文章编号：1674-2974-202X(0X)-0000-00　DOI:10.16339/j.cnki.hdxbzkb.XXXXXXX

基于 MDS-YOLO 模型的小目标检测问题研究

朱恩文，梁婴，肖进文，梁小林†

(长沙理工大学 数学与统计学院, 湖南 长沙, 410114)

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，本文将轻量级网络 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，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，并对小目标具有较好的检测性能。为了验证模型的有效性，本文分别在 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%，对于小目标的检测精度提升了 3.3%，检测速度也从 31 张/秒提升到了 36 张/秒；在 Visdrone2019 数据集上，MDS-YOLOv4 算法将平均检测精度从 YOLOv4 的 14.9%提升到了 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；YOLO 算法；轻量级网络 MobileNetV3；IF-EIoU 损失函数；MS COCO 数据集

中图分类号：O 213; TP 181　　文献标志码：A

Research of Small Object Detection Based on MDS-YOLO Model

Zhu enwen, Liang zhao，Xiao jinwen，Liang Xiaolin†

(School of Mathematics and Statistics Science, Changsha University of Science and Technology,Changsha 410114,China)

Abstract: In order to solve the problem of large computation and low accuracy of the current mainstream algorithms for small target 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 is defined for small object detection。 Therefore, MDS-YOLO target detection model is constructed. This model has a high detection speed and good detection performance for small targets. In order to verify the effectiveness of the model, experiments are carried out on MS COCO data set and Visdrone2019 data set respectively. Compared with the YOLOv4 algorithm, on MS COCO data set, the average precision of the MDS-YOLO algorithm is improved by 1.5%, the detection accuracy of small targets is increased by 3.3%, and the detection speed is also increased from 31 images/SEC to 36 images/SEC. On the Visdrone2019 data set, the MDS-YOLOv4 algorithm increased the average detection accuracy from 14.9% of YOLOv4 to 16.3%. The experimental results show that the MDS-YOLO algorithm proposed in this paper can effectively improve the detection accuracy of small targets.

Key words: Small object detection; YOLOv4 algorithm; lightweight network Mobile-NetV3; IF-EIoU loss function; MS COCO dataset

收稿时间:

基金项目: 国家自然科学基金重点项目（51839002），湖南省自然科学基金资助项目(2021JJ30734)，湖南省研究生创新性课题（CX20220952）。

作者简介：朱恩文(1976—)，男，湖南湘阴，长沙理工大学，教授，博士

†通信作者: 梁小林, liang@csust.edu.cn。

引言

目标检测是计算机视觉中的一个重要任务，主要是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；水下无人潜航对小型生物的探测；无人机领域中对高空视角下的建筑物、桥梁、交通标志等各类小目标的检测；医学图像领域中对于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]。可以看出，小目标检测在很多领域中发挥着重要的作用，对改善人们的生产和生活具有重要意义。

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，它将容易被正确分类的样本的权重降低，以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 FCOS（Fully Convolutional One-Stage Object Detection，全卷积单阶段目标检测）算法，它采用了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 训练策略增加模型对小目标的泛化能力，并使用了 BiFPN（Bi-directional Feature Pyramid Network，双向特征金字塔网络）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。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检测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其算法进行了优化和改进，提出了 MDS-YOLO（MobileNet+Depthwise Separable+YOLO，轻量级网络+深度可分离+YOLO）模型。主要贡献为：（1）输入图像：先将原始图像分割成 N 张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，可以尽可能降低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，提升了算法对小目标的检测精度。（2）网络结构：将 YOLOv4 算法[9]的主干特征提取网络 CSPDarknet-53 替换成轻量级网络 MobileNetV3[10]，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降低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 损失函数。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

本算法是在 YOLOv4 算法的基础上从输入图像、网络结构、损失函数三个部分入手对

算法进行优化和改进。

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，之后会对经过预处理后的图像进行多次下采样得到输出特征图[11]。这两个过程都会丢失掉大量的图像信息，原始图像越大，损失的信息就越多，特别是对于图上只占据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）根据原始图像大小确定分割数 N

式（1）给出了图像分割数量：

$$N_x = \left\lfloor \frac{c_x p_y}{300} \right\rfloor, \quad N_y = \left\lfloor \frac{c_y p_x}{300} \right\rfloor \qquad (1)$$

其中，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，图像的分割数量 $N = N_x * N_y$ 是由图像本身的大小和 c_x, c_y 这两个超参数确定。当 c_x, c_y 固定不变时，图像横向像素 p_x 越多，纵向分割数 N_y 越大；图像纵向像素 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。

2）对分割后的局部图像设置重叠区域

为了有效地降低目标物体被分割截断的风险，本文将分割后的局部图像进行扩充重叠区域的操作。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

$$L_x = \frac{p_x}{N_x} + L'_x, L_y = \frac{p_y}{N_y} + L'_y \qquad (2)$$

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 L_x 和纵向尺寸 L_y 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，始终保持在合理的大小范围。

3）将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，所以当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 NMS（Non-Maximum Suppression，非极大值抑制）[12]操作来去除重复的候选框。

1.2 网络结构设计

网络结构与 YOLOv4[9]相似，由三部分构成：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[13]和 MobileNetV2 的具有线性瓶颈的倒残差结构[14]，还引入了轻量级注意力模块 SE[10]和一种新的激活函数 H-Swish[10]。该模型不仅能大幅减少模型的参数量和计算量，而且能有效地提升了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，分为三部分，分别对应下采样 32 倍、16 倍、8 倍的三个不同尺寸的特征图[15]。

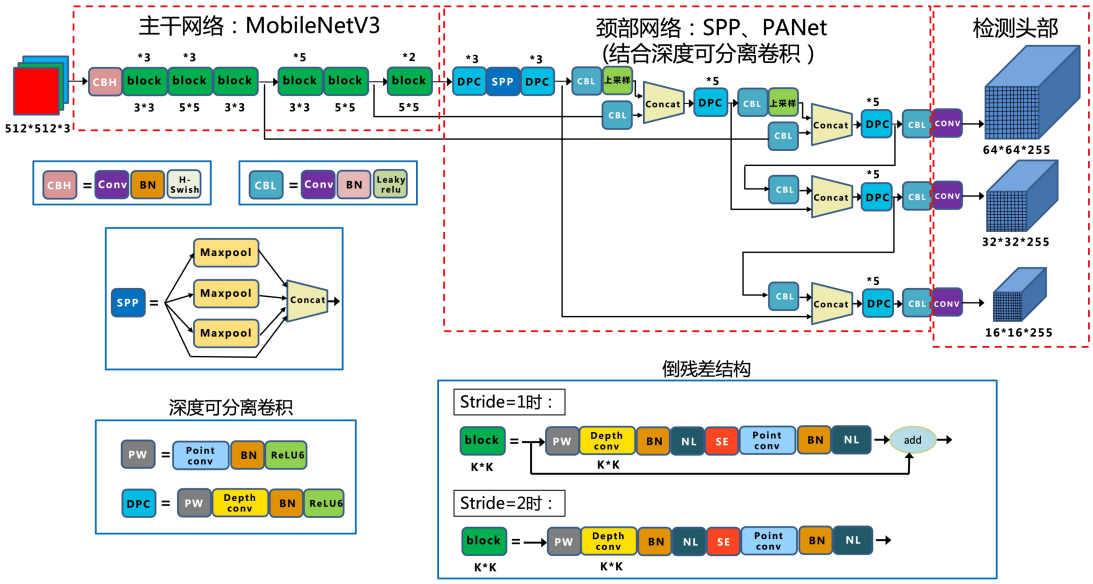


图 1 MDS-YOLO 的网络结构

1.3 损失函数优化

目标检测算法的损失函数由三部分组成：定位损失、置信度损失和类别损失。其中定位损失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；而类别损失用于衡量模型对检测框类别的预测精度[9]。本文主要针对 YOLOv4 损失函数的定位损失函数部分进行优化。YOLOv4 的定位损失函数为 CIoU Loss[16]，文献[17]通过整合 EIoU Loss 和 FocalL1 Loss，得到了最终的 Focal-EIoU 定位损失(见式（3）)

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

其中 $IoU(B,B^{gt})=\dfrac{\left|B\cap B^{gt}\right|}{\left|B\cup B^{gt}\right|}$，$EIoU(B,B^{gt})=1-IoU(B,B^{gt})+\dfrac{\rho^2(\mathbf{b},\mathbf{b}^{gt})}{c^2}+\dfrac{\rho^2(w,w^{gt})}{c_w^2}+\dfrac{\rho^2(h,h^{gt})}{c_h^2}$，

而 $\mathbf{b}, w, h$ 分别是预测框 B 的中心坐标及宽与高，$\mathbf{b}^{gt}, w^{gt}, h^{gt}$ 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。

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

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，将低质量预测框给予的损失权重越小，在训练时这会极大影响对于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，其定义如下：

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

与 Focal-EIoU Loss 的思想相反，FIoU Loss 对于 IoU 较小的低质量预测框给予较大的损失权重，使网络更加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

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

式（5）等式右边第一项为回归定位损失。其中，$K\times K$ 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）等式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）等式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）等式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，其它定义为中等目标）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了比较，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

配置	型号
操作系统	Win10
处理器	Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz
内存	64G
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。为了加快训练效率，采用了冻结训练的方式，在特征提取阶段使用了 MobileNetV3 在 MSCOCO 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 5e-4。

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）不同网络结构的推理速度、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）这两个操作的优化效果，本文在 MSCOCO 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）。

表 2 轻量级网络和深度可分离卷积对推理速度，参数量及 FLOPs 的影响

模型	网络结构	FPS（张/秒）	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（ours）	**61**	**19.10**	**10.16**

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，本文算法的 FPS 从 31 张/秒提升到了 61 张/秒，检测速度提升了约 97%，参数量由 64.36MB 减少到 19.10MB，FLOPs 由 45.84MB 下降到 10.16MB。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。

2）轻量级网络、深度可分离卷积、图像分割和定位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 损失函数这四个操作对于小目标检测精度的影响，本文在 MSCOCO 数据集上进行了平均精度（AP_{COCO}）与小目标检测精度（AP_S）[18]的对比实验。

表 3 轻量级网络、深度可分离卷积、图像分割和 IF-EIoU 损失函数对检测精度的影响

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
ours	**44.5**	**27.6**

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，其中，去“图像分割”对精度的影响最大，平均精度下降了 3.6%。小目标检测精度下降了 3.5%。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速度。

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 MSCOCO 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-YOLOv4（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[19]、Mask R-CNN[21]、SSD[23]、RetinaNet[5]、DETR[25]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-YOLOv4 算法的性能。

表 4 MDS-YOLOv4 与其他算法在 MSCOCO 数据集上的实验结果对比

算法	主干网络	输入尺寸 (平方像素)	AP_{COCO} (%)	AP_{50} (%)	AP_{75} (%)	AP_S (%)	AP_M (%)	AP_L (%)	FPS (张/秒)
两阶段算法：									

Faster R-CNN[19]	VGG-16	～1000×600	24.2	45.3	23.5	7.7	26.4	37.1	**4**
Faster R-CNN +++[20]	ResNet-101	～1000×600	34.9	55.7	37.4	15.6	38.7	50.9	—
Faster R-CNN w/ FPN[15]	ResNet-101	～1000×600	36.2	59.1	39.0	18.2	39.0	48.2	—
Mask R-CNN[21]	ResNeXt-101	～1300×800	39.8	62.3	43.3	22.1	43.2	51.2	2
PANet(multi-scale)[22]	ResNeXt-101	～1400×840	**47.4**	**67.2**	**51.8**	**30.1**	**51.7**	**60.0**	2
单阶段算法：									
YOLOv2[16]	DarkNet-19	544×544	21.6	44.0	19.2	5.0	22.4	35.5	21
SSD[23]	ResNet-101	513×513	31.2	50.4	33.3	10.2	34.5	49.8	—
RetinaNet[5]	ResNet-101-FPN	800×800	39.1	59.1	42.3	21.8	42.7	50.2	—
YOLOv3[24]	DarkNet-53	416×416	31.0	55.3	32.3	15.2	33.2	42.8	16
FCOS[7]	ResNet-101-FPN	800×1024	41.5	60.7	45.0	24.4	44.8	51.6	—
YOLOv4[9]	CSPDarknet-53	512×512	43.0	64.9	46.5	24.3	46.1	**55.2**	31
DETR[25]	ResNet-50	800×1066	42.0	62.4	44.2	20.5	45.8	**61.1**	—
Conditional-DETR[26]	ResNet-50	800×1066	43.0	64.0	45.7	22.7	**46.7**	**61.5**	—
Anchor-DETR[27]	ResNet-50	800×1066	42.1	63.1	44.9	22.3	46.2	**60.0**	—
YOLOv7-tiny-SiLU[28]	ELAN	640×640	38.7	56.7	41.7	18.8	42.4	51.9	—
MDS-YOLO（ours）	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 张/秒，无法运用于实际中的实时目标检测，而 MDS-YOLO 的 FPS 为 36 张/秒，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，可以满足实时检测要求。

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%上升到了 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%上升到了 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 张/秒提升到了 36 张/秒，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 1.4%，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，本文的结果优于其它五个模型。

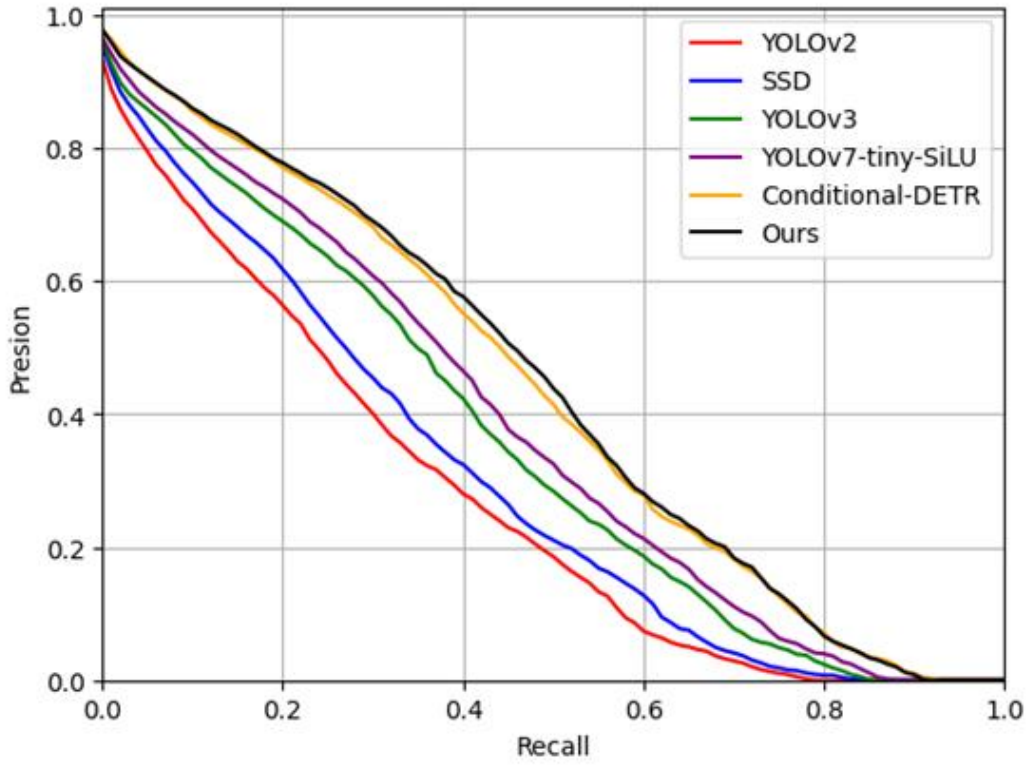


图 2　训练迭代 P-R 曲线对比图

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。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[19]、FCOS[7]、RetinaNet[5]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。

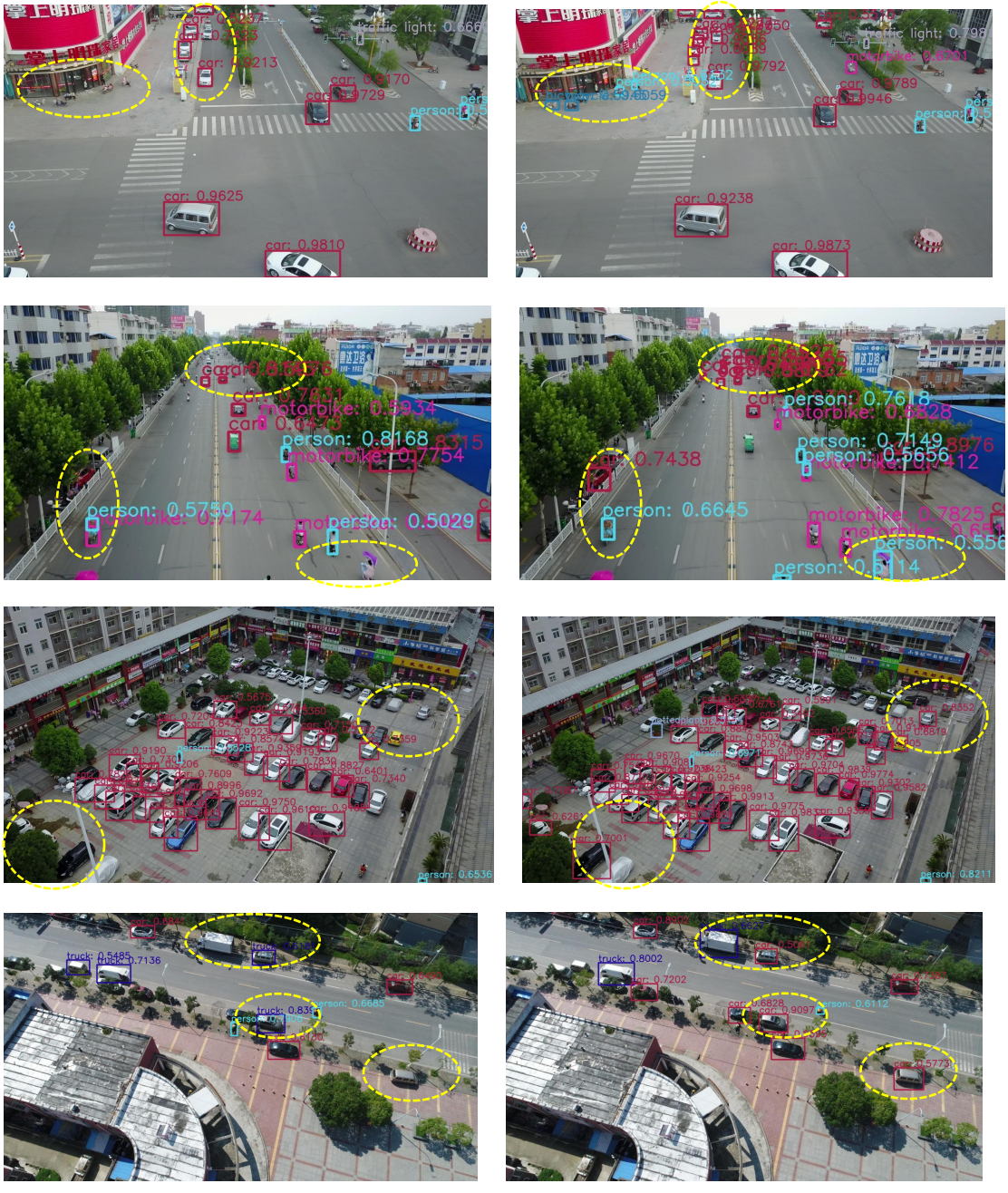
表 5 MDS-YOLOv4 与其他算法在 Visdrone2019 数据集上的实验结果对比

算法	主干网络	输入尺寸 (平方像素)	AP_{COCO} (%)	AP_{50} (%)	AP_{75} (%)	FPS (张/秒)
Faster R-CNN[19]	ResNet-50	600×600	12.7	22.8	11.9	7
SSD[23]	ResNet-101	513×513	14.3	25.7	12.1	—
RetinaNet[5]	ResNet-101-FPN	800×800	15.1	24.5	12.4	—
YOLOv3[24]	DarkNet-53	512×512	14.7	26.3	12.2	14
FCOS[7]	ResNet-101-FPN	800×800	15.6	28.7	14.8	

YOLOv4[9]	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-YOLOv4 算法在 Visdrone2019 数据集上的检测速度稍有下降，FPS 从 31 张/秒降到了 28 张/秒，原因可能是因为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 YOLO v4 的 14.9%上升到了 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%上升到了 30.1%，提升了 2.7 个百分点。总体来说在牺牲较小速度的情况下检测精度得到了提升。

2. 2. 4 实际检测效果对比



a) YOLOv4 检测效果　　b) MDS-YOLO 检测效果

图 3 YOLOv4 与 M-YOLOv4 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 YOLO，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 2(a)和 2(b)的最后一行图像可知，相比 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视角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。

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 MSCOCO 数据集上 MDS-YOLO 模型对于小目标的平均检测精度 APS 从 YOLOv4 的 24.3%上升到了 27.6%，提升了 3.3 个百分点，且检测速度也提升了约 19%；在 Visdrone2019 无人机数据集上，与 YOLOv4 相比，MDS-YOLO 模型以牺牲较小的检测速度为代价提升了 1.4 个百分比的检测精度。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, Wang X, 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, (01): 1-13.

[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(09): 162-166.

[4] Singh B, Davis S L. An analysis of scale invariance in object detection snip[C]//In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018: 3578-3587.

[5] Lin T Y, Goyal P, Girshick R, et al. Focal Loss for Dense Object Detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, PP(99): 2999-3007.

[6] Duan K, Bai S, Xie L, et al. CenterNet: Keypoint Triplets for Object Detection[C]//In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, 2019: 6569-6578.

[7] Tian Z, Shen C, Chen H, et al. FCOS: Fully Convolutional One-Stage Object Detection[C]//In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seoul, 2019: 9627-9636.

[8] Tan M, Pang R, Le Q V. EfficientDet: Scalable and Efficient Object Detection[C]//In Proceed-ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, 2020: 10781-10790.

[9] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. 2020. DIO：10.48550/arXiv: 2004.10934.

[10] Howard A, Sandler M, Chu G, et al. Searching for MobileNetV3[C]. International Conference on Computer Vision,2019:1314-1324.

[11] Krizhevsky A, Sutskever I, Hinton G. ImageNet Classification with Deep Convolutional Neural Networks[J]. Advances in Neural Information Processing Systems, 2012, 25(2): 1097-1105.

[12] Girshick R. Fast r-cnn[C]//In Proceedings of the IEEE International Conference on Computer Vision, Santiago, 2015: 1440-1448.

[13] Francois Chollet. Xception: Deep learning with depthwise separable convolutions[C]//In Procee-dings of the IEEE Confere nce on Computer Vision and Pattern Recognition, Honolulu，2017: 1800-1807.

[14] Sandler M, Howard A, Zhu M, et al. MobileNetV2: Inverted Residuals and Linear Bottlenecks[C]//In Proceedings of the I EEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018: 917-925.

[15] Lin T, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[C]//In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017: 2117-2125.

[16] Zheng Z, Wang P, Liu W, et al. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression[C]//In Proc eedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, 2019: 2146-2155.

[17] Yi-Fan Zhang, Weiqiang Ren, Zhang Zhanga, Zhen Jiaa, Liang Wanga, Tieniu Tan. Focal and Efficient IOU Loss for Accurate Bounding Box Regression. Neurocomputing,2022,506(9):146-157.https://doi.org/10.1016/j.neucom.2022.07.042.

[18] Chen C , Liu M Y , Tuzel O ,et al. R-CNN for Small Object Detection[C]//Asian Conference on Computer Vision.Sprin ger, Cham, 2016：214-230.DOI:10.1007/978-3-319-54193-8_14.

[19] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEE E transactions on pattern analysis and machine intelligence, 2016, 39(6): 1137-1149.

[20] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//In Proceedings of the Conference on Co mputer Vision and Pattern Recognition, Las Vegas, 2016: 770-778.

[21] He K, Gkioxari G, Dollar P, et al. Mask R-CNN[C]//In Proceedings of the IEEE International Conference on Computer Vision, Venice，2017: 2961-2969.

[22] Liu S, Qi L, Qin H, et al. Path Aggregation Network for Instance Segmentation[C]//In Proce-edings of the IEEE Confere nce on Computer Vision and Pattern Recognition, Salt Lake City, 2018: 8759-8768.

[23] Liu W, Anguelov D, Erhan D, et al. SSD: Single shot multibox detector[C]. In Proceedings of the European Conference on Computer Vision, Switzerland, 2016: 21-37.

[24] Farhadi A, Redmon J. Yolov3: An incremental improvement[J]. arXiv preprint arXiv: 1804.02767, 2018.

[25] Nicolas C, Carion N, Massa F, et al. End-to-end object detection with transformers[C]. European conference on computer vision. Cham: Springer International Publishing, 2020:213-229.

[26] Wang Y , Zhang X , Yang T, et al. Anchor DETR: Query Design for Transformer-Based Object Detection[J]. 2021.DOI: 10.48550/arXiv.2109.07107.

[27] Meng D, Chen X, Fan Z, et al. Conditional detr for fast training convergence[C]. In Proceedings of the IEEE/CVF intern ational conference on computer vision, 2021:3651-3660.

[28] Wang C Y, Alexey B, and Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023: 7464-7475.