

# SimpleNet: A Simple Network for Image Anomaly Detection and Localization

## 摘要

本文提出了一个简单且应用友好的网络（称为 SimpleNet），用于检测和定位异常情况。SimpleNet 由四个部分组成：(1) 一个预先训练好的特征提取器，生成局部特征；(2) 一个浅层特征适配器，将局部特征转换到目标域；(3) 一个简单的异常特征生成器，通过向正常特征添加高斯噪声来伪造异常特征；(4) 一个二进制异常鉴别器，区分异常特征和正常特征。

**关键词：**特征提取；特征适配；异常生成；异常鉴别

## 1 引言

在工业场景中，异常检测和定位尤其困难，因为异常样本很少，而且异常可以从细微的变化，如薄薄的划痕到大的结构缺陷，如缺失的零件。目前的方法是以无监督的方式解决这个问题，在训练过程中只使用正常样本。基于重建的方法 [3,6,9]，基于合成的方法 [5,8] 和基于嵌入的方法 [1,6,7] 是解决这个问题的三个主要趋势。

目前的方法是以无监督的方式解决这个问题，在训练过程中只使用正常样本。基于重建的方法 [3,6,9]，基于合成的方法 [5,8] 和基于嵌入的方法 [1,6,7] 是解决这个问题的三个主要趋势。

## 2 相关工作

目前主流的异常检测方法，主要可以分为基于重构的方法，基于嵌入的方法和基于合成的方法三种。本文采用基于嵌入的方法和基于合成的方法相结合，从而解决工业异常检测场景中异常样本少，且异常种类多的问题。

### 2.1 基于重构的方法

一些方法 [3] 利用生成模型，如自动编码器和生成对抗网络 [11] 来编码和重建正常数据。其他方法 [2,5,9] 将异常检测作为一个画图问题，图像中的斑块被随机掩盖。

### 2.2 基于合成的方法

DRÆM [8] 提出了一个网络，该网络以端到端的方式对合成的刚出炉的模式进行判别训练。CutPaste [4] 提出了一个简单的策略来生成用于异常检测的合成异常点，该策略在大图像

的随机位置剪切一个图像补丁并进行粘贴。

### 2.3 基于嵌入的方法

这些方法将正常特征嵌入到一个压缩的空间。异常特征在嵌入空间中远离正常集群。典型的方法 [1,2,6,7] 利用在 ImageNet 上预先训练好的网络进行特征提取。通过预训练的模型, PaDiM [1] 通过多变量高斯分布嵌入提取的异常补丁特征。PatchCore [6] 使用名义斑块特征的最大代表存储库。在测试中采用 Mahalanobis 距离或最大特征距离对输入特征进行评分。

## 3 本文方法

### 3.1 本文方法概述

SimpleNet 的整体架构如图 1所示:

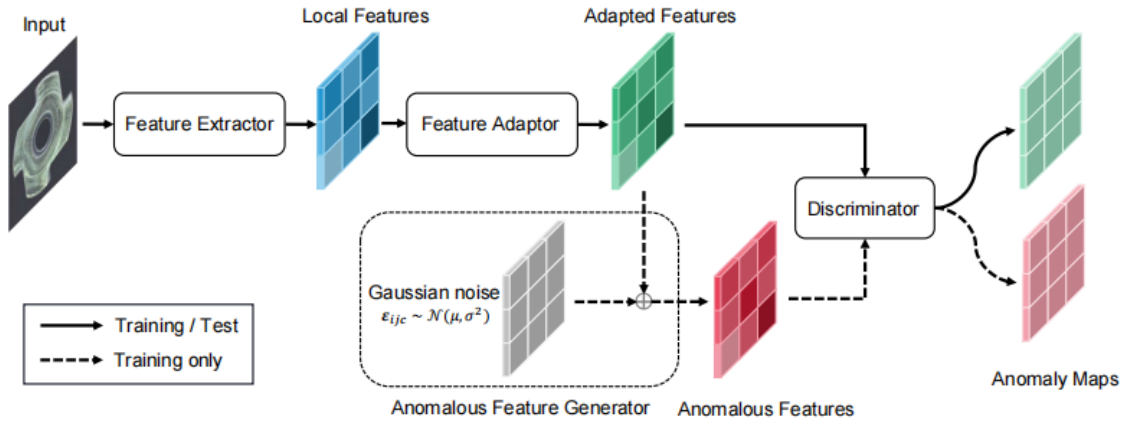


图 1. SimpleNet 架构示意图

### 3.2 特征提取

对于训练集和测试集中的所有图像, 预训练网络从不同层次提取特征。来自层次 1 的特征图表示为:

$$\phi^{l,i} \sim \phi(x_i) \in R^{H_l \times W_l \times C_l} \quad (1)$$

其中  $H_l$ ,  $W_l$  和  $C_l$  代表特征图的高度, 宽度, 通道数。对于在位置  $(h, w)$  上的条目, 其具有 patch 大小  $p$  的邻域被定义为:

$$N_p^{(h,w)} = \{(h', y') \mid h' \in [h - \lfloor p/2 \rfloor, \dots, h + \lfloor p/2 \rfloor], y' \in [w - \lfloor p/2 \rfloor, \dots, w + \lfloor p/2 \rfloor]\} \quad (2)$$

用聚合函数  $f_{agg}$  聚合邻域内的特征, 结果是本地特征, 可以表示为:

$$z_{h,w}^{l,i} = f_{agg} \left( \left\{ \phi_{h',w'}^{l,i} \mid (h', y') \in N_p^{h,w} \right\} \right) \quad (3)$$

为了融合不同层次的特征，将特征图调整为相同大小  $H_0, W_0$ ，并进行简单的连接。以上式子可以简述为：

$$o^i = F_\phi(x_i) \quad (4)$$

其中， $F_\phi$  为特征提取器。

### 3.3 特征适配

使用特征适配  $G_{theta}$  将局部特征  $o_{h,w}$  映射为  $q_{h,w}$ ：

$$q_{h,w}^i = G_\theta(o_{h,w}^i) \quad (5)$$

特征适配器可以使用简单的神经块，如全连接层或 MLP, SimpleNet 使用一个全连接层作为特征适配器。

### 3.4 异常特征生成

对正常特征添加高斯噪声得到异常特征，并对噪声进行采样：

$$q_{h,w}^{i-} = q_{h,w}^i + \epsilon \quad (6)$$

### 3.5 异常鉴别

对每个位置  $(h, w)$  进行判断，对正常特征期望正输出，对异常特征期望负输出。本文使用两层的 MLP 作为鉴别器。

### 3.6 损失函数和训练

本文使用简单的  $L_1$  截断函数作为损失函数：

$$l_{h,w}^i = \max(0, th^+ - D_\psi(q_{h,w}^i)) + \max(0, -th^- + D_\psi(q_{h,w}^{i-})) \quad (7)$$

其中  $th^+$  和  $th^-$  是防止过拟合的截断项，分别被设置为 0.5 和 -0.5。训练目标为：

$$L = \min_{\theta, \psi} \sum_{h,w} \frac{l_{h,w}^i}{H_0 * W_0} \quad (8)$$

SimpleNet 在实验部分对截断  $L_1$  损失函数与交叉熵损失进行实验评估。

### 3.7 推理和异常得分函数

异常得分由鉴别器提供，在推理过程中异常定位的异常图被定义为：

$$S_{AL}(x_i) := \{s_{h,w}^i \mid (h, w) \in W_0 \times H_0\} \quad (9)$$

以异常图的最大分数作为每幅图像的异常检测分数。

## 4 复现细节

### 4.1 与已有开源代码对比

SimpleNet 的作者在 github 上已公布源代码。但源代码中仅包含训练模型的部分。本文在 SimpleNet 源代码的基础上进行修改并添加了推理模块。

推理模块通过读取训练生成的模型，将模型参数导入网络结构，对输入图像进行异常检测，并输出异常检测生成的热图和推理所需的时间。

### 4.2 实验环境搭建

- python==3.8
- torch==1.12.1
- torchvision==0.13.1
- numpy==1.22.4
- opencv-python==4.5.1

### 4.3 使用说明

对于训练部分，源代码已经给出，根据源代码的自述文件即可进行模型的训练。此处重点介绍推理部分的使用。

本文已将推理模块 inference.py 加入源代码的文件夹中。在源代码目录下运行指令 `python inference.py` 即可进行测试。

训练的模型按照 SimpleNet 自述文件中的路径存放即可。测试的图片需要按照路径“./pictures/pic.png”存放。运行得到的热图存放在“./results/”中，推理一张图片的时间会直接显示在测试界面。

### 4.4 创新点

在源代码的基础上加入推理部分，通过读取训练生成的模型，对测试图像进行异常检测，并输出异常检测生成的热图和推理所需的时间。

## 5 实验结果分析

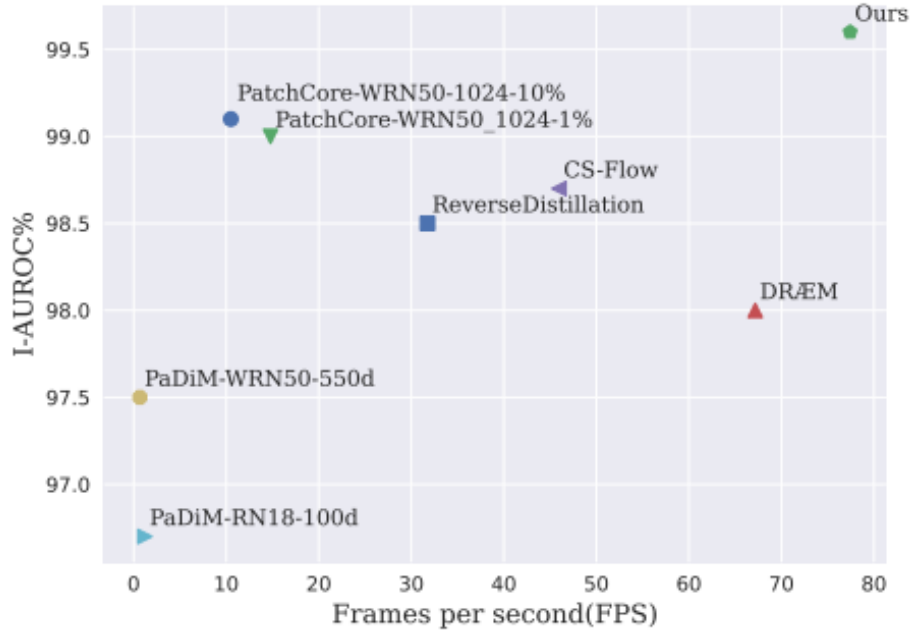


图 2. 不同方法在 MVTecAD 基准测试上的推理速度 (FPS) 与 I-AUROC

以上方法在相同的硬件上进行测量, Nvidia GeForce GTX 3080ti GPU 和 Intel(R)Xeon(R) CPU E5-2680 v3@2.5GHZ。这清楚地表明, SimpleNet 同时取得了最好的性能和最快的速度。

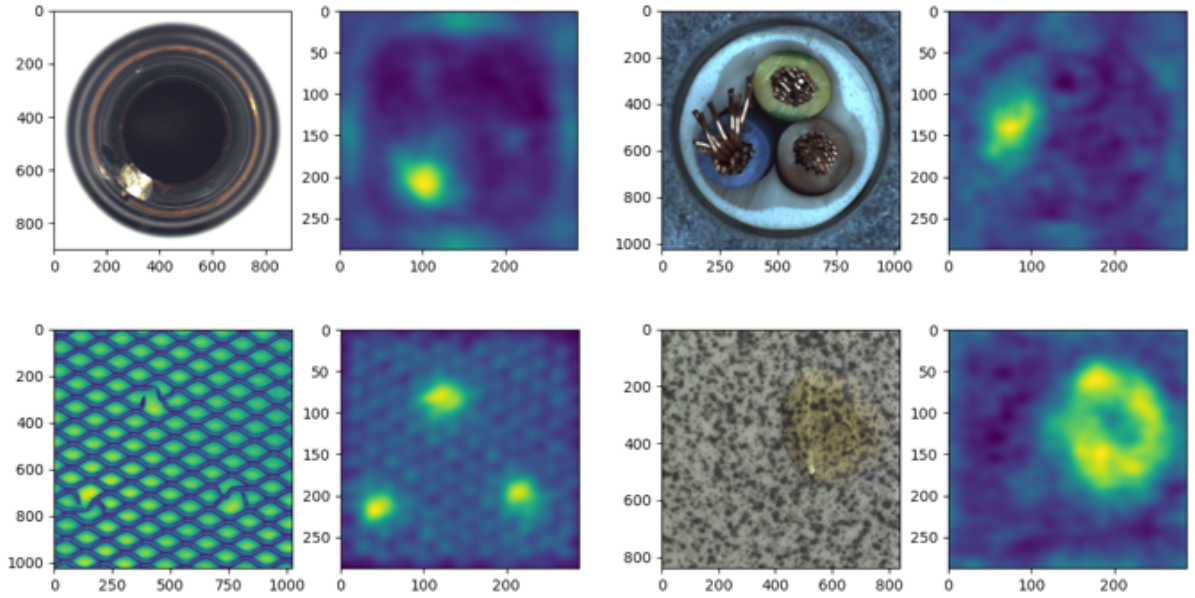


图 3. 异常图片 (左) 与推理得到的热图 (右)

上图为选取了 MVTecAD 中 bottle, cable, grid, tile 种类的部分异常图像进行异常检测, 从得到的热图中可以看出, 异常基本被检测出来, 本文训练的模型取得了不错的异常检测效果。

Type	Reconstruction-based		Synthesizing-based		Embedding-based					Ours
Model	AE-SSIM	RIAD	DRÆM	CutPaste	CS-Flow	PaDiM	RevDist	PatchCore	SimpleNet	SimpleNet
Carpet	87/64.7	84.2/96.3	97.0/95.5	93.9/98.3	100/-	99.8/99.1	98.9/98.9	98.7/99.0	99.7/98.2	99.3/98.0
Grid	94/84.9	99.6/98.8	99.9/99.7	100/97.5	99.0/-	96.7/97.3	100/99.3	98.2/98.7	99.7/98.8	99.7/96.6
Leather	78/56.1	100/99.4	100/98.6	100/99.5	100/-	100/99.2	100/99.4	100/99.3	100/99.2	100.0/99.2
Tile	59/17.5	98.7/89.1	99.6/99.2	94.6/90.5	100/-	98.1/94.1	99.3/95.6	98.7/95.6	99.8/97.0	99.9/93.2
Wood	73/60.3	93.0/85.8	99.1/96.4	99.1/95.5	100/-	99.2/94.9	99.2/95.3	99.2/95.0	100/94.5	100.0/94.0
Avg. Text.	78/56.7	95.1/93.9	99.1/97.9	97.5/96.3	99.8/-	95.5/96.9	99.5/97.7	99.0/97.5	99.8/97.5	99.8/96.2
Bottle	93/83.4	99.9/98.4	99.2/99.1	98.2/97.6	99.8/-	99.1/98.3	100/98.7	100/98.6	100/98.0	100.0/98.0
Cable	82/47.8	81.9/84.2	91.8/94.7	81.2/90.0	99.1/-	97.1/96.7	95.0/97.4	99.5/98.4	99.9/97.6	100.0/97.3
Capsule	94/86.0	88.4/92.8	98.5/94.3	98.2/97.4	97.1/-	87.5/98.5	96.3/98.7	98.1/98.8	97.7/98.9	97.8/98.9
Hazelhut	97/91.6	83.3/96.1	100/99.7	98.3/97.3	99.6/-	99.4/98.2	99.9/98.9	100/98.7	100/97.9	100.0/97.8
Metal Nut	89/60.3	88.5/92.5	98.7/99.5	99.9/93.1	99.1/-	96.2/97.2	100/97.3	100/98.4	100/98.8	100.0/98.8
Pill	91/83.0	83.8/95.7	98.9/97.6	94.9/95.7	98.6/-	90.1/95.7	96.6/98.2	96.6/97.4	99.0/98.6	99.0/98.4
Screw	96/88.7	84.5/98.8	93.9/97.6	88.7/96.7	97.6/-	97.5/98.5	97.0/99.6	98.1/99.4	98.2/99.3	98.1/99.2
Toothbrush	92/78.4	100/98.9	100/98.1	99.4/98.1	91.9/-	100/98.8	99.5/99.1	100/98.7	99.7/98.5	100.0/98.6
Transistor	90/72.5	90.9/87.7	93.1/90.9	96.1/93.0	99.3/-	94.4/97.5	96.7/92.5	100/96.3	100/97.6	100.0/97.0
Zipper	88/66.5	98.1/97.8	100/98.8	99.9/99.3	99.7/-	98.6/98.5	98.5/98.2	99.4/98.8	99.9/98.9	99.9/98.6
Avg. Obj.	91/75.8	89.9/94.3	97.4/97.0	95.5/95.8	98.2/-	96.0/97.8	98/97.9	99.2/98.4	99.5/98.4	99.5/98.3
Average	87/69.4	91.7/94.2	98.0/97.3	96.1/96.0	98.7/-	95.8/97.5	98.5/97.8	99.1/98.1	99.6/98.1	99.6/97.6

表 1. 本文复现的 SimpleNet 在 MVTec AD 上与原 SimpleNet 以及 SotA 的比较, 左边为 I-AUROC, 右边为 P-AUROC

上表包括了 SimpleNet 与目前最先进的异常检测技术在 MVTecAD 数据集上检测指标的对比, 以及本文复现出的 SimpleNet 在该数据集上的异常检测指标。从表中可以看出, 复现出的模型精度基本达到了原论文公开的异常检测指标, 说明本文得到了性能比较好的 SimpleNet 模型。

## 6 总结与展望

本文首先对 SimpleNet 模型进行了大致介绍, 并且和目前已有的异常检测方法进行对比。接着对 SimpleNet 使用的方法进行详细阐述, 该网络模型主要由四个部分组成, 分别是特征提取器, 用于提取特征; 特征适配器, 用于减少转移域的误差; 异常特征生成器, 通过向正常特征添加高斯噪声生成异常特征; 异常鉴别器, 对异常特征进行判断。然后对 SimpleNet 的实验进行介绍, 展示了原论文以及本文的工作结果以及本文的创新点。

目前文章主要的不足在于本文并没有对 SimpleNet 的网络结构进行更改, 只是在源代码的基础上添加了推理模块, 使之能够对异常图像进行检测, 而没有在异常检测的精度或者速度上有所突破。未来的研究可能会在 SimpleNet 的网络架构上进行优化创新, 以求达到更高的精度或者更快的速度。

## 参考文献

- [1] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489. Springer, 2021.

- [2] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9737–9746, 2022.
- [3] 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.
- [4] 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.
- [5] Nicolae-Cătălin Ristea, Neelu Madan, Radu Tudor Ionescu, Kamal Nasrollahi, Fahad Shahbaz Khan, Thomas B Moeslund, and Mubarak Shah. Self-supervised predictive convolutional attentive block for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13576–13586, 2022.
- [6] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022.
- [7] Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. Fully convolutional cross-scale-flows for image-based defect detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1088–1097, 2022.
- [8] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8330–8339, 2021.
- [9] Vitjan Zavrtanik, Matej Kristan, and Danijel Skočaj. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112:107706, 2021.