

基于改进 YOLOv4 的无人机目标检测方法

田 港, 张 鹏, 邹金霖, 赵晓林

(空军工程大学装备管理与无人机工程学院, 西安, 710051)

摘要 针对无人机平台由于内存、算力有限而导致检测模型部署困难、检测速度降低的问题,提出了一种基于 YOLOv4 的改进模型。首先,为了减小模型内存占用、节省计算资源,根据目标尺寸特点,对 YOLOv4 原模型的预测层进行了改进,将三尺度检测模型改进为双尺度检测模型;其次,对双尺度检测模型进行正常训练,然后将其 BN 层的缩放因子进行稀疏训练,最后通过裁剪一定比例的通道数以再次减小模型内存占用提升检测速度。实验分析表明,在与原模型检测效果基本一样的情况下,最终改进模型的内存占用减少了 60%,仅 103 M, FPS 提升了 35%,达到了 58 帧/s。

关键词 无人机;目标检测;YOLOv4;中小目标;双尺度检测模型;通道裁剪

DOI 10.3969/j.issn.1009-3516.2021.04.002

中图分类号 V279;TP391.4 **文献标志码** A **文章编号** 1009-3516(2021)04-0009-06

An UAV Target Detection Method Based on Improved YOLOv4

TIAN Gang, ZHANG Peng, ZOU Jinlin, ZHAO Xiaolin

(Equipment Management and UAV Engineering College, Air Force Engineering University,
Xi'an 710051, China)

Abstract As an application platform in target detection, unmanned aerial vehicles play an incomparable advantage and characteristics in reconnaissance missions. However, the limited memory and computing power of the UAV platform are difficult in detection model deployment and slow at detection speed. To solve the above problems, an improved model based on YOLOv4 is proposed. Firstly, in order to reduce the memory usage of the model and save computing resources, this paper improves the prediction layer of the original YOLOv4 model according to the characteristics of the target size. Secondly, the improved model is trained, and then sparse training and channel pruning on the scaling factor of the BN layer are made to reduce the memory usage of the model again to improve the detection speed. The experimental results show that with the detection results being basically the same, the memory usage of the improved model is reduced by 54%, and the FPS is increased by 35% compared with the original model, reaching 58 frames per second respectively.

Key words UAV; target detection; YOLOv4; small target; two-scale detection model; channel pruning

目标检测已经在车辆检测、人脸检测、自动驾驶、安全系统等领域广泛应用并发挥着重要的作用。

相比于传统检测方法,基于深度学习的检测模型具有检测精度高、速度快等优点。基于深度学习的检

收稿日期: 2021-03-04

基金项目: 国家自然科学基金(61703422)

作者简介: 田 港(1997—),男,陕西西安人,硕士生,研究方向:计算机视觉。E-mail:1228330196@qq.com

通信作者: 张 鹏(1979—),男,山西太原人,副教授,研究方向:无人机工程、目标检测。E-mail:peng1439@163.com

引用格式: 田港,张鹏,邹金霖,等.基于改进 YOLOv4 的无人机目标检测方法[J].空军工程大学学报(自然科学版),2021,22(4):9-14.
TIAN Gang, ZHANG Peng, ZOU Jinlin, et al. UAV Target Detection Method Based on Improved YOLOv4[J]. Journal of Air Force Engineering University (Natural Science Edition), 2021, 22(4): 9-14.

像素一直在减小,最终生成 3 个不同大小的特征图,分别为 52×52 , 26×26 , 13×13 。特征图中像素点和输入图像中像素点对应关系为:

$$n = \frac{X}{Y} \quad (2)$$

式中: X 表示输入图像边长的像素个数; Y 表示特征图边长的像素个数; n 表示特征图中一个像素点对应原图中像素点的个数。

YOLOv4 模型输入图像大小为 416×416 , 52×52 , 26×26 以及 13×13 特征图中一个像素点分别对应输入图像 8 个、16 个以及 32 个像素点,当输入图像被下采样至 13×13 时,图像中小于 32×32 像素的目标均会被压缩至不到一个像素点,这样目标特征信息丢失较为严重,因此在 13×13 特征图上检测 32×32 像素以下的目标意义不大。

本文使用的数据集,目标长宽都分布在 $0 \sim 38$ 像素,属于中小目标^[10],因此负责从 13×13 特征图

中检测的 YOLO Head3 模块对于中小目标的检测意义不大,反而会使得模型参数冗余,导致模型内存占用过大,消耗计算资源,影响检测速度。

针对目标尺寸特点,对原模型的检测层进行了改进,改进后的模型结构如图 2 所示,保留了负责从 52×52 和 26×26 特征图中检测中小目标的 YOLO Head1 和 YOLO Head2 模块。移除了负责检测大目标的 YOLO Head3 模块和其相关的采样卷积过程,但保留了与之对应的 13×13 特征图,该特征图属于深层特征图,里面包含较强的定位特征, 52×52 , 26×26 的浅层特征图里面包含较强的语义信息,通过上采样,将 13×13 特征图中的强定位信息传递到浅中层特征层中进行特征融合,以加强对中小目标的定位精度。模型中小尺度 YOLO Head1 对应的初始候选框大小为 $[12, 16, 19, 36, 40, 28]$, 中尺度 YOLO Head2 对应的初始候选框大小为 $[36, 75, 76, 55, 72, 146]$ 。

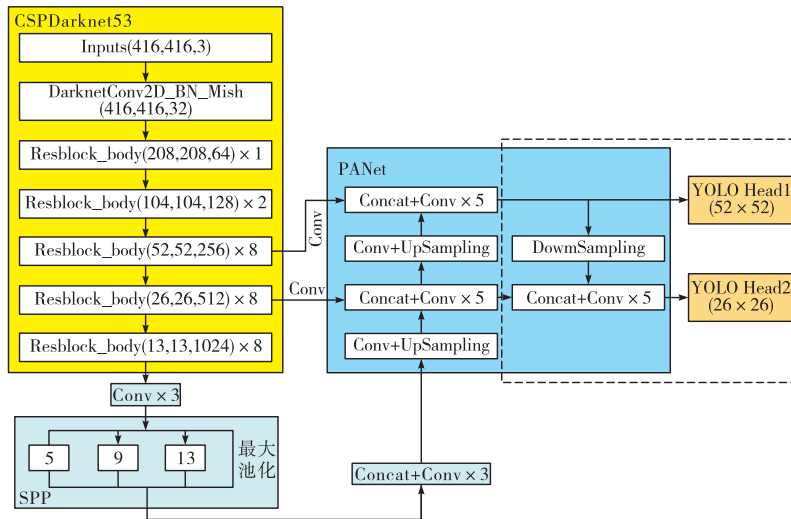


图 2 YOLOv4 改进模型

相比在 3 个尺度上预测输出的原模型,本文改进后的模型仅需要在 2 个尺度上进行检测,减少了模型参数,节省了计算资源,模型检测速度更快。

2.2 YOLOv4 模型裁剪

目前模型裁剪方法主要有权重裁剪^[11]、神经元裁剪^[12]、卷积核裁剪^[13]以及通道裁剪^[14]。

权重裁剪的核心思想是使权重稀疏化,该方法需要多次迭代,并且裁剪一次就需要对模型修复一次,耗时耗力。神经元裁剪是将模型中输出为零的神经元进行删除,该方法虽然操作比较简单,但是裁剪出来的模型性能较差。卷积核裁剪是对卷积核中的参数进行处理,从而降低运算消耗,但目前该方法的相关技术还不够成熟,对于模型速度提升很有限。

通道裁剪流程如图 3 所示,是将卷积层中不重要的通道全部删除。该方法裁剪力度大,效果明显,同时带来的精度损失也大,但是通过微调可以将裁剪后的模型恢复到很好的精度,并且该方法不需要

专门的软硬件支持。通道裁剪方法适用于含有批归一化层即 BN (batch normalization) 层的模型,如 YOLO 模型。对于没有批归一化层的模型该方法并不适用,如 R-CNN 模型。

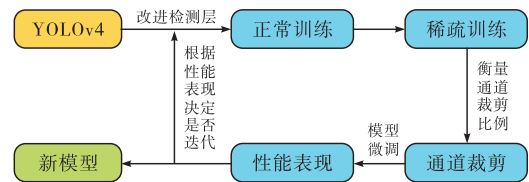


图 3 通道裁剪流程

2.2.1 BN 层稀疏化训练

在 YOLOv4 模型中,通过卷积层和激活函数中间的 BN 层对每一个输入的 mini-batch 进行归一化,使网络收敛更快,获得更好的训练效果。YOLOv4 模型中 BN 层的输入和输出之间关系为:

$$b_{out} = \gamma \times \frac{b_{in} - \mu_m}{\sqrt{\sigma_m^2 + \epsilon}} + \beta \quad (3)$$

式中: m 为每次输入的 mini-batch; γ 为 BN 层的缩

放因子; β 为 BN 层的偏置项; μ 为 BN 层的均值参数; σ 为 BN 层的方差参数; ϵ 项可避免分母为 0。

进行模型通道裁剪首先需要对 BN 层进行稀疏化训练, 本文通过在初始损失函数中引入 L1 正则化来对 BN 层的缩放因子 γ 进行稀疏化训练, 每个特征通道对应一个缩放因子 γ , 引入 L1 正则化的损失函数为:

$$L = \sum_{(x,y)} l(f(x,W), y) + \lambda \sum_{\gamma \in \Gamma} g(\gamma) \quad (4)$$

式中: x, y 分别为模型的输入和输出; W 为模型的参数, 第 1 项为初始损失函数, 第 2 项为 L1 正则损失项, λ 为两项的平衡因子。本文用稀疏后的缩放因子 γ 来判断特征通道对输出重要性。 γ 值越大说明对应的通道对输出越重要, 反之, 说明该通道对输出影响较小。因为引入 L1 正则化后, 稀疏化之后的模型中许多缩放因子 γ 都会趋近于零, 这说明卷积层输出值无论多大, 在输入到 BN 层之后, 经过 BN 层的变换输出 b_{out} 都会变为:

$$b_{out} = \gamma + \frac{b_{in} - \mu_m}{\sqrt{\sigma_m^2 + \epsilon}} + \beta = 0 \times \frac{b_{in} - \mu_m}{\sqrt{\sigma_m^2 + \epsilon}} + \beta \quad (5)$$

所以, 缩放因子 γ 趋于零对应的特征通道对模型输出贡献很小, 这些不重要的特征通道可以将其裁剪掉。

2.2.2 模型通道裁剪

如图 4 所示, 对已经稀疏化的缩放因子进行排序, 根据剪枝率求得缩放因子的阈值。当缩放因子大于或等于该阈值时, 该缩放因子对应的特征通道将被保留。当缩放因子小于该阈值时, 该缩放因子对应的特征通道将会被剪掉, 见图 4。

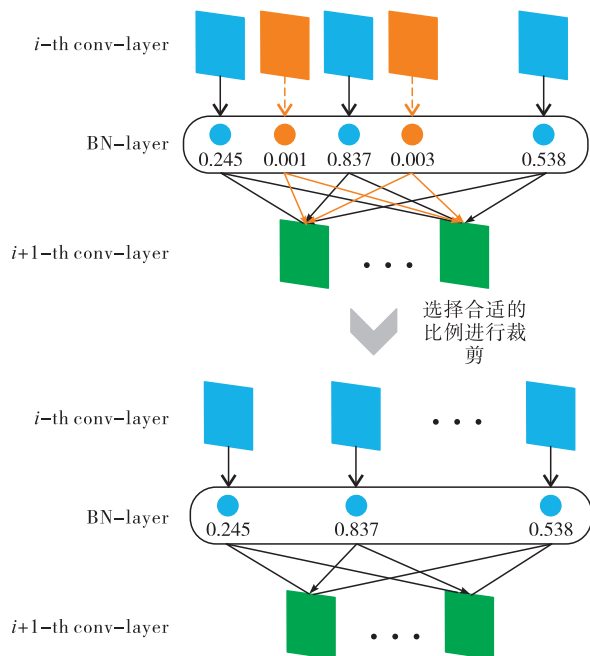


图 4 通道裁剪

通常情况下, 剪裁后的模型其检测精度会有不同程度的下降。剪裁的通道数越多, 参数越少, 检测精度下降的就越明显。为了恢复一定的检测精度,

需要用数据集将裁剪后的模型重新训练, 在这个过程中, 裁剪模型中的参数会根据训练样本数据进行调整, 以恢复至较高的检测精度。

3 实验结果与分析

3.1 数据采集

本文以坦克、飞机、舰艇作为被侦察对象, 使用高仿真模型, 按照 1:20 搭建缩比场景, 然后用四旋翼无人机进行视频采集, 采集完成后截取部分帧作为数据样本。对采集到的数据样本进行旋转、拼接、曝光、添加椒盐噪声等增广处理, 最终共制得图像 3 400 张, 其中 2 720 张用来训练, 680 张用来测试验证, 图 5 为部分数据样本。

3.2 数据标注

如图 6 所示, 本文用 Labelimg 软件对图像中的目标进行标注, 一共 3 类目标: 坦克(tank)、飞机(plane)、舰船(ship)。具体标注流程: ①打开图片, 在图片上创建包围目标的最小矩形框; ②对创建的不同矩形框进行命名; ③对单张图片中所有目标命名保存后, 软件自动生成对应的 xml 文件; ④使用格式转换代码将 xml 文件转换为 txt 文件, txt 文件包含用于训练的目标坐标位置等信息。

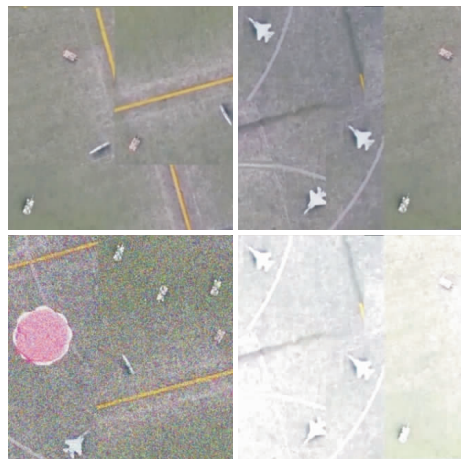


图 5 无人机航拍数据集样本

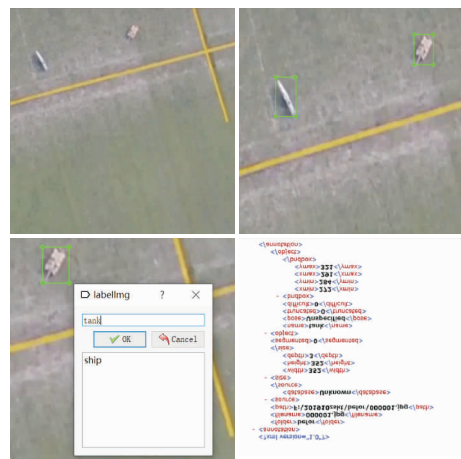


图 6 数据标注示意图

3.3 评价指标

目前在目标检测领域有多种性能评价指标,如精确度、召回率、平均精确率、平均精准率均值 mAP、检测速率等。以下分别为精确度(P)和召回率(R)的计算公式:

$$P = \frac{W_{TP}}{W_{TP} + W_{FP}} \quad (6)$$

$$R = \frac{W_{TP}}{W_{TP} + W_{FN}} \quad (7)$$

式中: W_{TP} 为被正确识别的目标个数; W_{FP} 为被错误识别的目标个数; W_{FN} 为没有被正确识别出来的目标个数。

AP 为单类目标 PR 曲线与坐标所围成的面积,该指标能够综合考虑精确度和召回率,对于单类目标的识别效果评价较为全面。mAP 为各类目标 AP 值的加和平均值,即表示模型整体检测精度,mAP 越高检测效果越好。FPS 为模型一秒能够检测的图片数量,FPS 值越大说明模型的检测速度越快。

3.4 实验配置与训练

本文实验所用配置如下:计算机系统采用 Ubuntu18.04;CPU 采用 Intel Core i5-9500;GPU 采用 NVIDIA GeForce RTX 2060 ti;深度学习框架采用 Darknet^[15]。训练时,模型输入大小为 416×416 ,batch-size 为 32,初始学习率为 0.001,稀疏率为 0.001。

训练过程中训练集损失函数和验证集损失函数变化情况如图 7 所示。当模型迭代至 150 个 epoch 后,两者的损失函数值均已收敛至 0.6 附近,因此实验中 epoch 取 160。

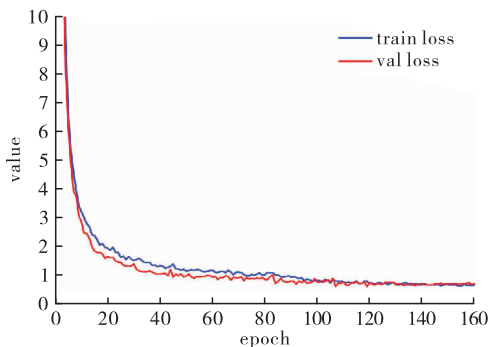


图 7 损失函数变化曲线

3.5 模型对比与分析

为了验证改进模型的有效性,本文设置了 3 组实验进行对比分析。

实验 1:使用 YOLOv4 原模型,模型结构见图 1。

实验 2:使用 2.1 节方法,对 YOLOv4 原模型预测层进行改进,生成 YOLOv4-A 模型,模型结构见图 2。

实验 3:使用 2.2 节方法,对 YOLOv4-A 模型

进行通道裁剪改进,微调生成 YOLOv4-B 模型。

在输入大小为 416×416 的情况下,对上述 3 种模型使用 3.2 节所述数据集进行训练测试,模型性能对比结果见表 1。

从表 1 的实验数据可以发现:

1) YOLOv4-A 模型比 YOLOv4 原模型 mAP 降低了 0.22%,但模型的内存占用少了 26.56%,检测速度提升了 11.63%。主要原因在于 YOLOv4-A 模型少了一个预测尺度,使得模型参数量减少。

2) YOLOv4-B 模型是在 YOLOv4-A 模型的基础上进行了通道裁剪,模型检测精度会有明显降低,为了恢复检测精度进行了微调,微调后 YOLOv4-B 模型较 YOLOv4-A 模型 mAP 降低了 1.30%,但模型内存占用减少了 85 MB,FPS 也提高了 10 帧/s。首先 mAP 的降低是因为裁剪模型在最后微调的时候参数量无法增加,这使得模型精度只能在有限的范围内进行调整提升。其次内存占用和 FPS 再次得到优化是因为对模型进行了通道裁剪,将不重要的通道进行了裁剪操作,因此参数量更少,推理速度更快。

3) 最终改进模型 YOLOv4-B 较 YOLOv4 原模型在 mAP 仅降低 1.52%的前提下,模型内存占用减少了 60%,仅 103 MB。FPS 增加了 15 帧/s,取得了 35%的提升。

表 1 模型性能对比

模型	裁剪率/%	模型大小/MB	mAP@0.5/%	FPS/(帧/s)
YOLOv4	0	256	99.03	43
YOLOv4-A	0	188	98.81	48
YOLOv4-B	0.3	103	97.51	58

表 2 为实验 1 原模型与实验 3 最终改进模型的召回率对比,召回率越高说明模型的检出效果越好。

表 2 各目标召回率对比

模型	Plane	Ship	Tank	%
YOLOv4	99.54	98.81	99.17	
YOLOv4-B	98.97	97.62	98.16	

从表 2 可看出,改进模型的飞机、舰船、坦克召回率较原模型分别下降了 0.57%、1.19%、1.01%。这说明 YOLOv4-B 在对飞机进行检测时,每检测 100 个目标,比原模型多漏检不到 1 个目标。同理,在对舰船、坦克每检测 100 个目标时,YOLOv4-B 比原模型仅多漏检 1 个目标。因此 YOLOv4-B 和 YOLOv4 模型在实际检测中效果应基本一样。

从图 8 的对比可以发现,YOLOv4-B 模型没有因为模型的大幅压缩,显著地影响到对目标的检测效果。YOLOv4-B 和 YOLOv4 原模型均可以将图像中的 3 类目标正确检测出来。

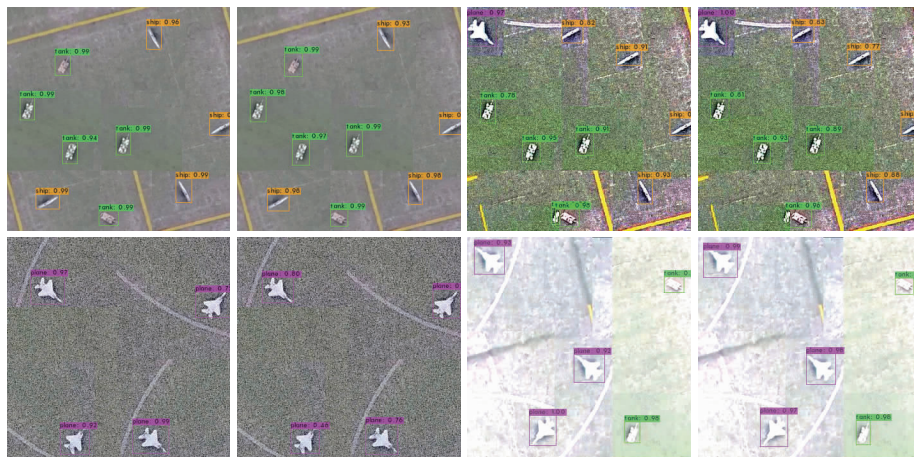


图8 实际检测效果对比

通过对 mAP、Recall 以及实际检测效果的分析,本文提出的最终改进模型 YOLOv4-B 在与 YOLOv4 原模型检测效果基本一致的情况下,内存占用和检测速度均取得了较大提升。

4 结语

本文针对检测模型内存占用大、上机部署时实时性差的问题,提出了一种基于 YOLOv4 的无人机目标检测模型。在保证检测精度的前提下,针对目标尺寸特点,将三尺度检测模型改进为双尺度检测模型;并进行了通道裁剪。通过实验数据能够看出,该方法减小了模型内存占用、提升了检测速度。

参考文献

- [1] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: Unified, Real-Time Object Detection [C] //Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition. [S. l.]: CVPR, 2016: 779-788.
- [2] LIU W, ANGELOV D, ERHAN D, et al. SSD: Single Shot Multibox Detector[C] //European Conference on Computer Vision (ECCV), 2016: 21-37.
- [3] FUCY, LIU W, RANGA A, et al. DSSD: Deconvolutional Single Shot Detector [Z]. ArXiv Preprint, ArXiv: 1701. 06659, 2017.
- [4] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C] //Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition. [S. l.]: CVPR, 2014: 580-587.
- [5] HE K M, ZHANG X Y, REN S Q, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern

Analysis and Machine Intelligence, 2015, 37 (9): 1904-1916.

- [6] GIRSHICK R. Fast R-CNN [C] //Proceedings of the IEEE Computer Society Conference on Computer Vision & Pattern Recognition. [S. l.]: CVPR, 2015: 1440-1448.
- [7] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks [Z]. ArXiv=1506. 01497, 2015.
- [8] HE K M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 386-397.
- [9] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: Optimal Speed and Accuracy of Object Tetection[Z]. ArXiv Preprint, ArXiv, 10934, 2004.
- [10] KISANTAL M, WOJNA Z, MURAWSKI J, et al. Augmentation for Small Object Detection [Z]. ArXiv Preprint, ArXiv: 1902. 07296, 2019.
- [11] LUO J, WU J. An Entropy-Based Pruning Method for CNN Compression [Z]. ArXiv Preprint, ArXiv: 1706. 05791, 2017.
- [12] HAN S, POOL J, TRAN J, et al. Learning both Weights and Connections for Efficient Neural Networks [Z]. ArXiv Preprint, ArXiv: 1506. 02626, 2015.
- [13] LI H, KADAV A, DURDANOVIC I, et al. Pruning Filters for Efficient ConvNets [Z]. ArXiv Preprint, ArXiv: 1608. 08710, 2016.
- [14] LIU Z, LI J, SHEN Z, et al. Learning Efficient Convolutional Networks Through Network Slimming [C] //IEEE International Conference on Computer Vision. [S. l.]: ICCV, 2017: 2755-2763.
- [15] REDMON J. Darknet: Open Source Neural Networks in C [EB/OL]. (2021-02-04). <http://pjreddie.com/darknet/>.

(编辑:徐敏)