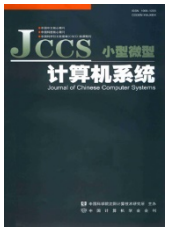


小型微型计算机系统
Journal of Chinese Computer Systems
ISSN 1000-1220,CN 21-1106/TP

《小型微型计算机系统》网络首发论文

题目：融合 CBAM 的 YOLOv4 轻量化检测方法
作者：任丰仪，裴信彪，乔正，白越
收稿日期：2021-08-25
网络首发日期：2022-03-02
引用格式：任丰仪，裴信彪，乔正，白越. 融合 CBAM 的 YOLOv4 轻量化检测方法[J/OL]. 小型微型计算机系统.
https://kns.cnki.net/kcms/detail/21.1106.tp.20220301.0935.002.html

网络首发时间：2022-03-02 18:25:22
网络首发地址：https://kns.cnki.net/kcms/detail/21.1106.tp.20220301.0935.002.html

融合 CBAM 的 YOLOv4 轻量化检测方法

任丰仪[1,2]，裴信彪[1]，乔　正[1,2]，白　越[1]

[1]（中国科学院长春光学精密机械与物理研究所，长春 130033）

[2]（中国科学院大学,北京 100039）

E-mail: renfengyii@163.com

摘要：基于深度学习的目标检测算法应用于无人机视觉中,会极大提升无人机的场景理解能力,但模型参数量和计算量巨大,难以应用于移动端或嵌入式平台。因此本文提出了一种效果较好的轻量级实时检测模型，采用 Yolov4 模型网络作为主要参考模型，使用 MobileNet 替换主干网络，并通过添加 CBAM 注意力机制以及 Soft-NMS 后处理策略来提高模型的准确性。选用 PASCAL VOC 数据集来测试所提出的轻量级 YOLOv4 模型，结果显示参数量只有原模型的一半，但速度 fps 提升了 26.48，精度 mAP 只下降了 0.52%。将所提出的轻量化 Yolov4 模型部署 Nvidia Jetson TX2 低功耗系统以及树莓派上，飞行试验显示在 TX2 上模型 fps 达到了 21.8，是原始的 Yolov4 的 4.74 倍，将本算法部署到无人机装载的嵌入式平台上，能够对航拍视野中的车辆目标进行实时识别和定位。

关键词：无人机图像；YOLOv4；MobileNet；CBAM；柔性非极大抑制策略

中图分类号：TP391　　**文献标识码：**A

YOLOv4 Lightweight Detection Method Based on CBAM

REN Feng-yi[1,2], PEI Xin-biao[1], QIAO Zheng[1,2], BAI Yue[1]

[1](Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China)

[2](University of Chinese Academy of Sciences, Beijing 100039, China)

Abstract: Object classification and detection algorithms based on deep learning will greatly improve understanding capabilities when applied to UAV vision,but the amount of model parameters and calculations are too huge to apply to mobile or embedded platforms. This paper investigated the feasibility and accuracy of a new lightweight real-time detection model with main reference model used the Yolov4 model network and MobileNet replaced the backbone network, as well as CBAM attention mechanism and Soft-NMS post-processing strategy. The PASCAL VOC data set was utilized to test the proposed lightweight YOLOv4 model. The results showed that the parameter amount was only half of the original model and the speed fps was increased by 26.48 just with 0.52% reduced mAP. After the model is deployed on the Nvidia Jetson TX2 low-power system, its fps has reached 21.8, which is 4.74 times that of the original Yolov4. The algorithm is applied to the embedded platform of the drone, which benefits to real-time recognition and positioning of vehicle targets in the aerial view.

Key words: UAV imagery; Yolov4; MobileNet; CBAM; Soft-NMS

收稿日期：2021-08-25 收修改稿日期：2022-01-12 基金项目：国家自然科学基金项目（11372309, 61304017）资助；吉林省科技发展计划重点项目（20150204074GX, 20160204010NY）资助；省院合作科技专项资金项目（2020SYHZ0031）资助；中科院轻型动力创新院重点基金项目（CXYJJ20-ZD-03）资助;中科院青促会项目（2014192）资助. 作者简介:**任丰仪**，女，1996 年生，硕士研究生，研究方向为无人系统视觉导航；**裴信彪**，男，1990 年生，博士研究生，助理研究员，研究方向无人机多传感器的数据融合与控制方法；**乔　正**，男，1997 年生，博士研究生，研究方向为无人机目标检测；**白　越**，男，1979 年生，博士研究生，研究员，研究方向为无人系统动力学、运动学、导航与控制.

1 引言

伴随着无人驾驶飞机的广泛普及，促进了城市管理[1]，军事侦察、灾害救援，土地变化监控[2]和交通监控[3]等多种应用。过去十年见证了无人机视觉技术的巨大进步，尤其是基于深度学习的目标检测算法，应用于无人机视觉领域会极大提高无人机的场景理解能力，对于无人机获取的航拍图像进行实时目标检测逐渐成为研究热点。

目标检测模型通常由三个部分组成：主干网络，颈部网络和头部网络。主干网络功能是进行初步特征提取，对于在 GPU 平台上运行的检测模型，可以选用复杂度高的网络，如 VGG，CSPDarknet53[4]，DenseNet，ResNet 或 ResNeXt。对于在 CPU 平台上运行的检测模型，要选择紧凑网络，如 SqueezeNet[5]，MobileNet[6-8]或 ShuffleNet[9]。颈部网络的作用是加强特征提取，具体包括 FPN，PANet，Bi-FPN 等模块。头部网络利用得到的特征进行预测，主要分为两类，一类是基于区域预测的 Two stage 算法，另一类是基于回归问题的 One stage 算法，Two stage 算法先生成候选框，再利用 CNN 进行特征提取与分类，其主要代表为 R-CNN [10]系列，包括 Fast R-CNN[11]，Faster R-CNN[12]，R-FCN[13]和 Mask R-CNN[14]。One stage 算法不再采用候选框，而是直接对目标物体边界框及类别进行回归，代表算法为 YOLO[15]，SSD[16]和 RetinaNet[17]。总体上，前者相对精度更高，后者检测速度更快。

近些年来，很多研究关注于在目标检测算法中添加功能模块，从而在增加少量推理成本的同时，提高目标精度。增强感受野的常见模块是 SPP[18]、ASPP[19]和 RFB[20]；物体检测中经常使用的注意力模块有 SE[21]、ECA[22]和 CBAM[23]；用于筛选模型预测结果的常见后处理方法是 NMS[24]，但原始的 NMS 没有考虑上下文信息，因此 2017 年提出了更加优化的 Soft-NMS[25]策略。

到目前为止，在无人机获取的图像上进行实时目标检测面临着挑战和困难，首要就是深度神经网络存在复杂性和存储量高的问题。本文针对这个问题，所做的工作主要分为三个方面：

1)以 YOLOv4 模型为基础，使用 MobileNet 模型替换 YOLOv4 本身的主干网络 CSPDarknet53，并利用 MobileNet 提出的深度可分离卷积思想，将原网络中 PANet 与 Head 模块的 3*3 标准卷积块替换为深度可分离卷积块。

2)在 MobileNet-YOLOv4 基础上进行模型优化，加入 CBAM 注意力网络，并在算法后处理部分引入 Soft-NMS 模块替代网络原先的 NMS 模块。在不影响模型运行速度的前提下，提高模型的检测精度。

3)将本文模型进行训练后部署到无人机装载的 Nvidia Jetson TX2 和 Raspberry Pi 低功耗嵌入式平台，通过飞行试验实时定位和识别无人机航拍图像中的车辆和行人。

2 改进的轻量化 YOLOv4 模型

2.1 YOLOv4 基本模型

如图 1 所示，YOLOv4 整体结构可以拆分成三部分：

1）主干网络：主干特征提取网络选用 CSPDarknet53，进行初步特征提取。可以获得三个初步的有效特征层，分别位于主干网络的中间层、中下层、底层，三个特征层的大小分别为(52,52,256)、(26,26,512)、(13,13,1024)，使用三个尺度的特征层进行分类与回归预测。

2）颈部网络：为了加强特征提取，从特征获取预测结果的过程可以分为两个部分，首先构建 SPP 模块、FPN+PAN 特征金字塔结构进行加强特征提取；接下来利用 YOLO Head 对三个有效特征层进行预测。YOLO Head 本质上是一次 3*3 卷积加上一次 1*1 卷积，作用分别是特征整合和调整通道数，可以对三个初步的有效特征层进行特征融合。

3）预测：第三部分为预测网络，利用更有效的特征层获得预测结果。

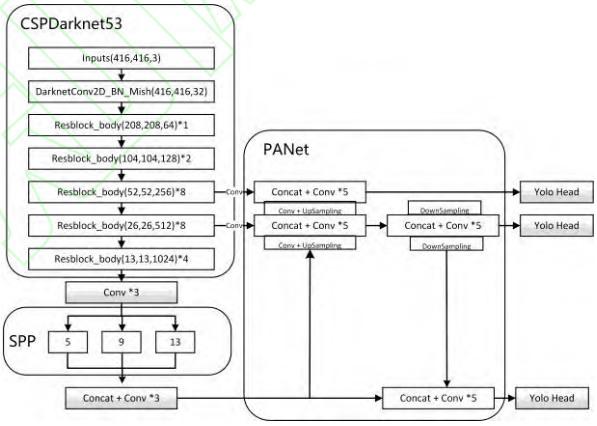


图 1 YOLOv4 的整体结构

Fig.1 The overall structure of YOLOv4

2.2 MobileNet 基本模型

2017 年 Google 提出了 MobileNet v1 网络结构，它使用了深度可分离卷积以及缩放因子，主要特点是模型小、计算速度快。MobileNet 卷积神经网络极低的参数量和运算量的优点使其更适合部署在运算量低的嵌入式设备上，可以更好的平衡检测模型的准确度和运行的效率。

MobileNet v1 是一种流水型网络结构，它的主要特点分为两方面：1）使用深度可分离卷积替代了传统的卷积操作，构建轻量级神经网络。如图 2 所示，深度级可分离卷积可以分解为两个更小的操作：深度卷积和逐点卷积。深度卷积将卷积核拆分成单通道形式，对每一通道进行卷积操作，得到与输入特征图通道数一致的输出特征图；逐点卷积就是 1*1 卷积，主要作用就是对特征图进行升维和降维。2）引入宽度 α 和分辨率 ρ 缩放因子。α

对网络输入和输出通道数进行缩减，ρ 用于控制输入和内部层表示，即控制输入的分辨率，都可以进一步缩小模型。

深度可分卷积操作在参数量与计算量上比标准卷积操作都具有更高优势。如表 1 所示，可知参数数量和乘加操作的运算量均下降为原来的$\frac{1}{N}+\frac{1}{D_k^2}$。通常使用的是 3*3 的卷积核，也就是会下降到原来的$\frac{1}{9}\sim\frac{1}{8}$。

表 1 深度可分离卷积与标准卷积的对比

Table 1 Comparison of factorized convolutions and standard convolution

名称	参数量	计算量
标准卷积	$Cal_s = D_K \cdot D_K \cdot M \cdot N$	$Para_s = D_K \cdot D_K \cdot M \cdot N \cdot D_W \cdot D_H$
深度可分离卷积	$Cal_d = D_K \cdot D_K \cdot M + M \cdot N$	$Para_d = D_K \cdot D_K \cdot M \cdot D_W \cdot D_H + M \cdot N \cdot D_W \cdot D_H$

除此之外，加入了更多的 ReLU6 激活函数，增加了模型的非线性变化，从而提高泛化能力。ReLU6 函数与其导函数如下：

$$relu6(x) = \min(\max(x, 0), 6) \in [0,6] \quad (1)$$

$$relu6'(x) = \begin{cases} 1, & 0 < x < 6 \\ 0, & \text{其他} \end{cases} \in \{0,1\} \quad (2)$$

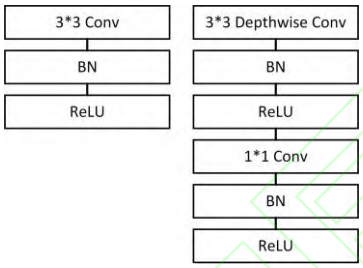


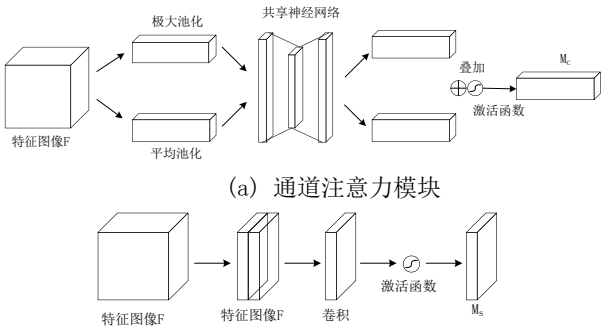
图 2 左图：标准卷积，右图：深度可分离卷积

Fig.2 **Left** standard convolution **right** depth separable convolution

本文将 MobileNet 作为 YOLOv4 的主干特征提取网络，利用 MobileNet 模型强大特征提取能力、极低参数量和运算量的优势，可以提高模型的运算效率，并且将 YOLOv4 中的部分 3*3 标准卷积替换为深度可分离卷积，在降低模型计算量的同时提高性能。

2.3 MobileNet-YOLOv4 优化模型

近年来，有很多研究尝试将注意力机制引入到卷积神经网络中，以提高其在大规模分类任务中的性能。CBAM 是一种能对特征图像局部信息聚焦的模块。它通过学习的方式在空间和通道上对特征图像进行权重分配，促使计算资源更倾向于重点关注的目标区域，从而加强感兴趣的信息，同时抑制无用信息。CBAM 包含两个模块，输入特征依次通过通道注意力模块、空间注意力模块的筛选，最后获得经过了重标定的特征，即强调重要特征，压缩不重要特征。模块划分如图 3 所示。



(a) 通道注意力模块

(b) 空间注意力模块

图 3 CBAM 模块划分

Fig.3 CBAM module division

将 CBAM、SE 注意力机制通过 Grad-CAM[26]方法进行可视化，并与 MobileNet 网络的可视化结果进行比较，结果如图 4 所示。可以看出 MobileNet 与 CBAM 方法同时存在时，Grad-CAM 掩码很好地覆盖了目标对象的区域,与 SE 算法相比，有效预测区域范围更大，结果也更加准确。使用 CBAM 注意力机制可以很好地学习利用目标区域中的信息并从中聚合特征。

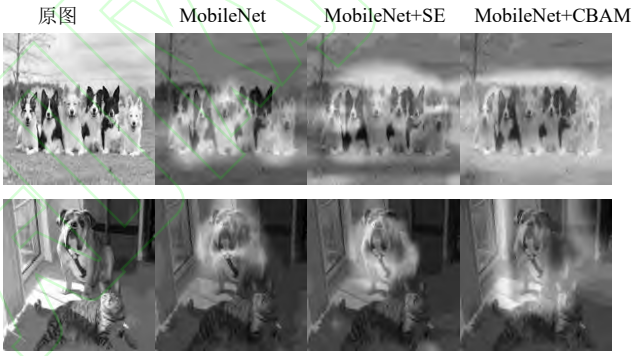


图 4 Grad-CAM 可视化结果

Fig.4 Grad-CAM visualization results

目标检测算法中，非极大值抑制策略(NMS)是很重要部分，对重叠框的处理方式如式(3)所示

$$s_i = \begin{cases} s_i & IoU(M, b_i) < N_t \\ 0 & IoU(M, b_i) > N_t \end{cases} \quad (3)$$

其中 IoU 表示重叠度，NMS 保留在其阈值内的检测框，它的问题在于它会将与目标框相邻的检测框的分数强制归零，导致漏检和目标定位错误，如图 5 所示，对于置信度不高的目标检测结果,容易出现因图像微小变化导致两个预测框置信度大小关系产生变化,使得在 NMS 阶段舍弃保留关系改变,最终第 1 帧检测结果偏左上方,第 2 帧偏右下方。

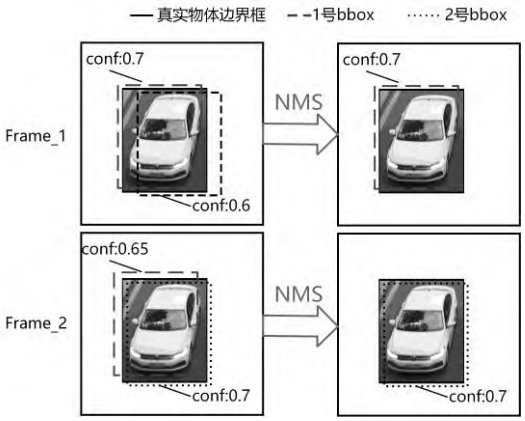


图 5 NMS 损害预测框定位稳定性的原因

Fig.5 Reason for original NMS reducing the stability of bounding box

本文在 YOLOv4 模型中引入 Soft-NMS 算法来代替 NMS 算法。Soft-NMS 同时考虑了得分和重合程度，对于与最高得分的检测框重叠度较高的框设置一个惩罚项，避免重叠框如果包含目标却被删除造成漏检的情况，同时不保留同一个目标两个相似的检测框。处理方式如式(4)所示:

$$s_i = \begin{cases} s_i & IoU(M,b_i) < N_t \\ s_i[1-IoU(M,b_i)] & IoU(M,b_i) > N_t \end{cases} \quad (4)$$

Soft-NMS 实现过程如下:

Input: $B=\{b_1,\dots,b_N\}, S=\{s_1,\dots,s_N\}, N_t$

begin

$D \leftarrow \{\}$

while $B \neq empty$ **do**

$m \leftarrow argmax\ S$

$M \leftarrow b_m$

$D \leftarrow D \cup M; B \leftarrow B - M$

for b_i in B **do**

if $iou(M,b_i) N_t$ **then**

$B \leftarrow B - b_i; S \leftarrow S - s_i$

end NMS

$s_i \leftarrow s_i f(iou(M,b_i))$ Soft-NMS

end

end

return D, S

end

提出的模型以 MobileNet-YOLOv4 为特征提取网络，并且将 PANet 和 YOLOHead 中的标准卷积替换为深度可分离卷积，进一步地减少计算量。此外，在模型三个不同尺度的预测部分加入 CBAM 注意力机制，从而能够增强空间维度和通道维度的有效特征，抑制无效信息的流动，并且将非极大抑制阶段的 NMS 替换为 Soft-NMS。模型整体框架如图 6 所示。

3 实验

为了训练和评估所提出的轻量级模型，实验使用深度学习框架 Pytorch，CPU 选用 Core i9-10900K，GPU 选用 Nvidia Geforce GTX 3080。此外，为了验证模型在无人机飞行时的适用性，在嵌入式系统 Nvidia Jetson TX2 和 Raspberry Pi 4B 上也进行了模型部署和实验结果的分析。

3.1 实验训练数据集

一般模型在进行训练和性能评估时，有很多数据集可以选择，其中最常用的是 ImageNet 数据集[27]、MS COCO 数据集[28]和 PASCAL VOC 数据集[29]。本次实验选用包含 20 个类别的 PASCAL VOC 作为模型训练和测试的数据集，划分验证子集和训练子集的比例为 1:9。为了降低各方面额外因素对识别的影响，对原始数据集进行数据增强。对构建好的模型进行训练微调时，设置 momentum=0.9，lr=0.001，batch_size=16，Init_Epoch=0，Freeze_Epoch=50，去除掉了优化器的权重衰减因子，即 weight decay=0。

3.2 实验过程与结果分析

通道剪枝和紧凑型网络设计，是轻量化网络的常见方法。为了完成对比实验，本文首先按照如图 7 所示的步骤在 YOLOv4 中应用通道修剪以获取 Slim YOLOv4。Network slimming[30]基本原理是对各个通道加入缩放因子 γ，将其与通道的输出相乘，通过训练进行网络权重和缩放因子的更新迭代，将小缩放因子对应的通道进行删减，然后进行网络微调，实现模型压缩。紧凑型网络实验是将 MobileNet 系列的三个网络分别替换原始 YOLOv4 本身的主干网络 CSPDarkNet53，实验结果如表 2 所示。可以看出利用 MobileNet 模型强大特征提取能力、极低参数量和运算量的优势，模型可以在小幅减少精度前提下，极大提高运算效率，比通道剪枝的效果更优，并且 MobileNetv1 作为主干网络时效果最佳。

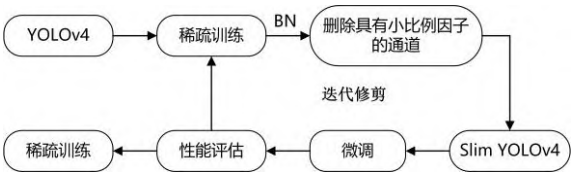


图 7 YOLOv4 通道剪枝流程图

Fig.7 Flow chart of YOLOv4 channel pruning

接下来对 MobileNet-YOLOv4 模型进行优化。首先将模型 PANet 和 YOLO Head 部分中的 3*3 标准卷积替换为深度可分离卷积块，再将 CBAM 卷积注意力机制嵌入 MobileNet-YOLOv4 模型，并在模型后处理阶段引入 Soft-NMS 算法来提高网络性能，NMS 和 Soft-NMS 策略时间消耗的差距几乎为零，但是后者检测精度有小幅度的提升。在各个阶段的优化之后，最终表现如表 3 所示。将本文算法的与 MobileNet-YOLO4 算法对比发现，改进后的算法时间消耗差距只增加了 0.0007s，速度 fps 减少了 4.57，但是 mAP 增长了 2.58%，记录每个实验的 20 个目标类的检测平均精度 AP 如下表 5。将本文算法的与原始 YOLO4 算法对比发现，本文模型参数量只有原模型的四分之一，速度 fps 提升了 26，精度 mAP 只下降了 0.52%。

可见本文基于 CBAM 机制的 MobileNet-YOLOv4 实时目标检测算法利用 MobileNet 网络中的深度可分离卷积网络层的技术，

在检测精度达到主流水平的同时，检测速度有了进一步的提升，同时参数量也大大减小，这有利于部署在计算能力和内存等资源有限的嵌入式设备上。利用 VOC 2007 和 VOC 2012 的训练集进行联合训练，然后基于 PASCAL VOC2007 测试集进行评估，图 8(a)展示了和该训练网络对 20 个目标类的检测平均精度 AP 值和总的 mAP。

为了进一步验证改进模型的性能，引入误检率 MR，并将本文算法与 MobileNet-YOLOv4 算法进行误检率比较，如图 8(b)所示，由图可知，本文算法明显优化了 MobileNet-YOLOv4，降低了大部分类别的误检率，提高了目标检测的平均检测精度。

为了更直观的体现改进的目标检测算法的性能，如图 9 所示，列举了本文设计的轻量化 YOLOv4 模型在航拍图片和视频上的运行结果。每行分别表示不同的航拍条件，从左到右依次是：正常路面情况、拍摄光照不足、画面有遮挡、相机视角倾斜、拍摄停下的车辆、拍摄实时视频。每列分别表示不同的检测算法，从上到下依次是：拍摄的原始图像、原始的 YOLOv4 模型、本文提出的优化后的 MobileNet-YOLOv4 模型。从图中可以看出，本文提出的模型能够有效地对图像、实时视频中所包含不同车辆的大小位置进行检测。相比于原算法来讲，检测结果并没有很大差别，但是实时性更佳。最后，将本文模型与近三年来目标检测模型在同一编程环境以及相同数据集上进行训练，结果对比如表 4 所示。

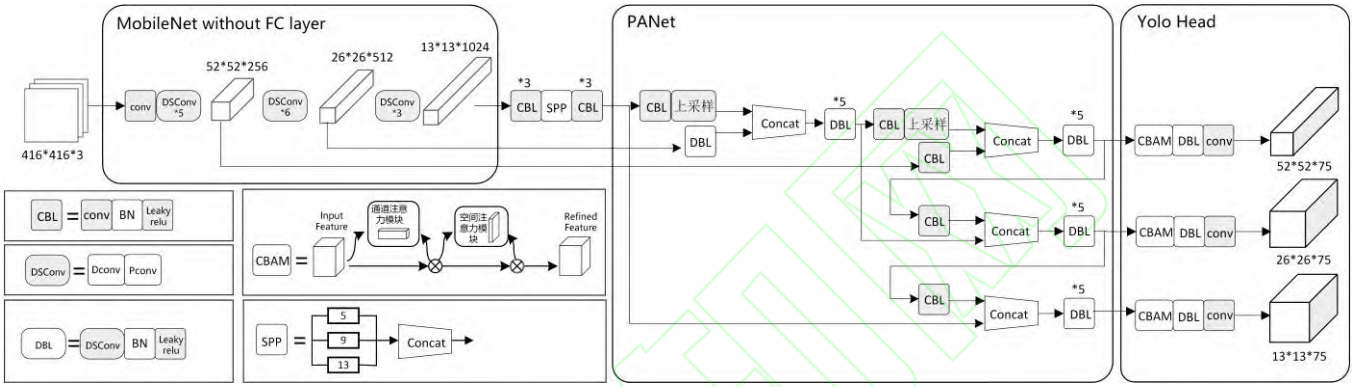


图 6 轻量级网络模型总体结构

Fig.6 Overall structure of lightweight network model

表 2 YOLOv4 网络轻量化前后对比

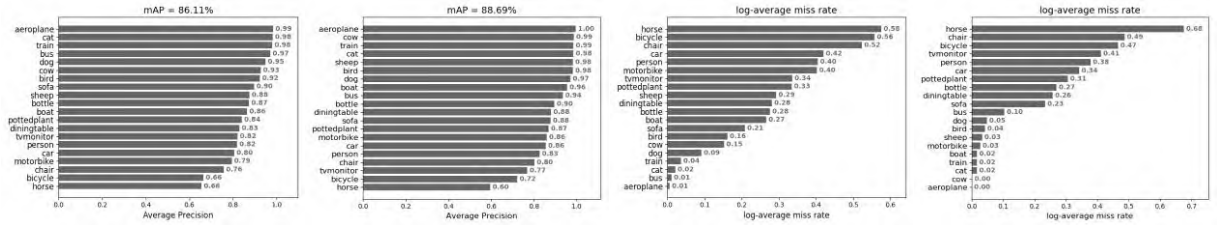
Table2YOLOv4 network lightweight before and after comparison

Model	Total params	mAP	FPS(GPU)	GPU time(s)
Yolov4	64.0M	89.21	48.33	0.0279
Slim Yolov4	14.4M	85.41	76.45	0.0195
Mobilenetv1-Yolov4	12.7M	87.11	79.37	0.0125
Mobilenetv2-Yolov4	12.9M	84.26	64.41	0.0160
Mobilenetv3-Yolov4	13.7M	86.08	57.80	0.0173

表 3 本文算法优化过程中各个阶段的模型对比

Table3 Comparison of models in each stage of the algorithm optimization process in this paper

Architecture	Total params	mAP	FPS(GPU)	GPU time(s)
Yolov4	64.0M	89.21	48.33	0.0279
Yolov4+MobileNet	12.7M	86.11	79.37	0.0125
Yolov4+MobileNet+CBAM+NMS	13.5M	88.19	74.81	0.0132
Yolov4+MobileNet+CBAM+Soft-NMS	13.5M	88.69	74.81	0.0132



(a) (b)

图 8(a) mAP 比较**左图** MobileNet-Yolov4 算法的 mAP **右图**改进后算法的 mAP；

(b)误检率比较**左图** MobileNet-Yolov4 算法的误检率**右图**改进后算法的误检率

Fig.8(a)Comparison of the mAP. **Left** The mAP of the MobileNet-Yolov4 algorithm **Right** The mAP of the improved algorithm；

(b) Comparison of false detection rates. **Left** False detection rate of MobileNet-Yolov4 algorithm **Right** False detection rate of improved algorithm

图 9 在不同拍摄条件下，不同算法得到的检测和分割结果

Fig.9 Detection and segmentation results obtained by different algorithms under different shooting conditions

3.3 基于无人机的飞行试验

为了找到所需的最低处理速度，本文研究了帧处理时间与车辆速度之间的关系。如图 10 所示，设 d 是航拍机视野中道路的长度，l 是移动车辆的长度，v 是其速度。那么，$t = \frac{d-l}{v}$是通过场所需的时间。t 必须足够大以处理至少一个帧。即检测系统的 fps 必须在$\frac{1}{t}$以上。例如，在 20 m 长的场地中，4 m 汽车以 100 km/h 的速度行驶，fps =2 就足够了。对于以 150 km/h 和 200km/h 行驶的同一辆车，所需的最低fps分别为 3 和 4。该结果证实所达到的处理速度很好地符合实时要求。

实验使用四旋翼无人机，无人机飞控为 Pixhawk，平台搭载 ZED 双目立体相机，获取丰富的环境信息，提高无人机的智能感知和场景理解能力。在无人机平台上，图像处理模块要使用体积较小且算力足够嵌入式平台，常用的是 Nvidia Jetson TX2 和 Raspberry Pi 4B 板。

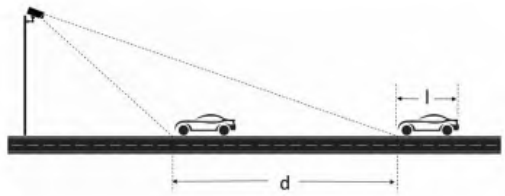


图 10 估计所需的最低帧处理速率

Fig.10 The minimum required frame processing rate

表 5 和表 6 展示了这两种嵌入式系统的规格。将轻量级模型分别部署到两个嵌入式系统中，对无人机航拍图像进行检测处理，所得到的检测结果如图 11 所示，检测速度如表 7 所示，飞行试验显示 TX2 上的 fps 达到了 21.8，相比于 YOLOv4 提高了 3.74 倍。故将本文算法部署到无人机装载的嵌入式平台上，能够对航拍视野中的车辆目标进行实时识别和定位。

表 4 本文模型与近年来目标检测模型的对比

Table 4 Comparison between the model in this paper and the t arget detection model in recent years

Model	mAP	FPS(GPU)	GPU time(s)
YOLOv2	67.40	22.05	0.0451
YOLOv3	75.44	66.89	0.0149
YOLOv4	89.21	48.33	0.0279
Faster RCNN	77.87	21.71	0.0496
本文模型	86.08	57.80	0.0173

表 5 NVIDIA Jetson TX2 的系统规格和软件

Table 5 NVIDIA Jetson TX2 system specs and software

Parameter	Value
GPU	256 core NVIDIA Pascal™ GPU
内存	8GB 128-bit LPDDR4 and 32 GB eMMC memory
尺寸	50mm × 87mm
CPU	HMP Dual Denver 2/2 MB L2+ Quad ARM® A57/2 MB L2
OS	Linux for Tegra®

表 6 Raspberry Pi 4B 的系统规格和软件

Table 6 Raspberry Pi 4B system specs and software

Parameter	Value
SOC	Broadcom BCM2711
尺寸	56mm×85mm
CPU	64-位 1.5GHz 四核
GPU	500 MHz VideoCore VI
内存	1 -4GB DDR4
最大分辨率	4K 60Hz+1080p 或 2*4K 30Hz

表 7 使用 Nvidia Jetson TX2 和 Raspberry Pi 的 FPS

Table 7 The FPS using Nvidia Jetson TX2

Model	FPS(Nvidia Jetson TX2)	FPS(Raspberry Pi)
YOLOv4	4.6	2.0
本文模型	21.8	8.5

图 11 对无人机采集的视频进行实时目标检测的结果

Fig.11 Target detection and insance segmentation on images collected by drones

4 总结与展望

本文提出的基于 CBAM 机制的 MobileNet-Yolov4 实时目标检测方法，首先将 MobileNet 替换为 Yolov4 的主干网络，并且利用 MobileNet 中的深度可分离卷积技术，将 YOLOv4 中的部分标准卷积替换为深度可分离卷积。接下来优化 MobileNet- YOLOv4 模型，通过嵌入卷积注意力机制 CBAM 提高了卷积神经网络输出特征图的全局特征；其次通过引入 Soft-NMS 有效地降低了因为传统非极大抑制 NMS 算法导致的密集物体的相邻框漏检问题。最终在 PASCAL VOC 数据集上的测试结果，表明算法在保证检测精度的前提下，有效地降低了参数量和复杂度，检测速度有了大幅度的提升。这有利于将算法部署到计算能力和内存等资源有限的无人嵌入式平台上，使得无人机能够对视野中的目标进行实时识别，提高无人机的场景理解能力，广泛应用于无人机检测和跟踪车辆、捕获违规行为的智能交通领域以及灾害救援、土地变化监控等应用。

参考文献

[1] Volpi M,Tuia D.Dense semantic labeling of sub-decimeter resolution images with convolutional neural networks[J].IEEE Transactions on Geoscience & Remote Sensing,2016,55(2):881-893.

[2] Wu C,Du B,Cui X,et al.A post-classification change detection method based on iterative slow feature analysis and Bayesian soft fusion[J].Remote Sensing of Environment,2017,199:241-255.

[3] Kopsiaftis G,Karantzalos K.Vehicle detection and traffic density monitoring from very high resolution satellite video data[C]// Geoscience & Remote Sensing Symposium,IEEE,2015:1881-1884.

[4] Wang C Y,Liao H,Wu Y H,et al.CSPNet:a new backbone that can enhance learning capability of CNN[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),IEEE,2020:390-391.

[5] Iandola F N,Han S,Moskewicz M W,et al.Squeezenet: alexnet-level accuracy with 50x fewer parameters and <0.5MB model size[J].CoRR,2016:2,arXiv:1602.07360.

[6] Howard A G,Zhu M,Chen B,et al.Mobilenets:efficient convolutional neural networks for mobile vision applications[J].CoRR,2017,arXiv:1704.04861.

[7] Sandler M,Howard A,Zhu M,et al.Mobilenetv2:inverted residuals and linear bottlenecks:mobile networks for classification,detection and segmentation[J].CoRR,2018,arXiv:1704.04861.

[8]Howard A,Sandler M,Chu G,et al.Searching for MobileNetV3[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019:1314-1324.

[9] Zhang X,Zhou X,Lin M,et al.Shufflenet:an extremely efficient convolutional neural network for mobile devices[J].CoRR,abs/1707.01083,2017:6848-6856.

[10] Gupta S,Girshick R,P Arbeláez,et al.Learning rich features from RGB-D images for object detection and segmentation[J].Springer International Publishing, 2014:345-360, doi:10.1007/978-3-319-10584-0_23.

[11] Girshick R.Fast R-CNN[C].IEEE International Conference on Computer Vision,2015.

[12] Ren S,He K，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.

[13] Dai J,Li Y,He K,et al.R-FCN:object detection via region-based fully convolutional networks[J].Advances in Neural Information Processing Systems,2016:379-387,arxiv:1605.06409.

[14] Kaiming H,Georgia G,Piotr D,et al.Mask R-CNN[C]//IEEE Transactions on Pattern Analysis & Machine Intelligence,2017,45(2): 386-397,arXiv:1703.06870.

[15] Bochkovskiy A,Wang C Y,Liao H.YOLOv4:optimal speed and accuracy of object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR)，2020,arXiv:2004.10934.

[16] Liu W,Anguelov D,Erhan D,et al.SSD:single shot MultiBox detector[J].European Conference on Computer Vision,2016:21-37,doi:10.1007/978-3-319-46448-0_2.

[17] Lin T Y,Goyal P,Girshick R,et al.Focal loss for dense object detection[J].IEEE Transactions on Pattern Analysis & Machine Intelligence,2017:2980-2988,arXiv:1708.02002.

[18] He K,Zhang X,Ren S,et al.Spatial pyramid pooling in deep convolutional networks for visual recognition[J].IEEE Transactions on Pattern Analysis & Machine Intelligence,2014,37(9):1904-1916.

[19] Chen L C,Papandreou G,Kokkinos I,et al.DeepLab:semantic image segmentation with deep convolutional nets,atrous convolution,and fully connected CRFs[J].IEEE Transactions on Pattern Analysis and Machine Intelligence,2018,40(4):834-848.

[20] Songtao Liu,Di Huang,et al.Receptive field block net for accurate and fast object detection[C]//Proceedings of the European Conference on Computer Vision (ECCV),2018:385–400,arXiv:1711.07767.

[21]Jie H,Li S,Gang S,et al.Squeeze-and-excitation networks[J].IEEE Transactions on Pattern Analysis and Machine Intelligence,2017:7132-7141,arXiv:1709.01507.

[22] Wang Q,Wu B,Zhu P,et al.ECA-Net:efficient channel attention for deep convolutional neural networks[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),IEEE,2020,arXiv:1709.01507.

[23] Woo S,Park J,Lee J Y,et al.CBAM:convolutional block attention module[J].Springer,Cham,2018:3-19,arXiv:1807.06521.

[24] Neubeck A,Gool L.Efficient Non-maximum suppression[C]// International Conference on Pattern Recognition.IEEE Computer Society,2006:850-855,doi:10.1109/ICPR.2006.479.

[25] Bodla N,Singh B,Chellappa R,et al.Soft-NMS-improving object detection with one line of code[C]//Proceedings of 2017 IEEE International Conference on Computer Vision，Venice,Italy，2017: 5562-5570,arXiv:1704.04503.

[26] Selvaraju R R,Cogswell M,Das A,et al.Grad-CAM:visual explanations from deep networks via gradient-based localization[J].International Journal of Computer Vision,2020,128(2):336-359.

[27] Russakovsky O,Deng J,Su H,et al.Imagenet large scale visual recognition challenge[J].International Journal of Computer Vision,2015,115(3):211-252.

[28] Lin T Y,Maire M,Belongie S,et al.Microsoft COCO:common objects in context[C]//European Conference on Computer Vision.Springer International Publishing,2014:740-755,doi: 10.1007/978-3-319-10602-1.

[29] Everingham M,Eslami S M A,Van Gool L,et al.The pascal visual object classes challenge:a retrospective[J].International Journal of Computer Vision,2015,111(1):98-136.

[30] Zhuang L,Li J,Shen Z,et al.Learning efficient convolutional networks through network slimming[C]//IEEE International Conference on Computer Vision (ICCV),2017:2736-2744,arXiv:1708.06519.