

# 基于 Faster 区域卷积神经网络的目标检测

吕本远<sup>1\*</sup>, 嵇真福<sup>2</sup>, 韩永赛<sup>1</sup>, 张立朝<sup>2</sup>

<sup>1</sup>空军工程大学研究生院一大队, 陕西 西安 710038;

<sup>2</sup>空军工程大学航空工程学院, 陕西 西安 710038

**摘要** 针对 Faster 区域卷积神经网络目标检测算法, 提出了一种自适应候选区域建议网络。在训练过程中根据当前损失反馈调节候选区域数目, 使候选区域在一定范围内动态变化, 进而节省开销, 并记录下表现最好的候选区域数目; 在测试时用记录的候选区域数目进行测试。针对 Softmax 函数对候选区域进行分类时需要人为选取置信度阈值带来的时间成本和小目标检测准确率下降问题, 提出了一种自适应置信度阈值选取算法。实验结果表明, 相比传统算法, 本算法的检测速度提升了 25%, 平均检测精度提高了 1.9 个百分点。

**关键词** 图像处理; 深度学习; 目标检测; 自适应区域; 卷积神经网络

中图分类号 TP183; TP751.1

文献标志码 A

doi: 10.3788/LOP202158.2210017

## Target Detection Based on Faster Region Convolution Neural Network

Lü Benyuan<sup>1\*</sup>, Zhuo Zhenfu<sup>2</sup>, Han Yongsai<sup>1</sup>, Zhang Lichao<sup>2</sup>

<sup>1</sup>The First Company, Graduate School, Air Force Engineering University, Xi'an, Shaanxi 710038, China;

<sup>2</sup>Aeronautics Engineering College, Air Force Engineering University, Xi'an, Shaanxi 710038, China

**Abstract** Aiming at the target detection algorithm based on Faster region-based convolutional neural network, we propose an adaptive candidate-region suggestion network. During training, the number of candidate regions is adjusted according to the current loss feedback to ensure that the candidate regions change dynamically in a certain range for cost savings. The number of candidate regions with the best performance is recorded. The recorded candidate regions are tested during testing. An adaptive confidence threshold selection algorithm is proposed to solve the time cost problem and the reduced accuracy of a small target detection caused by artificial confidence threshold selection when Softmax function is used for classifying candidate regions. Experimental results show that compared with the traditional algorithm, the detection speed of the algorithm improves by 25% and the average detection accuracy improves by 1.9 percentage points.

**Key words** image processing; deep learning; target detection; adaptive area; convolutional neural network

**OCIS codes** 100.2980; 100.4996; 150.0155; 100.3008

## 1 引言

随着深度学习的发展, 以目标检测和图像识别为主的技术成为人们的研究热点<sup>[1]</sup>, 对目标检测领域的研究也不断加深。传统目标检测算法大多运用滑动窗口法或图像分割产生大量候选框, 然后对候

选框进行特征提取, 包括方向梯度直方图 (HOG)<sup>[2]</sup>、尺度不变特征变换 (SIFT)<sup>[3]</sup>、Haar<sup>[4]</sup> 等特征, 最后将输出的特征传递给分类器进行判断, 如支持向量机 (SVM)<sup>[5]</sup>、Adaboost<sup>[6]</sup>、随机森林 (Random forest)<sup>[7]</sup> 分类器, 检测的准确率和速度不高, 难以满足商业化的要求。随着社会的发展, 工业

收稿日期: 2021-01-06; 修回日期: 2021-02-03; 录用日期: 2021-03-16

基金项目: 国家自然科学基金青年基金(62006245)

通信作者: \*1102936859@qq.com

生产的产品越发小巧精致,检测系统的应用场景也越来越丰富,如无人机航拍、街道路口监控,这种条件下采集的图像中存在大量小目标,因此,提升算法对小目标的检测能力显得尤为重要<sup>[8]</sup>。

目标检测算法的相关研究已经取得了丰硕的成果。Girshick 等<sup>[9]</sup>成功将卷积神经网络(CNN)<sup>[10]</sup>与目标检测相结合,提出了一种区域 CNN(RCNN)算法。Redmon 等<sup>[11]</sup>提出了 YOLO(You only look once)算法,仅需要一个 CNN 就可以预测目标的位置和种类,从而实现端到端的检测。随后又对 YOLO 算法进行改善,相继提出 YOLOv2<sup>[12]</sup>和 YOLOv3<sup>[13]</sup>算法,改进后的 YOLOv3 算法能满足实时检测对精度和速度的要求。Girshick 等<sup>[14-15]</sup>对 RCNN 算法进行改进,相继提出了 Fast-RCNN 和 Faster-RCNN 算法,将 Fast-RCNN 与区域生成网络(RPN)相结合,提高了算法的检测速度与精度。戴陈卡等<sup>[16]</sup>将 Faster-RCNN 与多部件方法相结合,提高了模型的检测速度。林莉<sup>[17]</sup>提出了一种基于候选框生成和多尺度自适应学习的小目标检测算法,提高了小目标的检测精度。董永峰等<sup>[18]</sup>提出了一种复杂背景下的小目标检测算法,以 Mask-RCNN 为基础架构,实现了对飞机遥感图像的识别。孙佳等<sup>[19]</sup>基于 YOLO 算法提出了一

种适应实时目标检测的网络结构及视频目标检测算法,提高了算法的检测速度和检测精度。现有的 RCNN 和 YOLO 系列检测算法各有优缺点,前者的检测精度较高,但速度较慢,无法达到实时性的要求;后者虽然能实现端到端的检测,检测速度较快,但精度不高,且对小目标的检测能力较弱。

针对 Faster-RCNN 算法检测速度较慢的问题,本文提出了一种自适应区域建议(ARP)算法,在训练过程中根据损失反馈调节候选区域数目,减少训练时间。此外,针对分类时人为选取置信阈值带来的时间成本和小目标检测不准确带来的精度损失问题,提出了一种自适应阈值算法,减少了人工选取阈值带来的误差和时间成本。

## 2 Faster-RCNN 算法

Faster-RCNN 算法的平均检测精度(mAP)较高,是目前主流的目标检测算法之一,但相比 YOLO 系列算法,Faster-RCNN 算法在检测速度方面略显不足。因此,将 ARP 和 Faster-RCNN 算法融合进一个网络模型——RPN 层,其整体框架如图 1 所示。其中, bbox\_pred 表示位置预测输出, cls\_prod 表示分类预测输出。

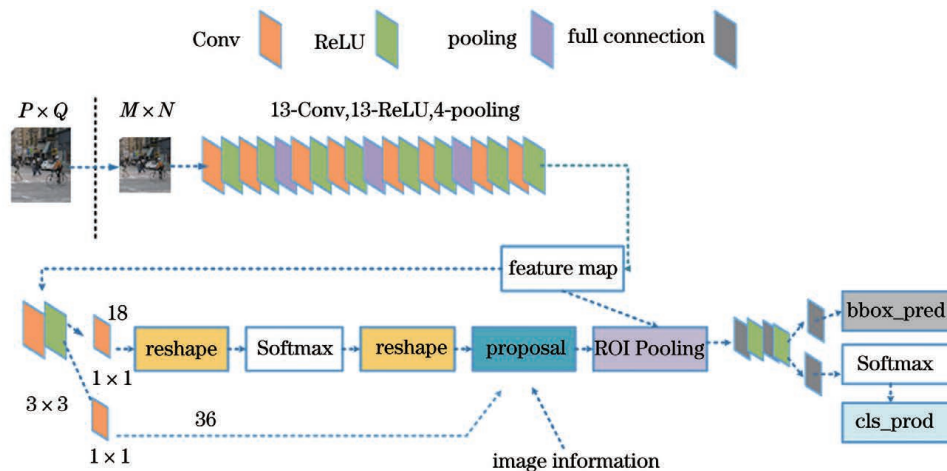


图 1 Faster-RCNN 算法的检测流程图

Fig. 1 Detection flow chart of the Faster-RCNN algorithm

Faster-RCNN 算法可分为特征提取层、RPN 层、感兴趣区域池化(ROI Pooling)层以及分类与回归四个部分。具体执行步骤如下。

1) 首先将尺寸为  $P \times Q$  的图像处理成尺寸为  $M \times N$  的图像,然后利用特征提取网络提取输入图像的特征。特征提取网络通常由卷积层(Conv)、池化层(Pooling)、修正线性单元(ReLU)激活层组成,

可使用训练好的网络架构,如 VGG (Visual geometry group) 网络<sup>[20]</sup>、残差网络(ResNet)<sup>[21]</sup>、Inception 架构<sup>[22]</sup>。

2) 将产生的特征图(Feature map)输入 RPN 中产生建议框,然后判断是否含有目标(二分类),同时将特征图传入 ROI Pooling 层进行池化操作,产生固定尺寸的候选区域特征图。

3) 对产生的候选区域特征图进行分类和回归, 得到物体的种类和位置。

Faster-RCNN 算法的核心是用 RPN 代替传统产生候选区域的方法, 实现端到端的训练, 并将整个物体的检测流程统一到同一个神经网络中, 使 RPN 和 Fast-RCNN 共享卷积特征, 减少了训练时间; ROI Pooling 层用最大值池化将特征图上的 ROI 固定为特定尺寸的特征图, 然后采用非极大值抑制 (NMS) 筛选候选框的数量。

### 3 Faster-RCNN 算法的改进

#### 3.1 自适应候选区域的调节

传统候选框选取方法大多采用滑动窗口法, 每张图像都需要消耗大量时间, 且每个步骤的参数无法共享。RPN 将物体检测流程整合到一个神经网络中, 提高了模型的检测效率, 具体检测步骤如图 2 所示。可以发现, 进入 RPN 之前卷积网络输出的通道数为 256 (256D), 256 个特征图上每个点会产生  $k$  个 anchor。RPN 先对输入特征图进行  $3 \times 3$  卷积操作, 输出的维度不变, 然后分为两条线路: 一条通过 Softmax 函数区分  $k$  个 anchor 分别是前景还是

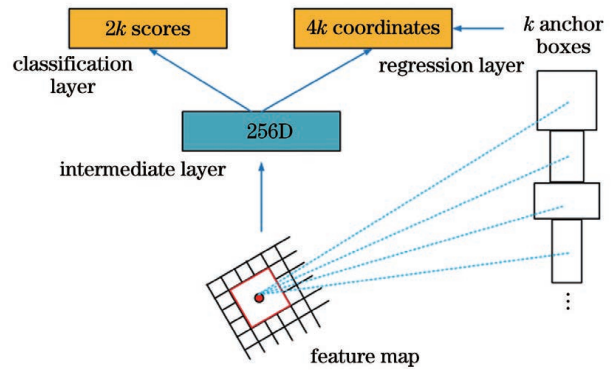


图 2 RPN 的检测步骤

Fig. 2 Detection steps of the RPN

背景的二分类, 即  $2k$  scores; 另一条线路是计算 anchor 的位置, 包括上下左右 4 个偏移量, 即  $4k$  coordinates。

为了尽可能地选取待检测目标, RPN 阶段通常会产生一万多个候选区域, 如果将这些候选区域都用于训练会大大增加训练时间。因此, 采用 NMS 算法对候选框进行筛选, 选出置信度较高的 2000 个候选框进行最后的训练。RPN 训练时采用的损失函数分为两部分, 第一个是分类损失 (Cls loss), 第二个是回归损失 (Regression loss)。损失函数可表示为

$$L = (\{p_i\}, \{t_i\}) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*), \quad (1)$$

$$p_i^* = \begin{cases} 0, & \text{negative label} \\ 1, & \text{positive label} \end{cases}, \quad (2)$$

式中,  $i$  为每个训练批次中第  $i$  个 anchor,  $p_i$  为第  $i$  个 anchor 预测为前景的概率,  $p_i^* = 0$  表示样本为负样本,  $p_i^* = 1$  表示样本为正样本,  $t_i$  为网络预测候选区域的坐标,  $t_i^*$  为对应正样本标注数据真实的坐标;  $L_{\text{cls}}(p_i, p_i^*)$  为分类损失,  $L_{\text{reg}}(t_i, t_i^*)$  为回归损失,  $p_i^* L_{\text{reg}}(t_i, t_i^*)$  表示预测结果为正样本时计算的回归损失,  $\lambda$  为权重因子。

传统 Faster-RCNN 算法在 RPN 层采用 NMS 算法筛选出 2000 个候选框进行最后的检测训练, 候选框较多会导致训练耗费大量时间。研究表明, 适当减少候选框的数量, 能加快检测速度。因此, 在 RPN 层引入 ARP 层, 使最终的候选框数量在 300~1600 之间动态波动, 进而减少训练时间。ARP 的数量可表示为

$$X_{\text{ARP}} = \begin{cases} A(1 + \beta_1), & L_j \geq \alpha L_{j-1} \\ A, & \lambda L_{j-1} < L_j < \alpha L_{j-1} \\ A(1 - \beta_2), & L_j \leq \lambda L_{j-1} \end{cases}, \quad (3)$$

式中,  $j$  为每  $N$  次训练的序号,  $A$  为第  $N \times j$  次到第  $(N+1)j$  次训练候选框的数量,  $L_j$  为训练前后两次的平均损失,  $\beta_1$  和  $\beta_2$  为惩罚因子。

引入 ARP 对候选框数目进行自适应调节, 设定每间隔  $N$  次训练计算回归损失的平均值以及不同的损失变化区间,  $(\alpha, \lambda)$  的取值分别为 (2.0, 0.8)、(1.8, 0.6)、(1.6, 0.4), 并记录不同变化区间下的总损失。对比发现, 当  $(\alpha, \lambda)$  的取值为 (1.8, 0.6) 时, 训练损失最小。因此, 设定当前平均损失  $L_i$  大于 0.6 倍前一次平均损失且小于 1.8 倍前一次平均损失为合理变化区间, 如果不在该区间, 则按照 (3) 式调整候选区域的数目, 使候选区域在一个安全的范围内。惩罚因子的取值遵循的基本原则: 为了避免候选框数量过低导致检测精度降低,  $\beta_2$  的取值不宜过大, 否则  $A(1 - \beta_2)$  过小, 会降低检测精度; 同理,  $\beta_1$  的取值不宜过小, 否则  $A(1 + \beta_1)$  过小, 也会降低检测精度。通过实验选定的候选框动态范围为 300~1600,  $(\alpha, \lambda)$  的取值为 (1.8, 0.6),  $\beta_1$



和  $\beta_2$  的取值为 (0.6, 0.4)。

### 3.2 自适应置信度阈值

根据 COCO 数据集对小目标的定义方法,将像素点数小于  $32 \times 32$  的目标定义为小目标,像素点数在  $32 \times 32$  与  $96 \times 96$  之间的目标定义为中目标,像素点数大于  $96 \times 96$  的目标定义为大目标。对大量出现漏警及误警的检测结果进行分析后发现,目标被遮挡面积超过 70% 时,检测错误率较大,因此,规定目标被遮挡面积超过 70% 时为被遮挡目标。网络模型对候选框进行分类的最后阶段,每个候选框都会得到该候选框是某种物体的置信度评分,在判断目标类别时需要设定一个阈值。如果候选框的评分高于该阈值,则认定候选框属于该类别;如果同时

超过多个目标阈值,取最高分数的目标类别。传统算法对置信度阈值的选取是不断通过手动选取尝试,直到找到合适的置信度阈值。这种方法耗时较长,且人工选取的阈值存在不合理问题,对小目标及被遮挡目标的检测能力不强。当目标较小或被遮挡时,传统 Faster-RCNN 算法检测“bottle”、“cup”、“person”、“horse”类别的目标时会出现漏检情况,如图 3 所示。原因是传统 Faster-RCNN 算法采用固定置信度阈值,阈值过高时会漏掉真目标,阈值过低时会检测到假目标;且人工选取的阈值只适应于当前数据集,更改数据集后需要调整置信度阈值。因此,引入自适应调节的思想,通过算法计算出合理的置信度阈值。



图 3 传统算法对小目标的检测结果。(a)原始图像;(b)漏警图

Fig. 3 Detection results of small targets by traditional algorithm. (a) Original image; (b) missing alarm image

为了确保阈值的准确性与合理性,首先舍弃置信度得分较低的候选框,然后利用算法对其余候选框生成合理的筛选阈值。真目标和假目标的得分差距很大,因此,舍弃置信度得分小于 0.2 的候选框。此外,假目标由于某些特征的特殊性,在检测时置信

度会超过 0.2。通过人工选取的固定阈值难以区分真目标和假目标。而根据该特点设置自适应阈值可辨别真目标和假目标,降低虚警率。

二阶差分在数学上能够反映出一组离散数组的变化趋势,对一组离散数组的二阶差分仿真中

发现,在最大值附近会出现峰值,将离散数组分开。候选框中是否含有目标两者之间的置信度差别较大,因此,对候选框的置信度得分进行二阶差分,选出最大值,作为置信度阈值。置信度变化函数可表示为

$$f(C_k) = \frac{(C_{k+1} - C_{k-1}) - (C_k - C_{k-1})}{C_k}, \quad (4)$$

式中,  $C_k$  为候选框的置信度得分。会产生  $M$  个候

选区域,经检测网络后,每个候选区域会得到属于 21 个类别(包括背景)的置信度得分,即  $M$  个  $21 \times 1$  的数组。将每个数组按照从大到小的顺序排列,取出最大值,并去掉小于 0.2 的值,得到  $m \times 1$  的数组  $C$ 。将  $f(C_k)$  取最大值时的  $C_k$  作为置信度阈值。改进后的 Faster-RCNN 结构如图 4 所示,其中,在 RPN 层加入 ARP,在分类层加入自适应置信度阈值筛选。

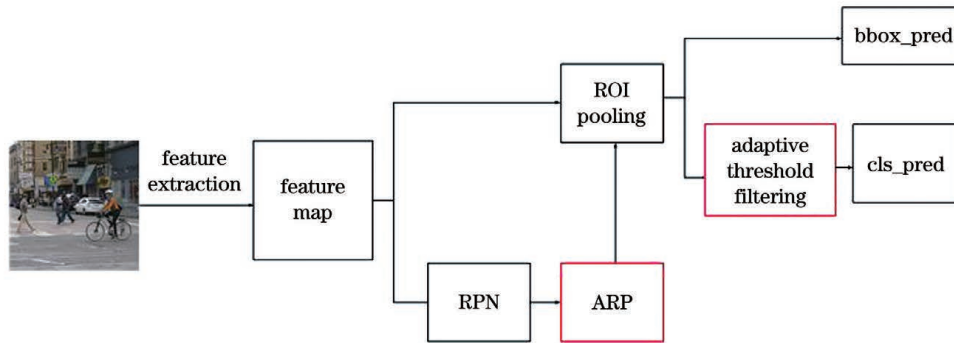


图 4 改进 Faster-RCNN 算法的结构

Fig. 4 Structure of the improved Faster-RCNN algorithm

## 4 实验结果与分析

### 4.1 实验环境

实验环境:采用英伟达 GeForce RTX2060 图形处理器(GPU),安装内存为 32 G,处理器为 Intel (R) Core(TM) i7-10750H CPU@3.6 GHz。深度学习框架使用 Python3.6 和 Tensorflow2.0。实验使用的数据集为公开数据集 PASCAL VOC2012,训练样本和测试样本共 4000 张图像,21 个不同对象类别(包括背景)。设置最大迭代次数为 60000,随机失活值为 0.6,所有实验的基础对比网络均为 VGG-16。

### 4.2 实验结果与分析

在特征提取网络模型 VGG-16 下,首先通过人工调整候选区域的数目,并记录平均检测精度(mAP)和检测时间。然后,用具有 ARP 的算法进

行实验,结果如表 1 所示。其中,  $\Delta t$  为检测速度的变化率,  $\Delta m$  为平均检测精度的变化百分点, ARP (300~1600)为采用自适应候选区域选取算法。可以发现,当候选区域从 2000 变化到 1500 时,平均检测精度没有变化;当候选框数量继续减少时,平均检测精度开始降低;当候选框数量为 50 时,检测速度提高了 56%,但平均检测精度下降了 3.7 个百分点。由于检测框在 1500~2000 变化时,检测精度基本没有变化,因此设置本算法的候选框数量在 300~1600 之间动态变化。为了验证所选候选框范围的合理性,对 3 个不同范围的候选框数量进行实验,结果如表 2 所示。可以发现,候选框数量在 200~1400 范围内时,检测速度更快,但出现了精度损失;候选框数量在 400~1800 范围内时,虽然没有出现精度损失,但检测速度没有达到最佳。因此,最终选择的候选框数量波动范围为 300~1600。

表 1 不同候选区域下的检测精度和速度

Table 1 Detection accuracy and speed under different candidate regions

Proposal	2000	1500	1000	500	50	APR(300-1600)
Detection time /s	0.215	0.194	0.160	0.135	0.095	0.157
mAP /%	73.5	73.5	72.6	71.1	69.8	73.5
$\Delta t$ /%	0	-11	-24	-37	-56	-27
$\Delta m$ /percentage point	0	0	-0.9	-2.4	-3.7	0

表 2 不同候选框波动范围内的检测精度和速度

Table 2 Detection accuracy and speed under different candidate frame fluctuation ranges

Proposal	200-1400	300-1600	400-1800
Detection time /s	0.143	0.157	0.168
mAP /%	73.1	73.5	73.5

图 5 为改进前后算法的检测结果,图 5(a)为

不同候选框下的平均检测精度;图 5(b)为不同候选框下的检测速度;图 5(c)为相比基础网络,不同候选框下传统算法的检测精度变化情况,负值表示检测精度降低;图 5(d)为不同候选框下改进算法相比传统算法的检测速度变化情况,负值表示检测速度提升。可以发现,改进后的 ARP 算法检测精度基本没有变化,但检测速度提升了 27%。

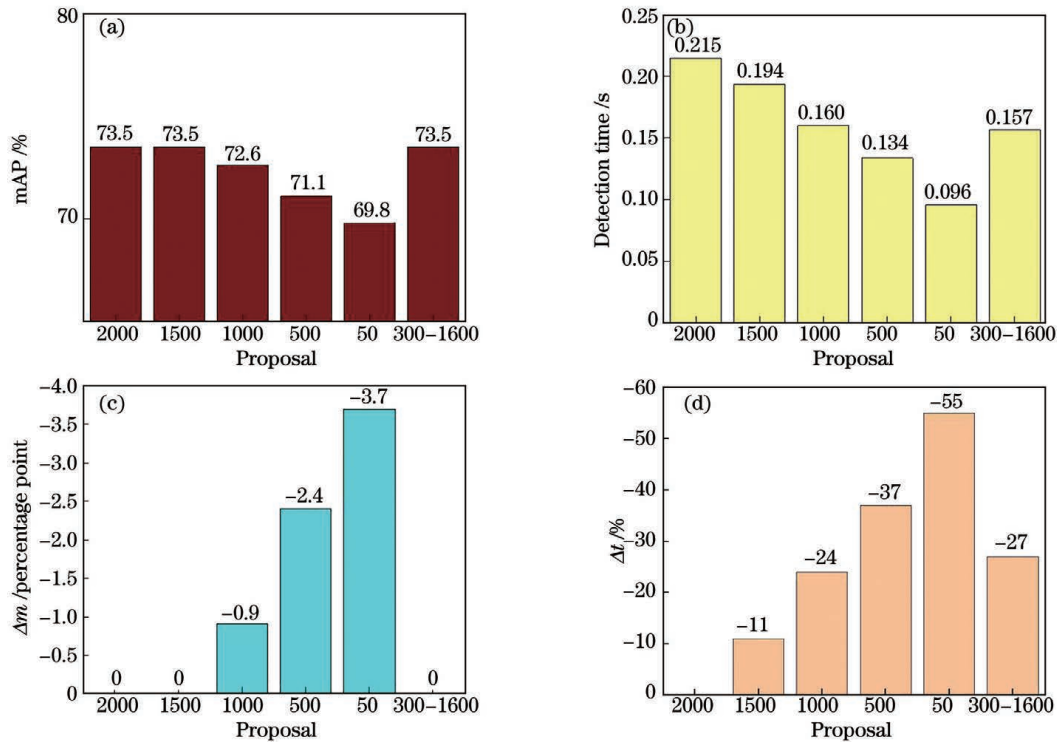


图 5 不同算法的实验结果。(a)平均检测精度;(b)检测速度;(c)精度对比结果;(d)速度对比

Fig. 5 Experimental results of different algorithms. (a) Average detection accuracy; (b) detection speed; (c) accuracy comparison result; (d) speed comparison

引入自适应置信度阈值算法(A1),以提高算法对小目标及被遮挡目标的检测效果,并另选 100 张目标密集的图像对算法改进前后的检测效果进行对比。表 3 为改进前后算法的检测精度,其中,  $S_{AP}$  为 100 张图像中小目标的检测精度。可以发现,相比传统算法,改进算法对小目标的检测精度提升了 6.9 个百分点;平均检测精度提高了 1.2 个百分点。图 6(a)为原始图像,图 6(b)为传统算法的检测效

表 3 自适应置信度阈值对算法性能的影响

Table 3 Influence of adaptive confidence threshold on algorithm performance

Algorithm	$S_{AP}$ /%	mAP /%
Faster-RCNN	13.2	73.5
Faster-RCNN+A1	20.1	74.7

果,图 6(c)为改进后算法对小目标以及被遮挡目标的检测效果。

综合以上改进,在 RPN 中引入 ARP 算法 (APR+A1),同时在分类阶段引入自适应置信度阈值选取算法,并对比该算法与传统算法的检测性能,结果如表 4 所示。可以发现,相比传统算法,改进算法的平均检测精度提高了 1.9 个百分点,检测速度提升了 25%。

表 4 APR 对算法性能的影响

Table 4 Influence of APR on algorithm performance

Algorithm	mAP /%	Detection time /s
Faster-RCNN	73.5	0.215
Faster-RCNN+APR+A1	75.4	0.161





图 6 改进算法的检测结果。(a)目标漏检图;(b)传统算法;(c)改进算法

Fig. 6 Detection results of the improved algorithm. (a) Target missed detection map; (b) traditional algorithm; (c) improved algorithm

## 5 结 论

相比传统 Faster-RCNN 算法,本算法主要有两个方面的改进:1)引入自适应候选区域选取算法,提

高模型的检测速度;2)提出自适应置信度阈值选取算法,以提高小目标以及被遮挡目标的检测能力,优化模型性能。实验结果表明,相比原始算法,改进算法的检测速度提升了 25%,检测精度提升了 1.9 个

百分点。Faster-RCNN 系列算法,在精度上比端到端的算法更具优势,但检测速度较慢,而本算法在 NMS 算法之后再次进行候选框筛选,通过在一定范围内减少候选框的数量提升检测速度,并保证检测精度不会下降。同时针对人工选取置信度阈值带来的时间成本以及检测过程中小目标与被遮挡目标漏检的问题,提出的自适应置信度阈值算法可以找出最合适的阈值,进而提高网络的检测能力。

### 参 考 文 献

- [1] Wang P, Liu R, Xin X J, et al. Scene classification of optical remote sensing images based on residual networks [J]. *Laser & Optoelectronics Progress*, 2021, 58(2): 0210001.  
汪鹏, 刘瑞, 辛雪静, 等. 基于残差网络的光学遥感图像场景分类算法 [J]. *激光与光电子学进展*, 2021, 58(2): 0210001.
- [2] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), June 20-25, 2005, San Diego, CA, USA. New York: IEEE Press, 2005: 886-893.
- [3] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. *International Journal of Computer Vision*, 2004, 60(2): 91-110.
- [4] Lienhart R, Maydt J. An extended set of Haar-like features for rapid object detection[C]//Proceedings of International Conference on Image Processing, September 22-25, 2002, Rochester, NY, USA. New York: IEEE Press, 2002: I.
- [5] Lienhart R, Maydt J. Support vector machines[M]//An introduction to support vector machines and other kernel-based learning methods. Cambridge: Cambridge University Press, 2000: 93-124.
- [6] Freund Y, Schapire E. Experiments with a new boosting algorithm[C]//Machine Learning, Proceedings of the Thirteenth International Conference (ICML'96), July 3-6, 1996, Bari, Italy. USA: IMLS, 1996: 148-156.
- [7] Svetnik V, Liaw A, Tong C, et al. Random forest: a classification and regression tool for compound classification and QSAR modeling [J]. *Journal of Chemical Information and Computer Sciences*, 2003, 43(6): 1947-1958.
- [8] Ju M R, Luo H B, Wang Z B, et al. Improved YOLO V3 algorithm and its application in small target detection [J]. *Acta Optica Sinica*, 2019, 39(7): 0715004.
- [9] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2014, Columbus, OH, USA. New York: IEEE Press, 2014: 580-587.
- [10] Uijlings J R R, Sande K, Gevers T, et al. Selective search for object recognition [J]. *International Journal of Computer Vision*, 2013, 104(2): 154-171.
- [11] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection [C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 779-788.
- [12] Redmon J, Farhadi A. YOLO9000: better, faster, stronger [C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 6517-6525.
- [13] Redmon J, Farhadi A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08)[2021-01-03]. <https://arxiv.org/abs/1804.02767>.
- [14] 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.
- [15] 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.
- [16] Dai C K, Li Y. Aeroplane detection in static aerodrome based on Faster RCNN and multi-part model [J]. *Journal of Computer Applications*, 2017, 37(S2): 85-88.  
戴陈卡, 李毅. 基于 Faster RCNN 以及多部件结合的机场场面静态飞机检测 [J]. *计算机应用*, 2017, 37(S2): 85-88.
- [17] Lin L. Small object detection based on deep learning [D]. Chengdu: University of Electronic Science and Technology of China, 2020.  
林莉. 基于深度学习的小目标检测 [D]. 成都: 电子科技大学, 2020.
- [18] Dong Y F, Zhang C T, Wang P, et al. Airplane detection of optical remote sensing images based on deep learning [J]. *Laser & Optoelectronics Progress*, 2020, 57(4): 041007.



- 董永峰, 仇长涛, 汪鹏, 等. 基于深度学习的光学遥感图像飞机检测算法 [J]. 激光与光电子学进展, 2020, 57(4): 041007.
- [19] Sun J, Guo D B, Yang T T, et al. Real-time object detection based on improved YOLOv3 network [J]. Laser & Optoelectronics Progress, 2020, 57(22): 221505.
- 孙佳, 郭大波, 杨甜甜, 等. 基于改进的 YOLOv3 网络的实时目标检测 [J]. 激光与光电子学进展, 2020, 57(22): 221505.
- [20] Wu J, M Y, Shi Q W, et al. Behavior recognition based on the fusion of 3D-BN-VGG and LSTM network [J]. High Technology Letters, 2020, 26(4): 372-382.
- [21] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition [C] // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 770-778.
- [22] Zhu Z K, Li J B. Multi-feature information fusion LSTM-RNN detection for OSA [J]. Journal of Computer Research and Development, 2020, 57(12): 2547-2555.
- 朱兆坤, 李金宝. 多特征信息融合 LSTM-RNN 检测 OSA 方法 [J]. 计算机研究与发展, 2020, 57(12): 2547-2555.