计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(12):159−168 [doi: 10.15888/j.cnki.csa.008866]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

面向无人机航拍场景的轻量化目标检测①

黄海生, 饶雪峰

(桂林航天工业学院 计算机科学与工程学院, 桂林 541004)
通信作者: 饶雪峰, E-mail: raoxuefeng@guat.edu.cn

摘　要: 针对无人机航拍场景下的实时目标检测任务, 以 YOLOv5 为基础进行改进, 给出了一种轻量化的目标检测网络 YOLOv5-tiny. 通过将原 CSPDarknet53 骨干网络替换为 MobileNetv3, 减小了网络模型的参数量, 有效提高了检测速度, 并进一步通过引入 CBAM 注意力模块和 *SiLU* 激活函数, 改善了因网络简化后导致的检测精度下降问题. 结合航拍任务数据集 VisDrone 的特性, 优化了先验框尺寸, 使用了 Mosaic, 高斯模糊等数据增强方法, 进一步提高了检测效果. 与 YOLOv5-large 网络相比, 以降低 17.4% 的 *mAP* 为代价, 换取 148% 的检测效率 (FPS) 提升, 且与 YOLOv5s 相比, 在检测效果略优的情况下, 网络规模仅为其 60%.

关键词: 无人机目标检测; 轻量化; YOLOv5; MobileNetv3; CBAM 注意力机制; *SiLU*

引用格式: 黄海生,饶雪峰.面向无人机航拍场景的轻量化目标检测.计算机系统应用,2022,31(12):159–168. http://www.c-s-a.org.cn/1003-3254/8866.html

Lightweight Object Detection for Drone-captured Scenarios

HUANG Hai-Sheng, RAO Xue-Feng

(School of Computer Science and Engineering, Guilin University of Aerospace Technology, Guilin 541004, China)

Abstract: A lightweight object detection network YOLOv5-tiny is given on the basis of YOLOv5 for real-time target detection tasks in drone-captured scenarios. The replacement of the original backbone network CSPDarknet53 with MobileNetv3 reduces the parameters of the network model and substantially improves the detection speed. Furthermore, the detection accuracy is improved by the introduction of the CBAM attention module and the *SiLU* activation function. With the characteristics of the aerial photography task dataset VisDrone, the anchor size is optimized, and data augmentation methods such as Mosaic and Gaussian blur are used to further improve the detection effect. Compared with the results of the YOLOv5-large network, the detection efficiency (FPS) is improved by 148% at the expense of a 17.4% reduction in *mAP*. Moreover, the network size is only 60% of that of YOLOv5 when the detection results are slightly superior.

Key words: drone object detection; light weight; YOLOv5; MobileNetv3; CBAM attention module; *SiLU*

随着无人机技术的普及, 在环境勘测、道路交通流量监管、安全巡检、疫情防控等领域, 利用无人机航拍实施远程监测已经屡现不鲜, 且采取深度学习算法对图像中的物体目标进行识别标记, 可以极大程度减轻人工负担, 提高监测效率. 传统以来, 两阶段检测算法是物体检测领域的主流方法, 其中最具代表性的是 R-CNN 系列[1]. 与两阶段检测算法相比, 单阶段检测算法可以同时预测物体的边界框和类别, 因此速度优势明显, 但精准度较低. 对于单阶段检测算法, 代表性的模型包括 YOLO 系列[2]、SSD[3] 和 RetinaNet[4], 其中 YOLO 系列经广泛使用和多年改进, 目前已经演进到 v5 版本, 是目标检测任务中应用最广泛算法之一.

① 基金项目: 广西教育厅中青年科研基础能力提升项目 (2019KY0805)
收稿时间: 2022-03-20; 修改时间: 2022-04-14, 2022-06-01; 采用时间: 2022-06-06; csa 在线出版时间: 2022-08-12

然而, 即便是 YOLOv5 中网络规模最小的 YOLOv5s, 其网络模型的参数规模和计算量, 对常规无人机平台所能搭载的低功耗嵌入式计算平台也不够友好, 难以满足实时性的需要. 虽然有专门针对低功耗嵌入式设备优化的轻量化网络, 如 MobileNet 系列[5], ShuffleNet 系列[6] 等, 能够满足实时性能需要, 却牺牲了精确率和查全率, 面对航拍图像多物体小尺寸的目标检测场景不太适应. 目前航拍场景下目标检测的高精度网络有如 Zhu 等人的研究[7], 虽然获得了较高的精度但是网络规模与运算量大, 难以部署到无人机平台上进行实时目标检测.

因此, 研究针对航拍场景的轻量化的目标检测网络和训练方法, 具有较强的工程应用价值.

1 相关工作

1.1 深度学习网络的结构组成

目前, 深度神经网络 (DNN) 通常由 3 部分组成: 用于提取图像特征的骨干网络 (backbone), 用于预测物体类别和边界框的检测头 (head) 以及介于 backbone 和 head 之间的 neck 层.

骨干 (主干) 网络 (backbone): 目前常用的 backbone 有如 MobileNet、EfficientNet[8]、CSPnet[9]、Deit[10]、Swin Transformer[11] 等, 它们的特征提取能力较为强大且经过诸多领域的工程验证, 发展较为成熟. 在工程实践中, 研发工程师会根据应用需要, 以某种特定的骨干网络为基础, 结合应用进行调优, 就能取得比较理想的任务效果.

头部网络 (head): head 负责对来自骨干网络的特征图进行处理, 检测物体的边界框位置、尺寸和类别. head 按检测阶段数量划分, 可分为单阶段检测头和二阶段检测头.

颈部网络 (neck): neck 通过对 backbone 在不同阶段提取的特征图进行重新处理并按比例使用, 可实现更好地利用骨干网络提取的特征, 获得更好的检测效果, 是目标检测框架中的关键环节.

1.2 YOLOv5 网络

YOLOv5 是目前基于回归的一阶段目标检测算法 YOLO 系列网络的最新迭代版本, 其网络结构如图 1 所示. 它是首个基于 PyTorch 编写的 YOLO 系列网络, 本文选择其作为改进对象.

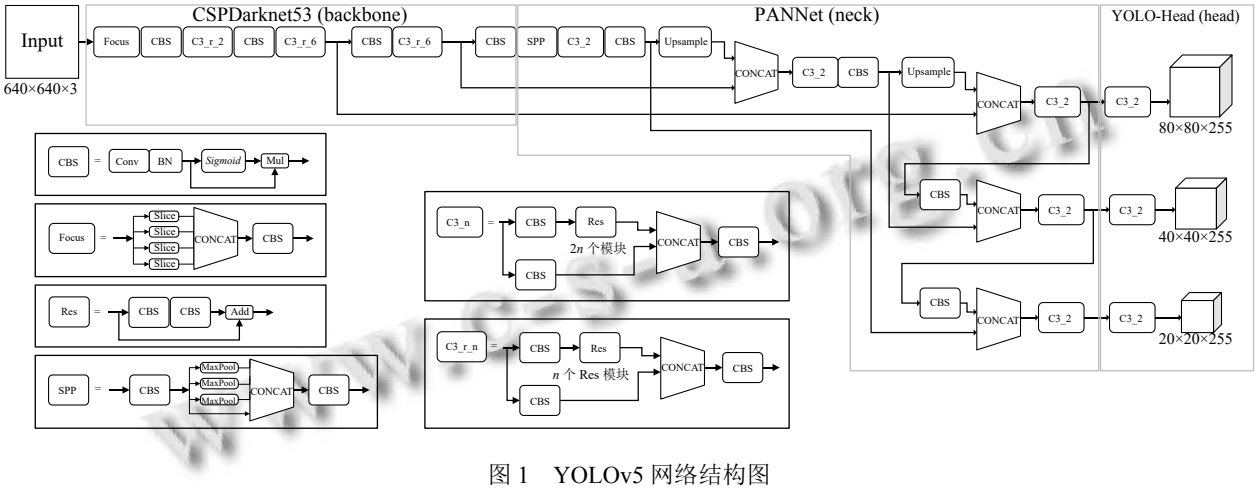


图 1 YOLOv5 网络结构图

YOLOv5 共有 4 个子版本: YOLOv5s、YOLOv5m、YOLOv5l 和 YOLOv5x, 分别表示 YOLOv5 的 small、middle、large 和 Extralarge 版本. YOLOv5 的整体结构如图 1 左方所示, 它们的总体结构与默认分辨率都相同, 使用两个参数进行变换: depth_multiple 与 width_multiple, 分别决定网络深度与卷积核个数, 开发者只要改变此参数即可快速调整网络模型规模. 其中, YOLOv5l 对应的参数 depth_multiple 与 width_multiple 均为 1, 因其综合性能更加均衡, 被作为本文改进工作选用的基准网络结构. 同时, 本文将与官方标准的轻量化实时推理网络 YOLOv5s 网络 (其 depth_multiple 与 width_multiple 分别为 0.33 与 0.5), 从计算复杂度和检测精度两个主要维度展开对比.

YOLOv5 的网络结构包含以下几部分.

图像输入端 (Input): 包含数据增强, 图像变换, 自适应锚框等部分.

主干网络 (backbone): YOLOv5 的 CSPDarknet53 特征提取网络是将 YOLOv3 的 Darknet53 融合 CSP (cross stage partial network) 模块演化而来, 与 YOLOv4 维持一致. Focus 端对经过输入端处理后的图像 (尺寸为 640×640×3) 进行切片处理, 再经过卷积、批标准化 (batch normalization, BN)、*Sigmoid* 激活函数与相乘 (Mul) 操作, 再通过多个基于 CSPDarkNet53 的 C3 模块进行特征提取, 最后经过 SPP (spatial pyramid pooling)[12] 结构进行池化操作, 以改善感受野和区分前后特征.

颈部 (neck): 使用 FPN (feature pyramid networks)[13] +PAN (pyramid attention network)[14] 结构, 其结构如图 1 中 PANNet 中所示, 在 FPN 结构基础进行上采样进行特征融合传递语义, 并使用 PAN 结构进行下采样传递定位特征. 该模块可以进一步把经过 backbone 处理的特征层进行特征融合.

头部 (head): 基于 YOLO 系列的 YOLO-Head[2] 改进而来的 head 包含了 Bounding box 损失函数和非极大值抑制 (NMS) 等机制, 能对预测阶段的目标检测框进行优化, 提高了预测框的回归速度以及与获得最佳尺寸的检测框.

1.3 MobileNetv3 网络

由 Google 公司提出的 MobileNetv3[5] 是其 MobileNet 系列网络的最新版本, 这系列网络初衷是为部署在有限算力的嵌入式平台, 如手机平台. MobileNetv3 是使用 NAS[15] 设计的网络, 包含 large 和 small 两个版本, 其中 large 的精度更同时兼顾速度, 因此本文选用其作为替代主干特征网络. MobileNetv3-Large 的 conv2d 模块包含普通卷积、BN、Hard-swish 激活函数操作, 而 bneck 模块 (图 2 中表示为 B_n_n 模块) 中的卷积为深度可分离卷积 (DWconv).

MobileNetv3 有许多优点特性, 突出体现在 bneck 模块中.

(1) 采用线性瓶颈状的残差倒置结构有效扩展输入的特征. 残差倒置结构的通道中间多两边少, 这与一般的残差结构相反. 该模块先使用 1×1 的 Conv 卷积进行升维操作, 接着进行 3×3 的 DWConv 卷积, 而传统的残差结构如 ResNet[16] 在此部分使用一般 Conv 卷积.

(2) 引入注意力模块机制. SEnet[17] 通过挤压 (squeeze) 和激励 (excitation) 操作来抑制信息与强调信息, 从而得到特征通道的权值, 以确定其重要程度. 具体的流程为: 全局池化、通过两个全连接层降维、再通过损失函数输出.

残差倒置模块与 SE 注意力模块对大部分卷积神经网络有着较好的普适性, 可以有效提高网络的检测精度.

(3) 采用深度可分离卷积 (DWconv). 不同于一般卷积块, bneck 模块中的 DWconv 卷积是由深度卷积 (DepthWise Conv) 与逐点卷积 (PointWise Conv) 组合而成的模块, 可以极大程度降低网络的运算量和参数量.

当 stride=1, padding=True, 输入的特征图大小=$H\times W\times M$, 输出特征图大小=$H\times W\times N$, 卷积核尺寸=$K\times K$ 时:

一般卷积的计算量 FLOPs (floating point operations) 为:

$$F1 = K^2\times M\times N\times H\times W \tag{1}$$

深度可分离卷积的计算量为第 1 步深度卷积的计算量与第 2 步点卷积的计算量之和, 为:

$$F2 = K\times M\times H\times W + M\times N\times H\times W \tag{2}$$

由式 (1) 和式 (2) 可得, 两者的计算量比值为:

$$P = \frac{F_2}{F_1} = \frac{1}{N} + \frac{1}{K^2} \tag{3}$$

当卷积核尺寸 K=3, 输出通道数 N=256 时, 深度可分离卷积的 FLOPs 远小于一般卷积, 仅为其的 1/9, 可大幅降低网络的运算量, 更好地在嵌入式计算平台上运行.

2 改进 YOLOv5 无人机航拍目标检测算法

2.1 本文给出的轻量化网络结构 YOLOv5-tiny

改进后的网络的结构如图 2 所示.

YOLOv5-tiny 对 YOLOv5l 进行了如下改进.

(1) 将原有的骨干网络替换为改进后的 MobileNetv3, 使网络结构紧凑高效.

原 MobileNetv3 网络后端的卷积和池化层与 YOLOv5 的 neck 网络 (PANNet) 的卷积层存在功能冗余, 因此可将其去除; 同时, 得益于 MobileNetv3 的深度可分离 DWconv 卷积, 改进后的网络参数量与模型大小远小于原 YOLOv5 网络.

(2) 使用 CBAM 注意力模块代替原有的 SE 注意力模块.

(3) 使用 *SiLU* 激活函数替代 *ReLU*.

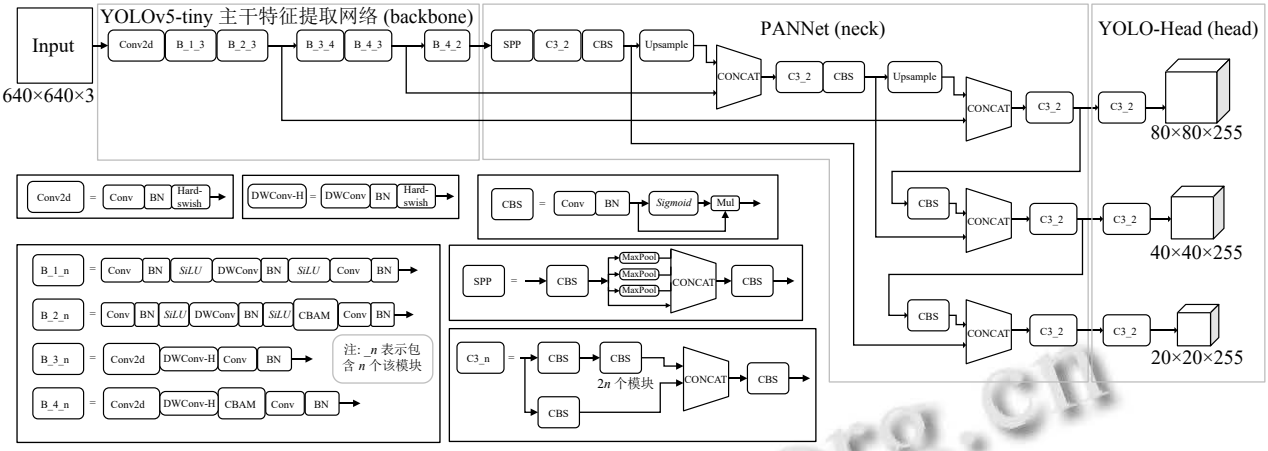


图 2 YOLOv5-tiny 网络结构图

2.2 使用 CBAM 模块替换 SE 注意力模块

CBAM (convolutional block attention module) 模块是由 Woo 等人[18] 提出, 包含有 CAM (channel attention module) 和 SAM (spatial attention module)两个子模块、它通过对图像的通道与空间方位两者的共同重要性评估以确定注意力区域, 以抑制无关背景信息或强调待检测目标的信息, 可以有效提高目标检测的精度. 本文选择使用 CBAM 模块是因为 CBAM模块相较于原有的 SE 模块在通道注意力机制的基础上增加了空间注意力的机制, 在对图像注意力处理的精准度会优于后者, 因为仅考虑通道的权重, 会出现同一通道不同坐标的像素的权重不一样的情况从而无法准确的判断该像素的重要性.

CBAM 模块的整体结构如图 3 所示, 其中输入的特征图先后经过 CAM 与 SAM 模块进行通道的权值评价后得到注意力图, 通过将输入的特征图与输出的注意力图相乘进行特征优化, 以突出图像的特征, 即待检测目标的位置与内容, 让网络更加注重检测到注意力区域的目标, 从而提高算法的目标识别能力.

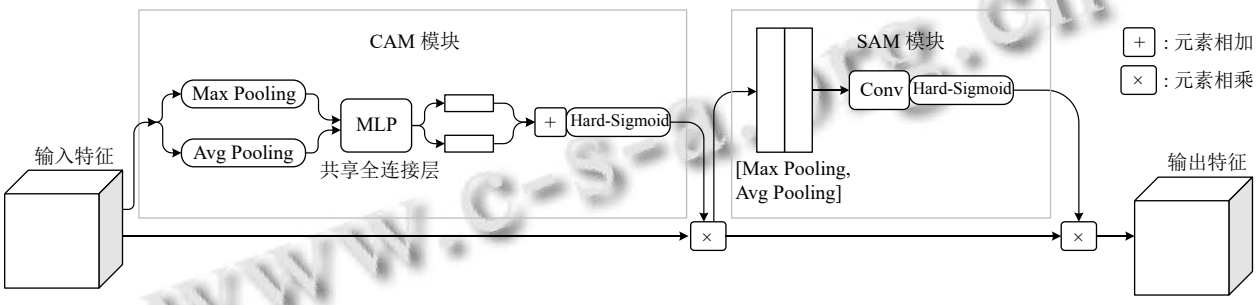


图 3 CBAM 注意力机制结构图, 包含通道注意力模块与空间注意力模块

在通道注意力模块中, 将输入的特征图 ($H\times W\times C$) 分别经过基于长度和宽度的全局最大池化与全局平均池化提取信息压缩空间信息, 分别生成尺寸都为 $1\times1\times C$ 的两个不同特征图, 接着将它们分别输入到一个含有一个隐藏层的两层神经网络中进行计算, 其中这个网络中的参数是共享的, 第 1 层有 C/r (通道降低率, 设置为 r=16) 个神经元, 经过激活函数 *ReLU* 后到有 C 个神经元的第 2 层. 然后将输出的特征图进行元素加和拼接操作后通过 Hard-Sigmoid 激活函数, 最终生成输入到空间注意力模块的特征图 ($1\times1\times C$).

在空间注意力模块中, 将 CAM 模块输出的特征图进行基于通道的最大池化与平均池化进行压缩合并, 得到两个尺寸为 $H\times W\times1$ 的特征图. 接着进行 Concat 拼接操作后经过一个 7×7 的卷积核操作后降维得到空间注意力特征图 ($H\times W\times1$).

CBAM 在通道注意力模块中的全局最大池化操控

与全局平均池化互补, 有效提取压缩信息. 空间注意力模块中使用 7×7 卷积而不是传统的多个 3×3 卷积, 能有效地增加感受野得到更好的空间信息. 因此使用 CBAM 模块替换 SE 注意力模块能更好更快地让网络找到待检测目标并精准检测, 该注意力机制能更好地改善无人机航拍图像存在的背景复杂、小目标间存在重合遮挡等问题, 有利于提升检测精度.

2.3 使用 *SiLU* 激活函数替换 *ReLU* 激活函数

YOLOv5-tiny 将原本 MobileNetv3 中部分 *ReLU* 激活函数更换为有一定相似度的 *SiLU* 激活函数, 这两个激活函数的性质如图 4、图 5 所示.

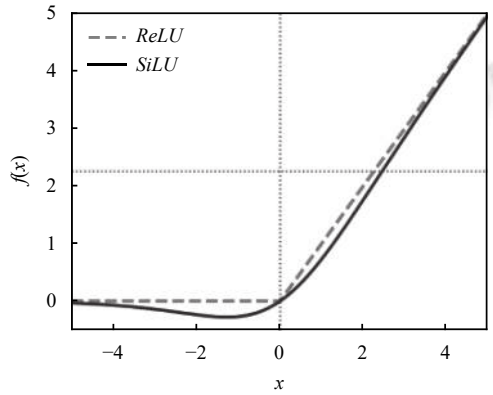


图 4 *SiLU* 与 *ReLU* 的函数图像

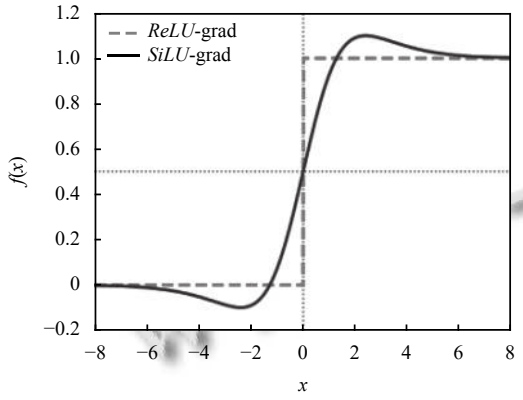


图 5 *SiLU* 与 *ReLU* 的一阶导数图像

ReLU 的详情如下:

$$ReLU = \max(0, x) \tag{4}$$

$$ReLU' = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \end{cases} \tag{5}$$

ReLU 激活函数运算简单只需要给定一个阈值 x 即可进行输出, *ReLU* 的特点是当输入负值时, 其输出与导数一直为 0, 这样会导致神经元中的参数无法更新, 这种情况被称为“神经元死亡”. *ReLU* 运算较为简单只需要指定的阈值就可以输出, 有利于使用梯度下降法 (gradient descent) 的训练的收敛速度提升.

本文选用的 *SiLU* 的详情如下:

$$SiLU = x \cdot Sigmoid(x) \tag{6}$$

$$Sigmoid(x) = \frac{1}{(1+e^{-x})} \tag{7}$$

$$SiLU' = SiLU + (1 - SiLU) \cdot Sigmoid(x) \tag{8}$$

两个函数的图像如图 4 所示, 相比于原本的 *ReLU* 激活函数, *SiLU* 在取极限时与其相似, 两函数交于 0 点, 但 *SiLU* 函数有最小值约为–0.28, 同时加以观察图 5 可得知 *SiLU* 相比 *ReLU* 有着更缓慢的变化率, 稳定性更好, 且导数有正负之分. *SiLU* 与 *ReLU* 在大于 0 时都为正值, 同时小于 0 时为负值, 这样既可以防止梯度消失又可以让激活函数模块的输出更趋于 0, 方差趋于 1, 从而得更好地正则化, 提高训练的收敛速度. Elfwing 等人的研究[19] 表明, 激活函数中的正大权值不断迭代相乘会导致梯度爆炸, 而激活函数中是最小值且导数为 0 的点可以有效抑制大权值的迭代进而防止梯度爆炸的发生. 因此 *SiLU* 相比 *ReLU* 有着更好的稳定性, 让模型收敛更快训练效果更好, 同时 *SiLU* 中增加的小量参数可以有效提高模型的精度.

3 训练策略改进

通过对 VisDrone 无人机航拍数据集[20] 的分析, 发现无人机航拍场景下的画面与主流的目标检测数据集存在明显差异.

(1) 无人机在复杂的城市中的场景航拍, 航拍所处的地域不同, 时间不同, 会导致地面图像背景情况复杂多变, 如同一场景不同天气与光照会导致目标的特征变化, 容易对目标的检测造成干扰.

(2) 无人机在空中远距离拍摄的图像中目标通常较小、模糊, 同时目标可能会存在大量重叠或遮挡现象.

(3) 视角具有特殊性, 图像多数为无人机在空中以不同俯角对地面进行远距离拍摄, 与常规数据集的水平视角近距离拍摄的图像不同, 所以该数据集中的特征与常规数据集中的同类物的体特征会有着较大差异.

因此, 可通过优化先验框尺寸和调整数据增强策略, 在不增加网络复杂度的情况下, 尽最大可能利用数

据集的特性, 实现对网络参数的优化.

3.1 针对小目标检测优化先验框尺寸

YOLOv5 官方提供的基于 COCO2017 常规数据集设定的 9 个先验框 (anchor) 尺寸从小到大 3 个尺度排序分别为 [10, 13, 16, 30, 33, 23]、[30, 61, 62, 45, 59, 119]、[116, 90, 156, 198, 373, 326], 适用于大部分的目标检测任务并且作为默认的模型训练参数. 本文参考原有的 9 个共 3 类先验框, 新算法在训练时会替换原有的先验框尺寸为使用 K-means 聚类算法计算获得的 3 个不同尺度的特征图, 包含 9 个不同尺寸的先验框. 更适配无人机航拍检测目标的先验框尺寸可以让算法获得更精准的边界框 (box), 从而提高检测效果.

K-means 计算先验框的步骤如下.

(1) 设置 9 个随机尺寸的锚定框为初始的先验框.

(2) 把每一个目标的锚定框样本归为与其最相似的先验框中.

(3) 再次计算先验框的高和宽的均值并把此设定为新的先验框.

(4) 重复步骤 (2)、(3), 直到先验框的尺寸固定或达到设定的最高迭代次数.

使用 K-means 聚类方法对 VisDrone 训练集的目标边界框进行计算, 得出用于检验小目标的小尺度先验框尺寸为 (2, 6)、(4, 13)、(8, 21), 中和大两个尺度的先验框尺寸分别为 (8, 10)、(14, 33)、(17, 16) 和 (24, 51)、(32, 27)、(55, 68). 改进后的先验框可以使训练后的网络对无人机航拍图像中的大量小目标进行更准的检测.

3.2 数据增强

针对无人机航拍图像中运动模糊和复杂背景、复杂光照条件下的目标识别, 进行针对性的数据增强处理, 可以有效提升算法训练后的精度. 本文在 YOLOv5 原有的数据增强的方式的基础上进行更加适配数据集的数据增强参数修改, 例如航拍中不会出现颠倒的图像, 故设置上下翻转的概率为 0; 航拍图像中有部分为光照不均匀、夜景或模糊的图像, 因此引入高斯模糊算法让其对不同光照下的图像有着更好的识别效果; 图像大部分为小目标且背景常是复杂的所以可以提高进行 Mosaic 处理的概率, 以提高在无人机航拍应用场景中所摄图像的多样化背景下的检测精度.

本文主要使用了以下两种数据增强策略.

(1) Mosaic 马赛克处理

Mosaic 数据增强方式[2] 是一种可以有效增广有限数据集合成新的大尺寸图片的算法, 具体方法如下: 选择 4 张图片; 把 4 张图片进行缩放、翻转和改变色域的操作, 然后把他们的一部分重叠后拼接在一起生成一张大的新图片; 再对新的图片标注目标锚定框.

Mosaic 算法可以生成更为复杂图片, 特别有利于提高算法检测复杂背景下的小目标的能力, 因此本文网络训练时把所有图片都进行 Mosaic 处理.

(2) 高斯模糊处理

无人机航拍图像时通常是在运动中的, 或者地面的待检测目标也在运动, 这样就容易造成获取的目标图像是模糊的, 因此多针对模糊的目标进行检测有着重要意义. 高斯模糊类似于均值滤波 (用周围像素点的均值最为中心点的像素值), 二维高斯函数滤波也是利用某点周围的数值进行高斯模型处理, 再将处理的数值作为该点的像素值.

二维高斯分布的概率如下:

$$G(x,y)=\frac{1}{2\pi\sigma^2}\mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}} \tag{9}$$

其中, 高斯函数的傅立叶变换频谱是单瓣的, 滤波器的宽度与和平滑度正相关, 由参数 σ 决定. 本文网络训练时设定 σ 的值为 1–100 随机, 进行高斯模糊操作的概率为 0.3.

4 实验过程与结果分析

4.1 实验平台

实验基于第 3.2 节的数据预处理, 操作系统为 Windows 10, 开发环境: PyTorch 版本为 1.10.1; Cuda 版本为 11.3; Python 版本为 3.8.8, 硬件配置: CPU 为 INTEL 11700K; GPU 为 NVIDIA RTX3060 12 GB; DRAM 为 16 GB; 进行网络模型训练与目标检测的平台也为同一平台, 图像输入分辨率统一为 640×640 像素, 默认网络训练时使用 YOLOv5 官方的初始数据增强方法与超参数进行训练, 包括使用余弦退火衰减[21] 等方法训练.

本文选择 VisDrone 数据集作为训练与检测的数据集. 旨在针对城市等复杂环境中的无人机航拍检测的轻量化网络的训练及优化, 主要考虑到 VisDrone 数据集相比于同类数据集如 Stanford Drone、AU-AIR 等, 有着更多背景复杂且小目标多的多视角图像, 使用 VisDrone 数据集进行训练和检验能获得检测效果更好、泛用性更高的网络模型.

4.2 评价标准

本文使用算法的均值平均精度 (*mAP*)[22]、召回率 (recall, *R*)、精确度 (precision, *P*)、FPS、参数量、计算量和模型大小作为指标来评估模型的性能. 其中, 精确度 (*P*) 与召回率 (*R*) 的定义公式如下:

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

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

其中, *TP* (true positive) 为正确预测的样本数, *FP* 为不存在目标的错误检测样本数, *FN* (false negative) 为漏检目标的样本数. *TP* 区为与 ground truth 区域的 IoU≥0.5 的区域 (此处设置 IoU 阈值=0.5, 下同), 则 *FP* 区为 IoU<0.5 的区域; *FN*区为遗漏的 ground truth 区域.

均值平均精度 (*mAP*)[22] 是目标检测算法性能的重要指标, 可以直观反映算法的综合精度, 其计算公式如下:

$$AP = \int_0^1 P(R)dR \tag{12}$$

$$mAP = \frac{\sum_{i=0}^{n} AP_i}{n} \tag{13}$$

4.3 消融实验

以 YOLOv5-Large 为基准方法, 逐一加入文中提及的改进点进行测试, 以评估各个改进点对 *mAP* 和实时性能的贡献. 测试使用一致的 VisDrone 数据集中的 test 测试集测试.

从表 1 可知, 方法①对比原本的 YOLOv5l 在使用了 MobileNetv3 轻量化网络作为主干特征提取网络后, 引入的 DWConv 卷积与残差倒置结构使运行速度有了 2 倍多的提升, 代价是精度下降为原网络的 60%; 方法②在此基础上优化了 MobileNetv3 的网络结构, 降低了参数与计算量同时去除多余的网络层, 使得新网络获得了速度与精度的双重提升, 使检测性能达到了 250 FPS, 精度略微提升 0.4% *mAP*; 方法③、④分别引入了 CBAM 注意力机制与 *SiLU* 激活函数, 以替代原算法中的 SE 注意力机制与 *ReLU* 激活函数, 这两种方法以小幅增加网络计算量的代价 (牺牲了 6.1、5.8 FPS) 换来了 8 FPS), 换来 2.8% 和 0.7% 的 *mAP* 增益. 网络结构④已经和 YOLOv5-tiny 一致. 对比表 2 可知, 此时网络的参数量、计算量、模型大小等指标不足 YOLOv5l 的 1/10, 且小于 YOLOv5s, 达到了实现轻量化的目的, 但精度稍低于 YOLOv5s. 方法⑤通过优化训练策略对精度指标进行改进通过对先验框尺寸的优化使得网络对航拍图像中的小目标的检测精度有效提高, *mAP* 提升 0.8%, 进一步, 方法⑥通过选用更适合的数据增强参数, *mAP* 提升 1.7%, 达到了 44.2%, 高于 YOLOv5s 的 42.7%.

表 1 消融实验测试

方法	MobileNetv3	删除池化层等层	CBAM	*SiLU*	改进先验框	数据增强	*mAP* (%)	FPS (帧)
YOLOv5l	—	—	—	—	—	—	61.6	96.2
①	√	—	—	—	—	—	37.9	222.2
②	√	√	—	—	—	—	38.3	250
③	√	√	√	—	—	—	41.1	243.9
④	√	√	√	√	—	—	41.7	238.1
⑤	√	√	√	√	√	—	42.5	238.1
⑥	√	√	√	√	√	√	44.2	238.1

表 2 不同网络的性能对比

网络模型	*mAP* (%)	平均召回率 (%)	平均精度 (%)	耗时 (ms)	FPS (帧)	参数量Params	计算量GFLOPS	理论模型大小 (MB)
YOLOv5l	61.6	57.8	67.1	10.4	96.2	46 563 709	109.3	89.5
YOLOv5s	42.7	41.5	52.3	4.7	212.8	7 235 389	16.5	13.7
YOLOv5-tiny	44.2	46.7	55.2	4.2	238.1	4 463 695	10.3	8.9

4.4 实验结果分析

YOLOv5s 与本文 YOLOv5-tiny 检测效果如图 6–图 9 所示, 其中, 图 6(a)、图 7(a)、图 8(a)、图 9(a) 为 YOLOv5s 的检测效果, 图 6(b)、图 7(b)、图 8(b)、图 9(b) 为 YOLOv5-tiny 的检测效果, 检测设置中 IoU 阈值为 0.45, 置信度阈值为 0.25. 对比后发现, 尽管两者都存在小目标漏检的情况, 中间部分经过针对小目标检测数据增强训练的 YOLOv5-tiny 的漏检率低于 YOLOv5s,

其精度也稍高于前者. 同时, 在低光照场景下, 两算法依然可以正常检测, 经过针对不同光照场景数据增强训练的 YOLOv5-tiny 检测到的右侧的小目标比 YOLOv5s 高, 同时平均精度也稍高于前者两算法在俯视场景中的中等尺寸目标检测精度都较高, 由图 8 和图 9 可知, YOLOv5s 漏检了中间车辆旁的行人, 而 YOLOv5-tiny 检测到了. 在俯视场景中的小尺寸目标检测中两算法精度一般, 但 YOLOv5-tiny 检测到了下方的行人和非机动车而 YOLOv5s没有.

(a) YOLOv5s (b) YOLOv5-tiny

图 6 白天低空俯视停车场场景检测效果

两个网络的混淆矩阵如图 10 所示, 其中, 图 10(a) 为 YOLOv5s, 图 10(b) 为 YOLOv5-tiny, 可见两者的大目标如 car 的检测精度相似, 但小目标如 tricycle、bicycle 等的召回率后者较高, 且精度也明显高于前者. 本文主要网络的性能如表 2 所示, YOLOv5-tiny 的参数量、计算量、理论模型大小是所有网络中最小的, *mAP* 精准度达到 44.2%. 综合来看, 虽然 YOLOv5s 与 YOLOv5-tiny 的检测效果大体相似, 但后者的网络参数量、计算量只有前者的 60%, 更适合部署在无人机嵌入式计算平台.

(a) YOLOv5s (b) YOLOv5-tiny

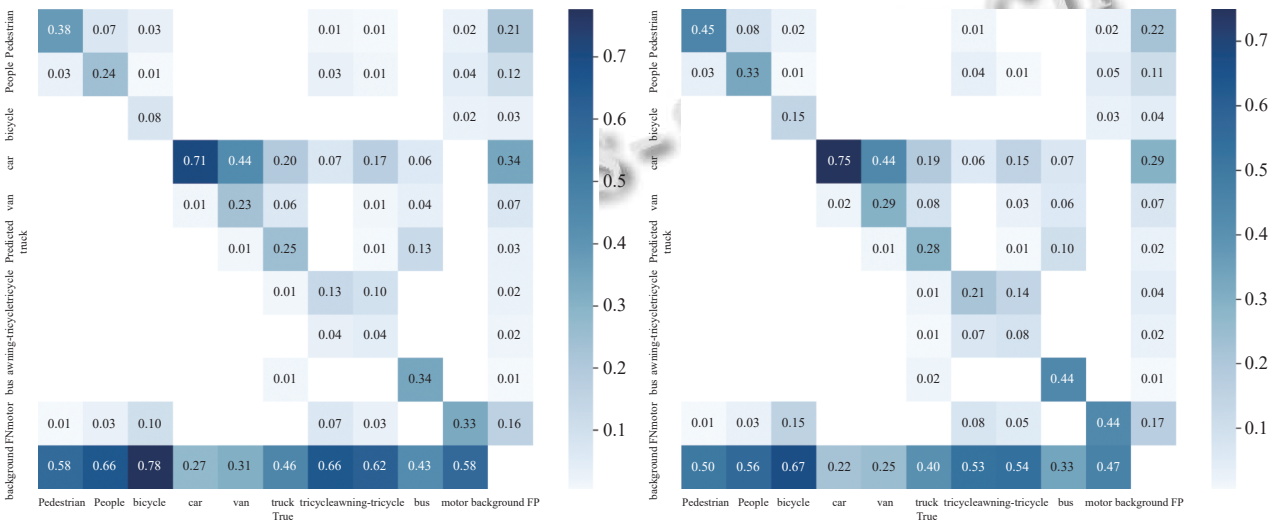
图 7 夜间道路车辆行人目标场景检测效果

(a) YOLOv5s (b) YOLOv5-tiny

图 8 傍晚广场复杂大量小目标场景检测效果

(a) YOLOv5s (b) YOLOv5-tiny

图 9 与阴天高空俯视道路场景检测效果



(a) YOLOv5s (b) YOLOv5-tiny

图 10 混淆矩阵图

5 结束语

本文针对目前主流目标检测算法难以在轻量化无人机嵌入式平台上执行实时目标检测任务的问题, 提出了一种适用于无人机航拍的轻量化改进算法 YOLOv5-

tiny. 相比于 YOLOv5l 在 GPU 平台上以降低了 17.4% 的精度 (*mAP*) 的代价, 获得了检测速度 (FPS) 提高 148% 的效果, 同时各方面的性能都优于 YOLOv5s, 检测速度与精度的提升使得本文模型适合部署在轻量化的无人机嵌入式平台完成实时目标检测任务.

主要改进如下.

(1) 针对 YOLOv5 的 CSPDarknet53主干特征提取网络规模、运算量大的问题, 采用以 MobileNetv3 替换原有的 backbone 的改进方法, 引入了 DWConv 卷积模块替换原算法的普通 Conv 卷积模块, 大幅度减少了运算量.

(2) 针对原 MobileNetv3 网络检测精度不足及原激活函数稳定性不足的问题. 把 MobileNetv3 原有的深层的池化层及后部的多余网络层去除, SE 注意力模块换成 CBAM 注意力模块, 同时把一部分激活函数 *ReLU* 替换为 *SiLU*, 提高了算法的检测精度与训练效果.

(3) 使用了 K-means 聚类计算出了适用于无人机航拍检测目标的先验框的尺寸, 提高了模型训练的效果. 数据增强方面, 基于数据集的特性优化了原有 Mosaic 等数据增强方法, 同时加入高斯模糊等数据增强的方法, 增广了有限的数据集提高了算法的训练效果, 使得算法得以适应更加复杂的现实场景.

现有的算法的精度在 VisDrone 数据集上的表现依然不尽人意, 即使是使用大规模的网络检测也无法达到很高的精度, 因此未来的改进工作包括: 如何在维持网络规模较小的同时, 进一步提升检测精度与如何进一步压缩网络模型, 进行量化剪枝操作将模型量化部署到轻量化无人机平台中.

参考文献

1 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

2 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934, 2020.

3 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot MultiBox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37. [doi: 10.1007/978-3-319-46448-0_2]

4 Lin TY, Goyal P, Girshick R, *et al*. Focal loss for dense object detection. Proceedings of the IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 2999–3007.

5 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324.

6 Ma NN, Zhang XY, Zheng HT, *et al*. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. Proceedings of the 15th European Conference on Computer Vision (ECCV). Munich: Springer, 2018. 122–138.

7 Zhu XK, Lyu SC, Wang X, *et al*. TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. Montreal: IEEE, 2021. 2778–2788.

8 Tan MX, Le QV. Efficientnet: Rethinking model scaling for convolutional neural networks. Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019. 6105–6114.

9 Wang CY, Liao HYM, Wu YH, *et al*. CSPNet: A new backbone that can enhance learning capability of CNN. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Seattle: IEEE, 2020. 1571–1580.

10 Touvron H, Cord M, Douze M, *et al*. Training data-efficient image transformers & distillation through attention. Proceedings of the 38th International Conference on Machine Learning. Online: PMLR, 2021. 10347–10357.

11 Liu Z, Lin YT, Cao Y, *et al*. Swin transformer: Hierarchical vision transformer using shifted windows. Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021. 9992–10002.

12 He KM, Zhang XY, Ren SQ, *et al*. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904–1916. [doi: 10.1109/TPAMI.2015.2389824]

13 Lin TY, Dollár P, Girshick R, *et al*. Feature pyramid networks for object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 936–944.

14 Liu S, Qi L, Qin HF, *et al*. Path aggregation network for instance segmentation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8759–8768. [doi: 10.1109/CVP

R.2018.00913]

15 Zoph B, Le QV. Neural architecture search with reinforcement learning. Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2017. 1–16.

16 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778.

17 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7132–7141.

18 Woo S, Park J, Lee JY, *et al*. CBAM: Convolutional block attention module. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 3–19.

19 Elfwing S, Uchibe E, Doya K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks, 2018, 107: 3–11. [doi: 10.1016/j.neunet.2017.12.012]

20 Zhu PF, Wen LY, Du DW, *et al*. Detection and tracking meet drones challenge. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. [doi: 10.1109/TPAMI.2021.3119563]

21 Loshchilov I, Hutter F. SGDR: Stochastic gradient descent with warm restarts. Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2017. 1–16.

22 Zheng L, Shen LY, Tian L, *et al*. Scalable person re-identification: A benchmark. Proceedings of the IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 1116–1124.

(校对责编: 孙君艳)