doi:10.12068/j.issn.1005－3026.2024.11.005

基于PP-PicoDet-XS的改进铝型材表面缺陷检测算法

马淑华，李立振，秦汉民，沙晓鹏
（东北大学秦皇岛分校 控制工程学院，河北 秦皇岛 066004）

摘 要：铝型材在生产加工过程中会产生特征不明显和尺度大小不一等多类型的表面缺陷，针对现有人工抽检方法准确率低、实时性差、主观性强等问题，提出一种基于PP-PicoDet-XS的改进铝型材表面缺陷检测算法．改进的算法在主干网络中嵌入无参注意力SimAM，增强对深层有效特征的提取能力；使用SIoU（Scylla intersection over union）损失函数对训练过程进行优化，提高预测框的定位能力；采用量化蒸馏策略对模型进行压缩，提高推理速度．结果表明，改进的算法平均精度均值在交并比（intersection over union，IoU）阈值为0.5时达到了98.93%，在IoU阈值0.5~0.95范围内达到了57.60%，较未压缩的原始模型分别提高了1.73%和4.13%．将该算法部署到骁龙865移动端平台上进行推理，推理速度可达116.82帧/s，较未压缩的原始模型提高了47帧/s.

关 键 词：铝型材；缺陷检测；SimAM；损失函数；量化；蒸馏
中图分类号：TP 391.4 **文献标志码：**A **文章编号：**1005-3026(2024)11-1557-08

Improved Surface Defects Detection Algorithm for Aluminum Profiles Based on PP-PicoDet-XS

MA Shu-hua, *LI Li-zhen*, *QIN Han-min*, *SHA Xiao-peng*
(School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China. Corresponding author: LI Li-zhen, E-mail: lilizhen559@163.com)

Abstract: During the production and processing of aluminum profiles, multiple types of surface defects such as unclear features and varying scales may generate. In response to the problems of low accuracy, poor real-time performance, and strong subjectivity in existing manual sampling method, an improved surface defects detection algorithm is proposed for aluminum profiles based on PP-PicoDet-XS. The SimAM attention was embedded in the backbone to enhance the ability of extracting deep effective features. The SIoU (Scylla intersection over union) loss function is used to optimize the training process to improve the positioning ability of the prediction boxes. The quantization and distillation were used to compress the model to improve the inference speed. The results show that the improved algorithm achieves a mean average precision of 98.93% at intersection over union (IoU) threshold of 0.5, and 57.60% across IoU thresholds ranging from 0.5 to 0.95, which is 1.73% and 4.13% higher than the uncompressed original model. Deploying this algorithm on the Snapdragon 865 mobile platform for inference, the inference speed can reach 116.82 frames per second, which is 47 frames per second higher than the uncompressed original model.

Key words: aluminum profiles; defects detection; SimAM; loss function; quantization; distillation

铝型材广泛应用于国防、航天、轨道交通、建筑、医疗等领域，我国经济进入高质量发展阶段以来，作为现代经济和高新技术发展支柱性原材料的铝型材需求愈发旺盛，铝型材生产加工企业

收稿日期：2023-06-16
基金项目：河北省自然科学基金资助项目（F2021501021）.
作者简介：马淑华（1967-），女，河北秦皇岛人，东北大学秦皇岛分校教授.

亟需智能化与数字化转型升级.铝型材表面缺陷检测作为生产加工中的重要环节,目前主要采用的人工抽检方法准确率低、效率不足、主观性强,且容易受外界干扰,造成大量误检和漏检[1].

随着图像处理技术的发展,以特征提取和模板匹配为代表的缺陷检测算法某种程度上取代了人工抽检并应用于工业现场[2-5],但是这类检测算法依赖于人工提取缺陷特征,当缺陷种类多样且形态多变时,检测过程将变得尤为复杂.

近年来,随着GPU算力的快速提升,基于深度学习的各类目标检测算法不断涌现,可以概括为二阶段目标检测算法和一阶段目标检测算法两大类.二阶段目标检测又称为基于候选区域的目标检测,此类算法先从输入图像中获取候选区域,然后利用卷积神经网络对候选区域进行分类识别,代表性的算法有R-CNN[6],Fast R-CNN[7],Faster R-CNN[8],Mask R-CNN[9],虽然此类算法的准确率较高,但速度较慢,难以实现实时检测.一阶段目标检测算法,如SSD[10],YOLO[11]系列,此类算法不需要产生候选框,仅通过一个网络同时完成目标的定位和分类问题,比二阶段目标检测算法具有更快的检测速度,更适用于工业现场的实时检测.

为了得到轻量化的模型以部署在嵌入式设备中,目前主流的做法,一种是将YOLO,SSD中的主干网络替换为轻量级的特征提取网络,如MobileNet[12],ShuffleNet[13]和GhostNet[14].王淑青等[15]使用改进的YOLOv5进行PCB板缺陷检测,采用ShuffleNetV2结构取代YOLOv5的主干网络中的原始卷积Conv与C3模块并进一步优化,使得改进的算法参数减少91%,浮点运算次数(floating point of operations,FLOPs)减少近70%,能够满足在小型计算平台上快速部署的要求.另一种是直接使用YOLOX-Nano[16],YOLOv5n,YOLOv7-Tiny[17],PP-PicoDet-XS[18]等轻量化模型,但是轻量化模型在追求推理速度的同时会造成一定的精度损失.

为了实现复杂场景下轻量化、实时、高精度的铝型材表面缺陷检测,本文提出了一种基于PP-PicoDet-XS的改进轻量级缺陷检测算法.首先将无参注意力SimAM引入主干网络,并通过实验探究其位置对检测精度的影响;然后使用SIoU Loss代替GIoU(generalized IoU) Loss作为边界框回归损失函数来优化预测框的回归过程;最后对全精度模型进行量化感知训练和知识蒸馏,减小模型体积,提高推理速度,并部署在骁龙865 CPU上进行精度与速度测试,以满足工业现场中的实际部署需求.

1 PP-PicoDet-XS算法

PP-PicoDet是基于anchor-free的轻量级目标检测算法,与其他算法相比,它更好地实现了精度与速度之间的均衡,在移动设备上实现了卓越的性能.最新发布的PP-PicoDet算法(2022.03.20版)提供了XS,S,M,L共4种不同的网络尺度,本文以PP-PicoDet-XS为基准进行改进,改进前其结构如图1所示.

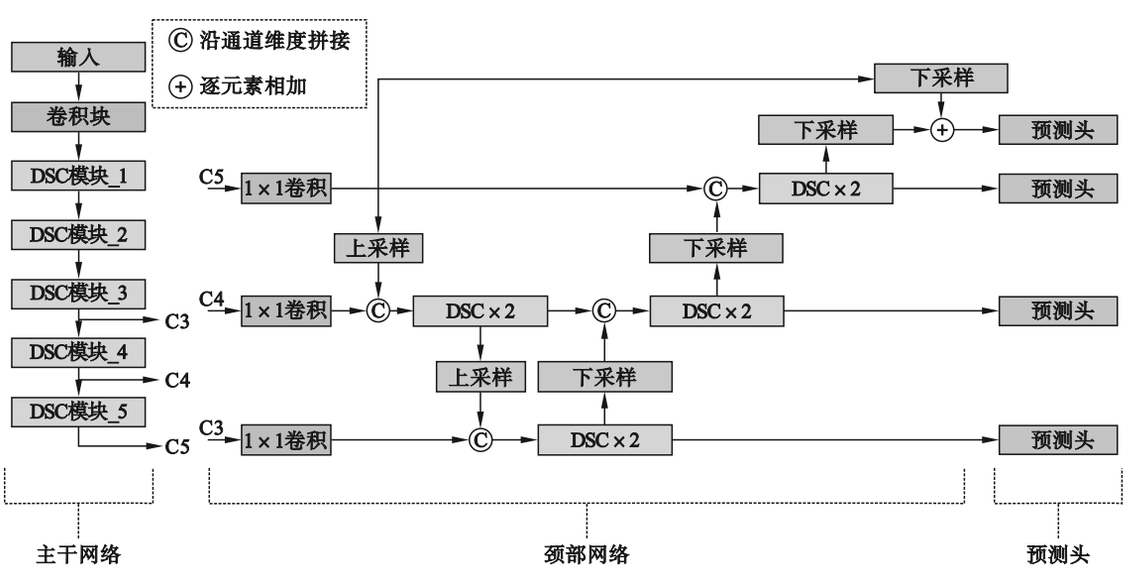

图1 PP-PicoDet-XS网络结构

Fig. 1 PP-PicoDet-XS network architecture

主干网络为LCNet[19],其中DSC(depthwise separable convolutions)代表深度可分离卷积,由逐通道卷积与逐点卷积组成.主干网络由5个DSC模块组成,分别为模块_1至模块_5,包含DSC的个数分别为1,2,2,6,2.表1为主干网络各层的具体实现,SE(squeeze-and-excitation)注意力机制被嵌入在模块_5中.

表1　PP-PicoDet主干网络结构

Table 1　PP-PicoDet backbone architecture

主干网络组成	算子	卷积核尺寸	步长	SE
卷积块	卷积	3×3	2	—
DSC模块_1	DSC	3×3	1	—
DSC模块_2	DSC	3×3	2	—
	DSC	3×3	1	—
DSC模块_3	DSC	3×3	2	—
	DSC	3×3	1	—
DSC模块_4	DSC	3×3	2	—
	DSC×5	5×5	1	—
DSC模块_5	DSC	5×5	2	√
	DSC	5×5	1	√

颈部网络首先通过3个1×1卷积将C3,C4,C5输出的特征图的通道数调整为最小通道数,然后通过特征金字塔网络和路径聚合网络来进行特征融合,得到4种不同尺度的特征图输入到预测头.

在预测头中,损失函数包含3部分,分别为$\mathrm{loss}_{\mathrm{vfl}}$,$\mathrm{loss}_{\mathrm{dfl}}$和$\mathrm{loss}_{\mathrm{giou}}$,如式(1)所示:

$$\mathrm{loss}=\mathrm{loss}_{\mathrm{vfl}}+0.5\cdot\mathrm{loss}_{\mathrm{dfl}}+2.5\cdot\mathrm{loss}_{\mathrm{giou}}. \tag{1}$$

其中:$\mathrm{loss}_{\mathrm{vfl}}$是用来表征物体类别的损失函数Varifocal Loss;$\mathrm{loss}_{\mathrm{dfl}}$和$\mathrm{loss}_{\mathrm{giou}}$是用来表征物体边界框回归的损失函数,分别为Distribution Focal Loss和GIoU Loss.其中GIoU Loss是基于IoU Loss[20]改进而来.GIoU Loss如式(2),式(3)所示:

$$\mathrm{GIoU}=\mathrm{IoU}-\frac{\left|C-B\cup B^{\mathrm{GT}}\right|}{|C|}, \tag{2}$$

$$\mathrm{loss}_{\mathrm{giou}}=1-\mathrm{GIoU}. \tag{3}$$

其中:

$$\mathrm{IoU}=\frac{\left|B\cap B^{\mathrm{GT}}\right|}{\left|B\cup B^{\mathrm{GT}}\right|}; \tag{4}$$

B,B^{GT}分别代表预测框和真实框;C代表B,B^{GT}的最小外接矩形框.

2　算法改进与模型压缩

2.1　嵌入SimAM注意力机制模型

SimAM注意力机制[21]基于神经科学理论提出,是一种能够在卷积神经网络中即插即用的无参注意力机制.

在神经科学理论中,信息量最大的神经元通常与周围神经元表现出不同的放电模式,一个活跃的神经元也会抑制周围神经元的活动,这种现象被称为空间抑制.表现出明显空间抑制效应的神经元在视觉加工中应该被认为具有更高的重要性.

基于上述理论,SimAM注意力机制为卷积神经网络中每个神经元定义了一个能量函数,用来测量目标神经元与其他神经元的线性可分性.当输入特征图X的通道数为C,高和宽分别为H,W时,每个通道有$M=H\times W$个能量函数,式(5)为经过一系列简化后得到的某通道中目标神经元t的最小能量函数:

$$e_t^*=\frac{4\left(\hat{\sigma}^2+\lambda\right)}{\left(t-\hat{\mu}\right)^2+2\hat{\sigma}^2+2\lambda}. \tag{5}$$

其中:

$$\hat{\mu}=\frac{1}{M}\sum_{i=1}^{M}x_i; \tag{6}$$

$$\hat{\sigma}^2=\frac{1}{M}\sum_{i=1}^{M}\left(x_i-\hat{\mu}\right)^2; \tag{7}$$

式中:$\hat{\mu},\hat{\sigma}$分别为某通道中所有神经元的均值和方差;λ为正则化系数.最小能量函数e_t^*越小,目标神经元t与周围神经元的线性可分性程度越高,该目标神经元也就越重要,其重要性可以通过$1/e_t^*$求得.最后通过式(8)计算所有神经元的注意力权重并进行加权,得到加权后的特征图$\tilde{X}$:

$$\tilde{X}=\mathrm{sigmoid}\left(\frac{1}{E}\right)\odot X. \tag{8}$$

其中:E表示输入特征图X所有神经元所对应的e_t^*构成的张量;sigmoid(·)为单调函数,用来限制$1/e_t^*$的大小;⊙为哈达玛积.

与通道注意力SE生成一维权重相比,SimAM注意力通过计算每个神经元的重要性生成三维权重,而且无需引入额外的参数.基于以上优点,本文在主干网络DSC模块_5中嵌入SimAM注意力替代原来的SE注意力,在使得目标检测网络更加轻量化的同时,能够更有效地找出特征图中的重点特征并进行增强,提高神经网

络的特征提取与表达能力.

2.2 改进损失函数

在PP-PicoDet算法中,边界框回归损失函数使用的是GIoU Loss.但是,GIoU存在一些不足,如图2所示,当预测框B完全处于真实框B^{GT}之内或真实框B^{GT}完全处于预测框B之内时,式(2)中GIoU与式(4)中IoU的值相等,这种情况下GIoU便退化为IoU.针对GIoU这种不足,DIoU(distance-IoU),CIoU(complete-IoU)相继被提出,DIoU在GIoU的基础上增加了两边界框中心点距离信息,CIoU在DIoU基础上进一步增加了边界框宽高比信息.本文使用更先进的SIoU Loss[22]作为边界框回归的损失函数,SIoU不仅考虑到了边界框的重叠面积、中心点距离和宽高比,而且进一步考虑到了预测框和真实框之间的向量角度.

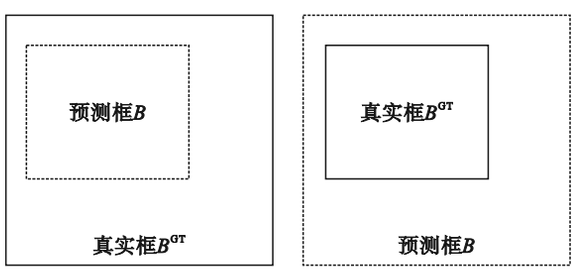

图2 GIoU退化为IoU示意图

Fig. 2 Schematic diagram of GIoU degradation to IoU

SIoU Loss由4部分组成:角度损失Λ、距离损失Δ、形状损失Ω和IoU损失.角度损失Λ计算如式(12)所示:

$$c_{\mathrm{h}}=\max\left(b_{c_y}^{\mathrm{gt}},b_{c_y}\right)-\min\left(b_{c_y}^{\mathrm{gt}},b_{c_y}\right), \tag{9}$$

$$\sigma=\sqrt{\left(b_{c_x}^{\mathrm{gt}}-b_{c_x}\right)^2+\left(b_{c_y}^{\mathrm{gt}}-b_{c_y}\right)^2}, \tag{10}$$

$$x=\frac{c_{\mathrm{h}}}{\sigma}=\sin\alpha, \tag{11}$$

$$\Lambda=1-2\times\sin^2\left(\arcsin x-\frac{\pi}{4}\right). \tag{12}$$

其中:$b_{c_x}^{\mathrm{gt}}$,$b_{c_y}^{\mathrm{gt}}$表示真实框中心点的坐标值;b_{c_x},b_{c_y}表示预测框中心点的坐标值;如图3所示,c_{h},c_{w}分别表示两框中心点的高度差和宽度差;σ表示两框中心点之间的距离.当角度$\alpha\leqslant\pi/4$时,优先最小化α,否则优先最小化β.

距离损失Δ计算如式(13)~式(16)所示:

$$\rho_x=\left(\frac{b_{c_x}^{\mathrm{gt}}-b_{c_x}}{d_{\mathrm{w}}}\right)^2, \tag{13}$$

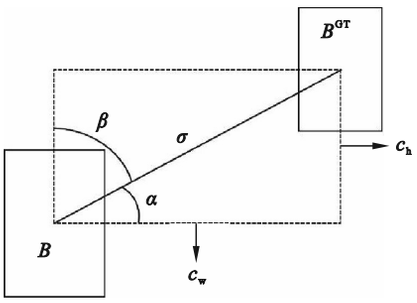

图3 角度损失计算示意图

Fig. 3 Schematic diagram of angle cost calculation

$$\rho_y=\left(\frac{b_{c_y}^{\mathrm{gt}}-b_{c_y}}{d_{\mathrm{h}}}\right)^2, \tag{14}$$

$$\gamma=2-\Lambda, \tag{15}$$

$$\Delta=\sum_{t=x,y}\left(1-\mathrm{e}^{-\gamma\rho_t}\right). \tag{16}$$

其中:d_{w},d_{h}分别表示预测框和真实框最小外接矩形的宽和高,如图4所示.

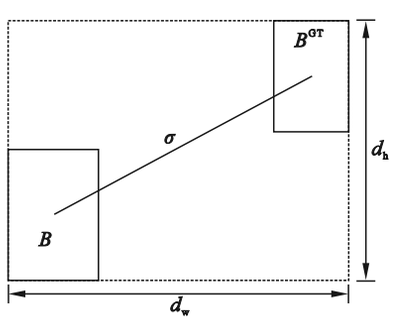

图4 距离损失计算示意图

Fig. 4 Schematic diagram of distance cost calculation

形状损失Ω计算如式(17)~式(19)所示:

$$\omega_{\mathrm{w}}=\frac{\left|w-w^{\mathrm{gt}}\right|}{\max\left(w,w^{\mathrm{gt}}\right)}, \tag{17}$$

$$\omega_{\mathrm{h}}=\frac{\left|h-h^{\mathrm{gt}}\right|}{\max\left(h,h^{\mathrm{gt}}\right)}, \tag{18}$$

$$\Omega=\sum_{t=w,h}\left(1-\mathrm{e}^{-\omega_t}\right)^{\theta}. \tag{19}$$

其中:w,h分别为预测框的宽和高;w^{gt},h^{gt}分别为真实框的宽和高;θ的值一般在2~6之间,用来限制形状损失的大小,本文中将θ值设为4. IoU损失计算同式(4),综上所述,SIoU Loss定义为

$$\mathrm{SIoU}=\mathrm{IoU}-\frac{\Delta+\Omega}{2}, \tag{20}$$

$$\mathrm{loss}_{\mathrm{siou}}=1-\mathrm{SIoU}. \tag{21}$$

本文使用SIoU Loss代替GIoU Loss,增加了对两框中心点之间距离、两框形状差别和两框向量角度的惩罚项,从多方面优化了预测框的回归过程,提高了定位精度.

2.3　模型压缩方案

随着深度学习的发展,深度神经网络模型变得越来越复杂,除了训练成本越来越高外,如何在移动端部署也更加受到关注.由于网络模型上百万的参数量以及浮点运算量,而移动端硬件设备的存储空间和算力有限,使得它们难以部署在Android和其他低功耗的边缘设备上,所以,一种有效的方法就是对模型进行压缩,提高推理速度.本文在推理部署阶段采用量化感知训练[23]和知识蒸馏策略[24]对模型进行压缩,框架如图5所示.

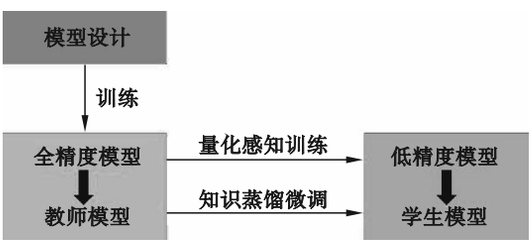

图5　量化蒸馏框架

Fig. 5　Framework for quantized and distillation

首先在训练好的全精度网络需要量化的算子前插入伪量化节点,然后训练少量轮数进行微调,最后导出量化预测模型.伪量化即先将全精度浮点数量化为8位定点数,然后再反量化为全精度浮点数,这一伪量化过程使得最后的损失中会存在量化误差分量,最后在数据集上训练少量轮数微调,从而减少精度损失.本文采用线性对称量化方法,如式(22),式(23)所示:

$$s=\frac{2^{b-1}}{\alpha}, \tag{22}$$

$$x_{\mathrm{q}}=\mathrm{clip}\left(\mathrm{round}\left(s\cdot x\right),-2^{b-1}+1,2^{b-1}-1\right). \tag{23}$$

其中:round(·)表示将数字四舍五入到最接近的整数;$\mathrm{clip}\left(a,a_\mathrm{min},a_\mathrm{max}\right)$表示将$a$的值限定在$a_\mathrm{min}$到$a_\mathrm{max}$之间;$s$为缩放系数;$b$为量化的比特数,此处为8;$\alpha$为全精度参数的表示范围,全精度参数被限定在$[-\alpha,\alpha]$内;$x$表示全精度权重;$x_{\mathrm{q}}$表示量化权重.反量化如式(24)所示:

$$\hat{x}=\frac{1}{s}\cdot x_{\mathrm{q}}. \tag{24}$$

其中,$\hat{x}$表示反量化后的全精度权重.量化是一个信息有损的压缩过程,量化感知训练的优点是在训练中调整权重分布以适应模拟量化训练,从而大幅降低量化模型的精度损失,优于离线量化的方法.

量化完成后采用自蒸馏的方式进行知识蒸馏,在自蒸馏中,教师模型和学生模型使用相同的网络结构.本文将全精度模型作为教师模型,将量化后的低精度模型作为学生模型,使用训练数据集和软标签进行微调,使量化模型能够学到全精度模型的知识,提高量化模型的精度,减小量化带来的精度损失.

3　实　　验

3.1　数据集准备

本文使用的数据集由第七届全国大学生集成电路创新创业大赛海云捷迅杯杯赛官方发布,共包含1 400张缺陷图片,缺陷类别为“针孔”、“擦伤”、“脏污”和“褶皱”,图片分辨率为640×480.数据集被划分为3部分:训练集共1 197张用于模型训练;验证集共133张用于模型评估;测试集共70张用于最终的推理部署测试.

训练时采用线上数据增强策略,由于过多的数据增强会提高正则化效果使得轻量化模型更难以收敛,所以本文沿用原PP-PicoDet-XS算法中采用的方法,即仅使用随机翻转、随机裁剪和随机像素内容更换作为训练中的单样本数据增强.批量样本增强采用多尺度训练方式,将一个批次的图片随机缩放到相同的尺寸,缩放尺寸为256,288,320,352,384,在验证和推理时图片调整为320×320作为模型的输入.

3.2　消融实验

本文使用未改进的PP-PicoDet-XS作为基准模型,使用平均精度均值(mean average precision,mAP)作为评价指标,mAP@0.5表示将IoU阈值设为0.5时的mAP,mAP@0.5∶0.95表示IoU阈值从0.5到0.95(步长为0.05)时的平均mAP.基准模型在验证集上的mAP@0.5∶0.95为53.47%,mAP@0.5为97.20%.

主干网络的作用是提取图像的特征表示,在主干网络中如果过早地应用注意力机制,可能会导致一些有用的信息被过早地过滤掉.而将注意力机制放在网络较深的位置可以确保所有可能有用的信息都被充分考虑.为了探究SimAM模块在主干网络中不同位置的影响,本文首先将原始网络中的SE模块去除,然后将SimAM模块分别添加在主干网络的5个DSC模块上进行单独实验,实验结果见表2.结果表明,SimAM在不同位置时mAP均有不同程度提升,当SimAM模块嵌入到DSC模块_5中时,mAP@0.5∶0.95与mAP@0.5达到了最高,分别为55.56%,98.81%,

相较于原始PP-PicoDet-XS分别提升了2.09%，1.61%，这说明将SimAM模块添加在主干网络末端时能更好地增强提取有效特征的能力，带来更高的精度收益.因此，本文最终将SimAM模块添加在DSC模块_5中的两个深度可分离卷积中.

表2 SimAM模块在不同位置的结果

Table 2 Results of SimAM module in different positions

网络模型	SimAM位置	mAP@0.5∶0.95 %	mAP@0.5 %
PP-PicoDet-XS	1 0 0 0 0	54.51	98.47
	0 1 0 0 0	54.12	98.02
	0 0 1 0 0	53.48	97.51
	0 0 0 1 0	53.55	97.22
	0 0 0 0 1	55.56	98.81
	1 1 1 1 1	53.66	97.94

尽管更先进的IoU损失函数往往更有利于提高检测精度，加快收敛，但是直接更换损失函数并不科学，为了探究不同IoU损失函数对模型性能的影响，本文对比了4种不同IoU损失函数下的模型精度，实验结果见表3.结果表明，当使用SIoU Loss时，检测精度超越了其他3种IoU损失函数，mAP@0.5∶0.95与mAP@0.5分别达到了56.22%，98.71%，相较于原始PP-PicoDet-XS分别提升了2.75%，1.51%，这说明SIoU Loss通过增加角度损失、距离损失和形状损失，有效增强了模型的回归能力，提高了预测框和真实框之间的位置匹配度，使检测精度得到显著提升.因此，本文最终选择使用SIoU Loss作为PP-PicoDet-XS的IoU损失函数.

表3 不同IoU损失函数的影响

Table 3 Influence of different IoU Loss

网络模型	IoU损失函数	mAP@0.5∶0.95 %	mAP@0.5 %
PP-PicoDet-XS	GIoU	53.47	97.20
	DIoU	54.16	97.73
	CIoU	54.06	97.89
	SIoU	56.22	98.72

3.3 与主流算法对比

本文将改进的算法与工业界主流的轻量级目标检测算法进行了比较，具体指标包括mAP、参数量、FLOPs和参数体积，实验结果如表4所示，表中PP-PicoDet-XS(+SimAM+SIoU Loss)为本文改进的算法.

结果表明，改进的PP-PicoDet-XS算法mAP@0.5∶0.95，mAP@0.5相比于改进之前分别提高5.57%，2.06%，且模型参数量有所减少、参数体积有所减小，这得益于SimAM生成的三维权重对主干网络末端更加抽象和语义化特征的进一步增强和使用SIoU Loss对训练策略的优化.

与YOLOX-Nano,PPYOLO-Tiny和SSDLite-MobileNet_v3算法相比，虽然FLOPs略大于这3种算法，但改进后的算法参数量和参数体积均达到了最小，mAP达到了最高.与YOLOv5n，SSD-MobileNet_v1和SSDLite-MobileNet_v1相比，所有指标均达到了最优.因此，在本文所用的铝型材缺陷检测数据集上，改进后的PP-PicoDet-XS算法与同类型轻量级目标检测算法相比具有显著优越性.

表4 不同算法对比

Table 4 Comparison with different algorithms

网络模型	mAP@0.5:0.95/%	mAP@0.5/%	参数量×10^{-6}	FLOPs×10^{-9}	参数体积/MB
YOLOv5n	51.16	95.78	1.7693	0.5246	6.81
YOLOX-Nano	55.58	97.53	0.8973	0.3124	3.53
PPYOLO-Tiny	46.75	95.07	0.9979	0.2510	3.90
PP-PicoDet-XS	53.47	97.20	0.6748	0.3204	2.67
SSD-MobileNet_v1	48.42	93.41	5.5621	1.1467	21.34
SSDLite-MobileNet_v1	49.53	93.78	5.6255	1.1550	21.63
SSDLite-MobileNet_v3	51.14	95.43	1.1737	0.1160	4.59
PP-PicoDet-XS(+SimAM+SIoU Loss)	59.04	99.26	0.6551	0.3204	2.65

3.4 推理部署

在进行模型部署之前，首先基于PaddleSlim对训练好的模型进行量化感知训练和知识蒸馏，得到压缩后的模型，然后基于PaddleLite将压缩后的模型转换为naive_buffer类型以便在ARM CPU端进行推理.模型推理部署所采用的移动端设备为搭载骁龙865处理器的小米10，骁龙865处理器采用Kryo 585架构，配备了1个主频为2.84 GHz的Cortex-A77大核、3个主频为2.42 GHz的Cortex-A77中核和4个主频为1.8 GHz的

Cortex-A55小核,搭载了Hexagon 698 DSP张量加速器,支持8位和16位定点数运算,集成了Adreno 650 GPU,支持16位和32位浮点数运算,支持新的AI混合精度指令,而且具有体积小、功耗低、价格便宜等优势.

本文采用测试集中的70张缺陷图片进行推理测试,算法改进前后与压缩前后的性能对比如表5所示.其中模型体积指经过PaddleLite转换为naive_buffer类型文件的体积,推理时间由对测试集70张图片总推理时间取平均得到.

表5　模型压缩前后性能对比

Table 5　Comparison of performance before and after model compression

网络模型	是否压缩	mAP@0.5:0.95	mAP@0.5	模型体积	推理时间	推理速度
		%	%	MB	ms	帧·s^{-1}
PP-PicoDet-XS	否	53.47	97.20	3.03	14.32	69.83
	是	52.63	95.92	1.21	8.82	113.38
PP-PicoDet-XS(+SimAM+SIoU Loss)	否	59.04	99.26	2.85	13.15	76.04
	是	57.60	98.93	1.19	8.56	116.82

从表5可以看出,对模型进行压缩后,模型体积会减小60%左右,推理速度能够提高44帧/s左右,但是精度会有一定的损失.改进后算法的mAP@0.5:0.95与mAP@0.5相比压缩前分别损失了1.44%和0.33%,降为57.60%和98.93%,但是此精度仍然高于表4中所比较的其他算法的精度,同时推理速度达到了116.82帧/s,相比于未压缩的原始PP-PicoDet-XS模型的推理速度提高47帧/s,完全能够满足工业现场的实际需求.

图6展示了模型改进和压缩前后的部分检测结果对比.可以看到铝型材背景环境整体偏暗,黑色脏污即使用肉眼也难以轻易检出,有时还存在反光,压缩前且改进前的模型对针孔、脏污、擦伤这几类缺陷出现了明显的漏检情况.相比之下压缩后且改进后的模型能够较为准确地检测出光线干扰背景下不同的缺陷,成功克服了由于背景昏暗造成的干扰,这说明本文的改进算法不仅大大提高了有效特征的提取能力,增强了模型在复杂光照条件下的鲁棒性,而且在提升检测性能的同时实现了模型的轻量化,进一步验证了改进方案的有效性.

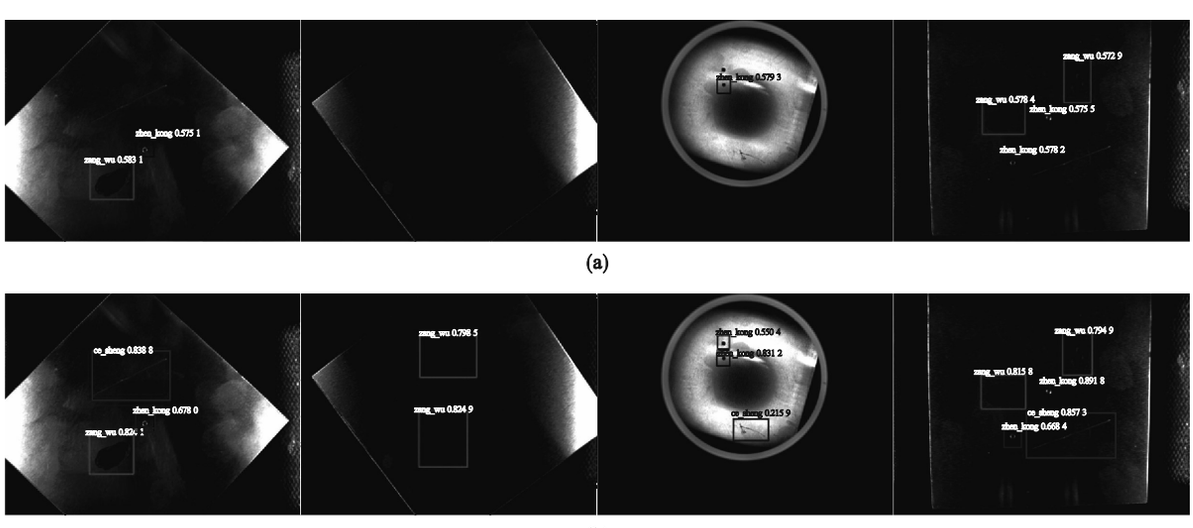

图6　模型改进和压缩前后的检测结果对比

Fig. 6　Comparison of detection results before and after model improvement and compression

(a)—压缩前且改进前模型的检测结果;(b)—压缩后且改进后模型的检测结果.

4　结　　论

1)为了解决铝型材表面特征缺陷不明显且与背景难以区分的问题,在PP-PicoDet-XS的主干网络中引入了SimAM注意力来增强对有效特征的提取能力,并放置在不同位置进行实验,验证了SimAM位于主干网络末端时能带来更高的

检测精度.

2)为了提高训练过程中预测框的定位能力,使算法更好地收敛,使用SIoU Loss代替GIoU Loss,增加了预测框和真实框中心点之间距离、两框形状差别和两框向量角度的惩罚项,优化了预测框的回归过程,进一步提高了检测精度.实验表明,在铝型材缺陷检测数据集上,改进后的算法与同类型轻量级目标检测算法相比在各项指标上具有显著优越性.

3)为了提高推理速度,减小模型体积,对训练好的模型进行量化蒸馏,最后部署到骁龙865移动平台上进行测试,结果表明推理速度和检测精度均有提升,完全满足工业现场的实际部署需求.

参考文献:

[1] Wei X K, Yang Z M, Liu Y X, et al. Railway track fastener defect detection based on image processing and deep learning techniques: a comparative study [J]. *Engineering Applications of Artificial Intelligence*, 2019, 80: 66-81.

[2] Jian C X, Gao J, Ao Y H. Automatic surface defect detection for mobile phone screen glass based on machine vision [J]. *Applied Soft Computing*, 2016, 52(3): 348-358.

[3] 赵翔宇,周亚同,何峰,等.工业干扰环境下基于模板匹配的印刷品缺陷检测[J].包装工程,2017,38(11):187-192.
(Zhao Xiang-yu, Zhou Ya-tong, He Feng, et al. Printing defects detection based on template matching under disturbing industrial environment [J]. *Packaging Engineering*, 2017, 38(11): 187-192.)

[4] 李永敬,朱萍玉,孙孝鹏,等.基于形状模板匹配的冲压件外形缺陷检测算法研究[J].广州大学学报(自然科学版),2017,16(5):62-66.
(Li Yong-jing, Zhu Ping-yu, Sun Xiao-peng, et al. Shape defect detection algorithm of stamping parts based on shape template matching [J]. *Journal of Guangzhou University (Natural Science Edition)*, 2017, 16(5): 62-66.)

[5] 孙光民,刘鹏,李子博.基于图像处理的带钢表面缺陷检测改进算法的研究[J].软件工程,2018,21(4):5-8.
(Sun Guang-min, Liu Peng, Li Zi-bo. Research on the detection algorithm of strip steel surface defects based on image processing [J]. *Software Engineering*, 2018, 21(4): 5-8.)

[6] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C]//IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014: 580-587.

[7] Girshick R. Fast R-CNN [C]//IEEE International Conference on Computer Vision. Santiago: IEEE, 2015: 1440-1448.

[8] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137-1149.

[9] He K M, Gkioxari G, Dollár P, et al. Mask R-CNN [C]// 2017 IEEE International Conference on Computer Vision (ICCV). Venice, 2017: 2980-2988.

[10] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector [C]//European Conference on Computer Vision. Berlin: Springer, 2016: 21-37.

[11] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [C]//IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 779-788.

[12] Howard A G, Zhu M L, Chen B, et al. MobileNets: efficient convolutional neural networks for mobile vision applications [EB/OL]. (2017-04-17) [2023-04-03]. http://arxiv.org/abs/1704.04861.

[13] Zhang X Y, Zhou X Y, Lin M X, et al. ShuffleNet: an extremely efficient convolutional neural network for mobile devices [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, 2018: 6848-6856.

[14] Han K, Wang Y H, Tian Q, et al. GhostNet: more features from cheap operations [C]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 1577-1586.

[15] 王淑青,鲁濠,鲁东林,等.基于轻量化人工神经网络的PCB板缺陷检测[J].仪表技术与传感器,2022(5):98-104.
(Wang Shu-qing, Lu Hao, Lu Dong-lin, et al. PCB board defect detection based on lightweight artificial neural network [J]. *Instrument Technique and Sensor*, 2022(5): 98-104.)

[16] Ge Z, Liu S, Wang F, et al. YOLOX: exceeding YOLO series in 2021 [EB/OL]. (2021-08-06) [2023-04-08]. https://doi.org/10.48550/arXiv.2107.08430.

[17] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors [C]//IEEE Conference on Computer Vision and Pattern Recognition. Vancouver: IEEE, 2023: 7464-7475.

[18] Yu G, Chang Q, Lyu W, et al. PP-PicoDet: a better real-time object detector on mobile devices [EB/OL]. (2021-11-01) [2023-04-15]. https://doi.org/10.48550/arXiv.2111.00902.

[19] Cui C, Gao T Q, Wei S Y, et al. PP-LCNet: a lightweight CPU convolutional neural network [EB/OL]. (2021-09-17) [2023-04-15]. https://doi.org/10.48550/arXiv.2109.15099.

[20] Rezatofighi H, Tsoi N, Gwak J, et al. Generalized intersection over union: a metric and a loss for bounding box regression [C]//IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2019: 658-666.

[21] Yang L, Zhang R Y, Li L, et al. SimAM: a simple, parameter-free attention module for convolutional neural networks [C]//International Conference on Machine Learning. [n.l]: PMLR, 2021: 11863-11874.

[22] Gevorgyan Z. SIoU loss: more powerful learning for bounding box regression [EB/OL]. (2022-05-15) [2023-04-23]. https://doi.org/10.48550/arXiv.2205.12740.

[23] Jacob B, Kligys S, Chen B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference [C]//IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 2704-2713.

[24] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network [J]. *Computer Science*, 2015, 14(7): 38-39.