第 59 卷 第 8 期/2022 年 4 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

针对弱小无人机目标的轻量级目标检测算法

蒋镕圻[1,2*]，叶泽聪[1,2]，彭月平[2**]，谢郭蓉[1,2]，杜衡[3]

[1]武警工程大学研究生大队，陕西 西安 710086；

[2]武警工程大学信息工程学院，陕西 西安 710086；

[3]新疆大学建筑工程学院，乌鲁木齐 新疆 830000

摘要 为解决无人机“滥用”带来的安全隐患，针对现有基于深度学习的无人机目标检测算法复杂度较高，导致模型训练耗时长、占用计算资源大、输入图像尺寸受限、检测速度慢等问题，提出了一种轻量级无人机目标检测(DTD-YOLOv4-tiny)算法。所提算法以YOLOv4-tiny为基础，通过K-means++聚类算法对Anchor box进行优化，并增加52×52尺寸特征图的检测头，拓展了算法对小目标的适用范围，再结合ShuffleNetv2轻量化骨干网络，使用reorg_layer下采样和sub-pixel上采样的方式，分别对YOLOv4-tiny算法的Backbone、Neck和Head进行优化，最终得到的模型大小仅为1.4 MB，浮点运算量(GFLOPs)仅为1.1的DTD-YOLOv4-tiny轻量级检测算法。实验结果表明，DTD-YOLOv4-tiny检测模型在不限制图像输入尺寸的同时，保证了较低的运算资源占用和高的检测实时性，同时降低参数量后的算法在面对原始大尺寸图像时也可以保持准确性。在Drone-vs-Bird 2017数据集上使用960×540尺寸的图像作为输入时，所提算法的平均精度(AP)@50值达到95%，在RTX2060显卡上的检测速度达到113 frame/s；在TIB-Net数据集上使用1920×1080尺寸的图像作为输入时，所提算法的AP@50值达到85.1%，在RTX2080Ti显卡上的检测速度达到119 frame/s。

关键词 图像处理；弱小无人机目标；DTD-YOLOv4-tiny；轻量级检测模型；实时目标检测

中图分类号 TP391.4 **文献标志码** A **doi**：10.3788/LOP202259.0810006

Lightweight Target Detection Algorithm for Small and Weak Drone Targets

Jiang Rongqi[1,2*], Ye Zecong[1,2], Peng Yueping[2**], Xie Guorong[1,2], Du Heng[3]

[1]*Graduate Team, Engineering University of PAP, Xi'an, Shaanxi* 710086, *China*;

[2]*School of Information Engineering, Engineering University of PAP, Xi'an, Shaanxi* 710086, *China*;

[3]*School of Civil Engineering, Xinjiang University, Urumqi, Xinjiang* 830000, *China*

Abstract To address the security risks associated with drone “abuse”, aiming at the high complexity of the existing deep learning-based drone target detection algorithm, which results in lengthy model trainings, large computing resources, limited input image size, and slow detection speed, a lightweight level drone target detection (DTD-YOLOv4-tiny) algorithm is proposed. The proposed algorithm is based on YOLOv4-tiny, and we optimized the Anchor box using the K-means++ clustering algorithm, added the detection head of the 52×52 size feature map to expand the scope of the algorithm for small targets, and combined it with the ShuffleNetv2 lightweight backbone network, and the reorg_layer downsample and sub-pixel upsample methods were used to optimize the Backbone, Neck, and Head of the YOLOv4-tiny algorithm. Eventually, we obtained the DTD-YOLOv4-tiny with a model size

收稿日期：2021-03-16；**修回日期**：2021-04-07；**录用日期**：2021-04-27

基金项目：武警工程大学科研创新团队课题(KYTD201803)、武警工程大学基础研究项目(WJY201905)

通信作者：*jjqqjjqq163@163.com；**percy001@163.com

of 1.4 MB and a floating-point calculation (GFLOPs) of 1.1, which is a lightweight detection technique. The experiments demonstrate that the DTD-YOLOv4-tiny detection model does not limit the image input size, while ensuring low computational resource occupation and high real-time detection. Simultaneously, the algorithm with reduced parameters can also maintain accuracy when facing the original large-scale image. When using 960×540 size image as input on the Drone-vs-Bird 2017 dataset, the average precision (AP) @50 of the proposed algorithm achieved 95%, and the detection speed on the RTX2060 graphics card attained 113 frame/s;when using 1920×1080 size image as input on the TIB-Net dataset, the AP@50 of the proposed algorithm achieved 85.1%, and the detection speed on the RTX2080Ti graphics card attained 119 frame/s.

Key words image processing; weak and small drone target; DTD-YOLOv4-tiny; lightweight detection model; real-time target detection

1 引 言

随着无人机技术的快速发展,无人机在多个行业占据了重要地位,而由无人机所引发的扰乱航空秩序、窃取隐私信息甚至恐怖袭击等事件也逐年攀升,已让其成为空中安防领域的重大隐患[1]。关于有效检测低空弱小无人机目标的研究是目前空中安防领域的重点问题。

随着深度学习理论的不断发展,以深度卷积神经网络为框架的视觉图像目标检测算法[2]相继被提出。目前基于深度学习的目标检测算法已被广泛研究且应用于多种任务,包括对视频图像中无人机目标的检测[3]等。但无人机目标在视频图像中占据像素极小,并与飞鸟等干扰物对比度极低,很容易混入背景图像中从而得不到有效的检测。因此,现阶段的无人机目标检测算法难以有效检测,存在较高的误检率和漏检率。针对此问题,不少学者基于目前两阶段目标检测算法 Fast R-CNN[4]、Faster R-CNN[5]和单目检测算法 YOLO[6-7]、SSD[8-9]给出了优化方案[10-15]。目前,研究人员的研究重点多放在如何提升算法检测精度上,但现阶段的算法模型复杂度较高,导致模型训练耗时长、占用计算资源大、输入图像尺寸受限、检测速度慢等问题,难以满足工业应用中的实时性要求和广泛的应用部署要求,且输入图像尺寸受限会丢失掉图像中小目标的大量信息,这无疑对小目标检测带来了巨大的限制。

本文针对现有算法中存在的问题,从尽可能降低模型复杂程度、尽可能保持原图像尺寸输入到网络模型、尽可能利用浅层特征图中小目标信息3个原则出发,提出了一种轻量级无人机目标检测(DTD-YOLOv4-tiny)算法。所提算法基于 YOLOv4-tiny 目标检测算法[16]进行优化,对 Drone-vs-Bird 2017 数据集[17]和文献[18]中提供的 TIB-Net 数据集中无人机目标情况进行了分析,总结了 YOLOv4-tiny 算法在检测无人机目标中存在的问题。针对 YOLOv4-tiny 算法存在的问题,首先通过 K-means++聚类算法对 Anchor box 进行优化,增加 52×52 尺寸特征图的检测头,拓展了算法对小目标的适用范围,并结合 ShuffleNetv2 轻量化骨干网络[19],使用 reorg_layer 下采样和基于 sub-pixel Conv 层优化得到的 sub-pixel 上采样的方式,分别对 YOLOv4-tiny 算法的 Backbone、Neck 和 Head 进行优化,最终得到模型大小仅为 1.4 MB,浮点运算量(GFLOPs)仅为 1.1 的 DTD-YOLOv4-tiny 轻量级检测算法。实验结果表明,DTD-YOLOv4-tiny 检测模型在不限制图像输入尺寸的情况下,保证了较低的运算资源占用和高的检测实时性,同时降低参数量后的算法在面对原始大尺寸图像时也可以保持准确性。

在 Drone-vs-Bird 2017 数据集上使用 960×540 尺寸的图像作为输入时,所提算法的平均精度(AP)@50 值达到 95%,在 RTX2060 显卡上的检测速度达到 113 frame/s;在 TIB-Net 数据集上使用 1920×1080 尺寸的图像作为输入时,所提算法的 AP@50 值达到 85.1%,在 RTX2080Ti 显卡上的检测速度达到 119 frame/s。

2 数据集

2.1 数据集介绍

由于目前缺乏公开可用的大规模无人机目标数据集[20],本实验组分别在 Drone-vs-Bird 2017 数据集[17]和 TIB-Net 数据集[18]两个无人机数据集进行了训练和测试。

Drone-vs-Bird 2017 数据集是“Drone-vs-Bird detection challenge”[17]大赛上中使用的数据集,该数据集并未公开,需要签署相关规定来获取此数据集,数据集由 EU-H2020-SafeShore 项目提供。

Drone-vs-Bird 2017 数据集(以下简称 Dataset A)由5个视频组成,数据集中目标包含5个不同场景、不同飞行距离下的无人机,共有 3345 帧图片,如图 1(a)所示。TIB-Net 数据集(以下简称 Dataset B)是 2020 年 Sun 等[18]为验证 TIB-Net 无人机目标检测算法使用的无人机目标数据集,如图 1(b)所示。该数据集共包含 2850 张不同场景下的图片,相较于 Dataset A 有更大的背景差和复杂的环境。

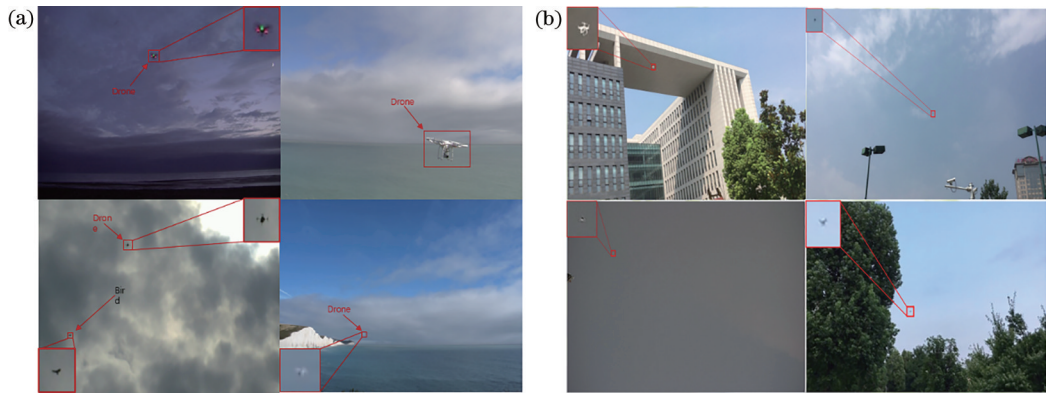

图 1　数据集部分图片。(a) Dataset A;(b) Dataset B

Fig. 1　Partial pictures of datasets. (a) Dataset A; (b) Dataset B

对两个数据集中的目标进行了统计分析,将 Dataset A 拆分成 3345 张 960 pixel×540 pixel 的图片,并生成对应的 3345 个 xml 标签文件,标签文件包含图片大小、标签信息“Drone”及目标“x_{min}”“y_{min}”“x_{max}”“y_{max}”位置等信息;Dataset B 使用原 VOC 格式标签。通过提取标签文件中每个目标的位置信息,对目标的先验框大小进行了统计分析,得到无人机目标 Bounding box 的像素面积(图 2 中的横坐标 Drone size),最终结果如图 2 所示。

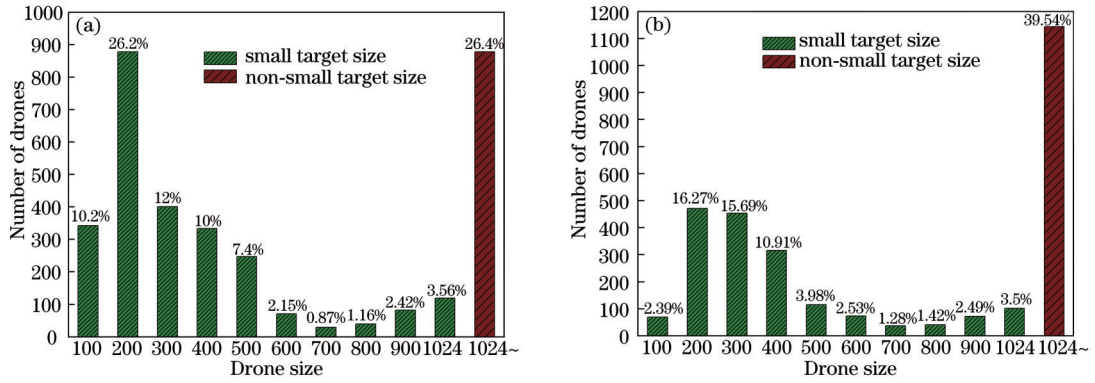

图 2　两个数据集中无人机目标大小情况分析。(a) Dataset A;(b) Dataset B

Fig. 2　Analysis of size of drone targets in two datasets. (a) Dataset A; (b) Dataset B

依据 COCO 数据集的评估标准,将小于 32 pixel×32 pixel 的目标认定为小目标[21]。从图 2 可以看出,无人机目标在两个数据集中基本都以小目标的形式出现,并且像素面积低于 500 的极小目标占到了数据集中一半。

2.2　训练集和测试集的划分

将 Dataset A 中的 3345 张图片按照 7∶3 的比例随机分配为训练集和测试集;Dataset B 则是按照 7.5∶2.5 的比例随机分配为训练集和测试集。两个数据集标签均处理成 YOLO 格式。

3　DTD-YOLOv4-tiny 算法介绍

3.1　YOLOv4-tiny 算法介绍

3.1.1　算法模型结构

YOLOv4-tiny 目标检测算法[16]是在 YOLOv4[6]基础上提出的一种轻量的单阶段目标检测算法。在 GeForce RTX 2070 GPU 硬件平台并通过 C++ 语言搭建的检测模型中,不同算法在 MS COCO 数据集的性能如表 1 所示。从表中可以看出,与 YOLOv4 相比,YOLOv4-tiny 的浮点运算量减小了 88.5%,检测速度提升至 330 frame·s^{-1},是 YOLOv4

表 1　不同算法在 MS COCO 数据集的性能对比

Table 1　Performance comparison of different algorithms in MS COCO dataset

Algorithm	Image size	mAP@50 /%	GFLOPs	Detection speed /(frame·s^{-1})
YOLOv4	416	65.7	60.1	55
YOLOv3-tiny	416	33.1	5.6	345
YOLOv4-tiny	416	40.2	6.9	330

检测速度的 6 倍；而检测精度较上一代 YOLOv3-tiny 算法有大幅度的提升，在 MS COCO 数据集上的 mean average precision（mAP）@50 达到了 40.2%。

YOLOv4-tiny 模型由骨干网络（backbone）、颈部（neck）、检测头（head）组成，模型结构如图 3（a）所示。YOLOv4-tiny 使用 CBL 卷积模块，即融合批量归一化（BN）和 Leaky ReLU 激活函数的卷积层，骨干网络为 CSPDarknet53-tiny，CSPBlock 是其核心模块，如图 3（b）所示。CSPBlock 将特征图通道数（C）等分为 A、B 两部分，并通过两次级联（Concat）特征融合对特征进行合并。这种类似于残差模块 ResBlock 的结构，能有效改善随着网络层次的加深所导致的梯度消失或爆炸的问题。同时，将特征图等分至两条路径上传播，不仅降低了模型整体的参数量和计算量，也增强了输出特征图梯度信息的相关性差异。且与残差模块 ResBlock 相比，CSPBlock 模块能提升网络的学习能力，提升检测精度。模型颈部使用特征金字塔网络（FPN）[22]来融合不同层次的特征图信息。检测头通过使用 13×13 和 26×26 两个不同尺寸的特征图来预测检测结果，减小了模型参数量，提升了检测速度。

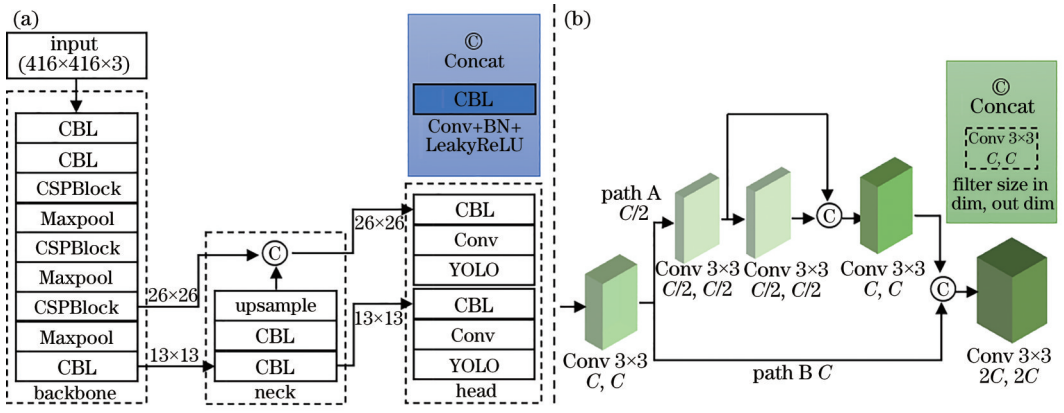

图 3　YOLOv4-tiny 结构图。（a）YOLOv4-tiny；（b）CSPBlock

Fig. 3　Structure diagram of YOLOv4-tiny algorithm. (a) YOLOv4-tiny; (b) CSPBlock

3.1.2　YOLOv4-tiny 算法检测流程

YOLOv4-tiny 以回归和分类的思想实现目标预测，检测原理[15]与 YOLOv3 算法相同。在 MS COCO 数据集上，YOLOv4-tiny 首先将图像的输入尺寸调整为 416×416，然后通过 CSPDarknet53-tiny 骨干网络的前 2 个卷积层和 3 个 Maxpool 层实现特征图的降采样，并输出 13×13 的特征图；其次利用网络颈部 FPN 结构进行浅层信息和高语义信息的特征融合；最后检测头进行预测时，对 13×13 和 26×26 两个不同尺度特征图进行预测，每个尺度的检测头分别预设 3 个通过聚类产生的先验框，并在特征图的每个栅格中预测边界框的 4 个偏移量、目标置信度和 80 类预测值。因此，最后的特征图是一个 $N\times N\times[(4+1+80)\times3]$ 的张量，其中 $N\times N$ 表示特征图的尺度。通过对每个栅格内的目标进行检测，并对检测出的先验框进行偏移量的调整和优化，可以得到多个预测的 Bounding box，并使用非最大值抑制（NMS）筛选出交并比（IoU）最大的预测框作为预测结果。

3.2　DTD-YOLOv4-tiny 算法

视频图像中，无人机通常以小目标的形式出现，而 YOLOv4-tiny 对于小目标的检测并不友好，主要原因如下。

1）输入图像尺寸被压缩。大多视频图像尺寸为 1920×1080，但由于目前检测算法模型复杂程度较高，且计算资源受限，通常会对输入图像进行压缩处理，但压缩图片尺寸会丢失掉大量的小目标信息，导致网络模型在输入端就限制了能检测到的目

标大小范围。

2) 小目标特征提取能力受限。当无人机目标尺寸为32×32时，进行5次下采样操作后，目标特征在骨干网络输出的13×13特征图中的尺寸仅为1×1；而当目标更小时，其在网络更深层次的特征图所表现出的有效特征几乎没有。

3) 检测头检测范围受限。由于YOLOv4-tiny只使用13×13和26×26这两种尺度的特征图进行检测，每个栅格可近似映射到原图像中32×32和16×16的尺度。而在无人机数据集Dataset A、Dataset B中像素面积小于300的无人机目标占据了数据集的1/3，由此可见，YOLOv4-tiny算法的检测范围并不能完全覆盖数据集中的目标。

4) 预设的Anchor box大小不匹配。由第3.1节可知，YOLOv4-tiny算法预设了6个Anchor box=(10,14)，(23,27)，(37,58)，(81,82)，(135,169)，(344,319)，这些Anchor box是在PASCAL VOC数据集上通过边框聚类得到的。从原预设的Anchor box宽高可见，其检测的目标差异较大，在普通场景的目标检测任务上有较好的适用性。但针对目标大小较为极端的数据集，原预设的Anchor box会导致检测器筛选不出合适的Bounding box，从而严重影响模型的性能。

针对上述问题，所提DTD-YOLOv4-tiny分别从Backbone、Head、Neck 3方面进行了改进，DTD-YOLOv4-tiny具体结构如图4所示。

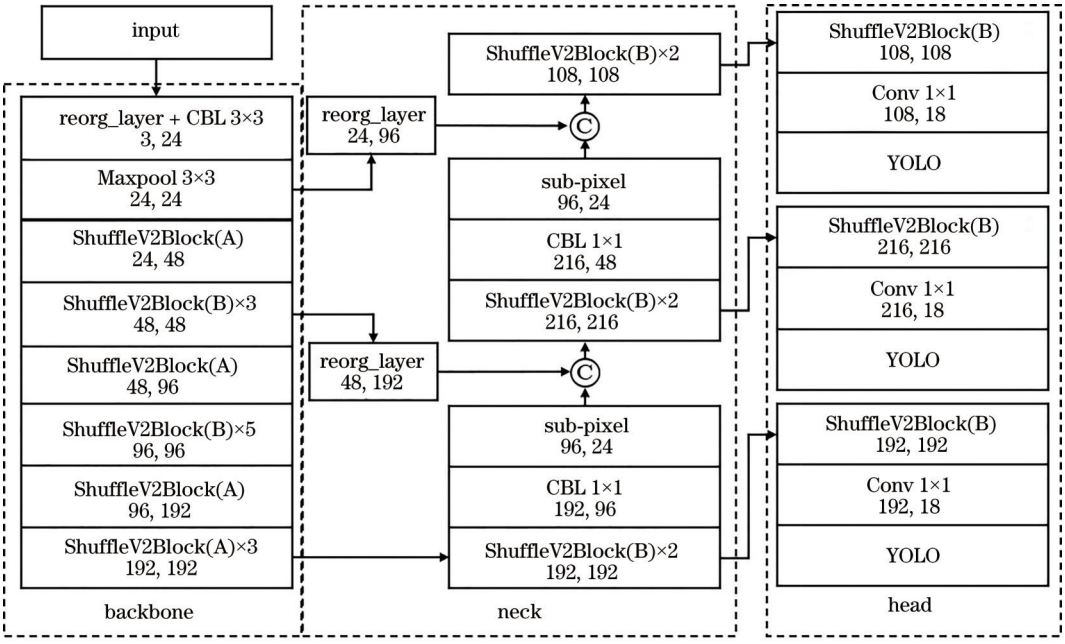

图4　DTD-YOLOv4-tiny模型结构图

Fig. 4　Structure diagram of DTD-YOLOv4-tiny model

3.2.1　Backbone改进

为尽可能地使输入图像尺寸更大，在保证图像中小目标信息不丢失的同时，平衡大尺寸输入图片消耗的计算资源，本实验组使用ShuffleNetV2对骨干网络进行优化。ShuffleNetV2[19]是目前轻量级神经网络中的常用网络。ShuffleNetV2主要由两种ShuffleV2Block模块构成，如图5(a)所示，ShuffleV2Block(A)用于下采样和通道数调整，将输入大小为$N\times N$，通道数为C的特征图分别通过步长为2的Depthwise Convolution (DWConv)分组卷积进行下采样，并通过Concat进行通道拼接，最后使用Channel Shuffle[23]完成通道混洗达到不同分组特征图之间信息融合的目的，最终输出大小为$N/2\times N/2$、通道数为$2C$的特征图。ShuffleV2Block(B)主要用于在保证降低参数量和计算量的情况下，提取特征信息。ShuffleNetV2的骨干网络结构如图5(b)所示，图中每一个模块下的数字表示输入与输出的通道数，stage表示一次下采样后的卷积阶段。

本实验组使用改进的ShuffleNetV2代替YOLOv4-tiny中使用的CSPDarknet53-tiny作为骨干网络，改进的ShuffleNetV2结构如图5(c)所示。为减少小目标信息的丢失，将原网络中第1层步长为2、大小为3×3的卷积层下采样改为reorg_layer

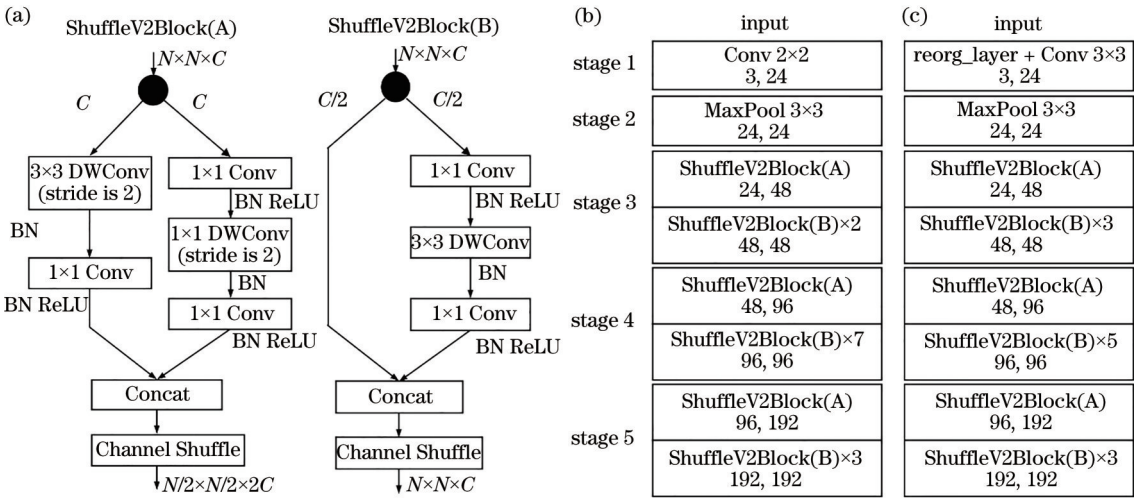

图 5　ShuffleNetV2 和改进后的骨干网络结构。(a) ShuffleV2Block;(b) ShuffleNetV2 骨干网络;(c)所提算法骨干网络

Fig. 5　ShuffleNetV2 and improved backbone network structure. (a) ShuffleV2Block; (b) backbone network of ShuffleNetV2; (c) backbone network of proposed algorithm

的下采样方式,并在其后增加一层大小为 3×3 的卷积来增大感受野,同时在第 4 次下采样阶段中将原网络中的 7 个 ShuffleV2Block(B)模块减少到 5 个,最终形成新的骨干网络。

3.2.2　Head 改进

为了解决 YOLOv4-tiny 中原预设 Anchor box 在小目标数据集上 Bounding box 检出率低的问题,本实验组针对数据集中目标使用 K-means++聚类算法进行边界框聚类分析。与传统的 K-means 聚类算法[24]相比,K-means++优化了初始点的选择,能显著改善分类结果的误差,从而获得更适合小目标数据集的 Anchor box,提高小目标检测的精度。K-means++算法首先随机选取某一个样本目标框区域作为初始聚类中心,然后计算每个样本 x_i 与已有聚类中心点的距离 $D(x)$,并计算每个样本被选为下一个聚类中心的概率 $P(x)$。

$$P(x)=\frac{D(x)}{\sum D(x)^2}。\tag{1}$$

然后通过轮盘法选出下一个聚类中心,重复计算距离 $D(x)$ 和概率 $P(x)$,直到得出 K 个目标框。最后不断重复计算每个样本到聚类中心点的距离,把该样本点划分到距离最小的聚类中心的类中并更新聚类中心,直到得到的 Anchor box 的大小不再发生改变。

为了拓展模型算法对极小目标的检测范围,本实验组将 YOLOv4-tiny 检测头从 2 个拓展到 3 个,增加了检测 52×52 尺寸特征图的 YOLO 检测头。同时为减小模型的参数量和计算量,将 CBL+Conv+YOLO 的 YOLO 检测头结构(如图 3 所示)改为 ShuffleV2Block(B)+Conv+YOLO 的结构(如图 4 所示)。

3.2.3　Neck 改进

FPN 作为网络模型的颈部结构,在提升模型对不同尺度的目标检测性能上发挥着至关重要的作用。FPN 以多尺度特征融合的方式,将骨干网络中浅层特征信息与深层语义信息进行融合,在不牺牲语义信息的同时,有效改善了小目标因卷积神经网络的下采样导致的特征信息丢失的问题,FPN[4]已经成为现阶段目标检测模型中 Neck 结构的标配。

图 6 为不同检测模型 FPN 的结构图,图中每个模型的输入尺寸默认为 416×416,S_i 表示第 i 次下采样后的阶段,P_i 表示颈部输出的特征图金字塔,特征图 P_i 的尺寸对应骨干网络中相同角标数值 S_i 的特征图尺寸。YOLOv4-tiny 模型使用 FPN 结构在一定程度上改善了其对小目标检测的性能,但对于极小目标的细节信息的利用依然受限,且随着网络不断下采样,无人机目标的特征信息依然被大量丢弃,如图 6(a)所示。由于降低骨干网络下采样频率会导致网络中感受野降低,因此本实验组保留骨干网络下采样次数,对网络颈部 FPN 进行优化,优化后的模型结构如图 6(c)所示。通过引入 reorg_layer 层将骨干网络中保留无人机目标基本特征的更浅层特征图与高层丰富的语义信息特征图相融合,并将 YOLO 中采用的 upsample 上采样层改为 sub-pixel 上采样,实现网络对浅层信息利用的最大化,

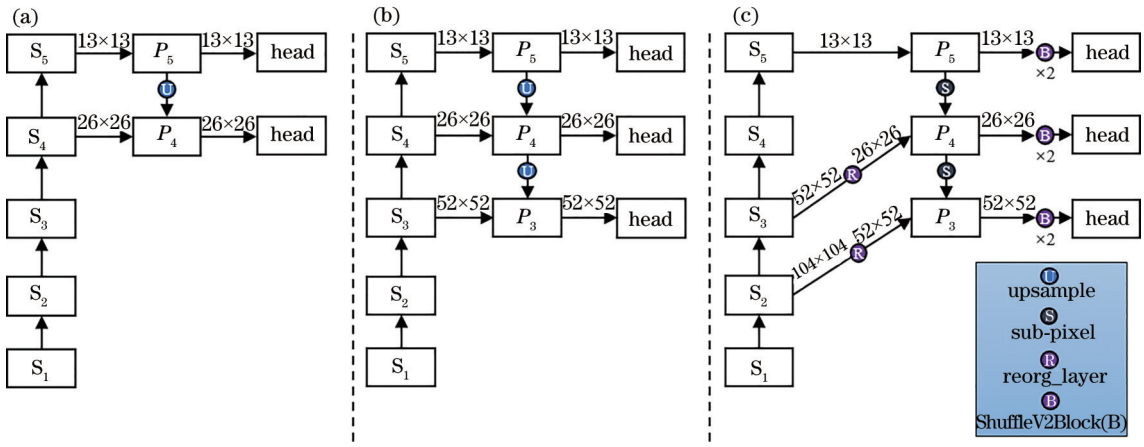

图 6　不同检测模型FPN结构对比。(a) YOLOv4-tiny；(b) YOLOv4-tiny (YOLO-Head enhancement)；(c) DTD-YOLOv4-tiny

Fig. 6　FPN structure comparison of different detection models. (a) YOLOv4-tiny; (b) YOLOv4-tiny (YOLO-Head enhancement); (c) DTD-YOLOv4-tiny

最后在每个Concat特征融合后加入两层ShuffleV2 Block(B)模块用于提高感受野并充分融合特征信息。

深度卷积网络中最初的几层特征图中包含了大量的目标细节信息，而对于小目标来说，高效地利用浅层细节信息能有效提升网络对小目标特征的学习能力，因此本实验组改变FPN的结构，利用S_2和S_3浅层特征图信息在颈部进行特征融合，如图6(c)所示，其中reorg_layer[25]是一种独特的下采样方式，可以将$N\times N\times C$的特征图转换为$(N/2)\times(N/2)\times 4C$的特征图，转换方式如图7所示。

图7中展示了$16\times16\times C$的特征图通过reorg_layer层后得到$8\times8\times4C$特征图的过程。从图7可以看出，相比于Maxpool或通过步长为3的卷积操作进行下采样的方式，reorg_layer可以更好地保留目标的细节信息。将S_2和S_3两个阶段得到的104×104和52×52特征图分别通过reorg_layer得到52×52和26×26的特征图，并与网络颈部相同尺度特征图进行Concat通道拼接的特征融合操作，

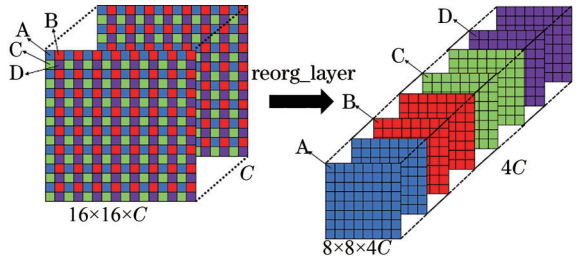

图 7　Reorg_layer工作原理

Fig. 7　Working principle of reorg_layer

以输出无人机目标细节特征信息更加丰富特征图用于检测。

为了还原出细节更加丰富的高分辨特征图，文献[26]提出了一种新的upscale方法：sub-pixel Conv。不同于YOLO中使用的upsample使用最近邻插值的上采样方式，sub-pixel Conv先将$N\times N\times C$的低分辨率特征图通过卷积的方式得到$N\times N\times 4C$的新特征图，再通过对特征图进行周期筛选的方式重新排列组合，从而得到新的$2N\times2N\times C$高分辨率特征图，具体过程如图8(a)所示。考虑到sub-

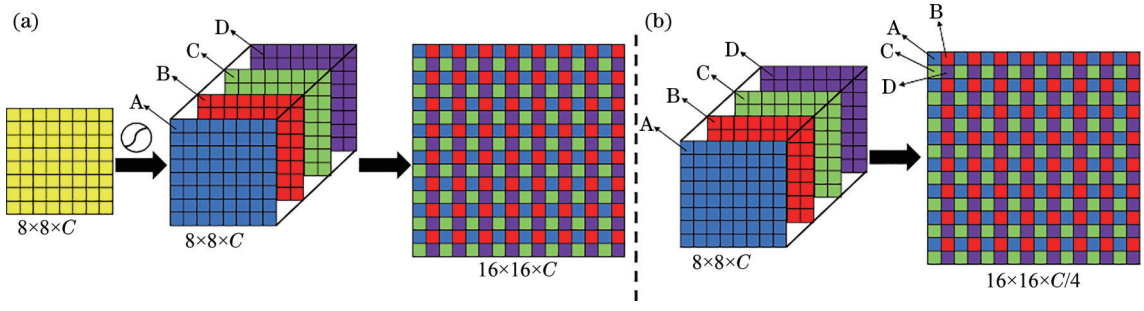

图 8　Sub-pixel Conv和sub-pixel的工作原理。(a) Sub-pixel Conv；(b) sub-pixel

Fig. 8　Working principles of sub-pixel Conv and sub-pixel. (a) Sub-pixel Conv; (b) sub-pixel

pixel Conv会带来额外的卷积操作，会增加模型参数量和计算成本，本实验组去掉了sub-pixel Conv中的卷积层，直接使用sub-pixel操作代替FPN中upsample上采样层，来实现特征图的尺度放大。sub-pixel操作的具体过程如图8(b)所示。

4　实验与结果分析

本实验硬件平台为Intel Core i7-8700K CPU，GeForce RTX 2060 6 GB GPU；软件使用Ubuntu系统，Python 3.7，PyTorch 1.5.0深度学习框架。

4.1　实施细节和评估指标

本实验基于Dataset A和Dataset B两个数据集。算法阶段初始学习率为0.01，使用模拟余弦退火学习率，动量为0.9，衰减系数为0.0005，使用Adam梯度优化算法进行epoch为200的迭代训练，并采用Mosaic[6]进行在线数据增强。

实验使用模型参数量(Parameters)、浮点运算量、检测速度和模型大小4项指标来对模型进行评估，当检测速度保持在25 frame/s以上时能基本满足实时性的要求。采用准确率(P)、召回率(R)、阈值IOU为0.5的平均精度(AP)3项指标来评判算法的检测性能[27]，P、R和AP的表达式分别为

$$P=\frac{X_{\mathrm{TP}}}{X_{\mathrm{TP}}+X_{\mathrm{FP}}},\tag{2}$$

$$R=\frac{X_{\mathrm{TP}}}{X_{\mathrm{TP}}+X_{\mathrm{FN}}},\tag{3}$$

$$R_{\mathrm{AP}}=\int_0^1 P(R)\mathrm{d}R,\tag{4}$$

式中：X_{TP}表示被正确检测出来的目标数；X_{FP}表示被错误检出的目标数；X_{FN}表示没有被检测出来的目标数。

4.2　消融实验

在数据集Dataset A上进行训练和测试，以YOLOv4-tiny得到的测试结果为Baseline，并以增量的方式逐步验证所提算法各优化方式的有效性。实验结果如表2所示，其中训练和测试的batch size设为16。

表2　DTD-YOLOv4-tiny算法消融实验结果

Table 2　Ablation experiment results of DTD-YOLOv4-tiny algorithm

Model	Head improve			Neck improve			Backbone Improve		AP / %	Parameters	GFLOPs
	Anchor box imporve	YOLO-Head enhancement	ShuffleV2 Block(B)	reorg layer	sub pixel	Shufflev2 Block(B)	ShuffleNetv2 Backbone	Proposed Backbone			
YOLOv4-tiny (Baseline)									43.1	5.8×10^6	12.1
Imporve A	√								84.1	5.8×10^6	12.1
Imporve B	√	√							91.3	6.1×10^6	14.3
Imporve C	√	√		√					93.6	6.8×10^6	17.1
Imporve D	√	√		√	√				94.5	6.5×10^6	16.1
Imporve E	√	√		√	√		√		87.1	0.65×10^6	1.9
Imporve F	√	√		√	√			√	87.4	0.64×10^6	2.2
Imporve G	√	√	√	√	√			√	85.3	0.27×10^6	0.9
DTD-YOLOv4-tiny	√	√	√	√	√	√		√	89.4	0.32×10^6	1.1

从表2中实验A可以看出：对于小目标数据集，Anchor box的大小很大程度上决定着检测模型在数据集上发挥的性能，通过预先对数据集中目标大小进行聚类，AP提升了41个百分点；对比实验A、B可以看出，拓宽检测头的检测范围也能改善模型的性能，AP又提升了7.2个百分点；实验C、D表明了sub-pixel上采样和reorg_layer下采样在改善小目标检测上的有效性；实验E、F、G通过将骨干网络替换为更轻量级的ShuffleNetV2 Backbone，轻量化了网络模型，模型参数和浮点运算量仅约为原网络模型的1/10，但AP也有所下降。通过使用改善后的ShuffleNetV2 Backbone，并在模型颈部和头部分别加入Shufflev2 Block(B)轻量级模块，所提DTD-YOLOv4-tiny算法最终将AP提升到89.4%，同时参数量仅有0.32×10^6，浮点运算量仅为1.1，使网络模型可以在消耗少量运算资源的同时处理更大尺寸的图像。

4.3 横向对比实验

为验证DTD-YOLOv4-tiny算法的有效性，分别在数据集Dataset A和Dataset B上对比了不同目标检测算法的性能，具体结果如表3和表4所示。

从表3和表4的对比可以看出，DTD-YOLOv4-tiny算法减去了原网络大量的参数，模型大小减小到1.4 MB，仅为目前最小的无人机目标检测算法TIB-Net的2倍。由于参数量的降低，所提算法在相同输入尺寸上的检测性能与其他算法有一定差距，但所提算法大大降低了浮点运算量和参数量。所提DTD-YOLOv4-tiny算法能够支持输入原始尺寸的图像，且仅占用少量的计算资源（表3中GPU指标一栏展示了在batch size为4时不同算法消耗的显存资源），当在数据集Dataset A上以原始960×540图像尺寸及在数据集Dataset B上以原始1920×1080图像尺寸输入时，AP@50分别达到了95%和85.1%。

表3　Dataset A数据集上不同算法性能对比

Table 3　4 Performance comparison of different algorithms on Dataset A

Algorithm	Image size	P /%	R /%	AP /%	Parameters	GFLOPs	GPU /GB
YOLOv3-SPP	416×234	93.5	96.2	96.8	625×10^6	116.9	3.85
YOLOv4	416×234	85.6	96.6	95.5	639×10^6	105.9	5.6
YOLOv4-tiny	416×234	81.5	87.6	84.1	5.87×10^6	12.1	0.7
DTD-YOLOv4-tiny	416×234	83.5	86.7	89.4	0.327×10^6	1.1	0.52
DTD-YOLOv4-tiny	608×342	88.3	93.9	94.2	0.327×10^6	1.1	0.583
DTD-YOLOv4-tiny	960×540	87.8	95.6	95.0	0.327×10^6	1.1	1.3

表4　Dataset B数据集上不同算法性能对比

Table 4　Performance comparison of different algorithms on Dataset B

Model	Image size	AP /%	Model size
YOLOv3[18]	1333×800	84.9	234.1 MB
Fast RCNN(MobileNet)[18]	1333×800	67.5	162.5 MB
Casacade RCNN(MobileNet)[18]	1333×800	78.0	384.9 MB
TIB-Net[18]	1333×800	89.2	697.0 KB
YOLOv4-tiny	1344×756	78.5	23 MB
DTD-YOLOv4-tiny	960×540	80.3	1.4 MB
DTD-YOLOv4-tiny	1344×756	83.3	1.4 MB
DTD-YOLOv4-tiny	1920×1080	85.1	1.4 MB

实验还对比了不同算法的精度与推理速度的平衡性，结果如图9所示。图9(a)、(b)分别是在数据集Dataset A和Dataset B上的对比结果，其中图9(a)中的结果均是在RXT 2060设备上得到的，

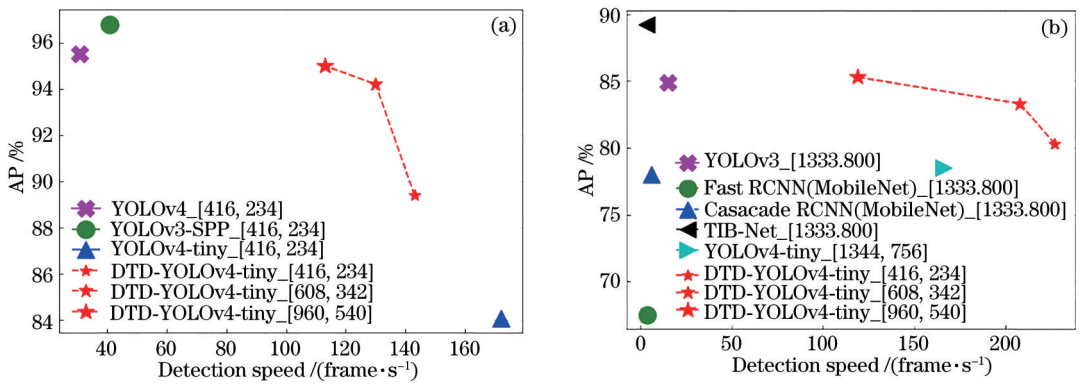

图9　不同数据集下不同目标检测模型准确性和实时性对比图。(a) Dataset A；(b) Dataset B

Fig. 9　Comparison of accuracy and detection speed of different target detection models under different datasets. (a) Dataset A; (b) Dataset B

图 9(b)中 DTD-YOLOv4-tiny 算法结果均是在 RTX 2080Ti 设备上得到的，其他算法结果均是在 NVIDIA TITAN Xp 设备上得到的，RTX 2080Ti 和 NVIDIA TITAN Xp 性能相近。从图 9 中可以看出，现有算法只有在输入图像尺寸较小时才能基本满足实时性的要求，一旦增大输入图像的尺寸，现有算法都难以做到实时检测，检测速度都在 10 frame/s 左右。而所提 DTD-YOLOv4-tiny 算法在精度与推理速度的平衡性上表现最好，在输入原始尺寸的图像时，可以在保证高实时性的同时达到较好的检测精度。在 Dataset B 数据集上，与 TIB-Net 无人机目标检测算法相比，所提 DTD-YOLOv4-tiny 算法精度仅低 4.1 个百分点，但检测速度却是 TIB-Net 的近 40 倍，达到 119 frame/s，完全满足实时性的要求。

图 10 展示了 YOLOv4-tiny 算法和改进后的 DTD-YOLOv4-tiny 算法在不同无人机数据集 Dataset A 和 Dataset B 的部分测试集上的检测结果。可视化检测结果表明，改进后的 DTD-YOLOv4-tiny 算法较 YOLOv4-tiny 算法有更高的准确性。

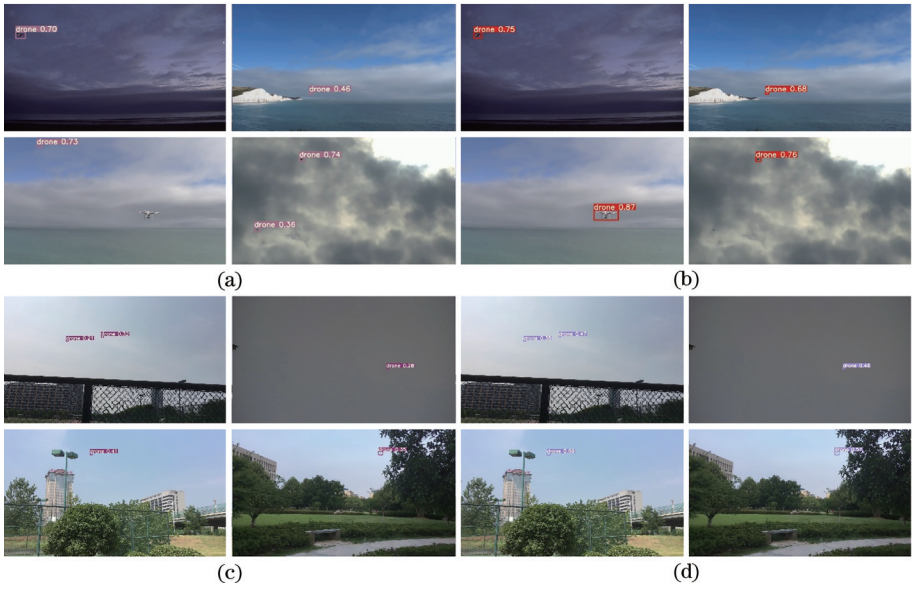

图 10　不同数据集上测试集部分检测结果对比图。(a) YOLOv4-tiny (Dataset A)；(b) DTD-YOLOv4-tiny (Dataset A)；(c) YOLOv4-tiny (Dataset B)；(d) DTD-YOLOv4-tiny (Dataset B)

Fig. 10　Comparison of partial detection results of test set on different datasets. (a) YOLOv4-tiny (Dataset A); (b) DTD-YOLOv4-tiny (Dataset A); (c) YOLOv4-tiny (Dataset B); (d) DTD-YOLOv4-tiny (Dataset B)

5　结　　论

针对现有基于深度学习的无人机目标检测算法复杂度较高，导致模型训练耗时长、占用计算资源大、输入图像尺寸受限、检测速度慢等问题，提出了一种轻量级检测算法 DTD-YOLOv4-tiny。实验结果表明，对于目标大小较为极端的数据集，预先优化 Anchor box 能很好改善算法的性能；同时通过增加检测头数量，能拓宽算法能检测到目标的范围，提升算法的精度；而实验中使用的 reorg_layer 下采样和 sub-pixel 上采样都能一定程度改善算法对小目标无人机的检测能力；通过 ShuffleNetv2 轻量化骨干网络，虽然会牺牲算法的部分精度，但能大幅度降低模型的参数量和浮点运算量，解决了现有算法难以输入原始尺寸图像导致极小目标的信息从输入时就丢失的问题。所提 DTD-YOLOv4-tiny 算法通过对模型 Backbone、Neck、Head 3 个区域的优化，最终实现图像以原始尺寸输入网络，保证在低运算开销和高实时性的同时，得到较好的检测精度。在下一步的工作中，将收集更庞大的无人机目标数据集用以训练，进一步优化网络，在保证高实时性和低运算开销的同时，提升算法的检测精度。

参　考　文　献

[1] Shi X F, Yang C Q, Xie W G, et al. Anti-drone system with multiple surveillance technologies: architecture, implementation, and challenges[J]. IEEE Communications Magazine, 2018, 56(4): 68-74.

[2] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing

Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012, December 3-6, 2012, Lake Tahoe, Nevada, United States. [S.l.: s. n.], 2012: 1106-1114.

[3] Aker C, Kalkan S. Using deep networks for drone detection[C]//2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), August 29-September 1, 2017, Lecce, Italy. New York: IEEE Press, 2017: 17287258.

[4] Girshick R. Fast R-CNN[C]//2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE Press, 2015: 1440-1448.

[5] 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.

[6] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: optimal speed and accuracy of object detection[EB/OL]. (2020-04-23)[2021-02-23]. https://arxiv. org/abs/2004.10934.

[7] Redmon J, Farhadi A. Yolov3: an incremental improvement[EB/OL]. (2018-04-08)[2021-02-23]. https://arxiv.org/abs/1804.02767.

[8] Liu W, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[M]//Leibe B, Matas J, Sebe N, et al. Computer vision-ECCV 2016. Lecture notes in computer science. Cham: Springer, 2016, 9905: 21-37.

[9] Fu C Y, Liu W, Ranga A, et al. DSSD: deconvolutional single shot detector[EB/OL]. (2017-01-23)[2021-02-23]. https://arxiv.org/abs/1701.06659.

[10] Peng J K, Zheng C W, Cui T, et al. Using images rendered by PBRT to train faster R-CNN for UAV detection[EB/OL]. [2021-02-23]. http://wscg. zcu. cz/WSCG2018/2018-papers/!!_CSRN-2802-3.pdf.

[11] Feng X Y, Mei W, Hu D S. Aerial target detection based on improved faster R-CNN[J]. Acta Optica Sinica, 2018, 38(6): 0615004.
冯小雨, 梅卫, 胡大帅. 基于改进 Faster R-CNN 的空中目标检测[J]. 光学学报, 2018, 38(6): 0615004.

[12] Saqib M, Khan S D, Sharma N, et al. A study on detecting drones using deep convolutional neural networks[C]//2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), August 29-September 1, 2017, Lecce, Italy. New York: IEEE Press, 2017: 17287272.

[13] Wang J Y, Wang X Y, Zhang K, et al. Small UAV target detection model based on deep neural network [J]. Journal of Northwestern Polytechnical University, 2018, 36(2): 258-263.
王靖宇, 王霰禹, 张科, 等. 基于深度神经网络的低空弱小无人机目标检测研究[J]. 西北工业大学学报, 2018, 36(2): 258-263.

[14] Li B, Zhang C X, Yang Y, et al. Drone target detection algorithm for depth representation in complex scene[J]. Computer Engineering and Applications, 2020, 56(15): 118-123.
李斌, 张彩霞, 杨阳, 等. 复杂场景下深度表示的无人机目标检测算法[J]. 计算机工程与应用, 2020, 56(15): 118-123.

[15] Ma Q, Zhu B, Zhang H W, et al. Low-altitude UAV detection and recognition method based on optimized YOLOv3[J]. Laser & Optoelectronics Progress, 2019, 56(20): 201006.
马旗, 朱斌, 张宏伟, 等. 基于优化YOLOv3的低空无人机检测识别方法[J]. 激光与光电子学进展, 2019, 56(20): 201006.

[16] Wang C Y, Bochkovskiy A, Liao H Y M. Scaled-YOLOv4: scaling cross stage partial network[EB/OL]. (2020-11-16)[2021-02-23]. https://arxiv. org/abs/2011.08036.

[17] Coluccia A, Ghenescu M, Piatrik T, et al. Drone-vs-Bird detection challenge at IEEE AVSS2017[C]//2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), August 29-September 1, 2017, Lecce, Italy. New York: IEEE Press, 2017: 17287276.

[18] Sun H, Yang J, Shen J Q, et al. TIB-net: drone detection network with tiny iterative backbone[J]. IEEE Access, 2020, 8: 130697-130707.

[19] Ma N N, Zhang X Y, Zheng H T, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[M]//Ferrari V, Hebert M, Sminchisescu C, et al. Computer vision-ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11218: 122-138.

[20] Taha B, Shoufan A. Machine learning-based drone detection and classification: state-of-the-art in research [J]. IEEE Access, 2019, 7: 138669-138682.

[21] Kisantal M, Wojna Z, Murawski J, et al. Augmentation for small object detection[EB/OL]. (2019-02-19)[2021-02-23]. https://arxiv.org/abs/1902.07296.

[22] 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), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 936-944.

[23] 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, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 6848-6856.

[24] Luo X Q, Pan S L. Improved YOLOV3 fire detection method[J]. Computer Engineering and Applications, 2020, 56(17): 187-196.
罗小权, 潘善亮. 改进YOLOV3的火灾检测方法[J]. 计算机工程与应用, 2020, 56(17): 187-196.

[25] Ju M R, Luo J N, Zhang P P, et al. A simple and efficient network for small target detection[J]. IEEE Access, 2019, 7: 85771-85781.

[26] Shi W Z, Caballero J, Huszár F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 1874-1883.

[27] Zhou Z H. Machine learning[M]. Beijing: Tsinghua University Press, 2015.
周志华. 机器学习[M]. 北京: 清华大学出版社, 2015.