

# 基于改进YOLOv8模型的PCB电路板缺陷检测方法研究

朱凯<sup>1\*</sup>, 李凌霄<sup>1#</sup>, 何宇楠<sup>2</sup>

<sup>1</sup>重庆理工大学理学院, 重庆

<sup>2</sup>重庆理工大学数学科学研究中心, 重庆

收稿日期: 2024年1月26日; 录用日期: 2024年2月22日; 发布日期: 2024年2月29日

## 摘要

针对现有PCB电路板缺陷检测方法普遍存在的准确率低、处理速度慢、适应性差等问题, 本文设计提出了一种基于YOLOv8的改进电路板缺陷检测模型YOLOv8-PCB。该模型通过引入注意力机制, 并结合数据增强和多尺度训练策略, 能够有效提升缺陷检测准确率和处理速度。与此同时, 考虑到PCB电路板的背景信息比较单一, 通用目标检测模型性能受限的问题, 本文进一步设计采用了一种LSKNet注意力机制, 通过在特征提取时自适应动态调整目标感受视野, 从而提升模型对小缺陷的目标检测能力。通过各项试验结果表明, 本文提出的算法模型在测试数据集下的平均准确率、召回率分别为95.0%和93.3%, 分别优于原始YOLOv8算法91.8%和90.9%。且模型参数量更小, 在提升检测性能的同时能够兼顾算法计算效率, 因此可以快速地、准确地实现PCB电路板的缺陷检测, 为智慧工厂、智能装备等领域提供技术支持。

## 关键词

PCB, YOLOv8s, LSKNet, 注意力机制, 目标检测, 机器学习

# Research on PCB Circuit Board Defect Detection Method Based on Improved YOLOv8 Model

Kai Zhu<sup>1\*</sup>, Lingxiao Li<sup>1#</sup>, Yunan He<sup>2</sup>

<sup>1</sup>College of Science, Chongqing University of Technology, Chongqing

<sup>2</sup>Research Center for Mathematical Sciences, Chongqing University of Technology, Chongqing

Received: Jan. 26<sup>th</sup>, 2024; accepted: Feb. 22<sup>nd</sup>, 2024; published: Feb. 29<sup>th</sup>, 2024

\*第一作者。

#通讯作者。

文章引用: 朱凯, 李凌霄, 何宇楠. 基于改进 YOLOv8 模型的 PCB 电路板缺陷检测方法研究[J]. 计算机科学与应用, 2024, 14(2): 501-516. DOI: 10.12677/csa.2024.142050

## Abstract

Aiming at the problems of low accuracy, slow processing speed and poor adaptability of existing PCB circuit board defect detection methods, this paper designed and proposed an improved circuit board defect detection model, YOLOV8-PCB, based on YOLOv8. By introducing an attention mechanism, combining data enhancement and a multi-scale training strategy, the model can effectively improve the accuracy and processing speed of defect detection. At the same time, considering that the background information of the PCB is relatively simple and the performance of the general target detection model is limited, this paper further designed and adopted a LSKNet attention mechanism to improve the model's target detection ability for small defects by dynamically adjusting the target perception field during feature extraction. The experimental results show that the average accuracy and recall rate of the proposed algorithm model under the test data set are 95.0% and 93.3%, respectively, which are better than 91.8% and 90.9% of the original YOLOv8 algorithm. Moreover, the number of model parameters is smaller, and the algorithm calculation efficiency can be taken into account while improving the detection performance, so the defect detection of PCB circuit boards can be quickly and accurately realized, providing technical support for smart factories, intelligent equipment and other fields.

## Keywords

PCB, YOLOv8s, LSKNet, Attention Mechanism, Object Detection, Machine Learning

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

印刷电路板(Printed Circuit Board)简称 PCB, 依靠其高度集成、轻量化、成本低和稳定性高等优势, 广泛地应用于电子产品领域。然而, 在制造过程中, 由于各种因素的影响, 如设备故障、操作失误、设计缺陷等, PCB 电路板上可能存在各种各样的缺陷, 例如短路、缺失孔、焊接不良、线路损伤等。这些缺陷如果未能及时检测和处理, 会严重影响 PCB 电路板的性能、质量和安全。

因此, 如何及时有效地对 PCB 电路板进行缺陷检测极为重要。早期的 PCB 缺陷检测主要依靠人工目测和在线仪器检测, 但这些方法通常存在成本高、效率低、误检率高等缺点, 且部分检测方法会接触到 PCB 电路板, 容易造成二次损坏, 使得目前检测难以满足现代 PCB 电路板生产效率高、性能好的检测要求[1]。随着计算机技术的发展, 深度学习在缺陷检测领域被广泛应用且取得了显著的成果。

基于深度学习的目标检测算法在步骤上包括双阶段(two-stage)和单阶段(one-stage)两种目标检测框架[2]。在双阶段的检测算法中, R-CNN [3]是一种基于区域的卷积神经网络, 通过使用选择性搜索算法来对输入图像提取潜在的候选区域。然后, 对每个候选区域进行特征提取, 并使用支持向量机(SVM)进行分类。这样就将目标检测问题转化为一个候选区域分类的任务。而 Faster R-CNN [4]在 R-CNN 的基础上进行了改进, 引入了区域建议网络(Region Proposal Network, RPN), 使得整个目标检测系统实现端到端地进行训练。但由于候选区域生成过程可能存在一些限制, 导致漏检或定位不准确。

而在单阶段算法中, SSD (Single Shot MultiBox Detector) [5]采用了基于卷积神经网络(CNN)的特征提取器, 并在多个不同尺度的特征图上进行目标检测。它通过在不同层级的特征图上应用不同大小的卷积

核来检测不同尺寸的目标。这种多尺度的检测策略使得 SSD 能够有效地检测不同大小的目标。而 YOLO (You Only Look Once) [6]也是一种单阶段的目标检测算法。与 SSD 不同的是, YOLO 是将目标检测问题转化为一个回归问题。它将输入图像分成一个固定大小的网格,并在每个网格单元中预测目标的边界框和类别。

YOLO 算法凭借检测速度快,且只需要一次前向传播就可以得到所有目标的检测结果的特点,实现了对目标的实时检测。YOLO 作为单阶段模型的代表之一,相较于更早提出的两阶段目标检测算法,不仅拥有更快的预测速度;对于背景图像(非物体)中的部分被包含在候选框的情况误检率更低,还拥有更好的算法通用性。以上这些特性,都使 YOLO 系列模型成为工业目标检测场景首选的算法。

### YOLOv1-YOLOv7

在各种物体检测算法中, YOLO (You Only Look Once)框架因其在速度和准确性方面的显著平衡而脱颖而出,能够快速、可靠地识别图像中的物体。自成立以来, YOLO 系列已经经历了多次迭代,见图 1,每次都是在以前的版本基础上解决局限性并提高性能[7]。

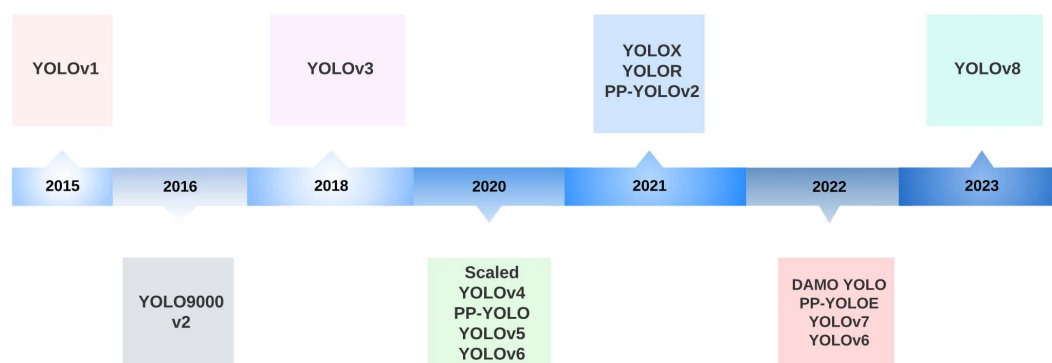


Figure 1. History of the YOLO series

图 1. YOLO 系列发展历史

**YOLOv1:** YOLOv1 采用全卷积神经网络结构,将目标检测问题转化为单次前向传播的回归问题。其核心思想就是将整张图片作为网络的输入(类似于 Faster-RCNN),直接在输出层对 BBox 的位置和类别进行回归。

**YOLOv2 [7]:** 在 yolo1 的基础上进行了改进,提出了一种检测与分类联合训练方法,这种方法使得模型可以检测超过 9000 种类别。

**YOLOv3:** 它引入了三种不同尺度的检测,采用了三种不同尺寸的检测核。这种改进的架构在对小物体进行预测时,使用更加精细的边界框,提高预测效果。

**YOLOv4:** YOLOv4 采用了一系列技术改进,包括使用更大和更深的模型 Darknet-53,引入 CSPDarknet53 和 PANet 等结构来提高特征表示能力。同时引入了多尺度推理、Mosaic 数据增强、CIOU 损失函数等技术来提高检测性能。

**YOLOv5 [8]:** YOLOv5 融入了 Ultralytics 的一种称为 AutoAnchor 的算法。该预训练工具检查并调整锚框,以使其适合于数据集和训练设置。它使用更加轻量级的网络结构作为主干网络,并使其更加易于训练,通过 Pytorch 进行开发以便在 GPU 和 CPU 上更快地运行。

**YOLOv6:** YOLOv6 采用了高效的主干网络(使用 RepVGG 或 CSPStackRep 块)、PAN 拓扑的 neck 和高效解耦头部,这些组件相互协作以实现更快、更准确的对象检测。

YOLOv7 [9]: YOLOv7 在架构上使用了扩展高效层聚合网络(E-ELAN)策略, 这种策略通过洗牌和合并不同组特征的 cardinality, 增强了网络的学习特征的能力, 提高准确率。通过基于串联的模型缩放, 减少硬件资源的使用, 保持模型的最佳结构。

## 2. YOLOv8 算法概述

Ultralytics 于 2023 年发布了 YOLOv8 [10]算法, 与之前的网络相比, YOLOv8 在减少网络参数数量的同时, 提高了检测精度和实时性。根据官方文档, YOLOv8 是一种(SOTA)模型, 在原有的 YOLO 版本基础上, 引入新的功能, 改进网络结构, 提升了模型的性能与灵活性。和 YOLOv5 一样, YOLOv8 也基于缩放系数也供了 n/s/m/l/x 五种尺度的模型, 不同模型大小用于满足不同场景需求。

YOLOv8 的具体创新包括一个新的骨干网络、一个新的 Ancher-Free 检测头和一个新的损失函数。骨干网络和 Neck 部分借鉴了 YOLOv7 的 E-ELAN 策略, 将 YOLOv5 中的 C3 结构替换为梯度流更丰富的 C2f 模块。增强特征的表达能力, 可以更好地捕捉图像中的细节信息。Head 部分相较于 YOLOv5, 耦合头换成了目前主流的解耦头结构, 将分类和检测头分离, 同时将 Ancher-Base 换成 Ancher-Free。简化算法流程的同时提高检测效率。损失函数的计算方面, YOLOv8 采用了 TaskAlignedAssigner 正样本分配策略, 同时使用 Distribution Focal Loss 损失计算函数, 以提高目标检测任务与分类任务时的性能表现, 使模型更好地适应复杂的数据分布和类别不平衡的情况。

### YOLOv8 模型结构

YOLOv8 模型结构主要由 Backbone、Neck、Head 三个部分组成, 其主要网络结构见图 2。

对于 Backbone 部分: 其主体是 CSPDarkNet 结构, 可以在计算效率与准确率之间取得良好的平衡。这个部分负责提取图像的特征信息。主要由 Conv、C2f 和 SPPF 三种层级组成。C2f 是将卷积特征图转换为全连接特征向量的层, SPPF 则是空间金字塔池化网络。

对于 Neck 部分: 指的是连接在骨干网络后面的部分, 用于进一步提取特征和融合不同尺度的特征图。Neck 部分是由一系列 Upsample、Concat 和 C2f 层组成的, 主要作用是将不同层级的特征图进行上采样、拼接和进一步处理, 以便 Head 部分更好地进行预测。

对于 Head 部分: 主要功能则是对特征进行处理和预测, 通过在不同层级上提取特征, 提高模型预测的准确度。主要由 Conv、Concat、C2f 和 Detect 层组成。Conv 层用于进一步提取特征信息, Concat 层则用于将不同尺度的特征图进行拼接, 以便更好地适应不同尺度的目标。C2f 层将卷积特征图转换为全连接特征向量, 并且可以通过 Dropout 操作来防止过拟合。Detect 层是检测层, 它在特征图上执行目标检测操作, 并输出检测结果, 包括目标的位置和类别等信息。

## 3. 基于 YOLOv8-PCB 的电路板缺陷检测算法

由于 PCB 电路板的检测目标尺寸较小, 且数据集数量有限, 因此在利用通用的 YOLOv8 模型进行缺陷检测时, 需要根据样本数据的特征分布进行针对性的改进优化。为了尽可能减少模型复杂度, 提高算法效率, 本文以参数量最少的原始 YOLOv8s 模型作为基准模型, 并在此基础上通过引入 LSK 注意力机制并调整模型结构等手段, 最后能够有效地检测出 PCB 图像中的各种缺陷, 包括短路、开路、伪铜等。下面对算法模型各部分进行详细说明。

### 3.1. LSKNet 注意力机制

注意力机制(Attention Mechanism)源于对人类视觉的研究。而计算机视觉中的注意力机制是指在图像识别、目标检测、语义分割等任务中, 通过对输入数据中的重要区域进行加权处理。通过引入注意力

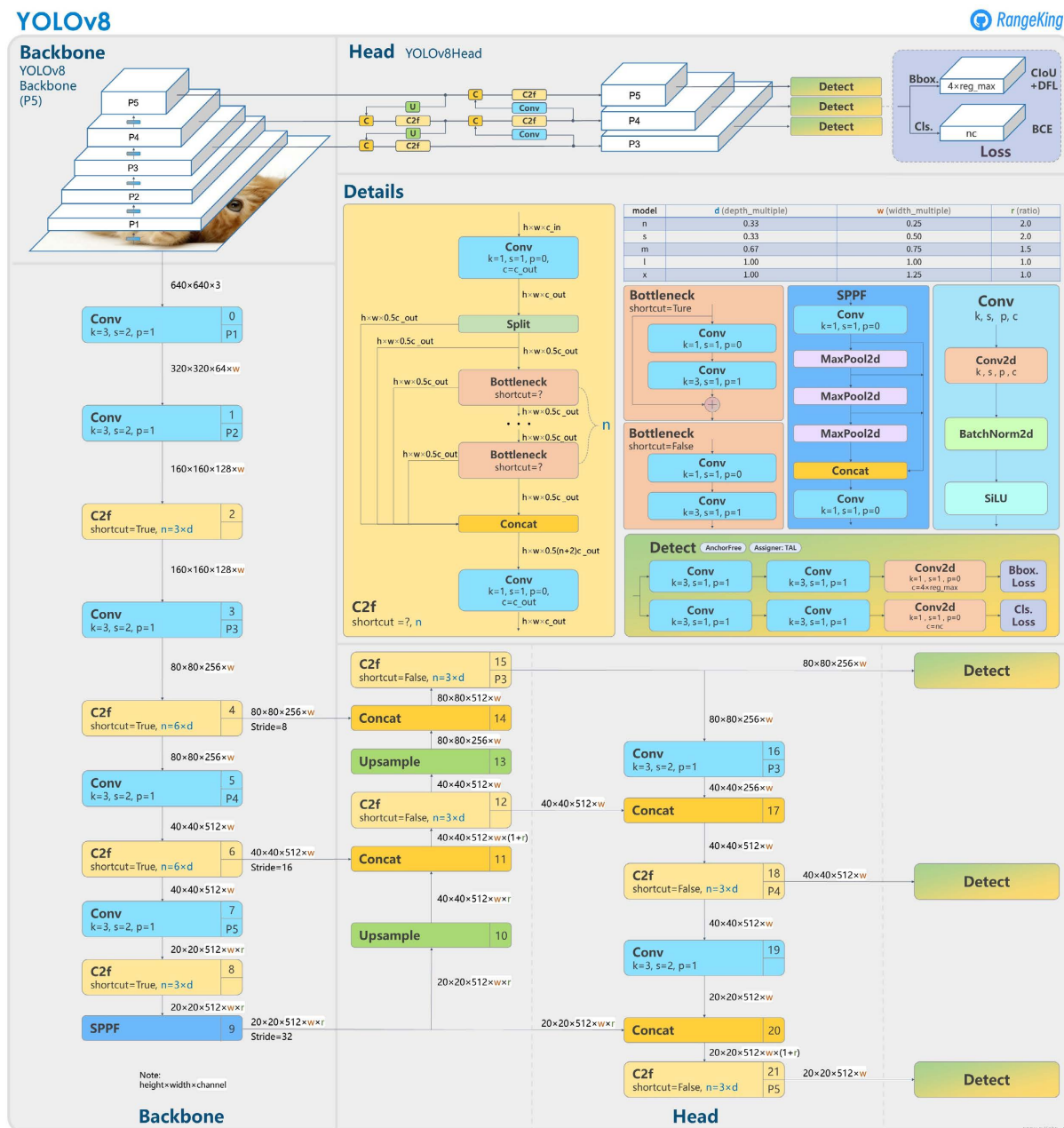
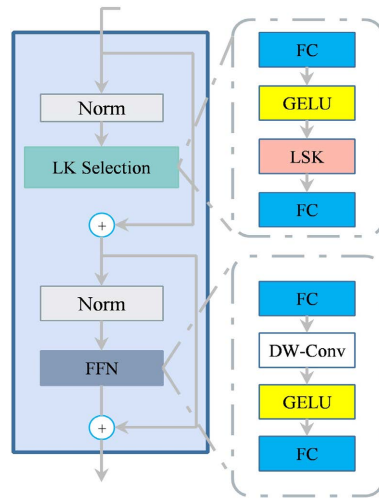


Figure 2. Network structure of the YOLOv8 mode

图 2. YOLOv8 模型网络结构图

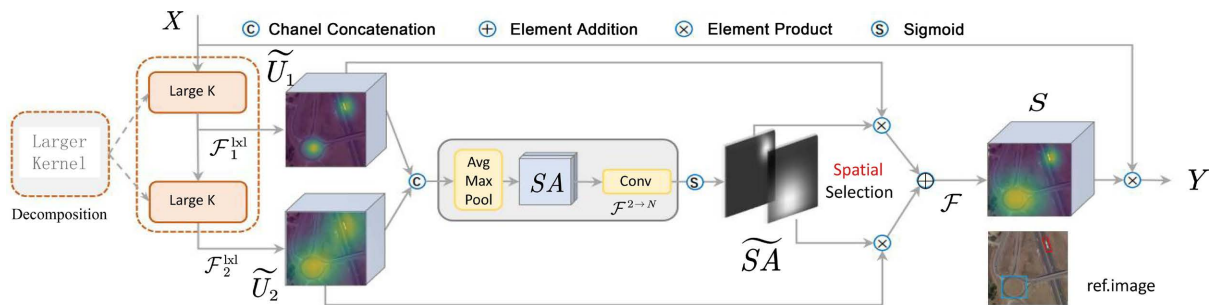
机制，对输入数据的不同部分赋予不同的权重，能有效地帮助网络模型对输入信息进行学习，使其能够更加关注重要的信息，从而提高模型的性能和准确率，同时也有助于避免过拟合的情况发生，提高模型的鲁棒性[11]。

本文在通用注意力机制模块的基础上，引入了一种叫 LSK (Large Selective Kernel) 的新型注意力结构 [12]，构成了 LSKNet 注意力机制模块。见图 3，LSKNet 结构简单，但能够获得优异的检测性能，可以动态地调整其大空间感受野，从而更好地建模遥感场景中各种物体的测距的场景。该模块也能在遥感目标检测等领域探索大的、有选择性的卷积核机制，可以广泛应用于动态的上下文特征信息提取当中。



**Figure 3.** Illustration of the LSKNet Block  
**图 3.** LSKNet Block 图示

LSKNet Block 是主干网络中的一个可重复堆叠的模块，每个 LSKNet Block 包括两个残差子块，即大核选择子块(Large Kernel Selection, LK Selection)和前馈网络子块(Feed-forward Network, FFN)，见图 3。LK Selection 子块根据需要动态地调整网络的感受野，FFN 子块用于通道混合和特征细化，由一个全连接层、一个深度卷积、一个 GELU 激活和第二个全连接层组成。



**Figure 4.** Concept art of the LSK Module  
**图 4.** LSK Module 模块概念图

LK Selection 子模块中(见图 4 橙色块)嵌入了核心模块 LSK Module，而 LSK Module 是由一个大核卷积序列(large kernel convolutions)和一个空间核选择机制(spatial kernel selection mechanism)组成(见图 3)。

由于要对一系列的多个长期上下文信息进行模型的自适应选择。因此通过显式地分解一个大的核卷积并将其分解为一个大的深度卷积序列来构造一个更大的核卷积。具体来说，该序列中第  $i$  个深度卷积的核大小  $k$ 、扩张率  $d$  和接受场  $RF$  的展开被细化如下：

$$k_{i-1} \leq k_i; d_1 = 1, d_{i-1} < d_i \leq RF_{i-1}, \tag{1}$$

$$RF_1 = k_1, RF_i = d_i(k_i - 1) + RF_{i-1}. \tag{2}$$

以上公式中核的大小与扩张率的增加确保了感受野有足够快的扩展，为保证扩张卷积不会在特征图之间引入空隙，对扩张率设定了一个上限，这样的设计有两个主要优点：首先，它能够明确地生成具有不同大感受野的多个特征，从而更容易进行后续内核的选择；其次，与简单地应用一个较大的核相比，

顺序分解技术更加高效和有效。最后，我们还采用了空间选择机制，以提高网络关注检测目标最相关的空间背景区域的能力。图4详细展示了LSK模块的概念，直观地展示了大选择核如何通过自适应地收集不同物体的相应大感受野而发挥作用，从而有效地将低了模型的参数量。

### 3.2. LOSS 损失函数

损失函数的设计是目标检测算法的重要组成部分，本文设计的YOLOv8-PCB算法的损失函数同样由分类损失VFL和回归损失CIOU + DFL两部分的三个损失函数加权组合而成，从而对缺陷目标进行更好地比较筛选。

#### 3.2.1. VFL

YOLOv5/v7均使用BCE(二元交叉熵)作为分类损失，每类别判断“是否为此类”，并输出置信度。

在YOLOv5中，由于存在对象损失，反向传播时只选择BCE分类输出的“置信度分数”中最大值作为置信度最高的类别，并直接输出。而在YOLOv8中，由于去除了对象损失，输出中也不再包含“对象置信度”，而是直接输出各个类别的“置信度分数”，然后从中选取最大值作为该锚框的“置信度”。这样的设计简化了模型输出的结构，并提高了算法的效率。具体计算如下所示：

$$VFL(p, q) = \begin{cases} -q(q \log(p) + (1-q) \log(1-p)) & q > 0 \\ -\alpha p^q \log(1-p) & q = 0 \end{cases} \quad (3)$$

$q$  是 label，正样本时候  $q$  为预测框和真实框的  $IoU$ ，负样本时候  $q = 0$ ，当预测结果为正样本时候没有采用 FL，而是普通的 BCE，只不过多了一个自适应  $IoU$  加权，用于突出主样本。而为负样本时候就是标准的 FL 了。可以明显发现其主要特点是正负样本非对称加权、突出正样本为主样本。

#### 3.2.2. CIOU

边界框回归损失函数在目标检测中至关重要。通过学习预测边界框的位置，模型可以尽可能地接近真实的边界框，从而提供检测目标的精确定位和区域的关键信息。这一过程不仅有助于提高检测精度，而且还使得模型能够更好地应对各种目标形状和大小的变化。YOLOv8-PCB 采用 CIOU 作为边界框回归损失函数，如下式(4)~(6)所示

$$CIoU = 1 - IoU + \frac{(x - x_{gt})^2 + (y - y_{gt})^2}{W_g^2 + H_g^2} + \alpha v, \quad (4)$$

$$\alpha = \frac{v}{L_{IoU} + v}, \quad (5)$$

$$v = \frac{4}{\pi^2} \left( \tan^{-1} \frac{w}{h} - \tan^{-1} \frac{w_{gt}}{h_{gt}} \right)^2, \quad (6)$$

其中  $IoU$  计算的是“预测的边框”和“真实的边框”的交叠率，即它们的交集和并集的比值。其中  $x, y$  为预测框的中心点坐标， $x_{gt}, y_{gt}$  为真实框的预测框的中心点坐标， $W_g$  和  $H_g$  是预测框与真实框构成的最小矩形框的宽高值， $\alpha$  为超参数， $w$  和  $h$  分别为预测框的宽高值。

#### 3.2.3. DFL

DFL 是一种优化方法，能够使网络更快地聚焦于目标位置  $y$  附近的值，并增大它们的概率。该方法通过使用交叉熵形式的损失函数，对与标签  $y$  最接近的左右两个位置的概率进行优化。这样可以使网络更快地集中学习目标位置周围的分布情况。

$$DFL(S_i, S_{i+1}) = -big((y_{i+1} - y) \log(S_i) + (y - y_i) \log(S_{i+1}))big. \quad (7)$$

换句话说,  $DFL$  的目标是让学习到的分布在理论上接近真实浮点坐标的附近, 并以线性插值的方式获得距离左右整数坐标的权重。通过引入  $DFL$ , 模型可以更加准确地定位目标, 提高目标检测的性能和效率。

### 3.3. YOLOv8-PCB 模型的结构设计

根据上述内容, 本节将利用设计引入的 LSK 注意力机制模块, 在原 YOLOv8 模型的 Backbone、Neck 以及 Global 全局不同位置处进行构造添加, 探索性能最优的改造优化方式, 最后利用 3.2 节的损失函数进行 YOLOv8-PCB 模型训练与预测输出。

#### 3.3.1. YOLOv8 Backbone 部分添加 LSK 模块

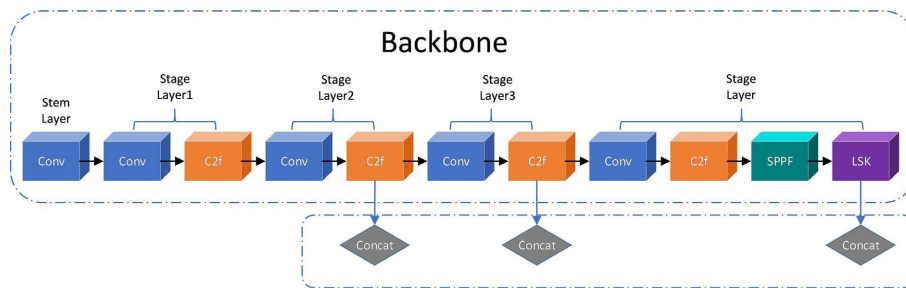


Figure 5. Backbone improved structure diagram  
图 5. Backbone 改进结构图

主干网络(Backbone)通过一系列的卷积层和池化层对输入图像进行多次下采样操作, 逐渐减小特征图的尺寸。这种逐渐降低特征图分辨率的操作有助于捕获不同尺度的目标信息, 使模型具备多尺度感知能力。

在 YOLOv8 的 Backbone 中, SPPF 模块可以对输入图像进行不同尺度的池化操作, 得到多个尺度的特征图, 这些特征图可以捕捉到不同大小的物体或缺陷。在 SPPF 模块后添加 LSKNet Block (见图 5, 紫色模块)可以更好的利用提取到的特征, 使网络更加关注缺陷部分, 以便更加准确地检测 PCB 电路板缺陷。

#### 3.3.2. YOLOv8 Neck 部分添加 LSK 模块

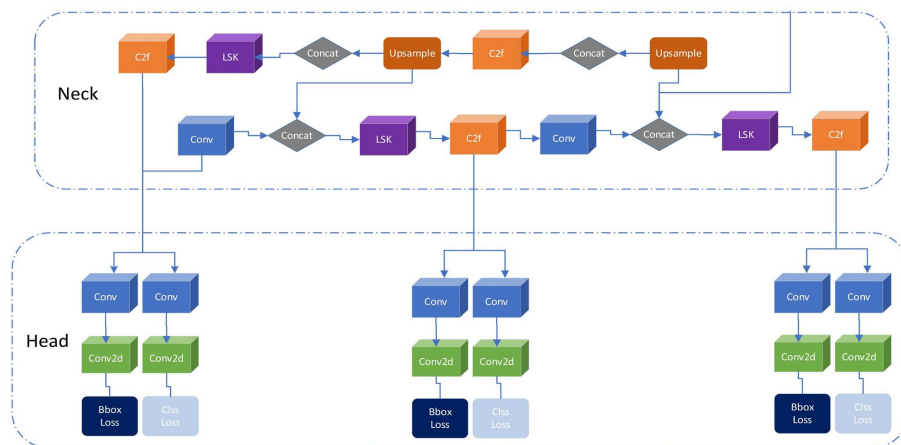


Figure 6. Neck improved the structure diagram  
图 6. Neck 改进结构图



Neck 部分位于 YOLOv5 的主干网络(Backbone)和检测头部(Head)之间。它的主要作用是在主干网络的基础上进行特征融合和降维操作,以提取更丰富的特征表示。具体来说,Neck 通过引入额外的卷积层,将来自主干网络不同层级的特征图进行融合,从而捕获到不同尺度的目标信息。这有助于提高模型对不同大小目标的检测能力,并增强模型对目标的多尺度感知能力。

见图 6, LSKNet Block 被添加在检测任务最后一个上采样层与最终的检测层之间,用于增强特征图的表示能力。由于模块可以动态地调整其大的空间感受视野,模型可以更加准确地学习到目标物体的局部特征和全局特征,并提高对复杂背景下目标物体的检测精度。

### 3.3.3. YOLOv8 多个位置(Global)添加 LSK 模块

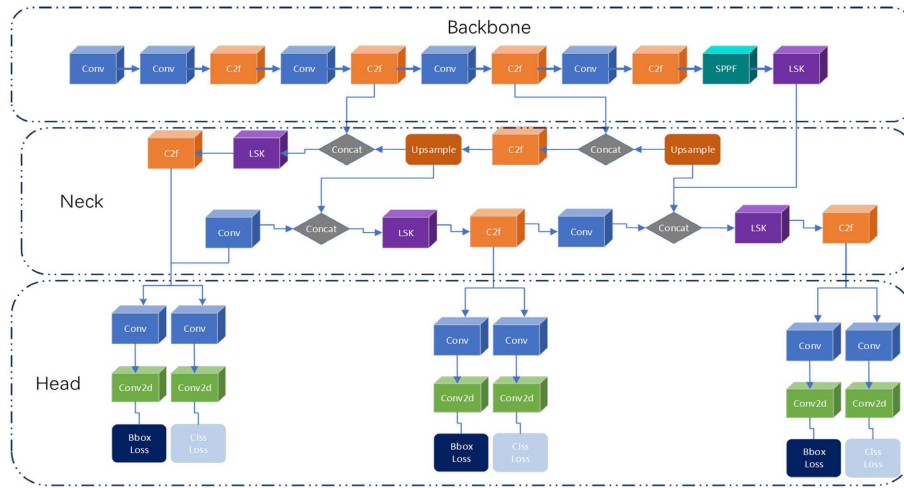


Figure 7. YOLOv8 network adds LSK modules in multiple locations

图 7. YOLOv8 网络多个位置添加 LSK 模块

YOLOv8 模型中的 Head 部分包含了三个检测头,分别对应于不同的特征层级,如低层、中层和高层特征图。这样可以实现多尺度目标检测,通过不同层级的特征来检测不同尺度的目标。在提高感受视野的同时,减少误检率,提高模型的整体性能。

LSKNet Block 会研究每个目标类别的感受野范围。这样可以使网络更加关注具有较高重要性的特征,而减弱对无关或冗余特征的关注,从而提高模型的表达能力和区分度。其中  $R_c$  为类别  $c$  的期望选择感受野的面积与地面边界框面积的比值:

$$R_c = \frac{\sum_{i=1}^{I_c} A_i / B_i}{I_c}, \quad (8)$$

$$A_i = \sum_{d=1}^D \sum_{n=1}^N |\tilde{S}_n^d \cdot RF_n|, B_i = \sum_{j=1}^{J_i} Area(GT_j), \quad (9)$$

$I_c$  为包含目标类别  $c$  的图像数量,  $A_i$  是输入图像  $i$  中所有 LSKNet Block 输出的空间选择激活的总和,  $d$  是 LSKNet 的 Block 数量,  $n$  是一个 LSK module 分解得到的卷积核数量,  $B_i$  是所有标注的地面真实目标框  $J_i$  的总像素面积。

通过在 C2f 前添加 LSKNet Block 模块,可以让模型更加关注重要的局部信息,从而提升目标的感知能力。

结合 Backbone 部分的改进, 在三个检测头前的 concat 连接处(图 7 灰色部分)添加 LSKNet Block, 提高模型整体的缺陷检测性能, 更加精确地检测 PCB 电路板的缺陷信息。

## 4. 实验

### 4.1. 数据集的准备

实验中用于模型训练和检测的图像数据集来自北京大学发布的 PCB, 其中包含六种(缺失孔, 老鼠咬坏, 开路, 短路, 杂散, 伪铜)缺陷一共 693 张 PCB 图片。数据集示例见图 8、图 9。

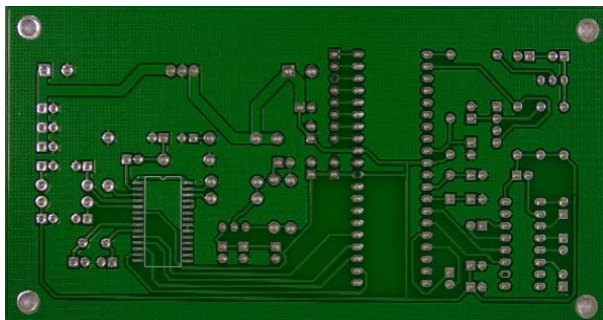


Figure 8. Dataset example

图 8. 数据集示例

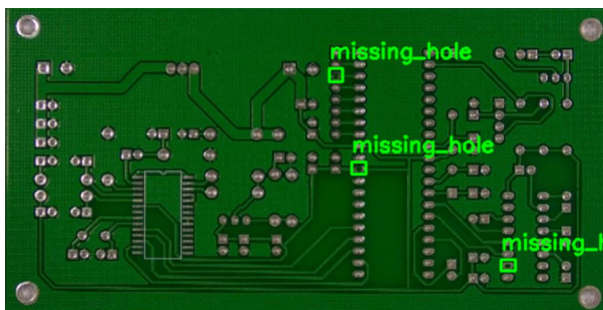


Figure 9. Example of dataset defect information (missing holes)

图 9. 数据集缺陷信息示例(缺失孔)

#### 4.1.1. 数据集的划分

根据 PCB 缺陷电路板数据集的大小, 将训练集、验证集、测试集划分比例设置为 8:1:1。即 554 张图片设置为训练集, 69 张设置为验证集, 70 张设置为测试集。为保证实验的可靠性和公平性, 在划分数据集的同时, 通过随机打乱数据集, 可以消除数据的顺序性和相关性, 减少模型受到数据排列顺序的影响。

由于数据过少, 对模型的训练有一定影响, 这里使用 SAHI(切片辅助超推理)的数据增强方法, 通过将图像切分成若干区域, 对各个区域分别进行预测, 并结合整张图片的预测结果进行合并, 最后使用 NMS 进行过滤, 以提高小目标检测的准确性和效率。

#### 4.1.2. 数据集的转换

由于 YOLO 数据集的格式是文本文件形式存储的, 而 PCB 数据集中的标签文件采用了 XML 格式存储。因此, 在使用 YOLO 模型训练前, 先将 XML 格式的标签文件转换为 YOLO 所需的文本文件格式, 以便模型能够正确读取和解析目标检测任务所需的信息。

## 4.2. 实验平台

### 4.2.1. 实验配置

本实验配置见表 1。

**Table 1.** Configuration of the experimental environment  
**表 1.** 实验环境配置

类型	参数
CPU	12th Gen Intel(R) Core(TM) i5-12600K
GPU	NVIDIA GeForce RTX 3080 Ti
操作系统	Linux
显存	12GB
Python 版本	3.8.17
Pytorch 版本	1.10.0
开发软件	VScode
加速环境	Cuda11.3

### 4.2.2. 实验参数设置

对于本次实验，训练参数设置如下。总共训练轮数 `epoch` 设置为 1500，批量大小 `batch_size` 设置为 -1，即根据可用资源自动确定。初始学习率设置为 0.01，训练设备为 GPU，输入图像尺寸为  $640 \times 640$  像素，数据加载器只使用主线程进行加载，即 `workers` 设置为 0。随机种子值 `seed` 设置为 249，不启用混合精度 `amp` 训练，即 `amp` 设置为 `True`。

### 4.2.3. 实验评价指标

实验采用精确率(Precision)、召回率(Recall)、 $F1$  值( $F1$ -score)、平均精度均值( $mAP$ )作为评估指标。

**精确率(Precision):** 在所有被分类为正例(正类别)的样本中，真正例(真实为正类别)的比例。它衡量了分类器将负例(负类别)错误地分类为正例的风险。精确率的计算公式为：精确率 = 真正例/(真正例 + 假正例)

$$P = \frac{TP}{TP + FP} \quad (10)$$

公式中：**TP (真正例):** 预测结果为正例，真实结果为正例，预测正确。**TN (假正例):** 预测结果为正例，真实结果为负例，预测错误。

**召回率(Recall):** 在所有真实为正例的样本中，被分类为正例的比例。它衡量了分类器正确地检测出正例的能力。召回率的计算公式为：召回率 = 真正例/(真正例 + 假反例)

$$R = \frac{TP}{TP + FN} \quad (11)$$

公式中：**FP (假负例):** 预测结果为负例，真实结果为正例，预测错误。**FN (真负例):** 预测结果为负例，真实结果为负例，预测正确。

**$F1$  值( $F1$ -score):**  $F1$  值即精确率和召回率的调和平均数，将两个指标综合考虑在内，可以更全面地评估模型的分类性能。 $F1$  计算公式： $2 * (Precision * Recall) / (Precision + Recall)$

$$F1 = \frac{2 * (P * R)}{(P + R)} \quad (12)$$

平均精度均值( $mAP$ ): 与  $F1$  值相比,  $mAP$  综合了所有类别的性能表现, 更全面地评估了模型在目标检测任务上的准确性和召回率。

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (13)$$

### 4.3. 实验结果

为验证添加 LSK 注意力机制的有效性, 在原始模型 YOLOv8s 的不同位置增加模块进行消融实验。实验分组分别是: 原始 YOLOv8s 模型、在原始 YOLOv8s 模型基础上 Backbone 部分添加 LSK 模块、在原始 YOLOv8s 模型基础上 Neck 部分添加 LSK 模块、在原始 YOLOv8s 模型基础上 Global (Backbone + Neck)部分添加 LSK 模块。训练结果见表 2。

**Table 2.** Experimental results

**表 2.** 实验结果

Model	Position	$P$	$R$	$mAP50$	$mAP50-95$	$F1$
YOLOv8s		0.918	0.909	0.937	0.518	0.913480
YOLOv8s + LSK	Backbone	0.926	0.900	0.945	0.504	0.912815
YOLOv8s + LSK	Neck	0.964	0.865	0.941	0.493	0.911821
YOLOv8s + LSK	Global	0.950	0.933	0.958	0.520	0.941423

注: Position 为改进的位置。

对于原 YOLOv8s 的训练数据, 在 Backbone 部分和 Neck 部分添加 LSK 模块,  $mAP$  平均精度分别提高 0.08、0.04, 精确率  $P$  分别提高 0.08、0.36。说明该注意力机制可以提升模型的精度。但是对于召回率, 对比元模型均有所下降, 模型可能错过了一些真实的样本。同时也导致  $F1$  值对比原模型没有明显的提高。

全局引入 LSK 模块后, 对比 YOLOv8s 原模型, 精确率  $P$  明显提高至 0.950, Recall 提高至 0.933,  $mAP$  提高至 0.958,  $mAP-50$  提高至 0.520,  $F1$  提高至 0.941。可以看出, 全局引入 LSK 模块改进后, 模型在 Recall、 $mAP$ 、 $mAP-50$  和  $F1$  上都有明显提高。说明改进后的模型, 在对于 PCB 缺陷检测的任务上, 有着更好的分类检测能力。

为验证模型的泛化能力与模型的可靠性, 现使用原模型 YOLOv8s 和添加 LSK 模块改进后的 YOLOv8s 模型, 在测试集上进行验证。结果见表 3。

**Table 3.** Comparison of test set results

**表 3.** 测试集结果对比

Model	Position	$P$	$R$	$mAP50$	$mAP50-95$	$F1$
YOLOv8s		0.958	0.872	0.932	0.477	0.912979
YOLOv8-PCB	Global	0.985	0.952	0.952	0.508	0.968219

通过对比表 3 与图 10 的结果, 可以看出 YOLOv8s-PCB 改进后的模型在多个指标上表现更好。

首先, 在所有类别的测试结果中, 改进后的模型在准确率  $P$ 、召回率  $R$ 、平均精度  $mAP50$ 、 $F1$  分数等指标上都有明显提升。例如, 准确率从 0.958 提高到了 0.985, 召回率从 0.872 提高到了 0.952,  $mAP50$  从 0.932 提高到了 0.952,  $F1$  分数从 0.912979 提高到了 0.968219。这意味着改进后的模型在识别目标物体方面更加准确、全面。

其次, 针对各个具体类别, 改进后的模型在大部分类别的指标上也有提升。以 missing\_hole 为例,

准确率提高到了 1，召回率提高到了 0.996， $mAP50$  提高到了 0.995， $F1$  分数提高到了 0.997996。而 `open_circuit` 和 `spur` 等类别的指标也都有提升。

综上所述，改进后的 `Yolov8s-PCB` 模型在目标检测任务中表现更好的原因可能是 `LSKNet block` 模块新的技术或优化策略，使得模型在目标检测方面的性能得到了提升。通过这个注意力机制，模型在准确性、召回率、平均精度等指标上都取得了明显的提高，具备了更好的泛化能力和识别能力。

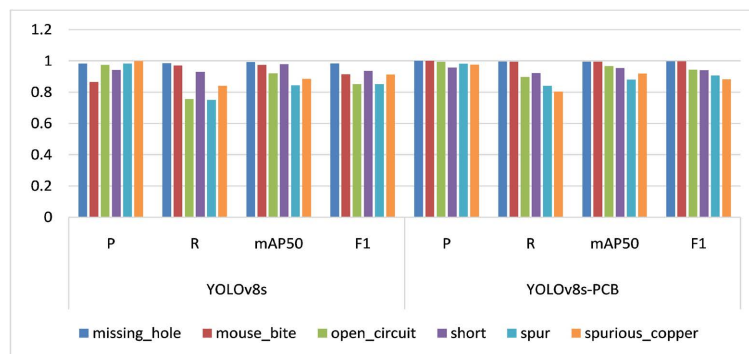


Figure 10. Comparison chart of test set metrics

图 10. 测试集指标对比图

预测效果对比见图 11~16，模型 `YOLOv8s-PCB` 的准确率也均要优于模型 `YOLOv8s` 的准确率。

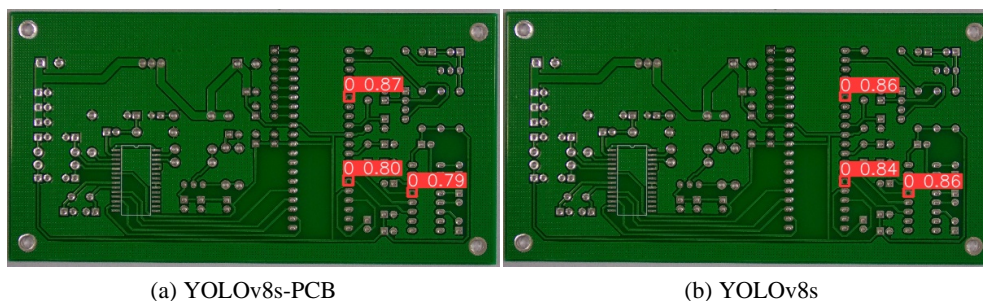


Figure 11. Comparison of the prediction effect of missing hole categories

图 11. 缺失孔类别预测效果对比

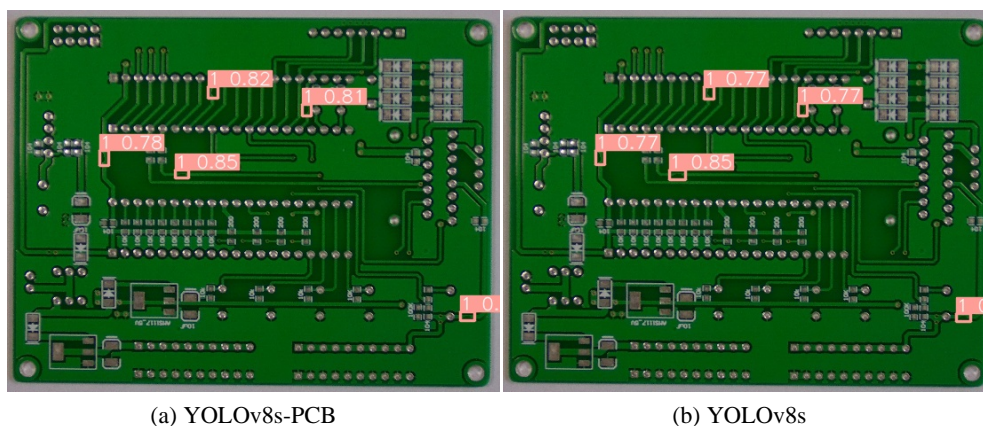


Figure 12. Comparison of the prediction effect of mouse bite categories

图 12. 老鼠咬坏类别预测效果对比

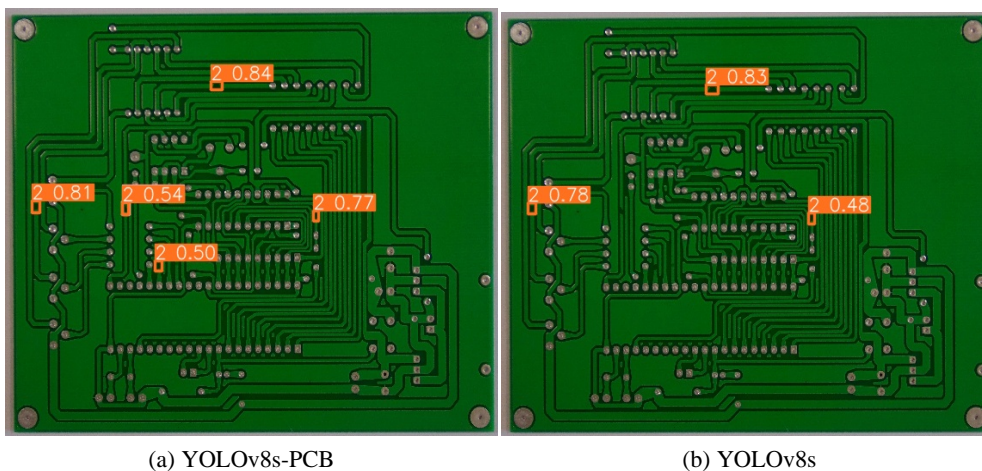


Figure 13. Comparison of the prediction effect of the open circuit category  
图 13. 开路类别预测效果对比

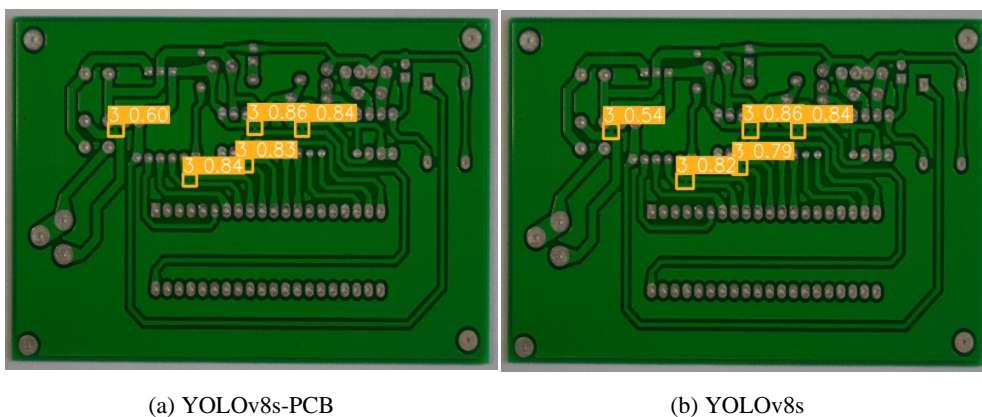


Figure 14. Comparison of the prediction effect of short circuit categories  
图 14. 短路类别预测效果对比

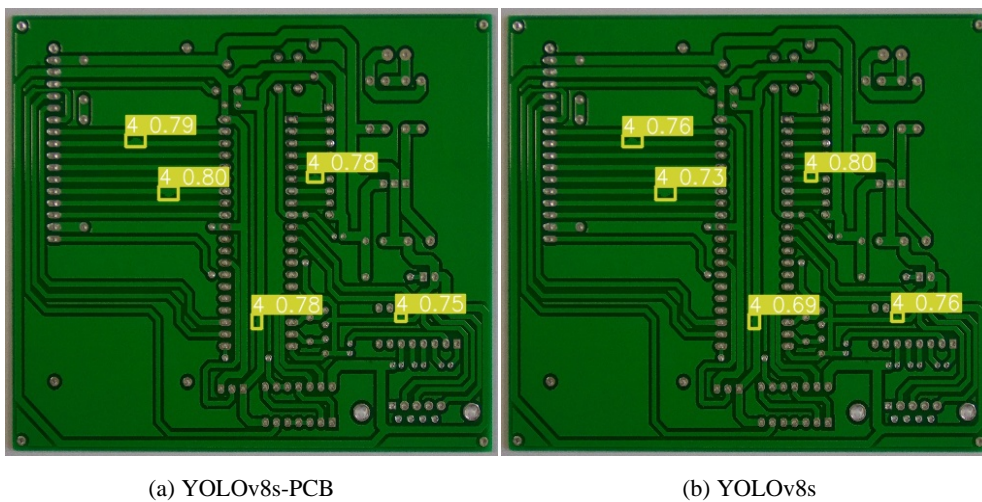
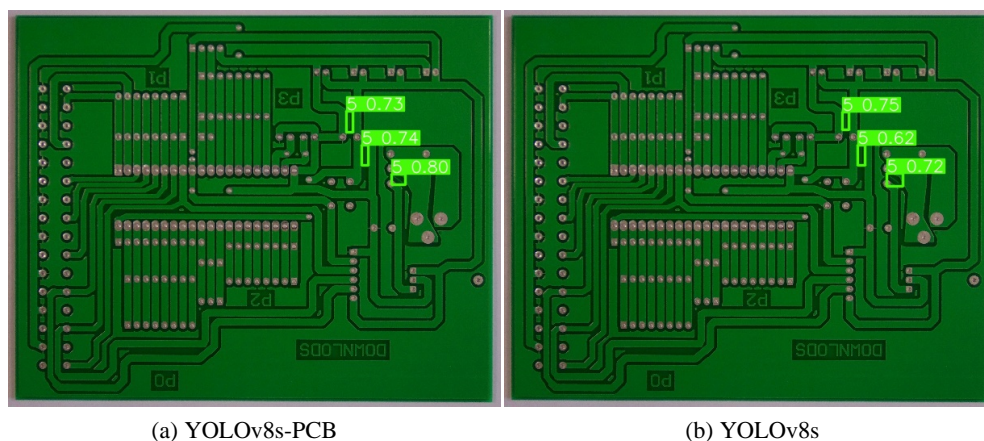


Figure 15. Comparison of spurious category prediction effects  
图 15. 杂散类别预测效果对比



**Figure 16.** Comparison of the prediction effect of spurious copper categories  
**图 16.** 伪铜类别预测效果对比

## 5. 展望

本文提出了一种基于改进 YOLOv8 模型的 PCB 电路板缺陷检测算法 YOLOv8-PCB，通过引入 LSK 注意力机制和调整网络模型结构，该模型可以快速准确地检测出 PCB 板上各种缺陷，大大提高 PCB 生产线的质量控制水平，节约人力物力成本，提高产品合格率。其次，YOLOv8-PCB 模型具有更好的泛化能力和识别能力，可以适应不同 PCB 板的检测需求。由于 PCB 板类型繁多，每种 PCB 板的缺陷形状、大小、位置等都可能存在差异，因此需要一个具有很强泛化能力的模型来适应各种情况。改进后的模型可以更好地适应不同 PCB 板的检测需求，具备更广泛的应用场景。后续研究会进一步考虑优化算法模型的复杂度，涉及更加轻量化的检测算法模型，并针对更多类型的电路板数据集(如柔性 FPCB 等)进行算法微调，使其具有更好的鲁棒性与适应能力。

## 基金项目

重庆理工大学大学生创新创业训练计划项目；重庆理工大学本科教育教学改革研究项目(2023YB116)；省级一流本科专业建设点 - 信息与计算科学(0101230246)。

## 参考文献

- [1] Lin, Y.L., Chiang, Y.M. and Hsu, H.C. (2018) Capacitor Detection in PCB Using YOLO Algorithm. 2018 *International Conference on System Science and Engineering (ICSSE)*, New Taipei, 28-30 June 2018, 1-4. <https://doi.org/10.1109/ICSSE.2018.8520170>
- [2] 宁健, 马淼, 柴立臣, 等. 深度学习的目标检测算法综述[J]. 信息记录材, 2022, 23(10): 1-4.
- [3] Girshick, R., Donahue, J., Darrell, T., et al. (2014) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 23-28 June 2014, 580-587. <https://doi.org/10.1109/CVPR.2014.81>
- [4] Jiang, H. and Learned-Miller, E. (2017) Face Detection with the Faster R-CNN. 2017 *12th IEEE International Conference on Automatic Face & Gesture Recognition*, Washington DC, 30 May-3 June 2017, 650-657. <https://doi.org/10.1109/FG.2017.82>
- [5] Liu, W., Anguelov, D., Erhan, D., et al. (2016) SSD: Single Shot Multibox Detector. *Computer Vision-ECCV 2016: 14th European Conference*, Amsterdam, 11-14 October 2016, 21-37. [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2)
- [6] Redmon, J., Divvala, S., Girshick, R., et al. (2016) You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vegas, NV, 27-30 June 2016, 779-788. <https://doi.org/10.1109/CVPR.2016.91>
- [7] Redmon, J. and Farhadi, A. (2017) YOLO9000: Better, Faster, Stronger. *Proceedings of the IEEE Conference on*

- Computer Vision and Pattern Recognition*, Honolulu, HI, 21-26 July 2017, 7263-7271. <https://doi.org/10.1109/CVPR.2017.690>
- [8] Jocher, G., Changyu, L., Hogan, A., *et al.* (2020) Ultralytics/Yolov5: Initial Release. Zenodo. <https://ui.adsabs.harvard.edu/abs/2020zndo...3908560J>
- [9] Li, Y., Hou, Q., Zheng, Z., *et al.* (2023) Large Selective Kernel Network for Remote Sensing Object Detection. 2023 *IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, 1-6 October 2023, 16748-16759. <https://doi.org/10.1109/ICCV51070.2023.01540>
- [10] Jocher, G., Chaurasia, A. and Qiu, J. (2023) Ultralytics Yolov8. Code Repository. <https://github.com/ultralytics/ultralytics>
- [11] 韩强. 面向小目标检测的改进 YOLOv8 算法研究[D]: [硕士学位论文]. 长春: 吉林大学, 2023.
- [12] Wang, C.Y., Bochkovskiy, A. and Liao, H.Y.M. (2023) YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, BC, 17-24 June 2023, 7464-7475. <https://doi.org/10.1109/CVPR52729.2023.00721>