文章编号:1674-2974(2024)12-0078-09 DOI:10.16339/j.cnki.hdxbzkb.2024285

基于MDS-YOLO模型的小目标检测问题研究

朱恩文[1],梁嬰[1,2],肖进文[1],梁小林[1†]
(1. 长沙理工大学 数学与统计学院,湖南 长沙,410114;
2. 湖南工程学院 计算科学与电子学院,湖南 湘潭,411104)

摘 要:针对目前主流算法对小目标检测存在计算量大与准确率较低的问题,本文以轻量级网络MobileNetV3代替YOLOv4中的主干网络,并将颈部网络中的一部分普通卷积用深度可分离卷积替代,同时针对小目标检测定义一个新的损失函数IF-EIoU Loss,由此构建了MDS-YOLO目标检测模型.该模型具有较高的检测速度,且针对小目标具有较好的检测性能.为了验证模型的有效性,分别在MS COCO数据集和Visdrone2019数据集上进行了实验.与YOLOv4算法相比,在MS COCO数据集上,MDS-YOLO算法的平均检测精度提升了1.5个百分点,对于小目标的检测精度提升了3.3个百分点,检测速度也从31帧/s提升至36帧/s;在Visdrone2019数据集上,MDS-YOLO算法将平均检测精度从YOLOv4的14.9%提升至16.3%.实验结果表明,本文提出的MDS-YOLO算法能有效提升小目标检测精度.

关键词:小目标检测;YOLOv4算法;轻量级网络MobileNetV3;IF-EIoU Loss;MS COCO数据集

中图分类号:O213;TP181 **文献标志码**:A

Research of Small Object Detection Problem Based on MDS-YOLO Model

ZHU Enwen[1],LIANG Zhao[1,2],XIAO Jinwen[1],LIANG Xiaolin[1†]
(1. School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha 410114, China;
2. School of Computational Science and Electronics, Hunan Institute of Engineering, Xiangtan 411104, China)

Abstract: To solve the problem of large computation and low accuracy of the current mainstream algorithms for small object detection, this paper replaces the backbone network in YOLOv4 with the lightweight network MobileNetV3, and replaces some ordinary convolutions in the neck network with depthwise separable convolutions. At the same time, a new loss function IF-EIoU Loss is defined for small object detection. Therefore, MDS-YOLO object detection model is constructed. This model has a high detection speed and good detection performance for small object. To verify the effectiveness of the model, experiments are carried out on MS COCO dataset and Visdrone2019 dataset, respectively. Compared with the YOLOv4 algorithm, on MS COCO dataset, the average detection accuracy of the MDS-YOLO algorithm is improved by 1.5 percentage points, the detection accuracy of

* **收稿日期**:2024-01-12
基金项目:国家自然科学基金重点资助项目(51839002,52338009), National Natural Science Foundation of China(51839002,52338009);国家杰出青年科学基金资助项目(52025085), National Science Fund for Distinguished Young(52025085);湖南省自然科学基金资助项目(2021JJ30734), Natural Science Foundation of Hunan Province(2021JJ30734);湖南省研究生创新性课题(CX20220952), Hunan Provincial Innovation Foundation for Postgraduate(CX20220952)
作者简介:朱恩文(1976—),男,湖南湘阴人,长沙理工大学教授,博士
† 通信联系人,Email: liang@csust.edu.cn。

small object is increased by 3.3 percentage points, and the detection speed is also increased from 31 frames per second to 36 frames per second. On the Visdrone2019 dataset, the MDS-YOLO algorithm increases the average detection accuracy from 14.9% of YOLOv4 to 16.3%. The experimental results show that the MDS-YOLO algorithm proposed can effectively improve the detection accuracy of small object.

Key words: small object detection; YOLOv4 algorithm; lightweight network MobileNetV3; IF-EIoU Loss; MS COCO dataset

目标检测是计算机视觉中的一项重要任务,主要用于识别和定位图像或视频中的特定物体或区域,具体过程是在输入的图像或视频中找出感兴趣的目标,将其框选出来,并给出它们的类别标签和位置信息[1].对于在图像或视频中占据像素较少的小目标物体的检测识别是目标检测的重要组成部分,小目标检测的主要应用领域如下:交通安防监控视频中对车牌、距离较远的车辆和行人等的检测;自动驾驶场景下对远距离环境中的路标、路牌、行人和各种障碍物的检测;水下无人潜航中对小型生物的探测;无人机领域中对高空视角下的生物、交通工具、交通标志等各类小目标的检测;医学图像领域中对早期较微小的病变区域进行精确检测,如肿瘤、微血管等[2-4].

基于深度学习的目标检测算法在大型数据集上的平均检测精度都比较理想,但对小目标的检测仍然存在一些困难.这主要是因为小目标所包含的特征信息比大目标要少得多,导致通用的目标检测算法无法准确地识别出图像或视频中的小目标.因此,专门针对小目标的检测算法应运而生.文献[5]提出了RetinaNet目标检测算法,它通过定义一种新的损失函数Focal Loss来解决在目标检测任务中出现的正负样本不平衡问题,将容易被正确分类的样本的权重降低,使模型更加关注难以分类的样本,这种权重调节的策略能够提高模型对于难以分类的小目标的检测能力.文献[6]通过在图像中搜索物体中心点,然后从中心点回归物体的边界框和类别来检测目标,对于小目标的检测效果比通用的目标检测算法好,运行速度也有所提升.文献[7]提出了全卷积单阶段目标检测(fully convolutional one-stage object detection, FCOS)算法,它采用特殊的正负样本采样策略,通过在特征图上分配正负样本,避免了小目标正负样本不均衡和标签不准确等问题,提高了小目标检测的准确性和稳定性.文献[8]提出了EfficientDet目标检测算法,该算法采用AutoAugment训练策略增强模型对小目标的泛化能力,并使用双向特征金字塔网络(bi-directional feature pyramid network, BiFPN)提取来自不同层级的特征并将它们结合在一起,使得网络在不同尺度下都能够有效地识别小目标.文献[9]提出了双结构金字塔场景解析网络模型,改善了由于模型结果连续进行下采样而导致的原始图像信息丢失和分辨率下降问题.尽管针对小目标的检测取得了一定的成果,但是各种检测算法在主流的目标检测数据集上对于小目标的平均检测精度都不高,如何提高小目标的检测精度和定位精度是本文关注的主要问题.

本文在YOLOv4算法的基础上从输入图像、网络结构、损失函数三个部分对该算法进行了优化和改进,提出了轻量级网络+深度可分离+YOLO(MobileNet+depthwise separable+YOLO, MDS-YOLO)模型.优化和改进可概括如下.

1)输入图像:先将原始图像分割成N帧局部图像并设置重叠区域,再分别输入检测网络进行目标检测,然后将网络生成的目标候选框和类别、位置信息整合在一起投射回原始图像上,最后经过非极大值抑制去除重复的候选框后输出预测结果.通过这样的操作,可以尽可能减少检测过程中图像特征信息的丢失,保留更多的细节信息,从而提取到更多的小目标特征,提升小目标的检测精度.

2)网络结构:将YOLOv4算法[10]的主干特征提取网络CSPDarknet-53替换成轻量级网络MobileNetV3[11],并将颈部网络的一部分普通卷积用深度可分离卷积替代.通过这样的改进可以显著减少网络的参数量和计算量,从而提升检测速度,减少图像分

割操作对检测速度的影响.

3)损失函数:提出了新的位置损失函数IF-EIoU Loss.该损失函数对于与真实边界框重合度较小的低质量预测框给予更大的损失权重,使网络更加关注难以检测的小目标物体,从而提升网络对于小目标预测框的回归精度.

1 小目标检测算法设计

本模型对YOLOv4算法的输入图像、网络结构、损失函数3个部分进行优化和改进.

1.1 图像分割

一般的目标检测算法会固定输入图像的尺寸,而在实际应用中需要检测的图像都是大小各异的,因此需要通过压缩、剪裁等方式将图像调整到合适的大小才能输入检测网络,然后对图像进行多次下采样得到输出特征图[12].这两个过程都会丢失大量的图像信息,原始图像越大,损失的信息就越多,特别是对于只占据图中少量像素的小目标物体来说,经过预处理和下采样后,网络很难提取到其有效特征信息,从而无法准确检测和定位目标.为了有效保留小目标信息,本文采用如下3步分割图像.

1.1.1 根据原始图像大小确定分割数N

式(1)给出了图像分割数:

$$N_x = \left\lfloor \frac{c_x p_y}{300} \right\rfloor, N_y = \left\lfloor \frac{c_y p_x}{300} \right\rfloor \tag{1}$$

式中:p_x, p_y分别代表图像在横向和纵向的像素个数;c_x, c_y分别是横向分割超参数和纵向分割超参数;N_x, N_y分别代表图像在横向和纵向的分割数.

由式(1)可以看出,图像的分割数$N = N_x N_y$是由图像本身的大小和c_x, c_y这两个超参数确定的.当c_x, c_y固定不变时,图像横向像素个数p_x越多,纵向分割数N_y越大;图像纵向像素个数p_y越多,横向分割数N_x就越大.横向分割超参数c_x和纵向分割超参数c_y的数值根据不同数据集中的图像大小灵活调整.

1.1.2 对分割后的局部图像设置重叠区域

为了有效降低目标物体被分割截断的风险,本文对分割后的局部图像扩充重叠区域.重叠区域的大小由分割后的局部图像的尺寸确定,若原始图像大小为$p_x p_y$,则分割后的局部图像尺寸L_x, L_y由式(2)确定:

$$L_x = \left\lfloor \frac{p_x}{N_x}(1 + L'_x) \right\rfloor, L_y = \left\lfloor \frac{p_y}{N_y}(1 + L'_y) \right\rfloor \tag{2}$$

重叠区域的横向尺寸L'_x和纵向尺寸L'_y分别取局部图像横向尺寸和纵向尺寸的15%,这样的设置使得重叠区域的大小随着图像尺寸和分割数的变化而灵活调整,始终保持在合理的范围内.

1.1.3 对目标检测网络得到的目标框进行非极大值抑制操作

所有的局部图像检测完成后,会将目标检测网络生成的候选框和类别及置信度信息全部投射到原始图像上,因为重叠区域是由相邻的局部图像共享的,所以重叠区域的目标可能会被不同的局部图像重复检测.投射到原始图像对应位置后可能会产生大量的对相同目标的重复候选框,所以将所有的目标预测框投射到原始图像后,对原始图像整体做一次非极大值抑制(non-maximum suppression, NMS)[13]操作,尽可能去除重复的候选框.

1.2 网络结构设计

MDS-YOLO网络结构与YOLOv4[10]相似,由3部分构成:主干网络、颈部网络与检测头部(见图1).

在主干网络设计中,将YOLOv4中的主干特征提取网络CSPDarknet-53替换为MobileNetV3(见图1).MobileNetV3是MobileNet系列网络的集大成之作,它不仅结合了MobileNetV1的深度可分离卷积[14]和MobileNetV2的线性瓶颈的倒残差结构[15-16],还引入了轻量级注意力模块SE[11]和一种新的激活函数H-Swish[11].该模型不仅能大幅减少模型的参数量和计算量,而且能有效提升网络的特征提取能力和表达能力.在颈部网络中,本文使用深度可分离卷积代替一部分YOLOv4中用到的普通卷积(见图1中DPC模块),从而减少参数量与计算量.而检测头部网络与YOLOv4一致,分为3部分,分别对应下采样的32倍、16倍、8倍3个不同尺寸的特征图[15].

1.3 损失函数优化

目标检测算法的损失函数由3部分组成:定位损失函数、置信度损失函数和类别损失函数.其中,定位损失函数用于衡量模型对检测框位置的预测精度,也就是模型预测的目标边界框与真实边界框之间的拟合程度;置信度损失函数则是用于度量模型对检测框置信度(即预测框是否包含目标)的预测精度;而类别损失函数用于衡量模型对检测框类别的

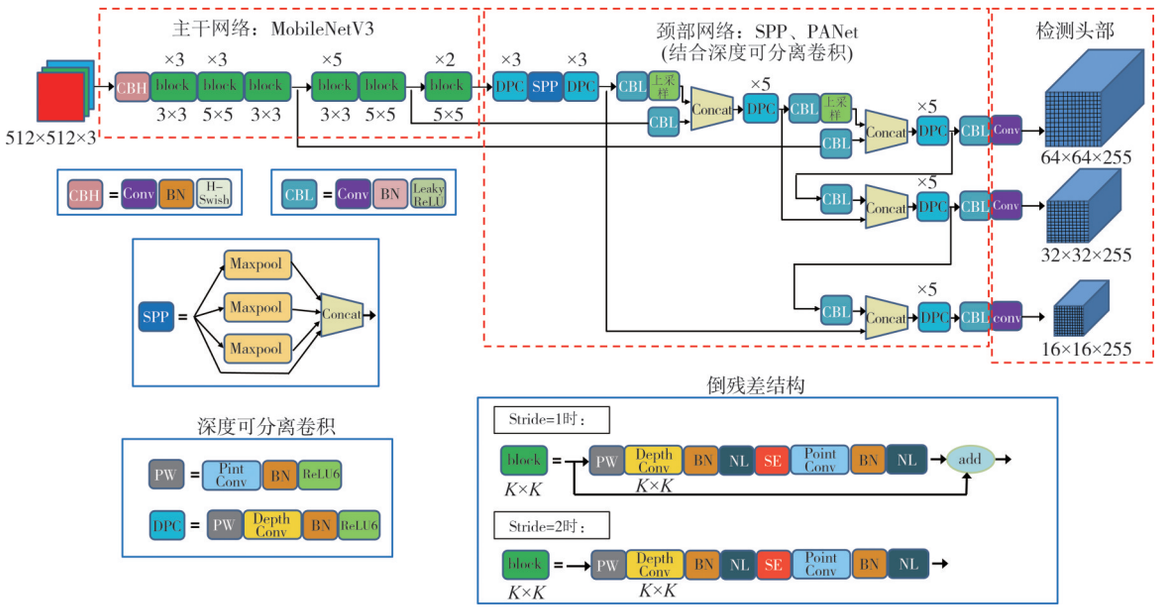


图 1　MDS-YOLO 的网络结构

Fig.1　Network Structure of MDS-YOLO

预测精度[10].本文主要针对YOLOv4损失函数的定位损失函数部分进行优化.YOLOv2的定位损失函数为CIoU Loss[17],文献[18]通过整合EIoU Loss和FocalL1 Loss,得到了最终的Focal-EIoU Loss,见式(3).

$$\text{Focal-EIoU Loss} = \text{IoU}^{\gamma} \cdot L(\text{EIoU}) \tag{3}$$

式中：$\text{IoU}(B, B^{gt}) = \dfrac{|B \cap B^{gt})|}{|B \cup B^{gt})|}$；$\text{EIoU}(B, B^{gt}) = 1 - \text{IoU}(B, B^{gt}) + \dfrac{\rho^2(b, b^{gt})}{c^2} + \dfrac{\rho^2(w, w^{gt})}{c_w^2} + \dfrac{\rho^2(h, h^{gt})}{c_h^2}$；而$b$, w, h分别是预测框B的中心坐标及宽与高；b^{gt}, w^{gt}, h^{gt}分别是真实框B^{gt}的中心坐标及宽与高；c, c_w, c_h分别是包含预测框与真实框的最小矩形的对角线及宽与高.

$$L(x) = \begin{cases} -\dfrac{e\beta x^2\left[2\ln(\beta x) - 1\right]}{4}, 0 < x \leqslant 1, 0 < \beta < 1/e \\ -e\beta(\ln\beta)x + e\beta(2\ln\beta + 1)/4, x > 1, 0 < \beta < 1/e \end{cases}$$

Focal-EIoU Loss对于IoU越大的高质量预测框给予的损失权重越大,对于IoU越小的低质量预测框给予的损失权重越小.但是现有的目标检测算法对于小目标的检测精度本身就很低,这就意味着小目标的预测边界框中误差大的低质量预测框占比较大,如果根据Focal-EIoU Loss的思想,对低质量预测框给予的损失权重较小,在训练时会极大影响对小目标预测边界框的回归精度,所以本文反其道而行之,基于Focal-EIoU Loss提出一种新的损失函数IF-EIoU Loss,其定义如下:

$$\text{IF-EIoU Loss} = (1 - \text{IoU})^{\gamma} \cdot L(\text{EIoU}) \tag{4}$$

与Focal-EIoU Loss的思想相反,IF-EIoU Loss对于IoU较小的低质量预测框给予较大的损失权重,使网络更多关注难以检测的小目标样本,从而提升网络对于小目标预测边界框的回归精度.

由此可得整体损失函数为

$$\begin{aligned} \text{Object Loss} = {} & \lambda_{\text{coord}} \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{\text{obj}} (1 - \text{IoU})^{\gamma} \cdot L(\text{EIoU}) - \\ & \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{\text{obj}} \left[C_i^j \ln(\hat{C}_i^j) + (1 - C_i^j)\ln(1 - \hat{C}_i^j) \right] - \\ & \lambda_{\text{noobj}} \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{\text{noobj}} \left[C_i^j \ln(\hat{C}_i^j) + (1 - C_i^j)\ln(1 - \hat{C}_i^j) \right] - \\ & \sum_{i=0}^{K\times K}\sum_{j=0}^{M} I_{ij}^{\text{obj}} \sum_{c \in \text{classes}} \left[p_i^j(c)\ln(\hat{p}_i^j(c)) + (1 - p_i^j(c))\ln(1 - \hat{p}_i^j(c)) \right] \end{aligned} \tag{5}$$

式(5)右边第一项为回归定位损失.式中:$K \times K$表示参与预测的特征图的大小;M表示每个对应的特征图网格生成的预测边界框数量;λ_{coord}是正样本权重系数;I_{ij}^{obj}是正样本指示函数,特征图中每个网格都会生成9个预测边界框,其中与真实边界框的IoU最大的预测边界框则负责对应目标物体的检测,对应的I_{ij}^{obj}为1,其他为0.

式(5)右边第二项为正样本的置信度损失.$\hat{C}_i^j$是预测边界框的置信度;C_i^j是真实边界框的置信度,当

某一边界框的 C_i^j 为 1 时,对应标号的 I_{ij}^{obj} 为 1,否则为 0.

式(5)右边第三项是负样本的置信度损失.λ_{noobj} 是负样本权重系数;I_{ij}^{noobj} 是负样本指示函数,目标预测边界框中与对应真实边界框的 IoU 小于 0.5 的视为负样本,对应的 I_{ij}^{noobj} 取 1,其他为 0.

式(5)右边第四项为类别损失.classes 是检测目标的类别集合;$\hat{p}_i^j(c)$ 表示边界框内物体属于某一类别的预测概率,$p_i^j(c)$ 表示边界框内物体是此类目标的真实概率(取值为 0 或 1).

2 实验结果与分析

为了验证目标检测算法 MDS-YOLO 对于小目标检测的性能,本文分别在经典的目标检测数据集 MSCOCO(分辨率小于 32 × 32 的目标定义为小目标,分辨率大于 96 × 96 的目标定义为大目标,其他定义为中等目标)和无人机拍摄图像数据集 Visdrone2019(绝大部分为小目标,不区分大中小目标)上进行了实验,并将实验结果与其他主流的目标检测算法进行比较,而且进行了消融实验.

2.1 实验环境及参数设置

实验的硬件平台和软件平台如表 1 所示.

表 1 实验运行环境配置

Tab. 1 Experimental running environment configuration

配置	型号
操作系统	Win10
处理器	Intel(R) Xeon(R) Silver 4210R CPU @ 2.40 GHz
内存	64 G
显卡	NVIDIA GeForce RTX 3060
深度学习框架	Pytorch
编程语言	Python 3.5

训练分为两个阶段,即冻结主干网络阶段和解冻阶段,使用 Adam 优化算法进行优化.为了提高训练效率,采用了冻结训练的方式,在特征提取阶段使用了 MobileNetV3 在 MS COCO 数据集上的训练权重.在冻结阶段,epoch 设为 50,batch size 设为 8;在解冻阶段,epoch 同样设为 50,但 batch size 降为 4,学习率均为 0.000 5.

2.2 实验结果及分析

2.2.1 消融实验

本节验证各种模块与优化方法对算法的贡献大小.

1) 不同网络结构的检测速度、参数量及 FLOPs 对比实验.

为了验证将主干特征提取网络替换成轻量级网络和将普通卷积替换成深度可分离卷积(DPC)的优化效果,本文在 MS COCO 数据集上做了关于检测速度与参数量及 FLOPs 的对比实验,其中,图像的输入尺寸均为 512 × 512(未进行图像分割).

从表 2 中可以看出,轻量级网络与深度可分离卷积均对优化模型有正面影响,特别地,由 1 号和 6 号模型可知,将 YOLOv4 的主干网络替换成 MobileNetV3 并加入深度可分离卷积后,FPS 从 31 帧/s 提升到了 61 帧/s,检测速度提升了约 97%,参数量由 64.36 MB 减少到 19.10 MB,FLOPs 由 45.84 MB 下降到 10.16 MB.实验结果充分说明了本文对于主干网络的优化,不仅可以减少参数量与 FLOPs,而且能有效提升检测速度.

表 2 轻量级网络和深度可分离卷积的检测速度、参数量及 FLOPs 对比结果

Tab. 2 Comparison results of detetion speed, parameter quantity and FLOPs between lightweight networks and depthwise separable convolutions

模型序号	网络结构	FPS/(帧·s^{-1})	参数量/MB	FLOPs/MB
1	CSPDarknet-53	31	64.36	45.84
2	CSPDarknet-53+DPC	44	42.95	35.19
3	MobileNetV2	46	39.94	21.21
4	MobileNetV2+DPC	56	18.54	10.56
5	MobileNetV3	52	40.51	20.81
6	MobileNetV3+DPC(本文)	**61**	**19.10**	**10.16**

2) 轻量级网络、深度可分离卷积、图像分割和 IF-EIoU 损失函数的精度对比实验.

为了验证轻量级网络、深度可分离卷积、图像分割和 IF-EIoU 损失函数对小目标检测精度的影响,本文在 MS COCO 数据集上进行了平均精度(AP_{COCO})与小目标检测精度(AP_S)[19]的对比实验.

从表 3 可知,轻量级网络、深度可分离卷积、图像分割和 IF-EIoU 损失函数都对精度有正面作用,

表3　轻量级网络、深度可分离卷积、图像分割和IF-EIoU损失函数的检测精度结果

Tab. 3　Detection accuracy results of lightweight network, depthwise separable convolution, image segmentation and IF-EIoU loss function

模型	AP_{COCO}/%	AP_S/%
去“MobilenetV3”	43.8	26.1
去“DPC”	42.6	25.4
去“图像分割”	40.9	24.1
去“IF-EIoU Loss”	43.8	26.5
本文	44.5	27.6

其中,去“图像分割”对精度的影响最大,相较于本文模型,平均精度下降了3.6个百分点,小目标检测精度下降了3.5个百分点.实验结果充分说明了轻量级网络、深度可分离卷积、图像分割操作和定位损失函数优化可以有效提升算法对于小目标的检测精度.

2.2.2　在MS COCO数据集上的检测结果

在保证实验环境一致的前提下,本文将改进后的MDS-YOLO算法和原YOLOv4算法在MS COCO数据集的训练集上进行训练并在测试集上进行检测,对比其检测速度和检测精度.MDS-YOLO的横向分割超参数c_x和纵向分割超参数c_y均设置为1,YOLOv4(未进行图像分割)和MDS-YOLO(加入图像分割)的输入尺寸都为512 × 512.本文还选择了一些具有代表性的两阶段目标检测算法和单阶段目标检测算法加入对比,如Faster R-CNN[20-21]、Mask R-CNN[22]、PANet[23]、SSD[24]、RetinaNet[6]、YOLOv3[25]、DETR[26-29]、YOLOv7-tiny-SiLU[30]等.此外,在评价指标方面,我们选择了不同IoU阈值和不同尺寸目标的多类别平均精度来全面衡量MDS-YOLO算法的性能.

表4　MDS-YOLO与其他算法在MS COCO数据集上的实验结果对比

Tab. 4　Comparison of experimental results between MDS-YOLO and other algorithms on the MS COCO dataset

算法	主干网络	输入尺寸/(像素×像素)	AP_{COCO}/%	AP_{50}/%	AP_{75}/%	AP_S/%	AP_M/%	AP_L/%	FPS/(帧·s^{-1})
两阶段算法:									
Faster R-CNN[20]	VGG-16	~ 1 000 × 600	24.2	45.3	23.5	7.7	26.4	37.1	4
Faster R-CNN+++[21]	ResNet-101	~ 1 000 × 600	34.9	55.7	37.4	15.6	38.7	50.9	—
Faster R-CNN w/ FPN[16]	ResNet-101	~ 1 000 × 600	36.2	59.1	39.0	18.2	39.0	48.2	—
Mask R-CNN[22]	ResNeXt-101	~ 1 300 × 800	39.8	62.3	43.3	22.1	43.2	51.2	2
PANet(multi-scale)[23]	ResNeXt-101	~ 1 400 × 840	47.4	67.2	51.8	30.1	51.7	60.0	2
单阶段算法:									
YOLOv2[17]	DarkNet-19	544 × 544	21.6	44.0	19.2	5.0	22.4	35.5	21
SSD[24]	ResNet-101	513 × 513	31.2	50.4	33.3	10.2	34.5	49.8	—
RetinaNet[6]	ResNet-101-FPN	800 × 800	39.1	59.1	42.3	21.8	42.7	50.2	—
YOLOv3[25]	DarkNet-53	416 × 416	31.0	55.3	32.3	15.2	33.2	42.8	16
FCOS[8]	ResNet-101-FPN	800 × 1 024	41.5	60.7	45.0	24.4	44.8	51.6	—
YOLOv4[10]	CSPDarknet-53	512 × 512	43.0	64.9	46.5	24.3	46.1	55.2	31
DETR[26]	ResNet-50	800 × 1 066	42.0	62.4	44.2	20.5	45.8	61.1	—
Conditional-DETR[27]	ResNet-50	800 × 1 066	43.0	64.0	45.7	22.7	46.7	61.5	—
Anchor-DETR[28]	ResNet-50	800 × 1 066	42.1	63.1	44.9	22.3	46.2	60.0	—
DN-DETR-R50[29]	ResNet-50	800 × 1 066	44.1	64.4	46.7	22.9	48.0	63.4	—
YOLOv7-tiny-SiLU[30]	ELAN	640 × 640	38.7	56.7	41.7	18.8	42.4	51.9	—
MDS-YOLO(本文)	MobileNetV3-Large	512 × 512	44.5	65.2	47.7	27.6	46.5	53.6	36

从表4可知,与单阶段目标检测算法相比,本文提出的MDS-YOLO无论在检测精度还是检测速度上都有优势.和两阶段算法中的PANet相比,MDS-YOLO无论是对于小目标的检测精度还是对于其他尺寸目标的检测精度都处于劣势,但是PANet的FPS为2帧/s,无法运用于实际中的实时目标检测;而MDS-YOLO的FPS为36帧/s,检测速度是PANet的18倍,在高性能的GPU上速度还会进一步提升,可以满足实时检测需求.

对比本文优化后的MDS-YOLO和原YOLOv4的实验结果可知,MDS-YOLO算法在MS COCO数据集上的平均检测精度AP_{COCO}从YOLOv4的43%上升至

44.5%,提升了1.5个百分点;而对于小目标的平均检测精度AP_S从YOLOv4的24.3%上升至27.6%,提升了3.3个百分点.同时,MDS-YOLO算法在MS COCO数据集上的检测速度也从31帧/s提升至36帧/s,提升了约16%.但是对于大目标的平均检测精度AP_L比YOLOv4下降了1.6个百分点,原因可能是图像分割操作导致特征图感受野减小,从而影响了对于大目标特征的识别和检测.但是总体来说,本文改进的MDS-YOLO算法在MS COCO数据集上无论是对于小目标的检测精度还是检测速度都有一定的提升.

另外,为了进一步说明本模型的有效性,本文绘制了不同模型的P-R曲线,从图2可看出,本文的结果优于其他五个模型.

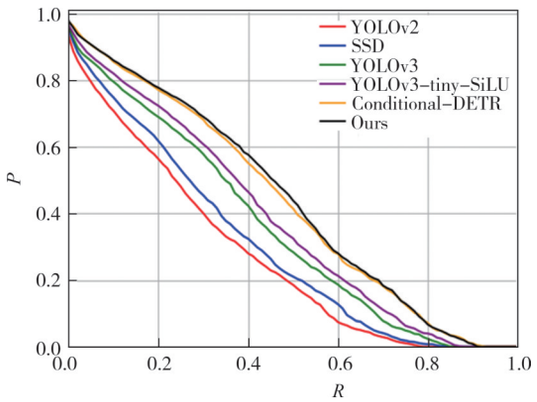


图2 训练迭代P-R曲线对比图

Fig.2 Comparison chart of P-R curves for training iterations

2.2.3 在Visdrone2019数据集上的检测结果

为了进一步验证MDS-YOLO算法对于小目标的检测性能,本文将改进后的MDS-YOLO算法和原YOLOv4算法在含有大量小目标的Visdrone2019无人机拍摄图像数据集的训练集上进行训练并在其测试集上进行检测,对比其检测速度和检测精度(见表5).数据集图像涵盖了居民区、城市街道、高速公路等复杂场景和不同光照、视角下的车辆目标,不同种类车辆无人机视角相似度较大,所以能更全面和深入地衡量算法对于小目标检测的性能.

YOLOv4和MDS-YOLO的输入尺寸均为512×512,因为Visdrone2019数据集的图像分辨率普遍较高,所以MDS-YOLO的横向分割超参数c_x和纵向分割超参数c_y均设置为2.同样地,本次实验也选择了一些具有代表性的目标检测算法加入对比,如Faster R-CNN[20]、FCOS[8]、RetinaNet[6]等.因为数据集中小目标占比极高,所以我们直接用多类别平均检测精度AP_{COCO}来衡量其对小目标的检测性能.

表5 MDS-YOLO与其他算法在Visdrone2019数据集上的实验结果对比

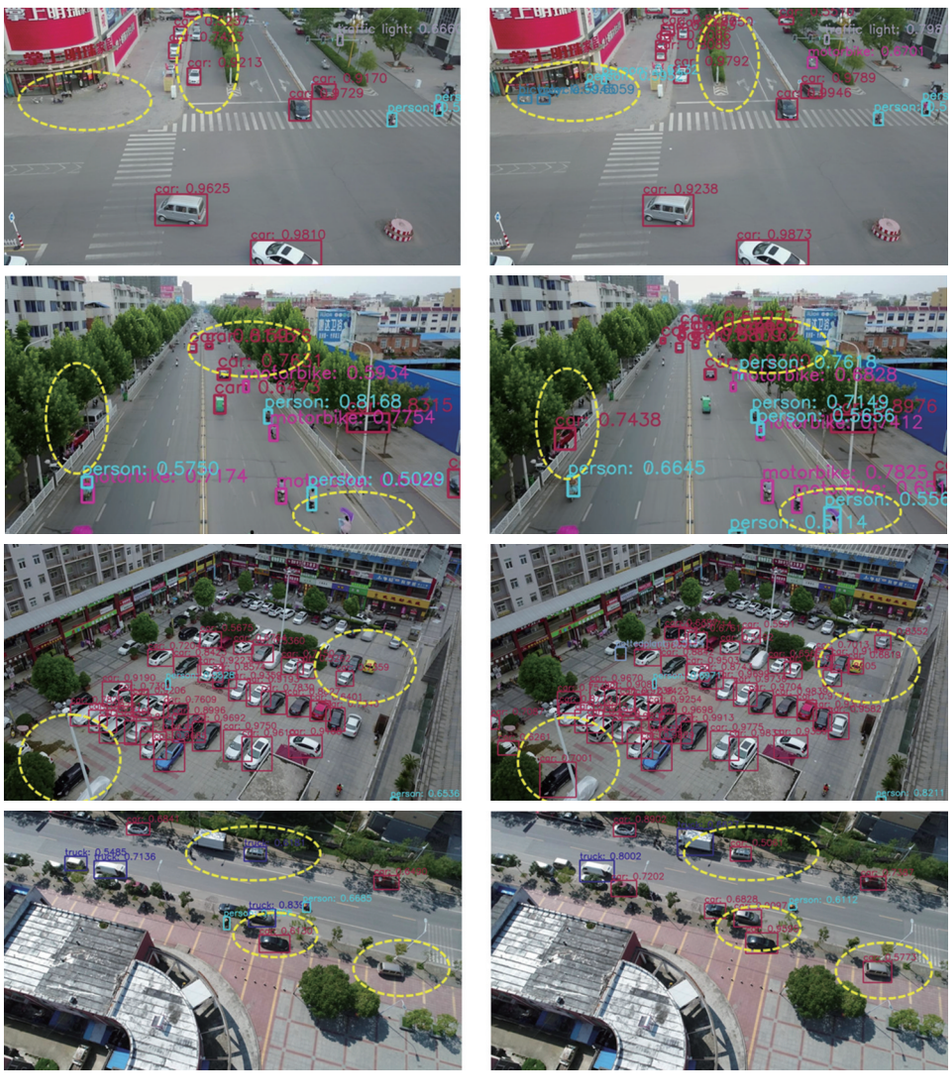
Tab. 5 Comparison of experimental results between MDS-YOLO and other algorithms on the Visdrone2019 dataset

算法	主干网络	输入尺寸/(像素×像素)	AP_{COCO}/%	AP_{50}/%	AP_{75}/%	FPS/(帧·s^{-1})
Faster R-CNN[20]	ResNet-50	600×600	12.7	22.8	11.9	7
SSD[24]	ResNet-101	513×513	14.3	25.7	12.1	—
RetinaNet[6]	ResNet-101-FPN	800×800	15.1	24.5	12.4	—
YOLOv3[25]	DarkNet-53	512×512	14.7	26.3	12.2	14
FCOS[8]	ResNet-101-FPN	800×800	15.6	28.7	14.8	
YOLOv4[10]	CSPDarknet-53	512×512	14.9	27.4	14.1	**31**
MDS-YOLO	MobileNetV3-Large	512×512	**16.3**	**30.1**	**14.9**	27

对比本文优化后的MDS-YOLO和原YOLOv4的实验结果可知,MDS-YOLO算法在Visdrone2019数据集上的检测速度稍有下降,FPS从31帧/s降到了27帧/s,原因可能是图像尺寸较大导致分割数增大,从而影响了检测速度.平均检测精度AP_{COCO}从YOLOv4的14.9%上升至16.3%,提升了1.4个百分点;IoU阈值为0.5时的检测精度AP_{50}从YOLOv4的27.4%上升至30.1%,提升了2.7个百分点.总体来说,在牺牲较少速度的情况下,检测精度得到了较大提升.

2.2.4 实际检测效果对比

图3中,左边是YOLOv4算法的实际检测效果,右边是本文改进后的MDS-YOLO算法的实际检测效果.对比黄色虚线区域(有矩形框表示识别到该目标)可以看出,MDS-YOLO对于高空视角图像中极小的目标,比如人、自行车、较远处的小车等的检测能力明显强于YOLOv4,特别是对处于图像边缘的小目标物体的识别率有显著提高,这与本文的实验结果相符.对比图3(a)和图3(b)的最后一行图像可知,相较于YOLOv4,MDS-YOLO能准确地区分不同角度和距离的小车与卡车,辨别相似物体的能力更强;不足之处在于对于图像中极远处的超小目标物体如小车、人等的检测还不够准确,且不能精准地区分摩托车和坐在摩托车上的人.



(a)YOLOv4检测效果　　(b)MDS-YOLO检测效果

图 3　YOLOv4 与 MDS-YOLO 实际检测效果对比

Fig.3　Comparison of actual detection performance between YOLOv4 and MDS-YOLO

3　结 论

为提升小目标检测性能的有效性,本文从输入图像、网络结构、损失函数三方面改进YOLOv4算法,提出了MDS-YOLO模型.通过实验验证,在MS COCO数据集上MDS-YOLO模型对于小目标的平均检测精度 AP_S 从 YOLOv4 的 24.3% 上升至27.6%,提升了3.3个百分点,且检测速度也提升了约16%;在Visdrone2019无人机数据集上,与YOLOv4相比,MDS-YOLO模型以牺牲较少的检测速度为代价,使检测精度提升了1.4个百分点.由此可见,本文模型的小目标检测效果更好、更高效.

参考文献

[1]　LIU L, OUYANG W L, WANG X G, et al. Deep learning for generic object detection: a survey[J]. International Journal of Computer Vision, 2020, 128(2): 261-318.

[2]　张伟, 庄幸涛, 王雪力, 等. DS-YOLO: 一种部署在无人机终端上的小目标实时检测算法[J]. 南京邮电大学学报(自然科学版), 2021, 41(1): 86-98.
ZHANG W, ZHUANG X T, WANG X L, et al. DS-YOLO: a real-time small object detection algorithm on UAVs[J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition), 2021, 41(1): 86-98. (in Chinese)

[3]　姚桐, 于雪媛, 王越, 等. 改进SSD无人机航拍小目标识别[J]. 舰船电子工程, 2020, 40(9): 162-166.
YAO T, YU X Y, WANG Y, et al. Improvement of small target recognition algorithm of aerial photography images based on SSD[J]. Ship Electronic Engineering, 2020, 40(9): 162-166.

(in Chinese)

[4] SINGH B, DAVIS L S. An analysis of scale invariance in object detection - SNIP[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA. IEEE, 2018:3578-3587.

[5] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection [C]//2017 IEEE International Conference on Computer Vision (ICCV). Venice, Italy. IEEE, 2017: 2999-3007.

[6] DUAN K W, BAI S, XIE L X, et al. CenterNet: keypoint triplets for object detection[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South). IEEE, 2019: 6568-6577.

[7] TIAN Z, SHEN C H, CHEN H, et al. FCOS: fully convolutional one-stage object detection [C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South). IEEE, 2019:9626-9635.

[8] TAN M X, PANG R M, LE Q V. EfficientDet: scalable and efficient object detection [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA. IEEE, 2020:10778-10787.

[9] 梁小林, 王欣怡, 黄雅娟, 等. 融合注意力模块的双结构金字塔场景解析网络[J]. 长沙理工大学学报(自然科学版), 2024, 21(5): 104-112.
LIANG X L, WANG X Y, HUANG Y J, et al. Two-structure pyramid scene parsing network with attention module[J]. Journal of Changsha University of Science & Technology (Natural Science), 2024, 21(5): 104-112.(in Chinese)

[10] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: optimal speed and accuracy of object detection[EB/OL]. 2020: 2004.10934. https://arxiv.org/abs/2004.10934v1.

[11] HOWARD A, SANDLER M, CHEN B, et al. Searching for MobileNetV3 [C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South). IEEE, 2019: 1314-1324.

[12] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [J]. Communications of the ACM, 2012, 60:84-90.

[13] GIRSHICK R. Fast R-CNN [C]//2015 IEEE International Conference on Computer Vision (ICCV). Santiago, Chile. IEEE, 2015: 1440-1448.

[14] CHOLLET F. Xception: deep learning with depthwise separable convolutions[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA. IEEE, 2017: 1800-1807.

[15] SANDLER M, HOWARD A, ZHU M L, et al. MobileNetV2: inverted residuals and linear bottlenecks [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA. IEEE, 2018:4510-4520.

[16] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection [C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA. IEEE, 2017:936-944.

[17] ZHENG Z H, WANG P, LIU W, et al. Distance-IoU loss: faster and better learning for bounding box regression[EB/OL]. 2019: 1911.08287.https://arxiv.org/abs/1911.08287v1.

[18] ZHANG Y F, REN W Q, ZHANG Z, et al. Focal and efficient IOU loss for accurate bounding box regression[J]. Neurocomputing, 2022, 506: 146-157.

[19] CHEN C Y, LIU M Y, TUZEL O, et al. R-CNN for small object detection [M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017:214-230.

[20] 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.

[21] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition [C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA. IEEE, 2016:770-778.

[22] HE K M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]// 2017 IEEE International Conference on Computer Vision (ICCV). Venice, Italy. IEEE, 2017:2980-2988.

[23] LIU S, QI L, QIN H F, et al. Path aggregation network for instance segmentation [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA. IEEE, 2018:8759-8768.

[24] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot MultiBox detector [M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016:21-37.

[25] FARHADI A, REDMON J. YOLOv3: an incremental improvement[EB/OL]. 2018: 1804.02767. https://arxiv. org/abs/1804. 02767.

[26] CARION N, MASSA F, SYNNAEVE G, et al. End-to-end object detection with transformers [M]//Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020: 213-229.

[27] WANG Y M, ZHANG X Y, YANG T, et al. Anchor DETR: query design for transformer-based detector [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(3): 2567-2575.

[28] MENG D P, CHEN X K, FAN Z J, et al. Conditional DETR for fast training convergence [C]//2021 IEEE/CVF International Conference on Computer Vision (ICCV). Montreal, QC, Canada. IEEE, 2021:3631-3640.

[29] LI F, ZHANG H, LIU S L, et al. DN-DETR: accelerate DETR training by introducing query DeNoising[C]//IEEE Transactions on Pattern Analysis and Machine Intelligence. IEEE, 2024: 2239-2251.

[30] 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]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Vancouver, BC, Canada. IEEE, 2023:7464-7475.