高的检测精度,因此在工业品缺陷检测中得到了广泛应用。在缺陷类型已知时,一般采用有监督深度学习方法,需要充足的样本标注。然而,在实际工业制造中,缺陷样本匮乏且大部分缺陷类型未知,因此无监督深度学习方法逐渐占据主流^[1]。

基于图像重建的方法早期使用自动编码模型^[2-4]进行图像重建。随着生成对抗网络(Generative Adversarial Networks, GANs)的广泛应用,基于 GAN 的方法^[5-7]也逐渐应用在缺陷检测中。然而,此类方法即使仅在正常样本上训练,仍然能准确地重建出缺陷,导致其检测性能不佳。

深度支持向量数据描述(Deep Support Vector Data Description, Deep SVDD)方法^[8],首次将深度学习与 SVDD 结合。通过训练模型将正常特征映射到超球面以内,而缺陷特征则会分布在超球面以外。在此基础上,文献^[9]以减小正常样本的特征之间的距离为目标进行优化,优化后无需人为指定特征中心,减少了人工操作。而 Patch SVDD^[10]使用聚类方法,在训练时自动生成多个特征中心,将方法扩展到复杂缺陷区域的定位任务。

为了解决缺陷数据匮乏的问题,基于内存库的工业品缺陷检测方法被提出。此类方法试图利用自然图像数据集上的预训练通用视觉表示,仅使用正常样本提取广义特征,然后将特征采样并存储到内存库中,而无需在目标数据集上训练。在测试阶段,通过计算输入特征与存储特征之间的距离,来定位输入样本的缺陷。在测试时,SPADE^[12]对输入样本的每个像素值进行 KNN 索引,计算每个像素点的缺陷得分。PaDim^[13]为所有像素位置估计一个多元高斯分布,而无需构造内存库,从而降低了测试的复杂度。PatchCore^[11]提出了一种局部感知的 Patch 特征,使用多层级特征融合机制,同时采用高效的贪婪子采样方法,提升了测试的效率。

此外,文献^[14]提出与单阶段目标检测算法原理相同的表面缺陷检测模型,通过加宽的特征提取网络提取丰富的上下文信息,然后进行多尺度特征融合,同时利用注意力机制提升缺陷检测的性能。文献^[15]使用小样本图像自适应滤波器去除小样本图像的噪声,优化图像质量,同时结合基于元学习

的小样本缺陷检测方法,实现小样本图像缺陷检测。CFA^[16]利用超球面映射思想^[8-9],在目标数据集上训练一个特征描述符。该描述符用于缩短提取特征与工业品目标域之间的距离。

2.2 对比表示学习

对比学习方法广泛应用于无监督表示学习任务上^[17-20],其核心思想是通过调整样本在嵌入空间中的位置,使正样本对靠近,负样本对分离,从而学习有意义的特征表示。MoCo^[21]模型使用动量更新策略来更新网络权重。动量更新将当前模型的权重与前一个状态的模型权重进行动量平均,然后更新网络参数。SimCLR^[22]模型引入一个投影网络,将图像嵌入映射到一个低维空间,以提高学习效率。

为了更充分地利用样本之间的相似度、类内变化和类间亲和力和信息, Kim 等提出了松弛对比学习^[23]。该方法使用松弛标签代替二元标签,通过计算样本对之间的相似度来生成伪标签,能够更全面、精确地学习特征表示。

ReContrast^[24]提出了将对比学习与重建相结合的方法,通过特征重构检测缺陷,同时优化编码器和解码器。ReConPatch^[25]方法在 PatchCore^[11]的基础上,额外提出了一个对比学习分支,通过训练线性网络来构建用于缺陷检测的判别特征。文献^[26]模拟异常数据和少量真实异常数据加入对比学习任务中,设计连续密集标准化流模型,增强生成式模型对分布的拟合能力。

3 基于 Patch 特征的对比表示方法—ConPatch

3.1 整体结构

模型整体可以分为 3 个阶段:数据预处理、训练以及测试阶段。数据预处理对第一个正常样本使用聚类方法生成初始特征中心,然后不断加入样本,逐步调整中心点的位置,最终内存库 C 中只存储了全部样本的特征中心,将其用作超球面映射模块的球心,同时为后续缺陷检测模块提供高质量的正常特征。具体过程在 3.2 节中描述。训练过程如图 1 所示。

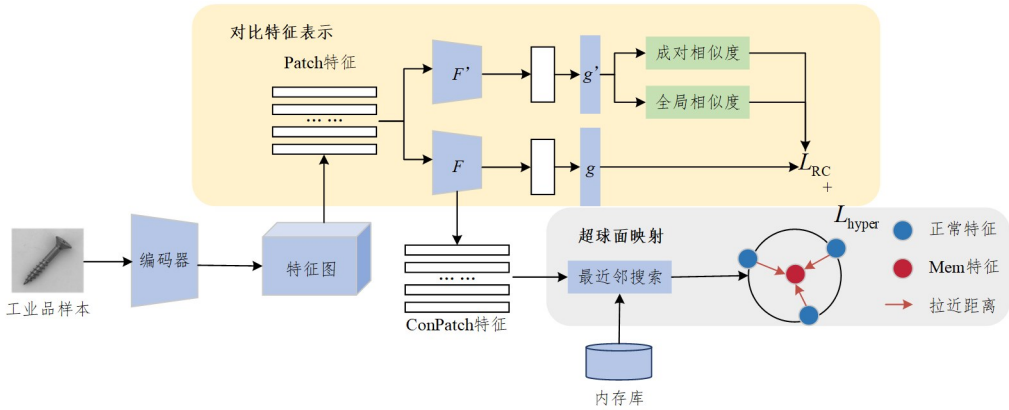


图 1 训练过程

Fig. 1 Training process

首先,使用在自然图像上的预训练模型(编码器)提取特征图,参照基线 PatchCore^[11]方法将不同层次特征图融合并提取 Patch 特征 $\Gamma \in \mathbb{R}^{H \times W \times D}$,其中每个 Patch 特征对应于原图的某一块区域,这里 H 和 W 分别代表 Patch 特征图的长度和宽度, D 表示特征图的维度。使用对比表示学习进行偏向工业品域的学习。对比表示学习分支由两个对称的特征表示网络 F 和投影网络 g 组成,可以将相似的 Patch 特征对聚集,不相似的特征对分离,并使用特征对之间的成对相似度和全局

相似度作为伪标签。对比特征表示模块生成 ConPatch 特征,对于该特征,使用超球面映射模块来监督。超球面映射模块被用于分离正常特征和缺陷特征,从而进一步训练特征表示。最终训练由对比特征表示和超球面映射模块提供损失,具体过程在 3.3 和 3.4 节描述。测试过程如图 2 所示,其中额外增加了缺陷检测模块。缺陷检测模块计算样本特征与记忆特征的距离并将其作为特征图的缺陷得分,根据得分进行缺陷判别以及进一步的像素级分割。具体过程在 3.5 节描述。

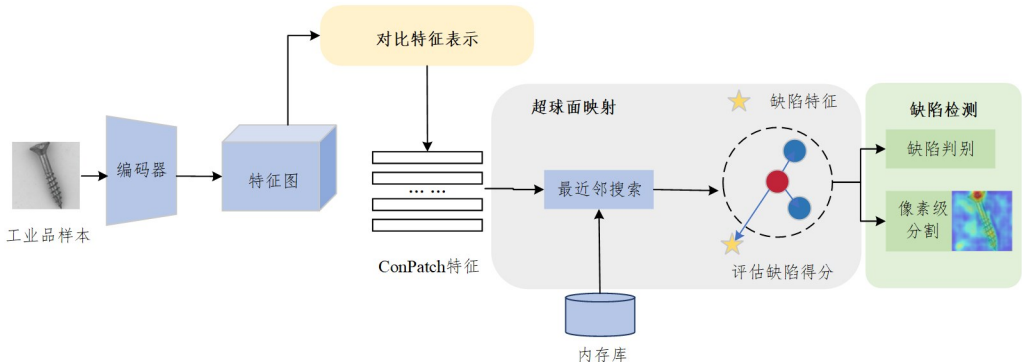


图 2 测试过程

Fig. 2 Testing process

3.2 内存库压缩算法

基于内存库的缺陷检测方法需要设置一个内存库来存储正常特征,内存库中存储的特征质量影响着缺陷检测的性能。本文统计了现有方法^[11-13]中内存库建模的复杂度以及内存库尺寸,结果如表1所列,其中 $|N|$ 为数据集的大小, H,W 和 D 分别代表特征图的长、宽和维度。

表1 内存库建模的复杂度和内存库尺寸的估计

Table 1 Estimates of memory bank modeling complexity and

memory bank size

方法	建模复杂度	内存库尺寸
SPADE ^[12]	$O(N HWD)$	$\Psi \in \mathbb{R}^{ N \times H \times W \times D^2}$
PaDiM ^[13]	$O(N HWD^2)$	$M \in \mathbb{R}^{ N \times H \times W \times D^2}$
PatchCore ^[11]	$O(N HWD)$	$N \in \mathbb{R}^{ N \times H \times W \times D}$
Ours	$O(HWD)$	$C \in \mathbb{R}^{\frac{H \times W \times D}{\gamma}}$

结果显示,现有算法^[11-13]内存库建模复杂度以及内存库尺寸与目标数据集的大小成正比。因此,现有算法的内存库需要随着数据集的增大而扩容。然而,大容量的内存库会增加测试时间和空间开销;同时,如果内存库尺寸较大,其中的低质量特征会导致将缺陷特征错误地估计为正常特征,即假阴性比例上升。

为了缓解这种现象,本研究在初始化阶段使用压缩方案来构建一个高效的内存库^[16]。压缩过程如算法1所示。

算法1 压缩过程算法

输入:特征表示网络 F ,Patch特征表示 \mathbf{p} ,数据集 N ,EMA参数 β ,内存库 C

输出:存储了正常样本特征中心的内存库 C

- $C_0 \leftarrow K$ Means $F_0(\mathbf{p})$ //初始化
- for $i \in \{1, \dots, |N|\}$ do //对每一个样本进行处理
- $C_i^{NN} \leftarrow \{\}$ //初始为空
- for $i \in \{1, \dots, |C|\}$ do //对内存库的每个状态
- $Y \leftarrow (F_i(\mathbf{p}) \cup C_i^{NN}) \cap (C_i^{NN})^c$
- $C_i^{NN} \leftarrow C_i^{NN} \cup \arg \min_{y \in Y} \|y - C_{i-1}^j\|_2$ //搜索最近Patch特征
- end for
- $C_i \leftarrow (1-\beta) \cdot C_{i-1} + \beta \cdot C_i^{NN}$ //混合更新新的内存库
- end for
- $C \leftarrow C_{|N|}$
- 返回 C

具体来说,压缩过程对训练集的第一个正常样本 N_0 得到 $F_0(\mathbf{p})$, $F_0(\mathbf{p})$ 是 $D \times H \times W$ 维特征,对该特征使用 K -means聚类算法。假设聚类算法的簇中心数目为 n_{clu} ,则该 $D \times H \times W$ 维特征聚类成 $D \times n_{clu}$ 维特征,将该特征存储得到初始内存库 C_0 。更新内存库的过程如下:对于第 i 个样本 N_i ,从上一个状态的内存库 C_{i-1} 中搜索最近的Patch特征 C_i^{NN} ,使用指数移动平均(Exponential Moving Average, EMA)的方式,通过将前一个状态的内存库 C_{i-1} 与 C_i^{NN} 进行混合,来计算新的内存库 C_i ,以逐步调整特征中心点的位置。通过对所有正常样本重复上述过程 $|N|$ 次,获得最终的内存库 C 。整个压缩过程的状态变化如图3所示。

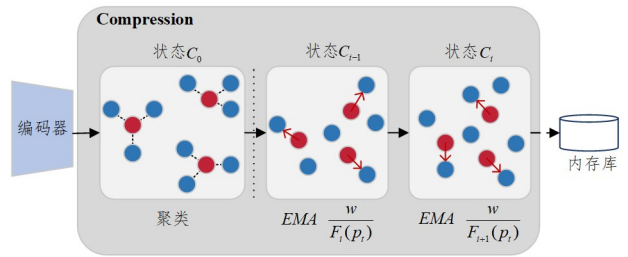


图3 压缩过程

Fig. 3 Compression process

由于初始得到的特征为 $D \times H \times W$ 维,因此更新过程的建模复杂度为 $O(H \times W \times D)$ 。而算法1在每个状态下都会更新内存库 C ,内存库中始终只有 $D \times n_{clu}$ 的特征,设置超参数 γ 为压缩比,表示由特征提取器得到的Patch数与簇中心数目的比值,即 $\gamma = \frac{H \times W}{n_{clu}}$ 。 γ 用于控制簇中心的数目, γ 越大,簇中心数目越少。这样,内存库中始终只有 $\gamma(H \times W) \times D$ 维特征。因此,本研究内存库尺寸不受表1中数据集大小 $|N|$ 的影响,极大地减小了空间复杂度。

3.3 对比特征表示

在有监督分类网络训练时,通常将模型预测输出与实际标签进行比较,然后通过计算损失来更新网络参数,使得网络能学习到与样本有关的特征。由于缺陷检测缺乏缺陷样本,因此PatchCore^[11]等方法使用无监督方式训练。这些方法未在目标数据集上训练,因此提取到的特征不适用于工业品目标。

ConPatch的目标是从Patch特征中学习面向目标的特征,从而更有效地区分正常和缺陷特征。为实现此目标,提出对比表示学习方法来聚合高相似度的特征,同时排斥低相似度的特征。通过计算相似度作为伪标签,计算特征向量的距离作为预测输出,然后将这两者进行对比来计算损失并更新网络参数,因此模型在工业品数据集上进行学习,并获得了面向工业领域的特征。同时,该方法旨在提高对缺陷区域的敏感性,使得模型能够更准确地识别和分割出工业品中的缺陷。

所提方法参照PatchCore^[11]提取图像的Patch特征,其中每个Patch特征代表输入图像的某个特定区域,然后将Patch特征送入对比表示学习模块ConPatch。如图4所示,该模块由两个对称分支组成。在进行对比监督时,缺陷数据匮乏,从而导致缺乏标记数据,因此本研究使用辅助分支计算特征之间的成对相似度和上下文相似度作为伪标签。辅助网络分支由对称的 F' 和 g' 组成,结构与表示网络相同。

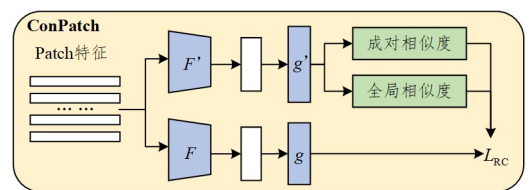


图4 对比表示学习模块

Fig. 4 Module of contrastive represents learning

对于任意两个Patch特征 \mathbf{p}_i 和 \mathbf{p}_j ,其经过投影空间得到的表示分别为 $\mathbf{Z}_i = g(F(\mathbf{p}_i))$ 和 $\mathbf{Z}_j = g'(F'(\mathbf{p}_j))$ 。那么,两个特征之间的成对相似度 $\omega_{ij}^{\text{pairwise}}$ 如式(1)所示:

$$\omega_{ij}^{\text{pairwise}} = e^{-\frac{\|z_i - z_j\|_2}{\sigma}} \quad (1)$$

其中, σ 为高斯核的带宽, 用于调整相似度的平滑度。

然而, 由于特征之间是否相似的分界并不清晰, 成对相似度不足以划分特征之间的关系, 因此不能准确聚集或者分离 Patch 特征, 从而导致假阳性和假阴性比例上升。工业品正常特征和缺陷特征的分界如图 5 所示, 其中矩形代表缺陷特征, 圆形代表正常特征, 虚线代表特征分界。

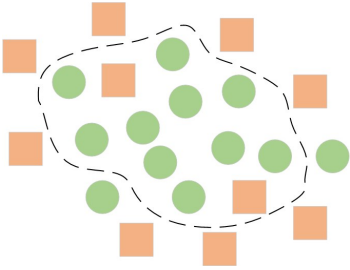


图 5 特征分界

Fig. 5 Features division

因此, 本研究采用了更精细的特征划分方法: 对于每对 Patch 特征 p_i 和 p_j , 通过对比其 k 近邻特征 N_k^i 和 N_k^j 重合的个数来计算相似度, 从而判断特征对是否相似。本研究将上述判断标准记为全局相似度 $\omega_{ij}^{\text{global}}$, 计算过程如式(2)所示:

$$\omega_{ij}^{\text{global}} = \begin{cases} \frac{|N_k^i \cap N_k^j|}{|N_k^i|}, & \text{if } j \in N_k^i \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

两个 Patch 特征 p_i 和 p_j 之间的最终相似度 ω 被定义为两个相似性的线性组合, $\alpha \in (0, 1)$, 如式(3)所示:

$$\omega = \alpha \omega_{ij}^{\text{pairwise}} + (1 - \alpha) \omega_{ij}^{\text{global}} \quad (3)$$

本研究采用松弛对比损失^[2], 由式(3)所得的相似度 ω 被视为伪标签进行对比损失计算, 使用松弛对比损失不需要缺陷标注, 同时也能利用特征相似度以及类内变化等信息。假设特征投影 δ_{ij} 表示得到的向量 z_i 和 z_j 在嵌入空间中的相对距离, 那么, 特征投影 δ_{ij} 的计算过程如式(4)所示:

$$\delta_{ij} = \frac{\|z_i - z_j\|_2}{\frac{1}{N} \sum_{n=1}^N \|z_i - z_n\|_2} \quad (4)$$

最终对比损失的计算过程如式(5)所示:

$$L_{RC} = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N \omega_{ij} (\delta_{ij})^2 + \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^N (1 - \omega_{ij}) \max(m - \delta_{ij}, 0)^2 \quad (5)$$

其中, N 代表样本数量, m 为正常与缺陷特征阈值, 使用 ω_{ij} 代替二元标签作为伪标签。

辅助网络的快速训练会降低 Patch 特征之间关系的一致性, 导致训练不稳定。因此, 本研究在训练过程中采用指数移动平均策略缓慢更新辅助网络参数。辅助网络参数的更新过程如式(6)所示:

$$\theta_{F', g'} \leftarrow \lambda \cdot \theta_{F', g'} + (1 - \lambda) \theta_{F, g} \quad (6)$$

其中, $\theta_{F', g'}$ 为辅助网络的参数, $\theta_{F, g}$ 为学习网络的参数, λ 为调节动量更新速率的参数。

3.4 超球面映射

文献[8-10]通过引入超球面思想来学习目标数据集的分布, 通过最小化训练数据表征的封闭球面, 可以使网络提取出

变化数据中的共有特征, 然后进行缺陷检测。

为了进一步解决预训练特征与目标域存在偏差的问题, 本研究将超球面映射和对比表示学习结合起来共同训练一个适应于工业品目标域的特征表示网络。

具体过程如下: 从内存库 C 中取出特征中心, 通过最近邻搜索算法计算 ConPatch 特征 $F(p_i)$ 与特征中心的 k 近邻 C_i^k , 通过特征表示网络 F 监督 p_i , 使其靠近以 C_i^k 为中心、超参数 r 为半径构成的超球面, 从而拉近 k 邻居特征与特征中心之间的距离, 使得正常特征高度集中在超球面以内, 而缺陷特征分布在超球面以外。损失函数 L_{hyper} 的计算过程如式(7)所示:

$$L_{\text{hyper}} = \frac{1}{TK} \sum_{t=1}^T \sum_{k=1}^K \max\{0, D(F(p_t), C_t^k) - r^2\} \quad (7)$$

其中, $T = H \times W$ 代表每个样本 Patch 特征的数量, 超参数 K 代表与 $F(p_i)$ 匹配的最近邻居数量, $D(F(p_i), C_i^k) = \sqrt{(F(p_i) - C_i^k)^2}$ 计算 ConPatch 特征与 k 邻居特征的距离。

训练时, 总损失为对比损失 L_{RC} 与超球面损失 L_{hyper} 之和, 如式(8)所示:

$$L_{\text{total}} = \varphi L_{RC} + L_{\text{hyper}} \quad (8)$$

其中, φ 是用于权衡两个损失项的超参数。

3.5 缺陷检测

在 3.4 节中, ConPatch 特征 $F(p_i)$ 与 k 近邻 C_i^k 之间的距离用欧氏函数计算, 则缺陷得分可以使用 $S = \min_k D(F(p_i), C_i^k)$ 来计算, 即计算 ConPatch 特征与特征中心之间的最小距离, 结果表示 ConPatch 特征的缺陷程度。

然而, 由于正常特征是连续分布的, 聚类之间的边界并不清晰, 仅用 S 很难准确地区分缺陷特征。因此, 提出了一种新的评估得分方法, 即衡量距离特征最近的特征中心与其余特征中心的接近程度, 并将其定义为确定性。通过将 S 与确定性相乘作为缺陷得分来解决假阴性的问题, 最终缺陷得分函数 Score 的计算过程如式(9)所示:

$$\text{Score} = \frac{e^{-S}}{\sum_{k=1}^K e^{-D(F(p_i), C_i^k)}} \cdot S \quad (9)$$

为了判别缺陷, 本研究通过最大化 F1 分数来确定阈值 (Threshold)。F1 分数综合考虑了准确率 (Precision) 和召回率 (Recall), 减小了假阳性和假阴性的影响。F1 分数的计算过程如式(10)所示:

$$\text{F1Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

之后, 在不同的分割阈值下, 计算模型的 F1 分数, 然后选择使 F1 分数最高的阈值作为最终的分割阈值, 过程如式(11)所示:

$$\text{Threshold} = \arg \max_k (\text{F1Score}) \quad (11)$$

在判别缺陷时, 使用各 Patch 特征得到的最大缺陷得分值作为图像级缺陷得分, 根据计算出的 Threshold, 将得分大于阈值的判定为缺陷样本, 而得分小于阈值的判定为正常样本。

对于像素级缺陷分割, 将各 Patch 缺陷得分上采样到原图尺寸, 得到像素级缺陷得分, 这样每个像素点都对应一个缺陷得分; 将每个像素点的缺陷得分与二元分类阈值进行比较, 判别缺陷, 从而得到像素级分割结果。此处上采样方法采用

双线性内插,并对插值结果使用高斯平滑。像素级缺陷得分的计算过程如式(12)所示:

$$PixelScores = GaussianSmooth(Upsample(Scores, ImageSize)) \quad (12)$$

3.6 模型训练和测试

3.6.1 训练

模型的训练过程如算法 2 所示。训练过程只使用正常样本提取特征,初始化过程将全部正常样本的特征中心存入内存库中;在训练阶段,利用从 ImageNet 预训练网络上提取到的 Patch 特征,使用对比表示学习进行偏向工业品目标域的学习。在这个过程中,通过计算 Patch 特征的成对相似度和全局相似度并将其作为伪标签,进一步计算对比损失。同时,使用最近邻搜索算法从内存库中得到输入特征的 k 邻居特征,使用基于 k 邻居特征为中心构成的超球面来进行监督,拉近 k 邻居特征与特征中心之间的距离,并计算超球面损失。最终的损失通过将两部分损失相加得到,然后计算梯度,更新表示网络 F 和 g 的参数,而辅助网络 F' 和 g' 的参数通过 EMA 策略更新。

算法 2 训练过程算法

输入:正常样本 N

输出:无

1. 初始化超参数
2. 使用算法 1 初始化成内存库 C
3. for number of industrial types do
4. for step do
5. $cp_i = \text{self. ConPatch}(\text{Patch_features})$ //计算 ConPatch 特征
6. 使用式(3)计算相似度作为伪标签
7. 使用式(4)计算对比损失
8. $C_k = \text{NNS}(cp_i, C)$ //在 C 中搜索 ConPatch 特征 k 近邻
9. 使用式(6)计算超球面损失;
10. 使用式(7)计算总损失;
11. end for
12. end for
13. 根据损失更新表示网络参数,并使用式(5)更新辅助网络参数

3.6.2 测试

在测试阶段输入正常或缺陷样本,从内存库存储的特征中心搜索 k 近邻特征,使用欧氏函数计算样本特征与 k 近邻特征的距离作为特征图的缺陷得分,通过比较特征图缺陷得分的最大值和阈值来判断图像是否具有缺陷,并将特征图平滑上采样到原图尺寸,判断每个像素是否是缺陷。

4 实验

4.1 实验设置

4.1.1 实验数据集

本研究聚焦于视觉工业品缺陷检测。为了评估模型在不同场景下的泛化能力以及检测多种缺陷的能力,采用真实工业场景下构建的 MVTec AD 数据集^[27]和 VisA 数据集^[28]进行实验。这两种数据集涵盖了多种缺陷类型,且缺陷样本相对较少。

1)MVTec AD 数据集^[27]共有 5354 张高分辨率图像,包含 15 种不同的工业品,分为物体和纹理两类,如皮革、牙刷和晶体管等。每个工业品类别数据样本被划分为训练集和测试集,

其中训练仅使用正常图像,而在测试时才使用具有像素级精确标注的缺陷图像。这些工业品包含 70 种不同类型的缺陷,例如划痕、凹痕、污渍以及结构的变化等。

2)VisA 数据集^[28]包含 12 个子数据集,对应于 12 种不同的对象,共有 10821 张图像,其中包含 9621 张正常样本和 1200 张缺陷样本。其中,4 个子集对应不同类型的印刷电路板(PCB),具有相对复杂的结构,集成了晶体管、电容器和芯片等组件。该数据集还考虑了一张图像可能包含多个实例的情况,多实例数据收集了胶囊、意大利面和蜡烛等工业品。

数据预处理过程将数据样本的大小调整为 256×256 ,并使用中心裁剪方法将其裁剪成尺寸为 224×224 的图像。

4.1.2 实验环境

本研究的实验环境设置如表 2 所列,使用了 80GB 内存的显卡,然而由于计算欧氏距离时占用内存过多,因此实验中 batch size 只能设置为 2,在每个工业品类别上预估训练 30 个 epoch。

表 2 实验环境

Table 2 Experimental environment

实验环境	环境配置
操作系统	CentOS Linux release 8.5.2111
处理器	AMD EPYC 7763
显卡	NVIDIA Tesla A800 80G PCIE
显存/GB	80
深度学习框架	Pytorch 1.10.0

4.1.3 参数设置

实验使用在 ImageNet 数据集上预训练得到的 Wide-ResNet50 和 ResNet18 作为特征提取器,训练过程中固定网络参数。融合生成 Patch 特征时,从 ResNet 的中间特征层 layer1, layer2 和 layer3 抽取多层次特征,特征图对应的空间分辨率分别是输入图像分辨率的 $1/4$, $1/8$ 和 $1/16$ 。

对比表示网络结构设置中,特征表示网络 F 使用 1×1 的 CoordConv 网络, g 使用线性网络,辅助网络与表示网络结构相同。网络训练中应用了 amsgrad 策略的 AdamW 参数优化器。学习率设置为 1×10^{-3} ,权重衰减设置为 5×10^{-4} 。

对比表示学习中超参数 m 为推离不相似特征对的阈值,本研究对 m 进行了超参数调优实验。在 MVTec AD 数据集上,使用 Wide-ResNet50 作为特征提取器。测试集上,检测性能 I-AUROC 随 m 的变化情况如图 6 所示。

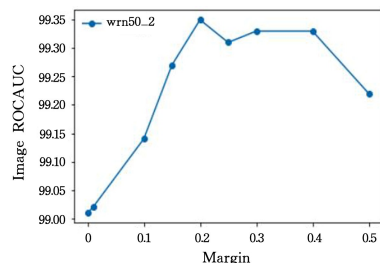


图 6 超参数 m 对检测性能的影响

Fig. 6 Effect of super parameter m on detection performance

当 $m=0.2$ 时,模型的 I-AUROC 最大,而 m 过大或者过小时都会影响正常样本与缺陷样本的界定,从而影响到模型的性能,因此在后续实验中将 m 设置为 0.2。

经过超参数调优的实验之后,其余超参数设置如下:压缩过程中 EMA 更新参数 β 设置为 0.1,对比特征表示模块相似度参数 α 设置为 0.8,高斯核 σ 设置为 4,辅助网络 EMA 更新参数 λ 设置为 0.4。超球面映射模块球面半径 r 设置为 1×10^{-5} , k 近邻搜索中 K 设置为 3。总 loss 计算参数 φ 设置为 0.6。

4.2 评价指标

缺陷检测中,正常样本数目通常远超缺陷样本,因此仅基于准确率的评价指标并不适用。本研究使用的评价指标 AUROC 是召回率 TPR 和误检率 FPR 的比例,对于不平衡类别的问题更加鲁棒。

ROC 曲线是在不同阈值下获得多组 (FPR, TPR) 绘制的曲线,以误检率 (FPR) 为横坐标轴,召回率 (TPR) 为纵坐标轴,如图 7 所示。

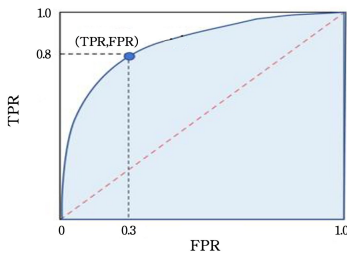


图 7 ROC 曲线

Fig. 7 ROC curve

对于 ROC 曲线,在理想情况下, $TPR=1, FPR=0$ 时分类效果最好,真实情况下 ROC 曲线越靠近 (0,1) 点分类效果越好。因此,本研究使用 ROC 曲线与 x 轴围成的面积作为

评价指标,记为 AUROC。在实验中,使用 I-AUROC 评估缺陷判别性能,使用 P-AUROC 评估缺陷分割性能。

4.3 对比实验

为衡量提出模型的性能,本研究复现了近几年主流的缺陷检测模型 CFLOW-AD^[29], SPADE^[12], PaDim^[13], PatchCore^[11] 和 CFA^[16] 等;使用 Wide-ResNet50 和 ResNet18 作为特征提取网络进行了定性实验和量化实验。

SPADE^[12], PaDim^[13] 和 PatchCore^[11] 是基于内存库进行缺陷检测的经典方法,在算法思想方面呈现递进关系。PatchCore^[11] 在前两种方法基础上提出了贪婪子采样方法,极大地提升了检测性能。除此之外,该方法提出的多层次特征融合提取 Patch 特征思想是本研究提取特征方法的基础。然而,这 3 种方法提取的记忆特征与工业品分布差异较大。

CFLOW-AD^[29] 使用条件归一化流进行无监督缺陷检测,用更低的复杂度进行多尺度特征映射,同时保持了良好的检测性能。CFA^[16] 使用超球面映射思想极大地提升了缺陷检测性能。

4.3.1 定性实验

本小节将所提模型分别使用 Wide-ResNet50 和 ResNet18 作为特征提取器,在 MVTec AD 和 VisA 数据集上进行实验,定性实验结果如图 8 和图 9 所示。图中,第一行是原始输入图像;第二行是输入图像对应的 Ground Truth;第三行是测试得到的热力图,热力图能够准确地关注缺陷所在位置,清晰地可视化了模型的关注重点,为进一步分割提供依据;第四行是测试最终的分割结果,使用红色曲线将缺陷所在位置框出。

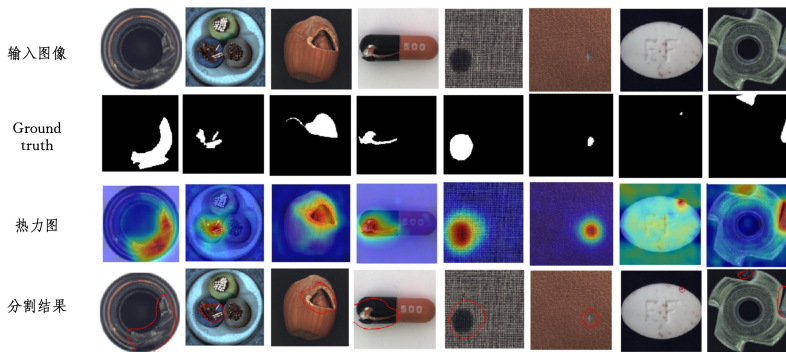


图 8 MVTec AD 数据集上测试结果(电子版为彩图)

Fig. 8 Test results on MVTec AD dataset

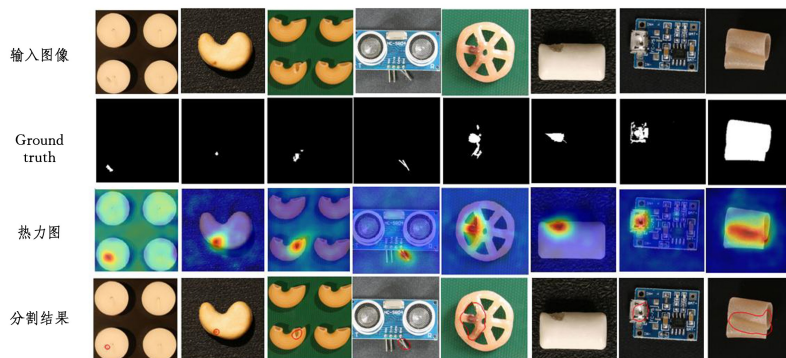


图 9 VisA 数据集上的测试结果(电子版为彩图)

Fig. 9 Test results on VisA dataset

定性实验结果显示,所提模型能够准确地各个工业品如钢缆、皮革和 PCB 板上定位缺陷位置(缺陷在热力图上体现为颜色更深的部分)并将其分割出来,缺陷部位也使用了红色曲线准确框出。

该实验结果验证了模型在两个真实工业品数据集上的有效性。即使在不同工业品上存在不同种类的缺陷,所提模型都能准确地将其分割出来。除此之外,训练过程只使用正常样本,而在测试时才使用缺陷样本,因此模型能学习正常样本的特征,从而检测出未知的缺陷类型。

此外,本研究将提出方法与复现方法的实验结果进行对比,对比结果如图 10 所示。图中第一行是原始输入图像,第二行是输入图像对应的 Ground Truth,第三行是使用 CFA 方法得到的测试分割结果,第四行是使用提出方法得到的测试分割结果,两种方法分割缺陷的结果都用红色曲线框出。

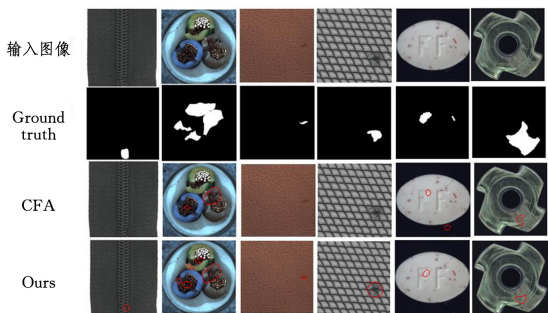


图 10 MVTEC AD 数据集测试结果对比

Fig. 10 Comparison of test results on MVTEC AD dataset

图 10 显示,所提模型在一些工业品类别(如拉链、皮革和钢缆等)上的分割结果更好。这是由于所提模型同时结合了对比表示模块和超球面映射共同来训练网络,而 CFA 只使用了超球面映射。对比表示模块可以更好地聚集相似特征而分离不相似特征,对于细节的检测更精准,因此提出模型的检测效果更佳。

本研究同样使用热力图来可视化测试结果,图 11 给出了所提方法与 CFA 方法在 MVTEC AD 数据集上的可视化对比结果,可以看出所提方法定位缺陷区域的精度更高。

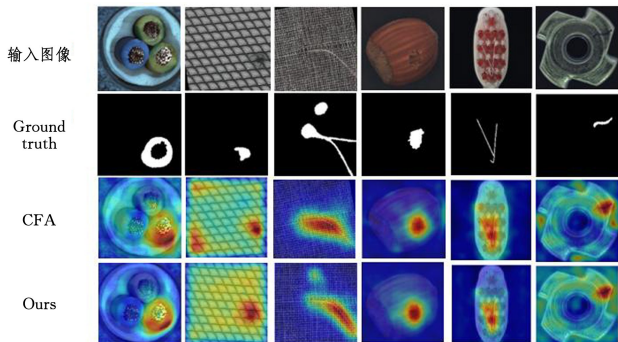


图 11 MVTEC AD 数据集上测试结果的可视化

Fig. 11 Visualization of test results on MVTEC AD dataset

4.3.2 量化实验

本小节介绍模型的量化性能。表 3 和表 4 列出了使用 ImageNet 预训练的 Wide-ResNet50 和 ResNet18 在

MVTEC AD 和 VisA 数据集上的 I-AUROC 和 P-AUROC 结果,并且将其与主流方法 PatchCore 和 CFA 等进行了对比。

表 3 MVTEC AD 数据集上的实验结果

Table 3 Experimental results on MVTEC AD dataset

(%)			
方法	Backbone	I-AUROC	P-AUROC
CFLOW-AD ^[29]	WRN_50	98.26	97.87
SPADE ^[12]	WRN_50	85.50	96.00
PaDim ^[13]	WRN_50	95.30	97.50
PatchCore ^[11]	WRN_50	99.10	98.10
FAIR ^[30]	WRN_50	98.60	98.20
TransFusion ^[31]	WRN_50	99.20	—
CFA ^[16]	ResNet18	98.46	97.74
	WRN_50	99.28	98.12
Ours	ResNet18	98.76	97.87
	WRN_50	99.35	98.26

在 MVTEC AD 数据集上的实验结果显示,提出方法在同样使用 Wide-ResNet50 作为特征提取器的基础上,I-AUROC 和 P-AUROC 分别比 CFA 提升了 0.07% 和 0.14%。与理想情况 $TPR=0, FPR=1$ 相比,CFA 的检测性能相差 0.72%,而提出模型仅相差 0.65%,因此提出模型与 CFA 相比,分类错误的概率降低了 9.73%;同理,提出模型与 CFA 相比,分割出错的概率降低了 13.2%。

表 4 VisA 数据集上的实验结果

Table 4 Experimental results on VisA dataset

(%)			
方法	Backbone	I-AUROC	P-AUROC
SPD ^[28]	WRN_50	87.80	93.80
PNI ^[32]	WRN_101	95.20	98.80
CFA ^[16]	WRN_50	93.92	98.11
Ours	WRN_50	95.50	98.21

在 VisA 数据集上,提出模型的 I-AUROC 和 P-AUROC 分别达到了 95.50% 和 98.21%,与使用了 Wide-ResNet101 的 PNI 方法得到的 95.2% 和 98.8% 相比,I-AUROC 提升了 0.3%,P-AUROC 降低了 0.59%。所提模型的分割性能不如 PNI 方法,但却相差不大。然而,所提模型只使用了 Wide-ResNet50 作为特征提取器,PNI 方法则使用了特征提取能力更好的 Wide-ResNet101,因此最终结果相差不大也从侧面验证了所提模型的有效性。

除此之外,本文详细统计了模型在 MVTEC AD 数据集各个工业品类别上的 I-AUROC 和 P-AUROC 实验结果,如表 5 和表 6 所列。将工业品分为物体类和纹理类两大类,物体类包括瓶子、钢缆和胶囊等,纹理类包括毛毯、铁网和皮革等。结果显示,CFLOW-AD 和 FAIR 方法在物体类的检测中性能较差,但 FAIR 在纹理类物体检测中更加精准。在物体类工业品检测中,提出方法的性能提升较少;而在纹理类的工业品检测中,其性能提升较大,相比 CFA 方法,I-AUROC 提升了 0.32%。这是由于提出的对比特征表示网络可以收集相似特征和分离不相似特征,在纹理类工业品检测中更容易区分是否与背景一致,从而判断出是否存在缺陷区域。

表 5 MVTEC AD 数据集各工业品类别的 I-AUROC 实验结果

Table 5 I-AUROC experimental results of each industrial category on MVTEC AD dataset (%)

工业品	CFLOW-AD ^[29]	PatchCore ^[11]	CFA ^[16]	TransFusion ^[32]	FAIR ^[31]	Ours
瓶子	100.00	100.00	100.00	100.00	100.00	100.00
钢缆	97.59	99.50	99.80	97.90	98.10	99.85
胶囊	97.68	98.10	97.30	98.50	97.00	97.69
榛子	99.98	100.00	100.00	100.00	99.20	100.00
螺帽	99.26	100.00	100.00	100.00	98.00	100.00
药丸	96.82	96.60	97.90	98.30	99.00	98.09
钉子	91.89	98.10	97.30	97.20	91.60	96.91
牙刷	99.65	100.00	100.00	100.00	100.00	100.00
晶体管	95.21	100.00	100.00	98.30	98.60	100.00
物体类平均	97.56	99.14	99.14	98.91	97.94	99.17
毛毯	98.73	98.70	97.30	99.20	99.70	99.92
铁网	99.60	98.20	99.20	100.00	99.70	99.58
皮革	100.00	100.00	100.00	100.00	100.00	100.00
瓷砖	99.88	98.70	99.40	99.80	100.00	98.86
木板	99.12	99.20	99.70	99.40	100.00	99.74
拉链	98.48	99.40	99.60	100.00	98.50	99.68
纹理类平均	99.30	99.03	99.30	99.73	99.65	99.62
平均	98.26	99.10	99.28	99.20	98.60	99.35

表 6 MVTEC AD 数据集各工业品类别的 P-AUROC 实验结果

Table 6 P-AUROC experimental results of each industrial category on MVTEC AD dataset (%)

工业品	CFLOW-AD ^[29]	SPADE ^[12]	PaDim ^[13]	PatchCore ^[11]	CFA ^[16]	FAIR ^[31]	Ours
瓶子	98.76	98.40	98.30	98.60	98.50	98.30	98.98
钢缆	97.64	97.20	96.70	98.80	98.70	98.50	98.72
胶囊	98.98	99.00	98.50	98.40	98.50	93.90	98.96
榛子	98.82	99.10	98.20	98.70	98.50	99.40	99.03
螺帽	98.56	98.10	97.20	98.40	97.60	98.10	99.21
药丸	98.95	96.50	95.70	97.40	98.30	98.40	98.57
钉子	98.10	98.90	98.50	99.40	98.80	98.80	99.04
牙刷	98.56	97.90	98.80	98.70	98.30	99.20	98.74
晶体管	93.28	94.10	97.50	96.30	98.00	95.40	98.67
物体类	97.96	97.69	97.71	98.30	98.23	97.78	98.88
毛毯	99.23	97.50	99.10	99.00	98.40	99.60	98.39
铁网	96.89	93.70	97.30	98.70	94.90	99.40	95.22
皮革	99.61	97.60	99.20	99.30	98.60	99.60	98.44
瓷砖	97.71	87.40	94.10	95.60	98.40	98.40	98.40
木板	94.49	88.50	94.90	95.00	94.10	97.30	94.71
拉链	98.41	96.50	98.50	98.80	98.50	99.40	98.77
纹理类平均	97.72	93.53	97.18	97.73	97.15	98.95	97.32
平均	97.87	96.00	97.50	98.10	98.18	98.20	98.26

4.4 消融实验

4.4.1 对比表示模块和超球面映射模块有效性验证

提出的模型将对表示学习与超球面映射结合,以提高缺陷检测的性能。本节介绍对比表示模块和超球面映射模块的有效性验证。消融实验使用 Wide-ResNet50 作为特征提取器,在 MVTEC AD 数据集上进行了实验,实验的数据预处理操作与 4.1 节保持相同,同时超参数选择最优值。

表 7 显示,同时使用两个模块比单独使用一个模块的性能更好,其中 $I-AUROC$ 至少提升了 0.18%, $P-AUROC$ 提升了 0.19%,从而验证了结合两个模块的有效性。

表 7 对比表示模块和超球面映射模块的有效性验证

Table 7 Validity verification of comparison representation module and hypersphere mapping module (%)

ConPatch	超球面映射	$I-AUROC$	$P-AUROC$
√	—	99.17	98.05
—	√	99.04	97.95
√	√	99.35	98.26

4.4.2 多层次 ResNet 特征有效性验证

对于计算机视觉任务来说,全面理解特征在不同层次的影响,分析不同感受野带来的信息差异非常重要。本研究使用不同层次的 ResNet 特征进行融合,观察模型是否有性能提升。消融实验在 PatchCore^[11] 的基础上,对于 ResNet 特征层,对比使用 layer2, layer3 和同时使用 layer1, layer2, layer3 进行特征融合两种设置,以不同层次特征上采样后相加的方式进行融合。实验中同时使用了对比表示学习和超球面映射模块,实验的最终结果如表 8 所列。

表 8 融合多层次 ResNet 特征

Table 8 Fusion of multi-levels of ResNet features (%)

ResNet 特征融合的层级	$I-AUROC$	$P-AUROC$
layer(2+3)	98.94	98.10
layer(1+2+3)	99.35	98.26

实验结果显示,使用 layer1, layer2 和 layer3 进行特征融合比仅使用 layer2 和 layer3 的性能有提升。其中, $I-AUROC$ 提升了 0.39%, $P-AUROC$ 提升了 0.16%,从而验证了融合

多层次 ResNet 特征对于模型的有效性。

4.4.3 成对相似度和全局相似度有效性验证

本研究将 Patch 特征对之间的成对相似度和全局相似度作为伪标签来进行对比表示学习,使用相似度作为伪标签可以学习到二元标签之外的丰富信息。消融实验超参数采用上述实验的最优设置,将两种相似度作为变量,对同时使用两种相似度以及单一使用成对相似度或全局相似度进行对比,得到的消融实验结果如表 9 所列。

表 9 两种相似度结合的有效性验证

Table 9 Validity verification of combining two similarity measures

(%)			
成对相似度	全局相似度	I-AUROC	P-AUROC
✓	—	99.25	97.95
—	✓	99.15	97.75
✓	✓	99.35	98.26

结果显示,使用成对相似度比使用全局相似度的性能更好,结合两种相似度可以更好地区分正常样本和缺陷样本,从而提升模型的性能。

4.4.4 特征降维的有效性验证

所提模型的特征提取器使用 CoordConv 层,投影层使用线性层。经过 CoordConv 层之后特征维度会减少,因此本研究对特征维度进行消融实验,以观察特征维度是否对相似度计算以及最终的性能有影响。

此消融实验超参数仍采用最优实验设置,将特征维度作为唯一变量,观察检测性能的变化,得到的消融实验结果如表 10 所列。

表 10 特征维度对性能的影响

Table 10 Effect of feature dimensions on performance

(%)		
特征维度	I-AUROC	P-AUROC
224	99.10	97.98
896	99.16	97.91
448(Ours)	99.35	98.26

实验结果显示,特征维度设置为 448 时比设置为 224 和 896 时性能更好,I-AUROC 分别提升了 0.25% 和 0.19%,P-AUROC 分别提升了 0.28% 和 0.35%。因此,选择恰当的特征维度可以提升缺陷检测性能,而过多或者过少的特征维度都会影响实验性能。

4.4.5 压缩算法中簇中心数目的有效性验证

3.2 节中压缩算法初始化生成的内存库质量会影响缺陷检测的性能。本研究初始生成内存库的方法使用聚类生成初始特征中心。在聚类算法中,簇中心数目增加,能够更好地捕捉数据的内在模式,但同时也带来了过拟合的可能;簇中心数目少,又可能导致聚类结果过于简化。本节探讨簇中心数目对最终检测性能的影响。

本研究通过设置超参数 γ 来调整簇中心的数目, γ 越大,则簇中心数目越少,将 γ 作为变量来探究簇中心数目对检测性能的影响。实验结果如表 11 所列, γ 为 4 时,比 γ 为 2 或 8 时的缺陷检测性能更好,因此合理设置簇中心数目可以提升缺陷检测性能。

表 11 γ 对性能的影响

Table 11 Effect of γ on performance

(%)		
γ	I-AUROC	P-AUROC
2	99.31	98.21
8	99.29	98.22
4(Ours)	99.35	98.26

结束语 本研究将对比表示学习和超球面映射结合,来训练面向工业品目标域的特征表示。对比表示模块重新设计了损失函数,使用成对相似度和全局相似度为特征表示层提供伪标签来计算对比损失,同时将正常特征拉近到超球面来共同监督。此外,使用了压缩的内存库,只存储少量特征,减小了空间复杂度以及内存库的尺寸。本研究在 MVTec AD 和 VisA 数据集上进行了实验,结果显示,提出模型在缺陷类型多、缺陷样本短缺的工业品缺陷检测中取得了良好的性能。但是本研究还存在以下不足:1)缺乏对高光场景和复杂工业品中出现遮挡问题的研究,针对这些问题,后续会将同一图像的多个视角结合,利用不同视角的互补信息,进一步提升缺陷检测的性能;2)使用超球面映射方法来聚集正常特征,然而正常特征与缺陷特征并不总是有清晰的界限,因此单一超球面不能准确区分特征。对于该问题,后续研究将参考应对图 5 中特征分界不清晰时所采用的策略,即将 k 近邻重合的个数作为相似度,在后续研究中提出更有效的划分方法。

参考文献

- [1] LUO D L, CAI Y X, YANG Z H, et al. A review of deep learning methods for industrial defect detection [J]. Chinese Science: Information Science, 2022, 52(6): 1002-1039.
- [2] DAVLETSHINA D, MELNYCHUK V, TRAN V, et al. Unsupervised anomaly detection for X-ray images [J]. arXiv: 2001.10883, 2019.
- [3] NGUYEN D T, LOU Z, KLAR M, et al. Anomaly detection with multiple-hypotheses predictions [C] // International Conference on Machine Learning. 2019: 4800-4809.
- [4] SAKURADA M, YAIRI T. Anomaly detection using autoencoders with nonlinear dimensionality reduction [C] // Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis. 2014: 4-11.
- [5] PIDHORSKYI S, ALMOHSEN R, DORETTO G. Generative probabilistic novelty detection with adversarial autoencoders [C] // Advances in Neural Information Processing Systems. 2018: 6822-6833.
- [6] SABOKKROU M, KHALOUEI M, FATHY M, et al. Adversarially learned one-class classifier for novelty detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 3379-3388.
- [7] ZHANG Y, CHEN X W, CHEN M D, et al. Unsupervised industrial surface anomaly detection using contrastive learning generative adversarial networks [J]. Journal of Electronic Measurement and Instrumentation, 2023, 37(10): 193-201.
- [8] RUFF L, VANDERMEULEN R, GOERNITZ N, et al. Deep one-class classification [C] // Proceedings of the International Conference on Machine Learning. 2018: 4393-4402.

- [9] PERERA P, PATEL V M. Learning deep features for one-class classification[J]. *IEEE Transactions on Image Processing*, 2019, 28(11):5450-5463.
- [10] YI J, YOON S. Patch svdd; Patch-level svdd for anomaly detection and segmentation[C]// *Proceedings of the Asian Conference on Computer Vision*. 2020:375-390.
- [11] ROTH K, PEMULA L, ZEPEDA J, et al. Towards total recall in industrial anomaly detection[C]// *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022:14318-14328.
- [12] COHEN N, HOSHEN Y. Sub-image anomaly detection with deep pyramid correspondences[J]. *arXiv*:2005.02357, 2020.
- [13] DEFARD T, SETKOV A, LOESCH A, et al. Padim; a patch distribution modeling framework for anomaly detection and localization[C]// *Proceedings of the International Conference on Pattern Recognition*. 2020:475-489.
- [14] HE S R, ZHANG S J, WANG Y X. Industrial Defect Detection Model Integrating Multi-scale Features[J]. *Journal of Chinese Computer Systems*, 2023, 44(5):1029-1034.
- [15] HUANG P Q Z, DUAN X J, HUANG W W, et al. Asymmetric defect detection method of small sample image based on meta learning[J]. *Journal of Jilin University(Engineering and Technology Edition)*, 2023, 53(1):234-240.
- [16] LEE S, LEE S, SONG B C. Cfa; Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization [J]. *IEEE Access*, 2022, 10:78446-78454.
- [17] WU Z, XIONG Y, YU S X, et al. Unsupervised feature learning via non-parametric instance discrimination[C]// *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018:3733-3742.
- [18] CHEN T, KORNBLITH S, NOROUZI M, et al. A simple framework for contrastive learning of visual representations [C]// *International Conference on Machine Learning*. 2020:597-1607.
- [19] CHEN X, HE K. Exploring simple siamese representation learning[C]// *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021:15750-15758.
- [20] YE M, ZHANG X, YUEN P C, et al. Unsupervised embedding learning via invariant and spreading instance feature[C]// *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019:6210-6219.
- [21] HE K, FAN H, WU Y, et al. Momentum contrast for unsupervised visual representation learning [C] // *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020:9729-9738.
- [22] CHEN T, KORNBLITH S, NOROUZI M, et al. A simple framework for contrastive learning of visual representations [C]// *Proceedings of the International Conference on Machine Learning*. 2020:1597-1601.
- [23] KIM S, KIM D, CHO M, et al. Embedding transfer with label relaxation for improved metric learning[C]// *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021:3967-3976.
- [24] GUO J, LU S, JIA L, et al. ReContrast; Domain-specific anomaly detection via contrastive reconstruction [J]. *arXiv*. 2306.02602, 2023.
- [25] HYUN J, KIM S, JEON G, et al. Reconpatch; Contrastive patch representation learning for industrial anomaly detection [C] // *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024:2052-2061.
- [26] ZHANG Z Q, ZHANG H, WU T Y, et al. Continuous dense normalized flow model for industrial image anomaly detection [J]. *Computer Science*, 2023, 50(12):212-220.
- [27] BERGMANN P, FAUSER M, SATTLEGGER D, et al. MVTec AD—A comprehensive real-world dataset for unsupervised anomaly detection[C]// *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019:9592-9600.
- [28] ZOU Y, JEONG J, PEMULA L, et al. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation [C]// *Proceedings of the European Conference on Computer Vision*. 2022:392-408.
- [29] GUDOVSKIY D, ISHIZAKA S, KOZUKA K. Cflow-ad; Real-time unsupervised anomaly detection with localization via conditional normalizing flows [C] // *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2022:98-107.
- [30] LIU T, LI B, DU X, et al. FAIR; Frequency-aware image restoration for industrial visual anomaly detection[J]. *arXiv*:2309.07068, 2023.
- [31] GUO H W, REN L P, FU J J, et al. Template-guided hierarchical feature restoration for anomaly detection[C]// *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023:6447-6458.
- [32] BAE J, LEE J H, KIM S. PNI; Industrial anomaly detection using position and neighborhood information[C]// *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023:6373-6383.



LUO Hangyu, born in 2000, postgraduate. His main research interests include computer vision and industrial defect detection.



WANG Xiaoping, born in 1965, Ph.D., professor. His main research interests include AI algorithms, deep learning and computer vision.