

基于改进的 Faster R-CNN 目标检测算法

周兵, 李润鑫*, 尚振宏, 李晓武

昆明理工大学信息工程与自动化学院, 云南 昆明 650500

摘要 目标检测是计算机视觉研究中的热门问题, 其中加速区域卷积神经网络(Faster R-CNN)对目标检测具有指导意义。针对 Faster R-CNN 算法在目标检测中准确率不高的问题, 先对数据进行增强处理; 然后对提取的特征图进行裁剪, 利用双线性插值法代替感兴趣区域池化操作, 分类时采用软非极大值抑制(Soft-NMS)算法。实验结果表明, 该算法在 PASCAL VOC2007、PASCAL VOC07+12 数据集下的准确率分别为 76.40% 和 81.20%, 相较 Faster R-CNN 算法分别提升了 6.50 个百分点和 8.00 个百分点。没有进行数据增强的情况下, 在 COCO 2014 数据集上的准确率相较 Faster R-CNN 算法提升了 2.40 个百分点。

关键词 目标检测; 加速区域卷积神经网络(Faster R-CNN); 感兴趣区域池化; 软非极大值抑制(Soft-NMS)

中图分类号 TP391

文献标志码 A

doi: 10.3788/LOP57.101009

Object Detection Algorithm Based on Improved Faster R-CNN

Zhou Bing, Li Runxin*, Shang Zhenhong, Li Xiaowu

Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, Yunnan 650500, China

Abstract Object detection is a hot topic in computer vision research, among which faster region-based convolutional neural network (Faster R-CNN) has guiding significance for object detection. Aiming at the problem of the low accuracy of the Faster R-CNN algorithm in object detection, the data is enhanced first. Then, the extracted feature map is trimmed, and bilinear interpolation is used to replace the region of interest pooling operation. Soft-non-maximum suppression (Soft-NMS) algorithm is used for classification. Experimental results show that the accuracy of the algorithm is 76.40% and 81.20% in PASCAL VOC2007 and PASCAL VOC07+12 datasets, which is 6.50 percentage points and 8.00 percentage points higher than that of the Fast R-CNN algorithm, respectively. Without data enhancement, the accuracy on the COCO 2014 dataset is improved by 2.40 percentage points compared with that of the Faster R-CNN algorithm.

Key words object detection; faster region-based convolutional neural network (Faster R-CNN); region of interest pooling; soft-non-maximum suppression (Soft-NMS)

OCIS codes 100.3008; 100.4996

1 引言

目标检测在自动驾驶、视频、图像索引等方面有广泛应用^[1], 随着深度学习的快速发展, 区域卷积神经网络(R-CNN)^[2]系列算法将传统手工提取特征的方式转变为特征学习的方式, 在目标检测领域中有重要地位。

R-CNN 算法采用选择性搜索^[3](SS)对图像提

取 1000 ~ 2000 个建议框, 每个候选框周围加 16 pixel 作为建议框像素平均值的边框, 用所有候选框像素减去该建议框像素平均值后, 将结果输入 AlexNet^[4]网络进行特征提取; 再使用支持向量机(SVM)分类确定候选框的类别, 用非极大值抑制(NMS)算法减少冗余的候选框数量, 最后将剩余的候选框通过检测框回归模型, 修正最终的位置。R-CNN 的训练阶段局限于候选框的选择, 存在的问题: 1) 由

收稿日期: 2019-08-28; 修回日期: 2019-10-11; 录用日期: 2019-10-18

基金项目: 国家自然科学基金(61462052)、昆明理工大学引进人才科研启动项目(KKSY201603016)

* E-mail: rxli@kmust.edu.cn

于 R-CNN 读取图像信息时需要固定图像尺寸,但对图像进行裁剪会导致图像信息的缺失;2) CNN 需要对每个区域的候选框进行计算,其中重复的特征提取会带来巨大的计算浪费。相比于 R-CNN 算法, Fast R-CNN^[5] 算法对每张图像只进行一次特征提取,将提取的特征在计算中共享,以提升训练和测试速度。针对 Fast R-CNN 不能达到实时性检测和端到端训练测试的问题, Ren 等^[6] 提出了 Faster R-CNN 算法,利用区域建议网络(RPN)代替 SS 算法,可有效提取出原始图像中的候选区域。

Faster R-CNN 算法中的感兴趣区域(ROI)池化只能进行整数的计算,因此必须将产生的浮点数转换为整数进行池化,从而导致原图和特征图中的感兴趣区域位置不匹配,可通过 crop_and_resize^[7] 操作解决该问题。此外, Faster R-CNN 在分类时采用 NMS^[8] 算法删除了所有交并比(IOU)大于阈值的候选框,导致置信度降低,可通过软非极大值抑制(Soft-NMS)^[9] 算法解决该问题。为进一步提高算法的精度和鲁棒性,本文在数据预处理阶段,采用多种数据增强方式,实验结果表明,该方法可有效提升目标的检测精度。

2 数据增强

目标检测技术在实际中,经常会出现数据集不够导致检测精度不高的问题。传统的数据增强技术只是简单对图像进行缩放、颜色变换、色彩抖动等,本方法结合了三种数据增强^[10] 技术,提升了模型的泛化能力和鲁棒性。图像尺寸可缩放为 256 pixel ×

256 pixel, 448 pixel × 448 pixel, 512 pixel × 512 pixel 等,可随机改变图像亮度、饱和度、对比度。实验主要针对 RASCAL VOC 数据进行增强,同时生成了对应的 XML 文件。

3 Faster R-CNN 框架分析

3.1 整体框架

Faster R-CNN 的算法结构如图 1 所示,首先利用 CNN 提取输入图像的特征图^[11-12],然后通过共享卷积层将特征图分别送入 Fast R-CNN 和 RPN 中。一方面将输入 RPN 的特征图经滑动窗口产生的特征向量送入全连接层(FC1, FC2),再进行分类和回归生成区域建议;另一方面通过 CNN 提取特征图,然后把区域建议和特征图送入 ROI 以固定其尺寸,最后通过两个损失函数进行分类和回归,生成预测坐标值,从而得到检测目标框。

3.2 区域建议网络

RPN 是通过一个全卷积网络来实现的,具体结构如图 2 所示,其中 512-d 表示 512 维向量。可以看到,在 RPN 结构中,滑动窗口对特征图进行卷积操作产生的 n 维特征向量会输入到回归层和分类层。滑动窗口每滑动一次会预测出 k 个区域建议,通常每个滑动位置对应着 3 种尺度和 3 种宽高比(三种尺度分别为 128^2 、 256^2 和 512^2 ,三种宽高比分别为 1:1, 1:2 和 2:1),回归层需要预测出包围框(包含检测物体的矩形框)的宽高和中心坐标,分类层需要辨别建议框是前景或背景的概率,因此回归层需要输出 4×10^3 个坐标,分类层包含 2×10^3 个得分。

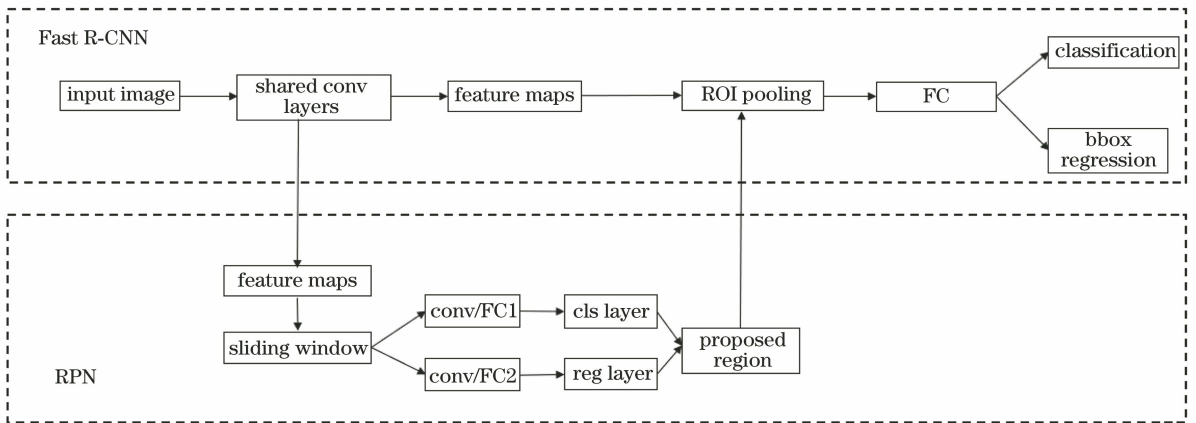


图 1 Faster R-CNN 算法结构图

Fig. 1 Structure of Faster R-CNN algorithm

3.3 感兴趣区域池化的改进

ROI 的具体操作:将输入图像的感兴趣区域映射到特征图对应的位置,将其固定成 7 pixel ×

7 pixel 的大小,对该区域进行最大池化操作。ROI 存在很多问题,如图 3 所示,输入一张尺寸为 960 pixel × 960 pixel 的图像,有一个尺寸为

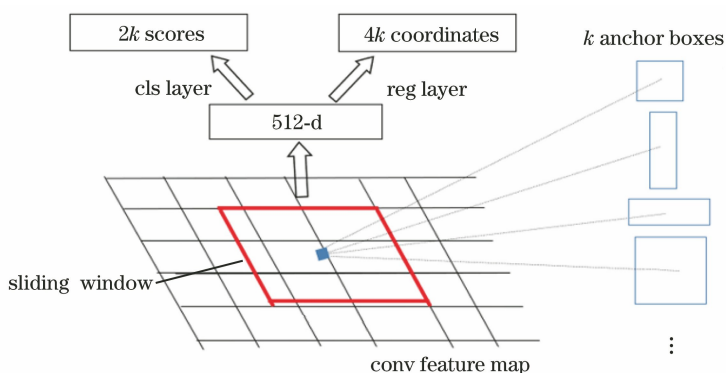


图 2 区域建议网络

Fig. 2 Region proposal network

775 pixel×775 pixel 的框(一架飞机)。图像经 VGG-16 网络(步长为 32)后得到的图像边长较原图缩小了 1/32,将边界区域继续分割成 7 pixel×7 pixel 的单元。由于 ROI 只能进行整数运算,因此

将两次产生的浮点数都进行了整数化,量化后的候选框相对首次产生的候选框已经发生了位置偏差,因此采用 crop_and_resize 操作,将特征图尺寸固定为 14 pixel×14 pixel。

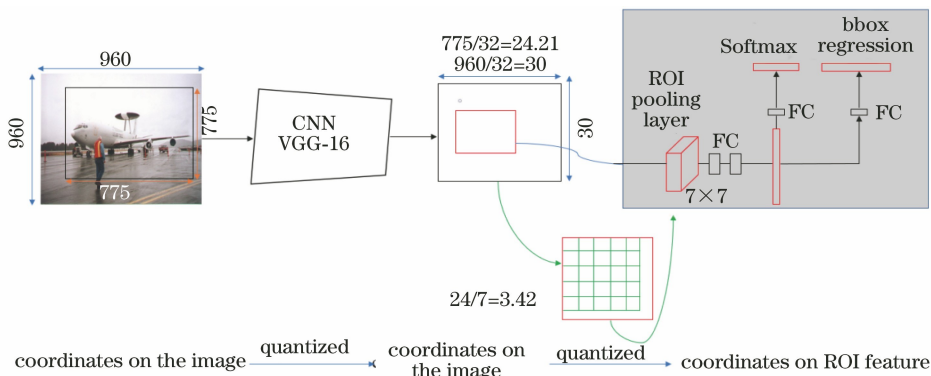


图 3 ROI 池化存在的问题

Fig. 3 Problem with ROI pooling

crop_and_resize 的操作方式:在 ROI 层前添加一层 warping layer,对每一个感兴趣区域都进行 crop 和 warp 的操作,将感兴趣区域转换成尺寸为 14 pixel×14 pixel 的特征图,可表示为

$$F_i(u',v') = \sum_{(u,v)}^{W \times H} G(u,v;u',v' | B_i) F_i(u,v), \quad (1)$$

式中, W, H 分别为感兴趣区域宽和高, F_i 为经 ROI 后的第 i 个特征图, (u, v) 为 ROI 在原始图像中的坐标, (u', v') 为进行 crop 和 resize 操作后的 ROI 坐标,函数 G 为双线性插值函数,目的是对宽和高进行相乘, $F_i(u, v)$ 为原始的 ROI, B_i 为第 i 个预测框。

3.4 非极大值抑制算法的改进

NMS 算法广泛应用于边缘和目标检测中,可解决分类器分类时目标被大量候选框包围情况下,出现候选框大量重叠的问题,NMS 算法的步骤:

1) 将全部的候选框按得分大小排序,选择出得

分最高的候选框。

2) 将剩余的候选框依次与得分最高的候选框对比,如果两者重叠面积大于一定阈值时,就删除该框。

3) 在未处理的候选框(与得分最高框没有重叠的框)中继续选一个得分最高的,重复步骤 2),只留下最优框中的目标。

NMS 算法在目标检测中可理解为一个对边界框进行评分的过程,其线性加权函数可表示为

$$s_i^{NMS} = \begin{cases} s_i^{NMS}, & X_{IOU}(M, b_i) < N_t \\ 0, & X_{IOU}(M, b_i) > N_t \end{cases}, \quad (2)$$

式中, s_i^{NMS} 为第 i 个预测框对应的 IOU, N_t 为抑制阈值, b_i 为待过滤的第 i 个预测框, M 为当前得分最大的边界框, X_{IOU} 为预测边界框面积 A 与实际边界框面积 B 的交集比上其并集,可表示为

$$X_{IOU} = \frac{A \cap B}{A \cup B}. \quad (3)$$

由上述步骤可知,检测同一类物体,且物体间重叠部分较多时,会将原本的预测框删除。如图4所示,检测算法本应输出两个框,分别是置信度为0.80和0.95的框。但由于传统的NMS算法可能会把置信度较低的实线框删除,导致置信度为0.80的框的置信度变为0.00或降低为0.40。可以看出,真实检测出的输出有两个框,若NMS下实线框和虚线框的IOU大于设定的阈值,且实线框的分数较低就会被删除,导致只能检测出虚线框中的目标(完整的长颈鹿),降低了物体的召回率。为了解决该问题,采用Soft-NMS算法替代NMS,可以根据当前得分重新递归评分,而不是直接删除分数较低的相邻方框,在同一类物体高度重叠时降低了误删预测框的情况,且Soft-NMS算法在训练阶段不需要引入任何超参数,用于调整Soft-NMS算法的超参数只会出现在测试或演示阶段,不会增加计算的复杂度。Soft-NMS算法的步骤:

1) 按照不同的种类对标签分组,预测出不同标签中的所有候选区域;

2) 将每一类的所有框记作 E ,筛选后的框集

合记作 D 。a)先选择得分最高的框 M ,加入到 D 中;b)计算剩余框与 M 的重叠面积,若大于设定阈值 N_t ,则舍弃,反之保留;c)若步骤b)中得到的所有框为空,则返回到步骤2),反之继续执行步骤a);

3) 处理完成之后,将所有类别保留在有效框的集合内。

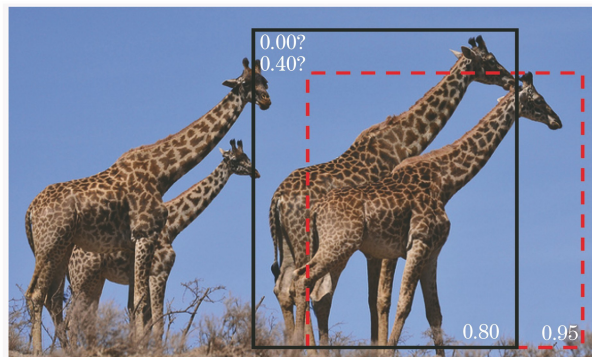


图4 非极大值抑制算法存在的问题

Fig. 4 Problem with non-maximum suppression algorithm

Soft-NMS中的线性加权函数可表示为

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

式中, s_i^{Soft} 为第 i 个预测框对应的分类得分,文献[10]在相同数据下,对比了Soft-NMS与NMS在不同的IOU阈值下的平均精度均值(mAP),结果表明IOU阈值设置为0.45~0.60时效果最佳,因此实验将IOU阈值设为0.60。

4 实验结果与分析

所有实验均在Ubuntu16.04系统下进行,其中深度学习框架为tensorflow,硬件环境: Intel Corei7-7800X 3.50 GHz CPU, GTX Titan X GPU, 显存为12 GB。为验证本算法的有效性,采用PASCAL VOC2007^[13]、PASCAL VOC2012和COCO2014^[14]数据集进行测试。

4.1 定量分析

表1是PASCAL VOC2007数据集的检测结果,该数据集包含20个类别,有9963张图片,4981张用于训练,4982张用于测试。均采用VGG-16算法^[15]利用ImageNet数据集^[16]对网络进行预训练,将原

PASCAL VOC2007的训练集、验证集和数据增强1倍的PASCAL VOC2007图像(VOC07++)作为实验的训练集,原PASCAL VOC2007的测试集作为测试集(VOC2007)。从表1可以看出,与Faster R-CNN算法相比,只考虑数据增强技术算法的mAP提高了1.00个百分点;只考虑改进ROI和NMS算法的mAP提高了3.20个百分点;将上述两种方式结合后算法的mAP值提高了6.50个百分点,与表中的其他算法相比,优势也比较明显。

Faster R-CNN区域建议包含三种尺度(128^2 、 256^2 和 512^2)、三种宽高比(1:1、1:2和2:1)。在宽高比不变的情况下,改变尺度的大小,从而选择出最优尺度。表2为不同尺度下对PASCAL VOC07++数据集的检测结果,该数据集含有19926张图片,取14944张作为训练集和验证集(VOC07++),4982张作为测试集(VOC2007)。可以看出anchor的尺度在 64^2 、 128^2 、 256^2 和 512^2 时效果最佳,因此在实验中选择这四种尺度。

表1 PASCAL VOC2007 测试结果

Table 1 Test results on the PASCAL VOC2007

Algorithm	Backbone	Training set	Testing set	mAP / %
Fast R-CNN	VGG-16	VOC2007	VOC2007	66.90
Faster R-CNN	VGG-16	VOC2007	VOC2007	69.90
SSD300	VGG-16	VOC2007	VOC2007	68.00
YOLO	GoogleNet	VOC2007	VOC2007	63.40
Data enhancement	VGG-16	VOC2007	VOC2007	70.90
Soft-NMS+crop_and_resize	VGG-16	VOC07++	VOC2007	73.10
Ours	VGG-16	VOC07++	VOC2007	76.40

表2 不同尺度下 PASCAL VOC07++数据集的检测结果

Table 2 Detection results on PASCAL VOC07++ data set at different scales

Enter size	Backbone	Training set	Testing set	mAP / %
$128^2, 256^2, 512^2$	VGG-16	VOC07++	VOC2007	76.40
$64^2, 128^2, 256^2, 512^2$	VGG-16	VOC07++	VOC2007	77.69
$32^2, 64^2, 128^2, 256^2, 512^2$	VGG-16	VOC07++	VOC2007	77.63

将 PASCAL VOC2012 的训练集、验证集和 PASCAL VOC2007 的训练集、验证集、测试集作为实验的数据集 (PASCAL VOC07+12), 其中 PASCAL VOC2012 的训练集、验证集和 PASCAL VOC2007 的训练集、验证集作为实验的训练集 (VOC07+12), PASCAL VOC2007 作为实验的测试集 (VOC2007), 结果如表 3 所示。可以发现, 与

Faster R-CNN 算法相比, 只考虑数据增强技术的算法 mAP 提高了 2.60 个百分点; 只考虑改进 ROI 和 NMS 的算法的 mAP 提高了 5.20 个百分点; 同时融合上述两种方式的算法与 Fast R-CNN、Faster R-CNN 和 SSD^[17] 算法的 mAP 相比均有明显的提升, 这证明了该算法的有效性。

表3 PASCAL VOC07+12 测试集的检测结果

Table 3 Test results on PASCAL VOC07+12 test set

Algorithm	Backbone	Training set	Testing set	mAP / %
Fast R-CNN	VGG-16	VOC07+12	VOC2007	70.00
Faster R-CNN	VGG-16	VOC07+12	VOC2007	73.20
Faster R-CNN	ResNet-101	VOC07+12	VOC2007	76.40
MR-CNN	ResNet-101	VOC07+12	VOC2007	78.20
ION	VGG-16	VOC07+12	VOC2007	76.50
YOLO	GoogleNet	VOC07+12	VOC2007	63.40
YOLOV2	Darknet-19	VOC07+12	VOC2007	78.60
SSD300	VGG-16	VOC07+12	VOC2007	77.20
Data enhancement	VGG-16	VOC07+12	VOC2007	75.80
Soft-NMS+crop_and_resize	VGG-16	VOC07+++12	VOC2007	78.40
Ours	VGG-16	VOC07+++12	VOC2007	81.20

将 PASCAL VOC2012 中的训练集、验证集, PASCAL VOC2007 中的训练集、验证集、测试集以及 PASCAL VOC2007 数据集增强一倍的图像作为本实验的数据集 (PASCAL VOC07+++12), 其中 PASCAL VOC2012 中的训练集、验证集, PASCAL VOC2007 数据集中的训练集、验证集以及 PASCAL VOC2007 数据集增强一倍的图像作为实验的训练集 (VOC07+++12), PASCAL VOC2007 的测试集作为实验的测试集 (VOC2007), 结果如表 4 所示。可以看出 anchor 的

尺度在 $64^2, 128^2, 256^2$ 和 512^2 效果最佳, 因此在实验中选择四种尺度。

表 5 为不同算法在 COCO2014 数据集下的 mAP, 表中除第三行中的算法采用 ResNet-101 网络, 其余算法均采用 VGG-16 网络。其中 COCO2014 中含有 80 个类别, 训练集 82783 张图片, 验证集 40504 张图片, 测试集 81434 张图片。表中的 train 表示训练模型为训练集中的 82783 张图片, trainval35 表示训练模型为训练集中的 82783 张图片和验证集中的 35000 张图片。其中 0.50:0.95 表

表 4 不同尺度下 PASCAL VOC07++12 的检测结果

Table 4 Detection results on PASCAL VOC07++12 at different scales

Enter size	Backbone	Training set	Testing set	mAP / %
$128^2, 256^2, 512^2$	VGG-16	VOC07++12	VOC2007	81.22
$64^2, 128^2, 256^2, 512^2$	VGG-16	VOC07++12	VOC2007	83.00
$32^2, 64^2, 128^2, 256^2, 512^2$	VGG-16	VOC07++12	VOC2007	82.94

表 5 不同算法在 COCO2014 的 mAP

Table 5 mAP of different algorithms on COCO2014

unit: %

Algorithm	Training set	IOU			Image size		
		0.50:0.95	0.50	0.75	S	M	L
Fast R-CNN	train	19.70	35.90	-	-	-	-
Faster R-CNN	train	20.50	39.90	19.40	4.10	20.00	35.80
Faster R-CNN	train	21.90	42.70	-	-	-	-
ION ^[18]	train	23.60	43.20	23.60	6.40	24.10	38.30
Faster R-CNN	trainval35	24.20	45.30	23.50	7.70	26.40	37.10
SSD300	trainval35	23.20	41.20	23.40	5.30	23.20	39.60
SSD512	trainval35	26.80	46.50	27.80	9.00	28.90	41.90
YOLOV2 ^[19]	trainval35	21.60	44.00	19.20	5.00	22.40	35.50
Ours	trainval35	26.60	47.20	27.00	11.40	30.80	37.10

示 IOU 以 0.05 的步长从 0.50 到 0.95 时, 在 COCO2014 数据集下检测的 mAP; S、M、L 分别表示检测物体面积小于 $32 \text{ pixel} \times 32 \text{ pixel}$ 、在 $32 \text{ pixel} \times 32 \text{ pixel}$ 到 $96 \text{ pixel} \times 96 \text{ pixel}$ 之间、大于 $96 \text{ pixel} \times 96 \text{ pixel}$; -表示参考算法未给出相关结论。可以发现该算法在检测中小物体效果最优,

相较原始的 Faster R-CNN 算法精度提高了 2.40 个百分点左右, 这验证了本算法的可行性和鲁棒性。

表 6 为不同算法在 COCO2014 上的平均召回率(mAR), 可以发现, 本算法相比表中其他算法的 mAR 有明显提高。

表 6 不同算法在 COCO2014 的 mAR

Table 6 mAR of different algorithms on COCO2014

unit: %

Algorithm	Training set	Number of iterations			Image size		
		1	10	100	S	M	L
Faster R-CNN	train	21.30	29.50	30.10	7.30	32.10	52.00
ION	train	23.20	32.70	33.50	10.10	37.70	53.60
Faster R-CNN	trainval35	23.80	34.00	34.60	12.00	38.50	54.40
SSD300	trainval35	22.50	33.20	35.30	9.60	37.60	56.50
SSD512	trainval35	24.80	37.50	39.80	14.00	43.50	59.00
YOLOV2	trainval35	20.70	31.60	33.30	9.80	36.50	54.40
Ours	trainval35	25.50	38.30	39.30	19.70	45.50	55.40

4.2 定性分析

为了更直观地展示本算法的实验效果, 与其他

三种相关算法在数据集 PASCAL VOC07+12 进行了对比实验, 结果如图 5~图 7 所示。从图 5 中

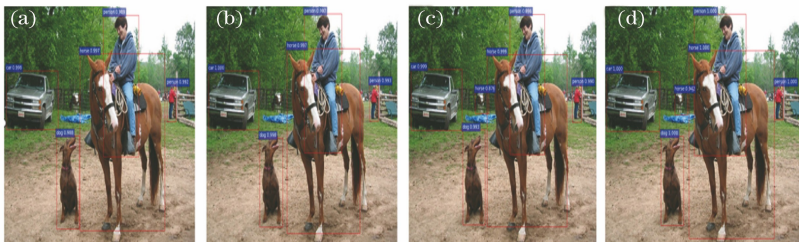


图 5 正常条件下的检测结果。(a) Faster R-CNN; (b) 添加 Soft-NMS+crop_and_resize; (c) 添加数据增强; (d) 本算法

Fig. 5 Detection results with normal conditions. (a) Faster R-CNN; (b) add Soft-NMS+crop_and_resize;

(c) add data enhancement; (d) our algorithm

可以看出,与原始 Faster R-CNN 算法、加入 Soft-NMS+crop_and_resize 的算法相比,本算法可以检测出图中的小马;与加入数据增强的算法相比检测精度也得到了提高。从图 6 中可以看出,与其他三

种算法相比,本算法检测出图中的两个人。从图 7 中可以看出,与其他三种算法相比,本算法能检测出的物体数量更多。这表明本算法在实际中是可行的。



图 6 灰度图像下的检测结果。(a) Faster R-CNN;(b)添加 Soft-NMS+crop_and_resize;(c)添加数据增强;(d)本算法

Fig. 6 Detection results with grayscale image. (a) Faster R-CNN; (b) add Soft-NMS+crop_and_resize; (c) add data enhancement; (d) our algorithm



图 7 多目标重叠下的检测结果。(a) Faster R-CNN;(b)添加 Soft-NMS+crop_and_resize;(c)添加数据增强;(d)本算法

Fig. 7 Detection results with multiple targets overlapping. (a) Faster R-CNN; (b) add Soft-NMS+crop_and_resize; (c) add data enhancement; (d) our algorithm

5 结 论

针对 Faster R-CNN 算法在目标检测中存在的问题,首先采用 crop_and_resize 操作解决了 ROI 因量化导致的位置不匹配问题;然后使用 Soft-NMS 算法提升了 Faster R-CNN 算法检测同一类物体时的精度;最后在数据预处理阶段结合了三种数据增强技术,解决了算法中原始数据不足的问题。实验结果表明,改进后的算法 mAP 值相较于 Faster R-CNN 有明显提升,且呈现的视觉效果更好。但该算法的检测速度并没有达到实时检测的要求,未来工作将针对该问题进行研究。

参 考 文 献

[1] Franke U, Gavrila D, Gorzig S, et al. Autonomous driving goes downtown [J]. IEEE Intelligent Systems, 1998, 13(6): 40-48.

[2] 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, 2014: 580-587.

[3] Uijlings J R R, van de Sande K E A, Gevers T, et al. Selective search for object recognition [J]. International Journal of Computer Vision, 2013, 104(2): 154-171.

[4] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]. Advances in Neural Information Processing Systems: Proceedings of the 25th International Conference on Neural Information Processing Systems-Volume 1, New York: Curran Associates Inc.5, 2012: 1097-1105.

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

[6] 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.
- [7] Chen X L, Gupta A. An implementation of faster RCNN with study for region sampling [EB/OL]. [2019-07-30]. <https://arxiv.org/abs/1702.02138>.
- [8] Neubeck A, van Gool L. Efficient non-maximum suppression[C] // 18th International Conference on Pattern Recognition, August 20-24, 2006, Hong Kong, China. New York: IEEE, 2006, 3:850-855.
- [9] Bodla N, Singh B, Chellappa R, et al. Improving object detection with one line of code [EB/OL]. [2019-08-01]. <https://arxiv.org/abs/1704.04503>.
- [10] Qiao T, Su H S, Liu G H, et al. Object detection algorithm based on improved feature extraction network [J]. Laser & Optoelectronics Progress, 2019, 56(23): 231008.
乔婷, 苏寒松, 刘高华, 等. 基于改进的特征提取网络的目标检测算法[J]. 激光与光电子学进展, 2019, 56(23): 231008.
- [11] He Z C, Zhao L Z, Chen C. Convolution neural network with multi-resolution feature fusion for facial expression recognition[J]. Laser & Optoelectronics Progress, 2018, 55(7): 071503.
何志超, 赵龙章, 陈闯. 用于人脸表情识别的多分辨率特征融合卷积神经网络[J]. 激光与光电子学进展, 2018, 55(7): 071503.
- [12] Zhou S R, Tan F, Zeng D J. Human action recognition based on multi-size pooling of convolutional neural network[J]. Journal of Chinese Computer Systems, 2017, 38(8): 1893-1898.
周书仁, 谭凤, 曾道建. 多尺寸池化卷积神经网络的人体行为识别研究[J]. 小型微型计算机系统, 2017, 38(8): 1893-1898.
- [13] Everingham M, van Gool L, Williams C K I, et al. The pascal visual object classes (VOC) challenge[J]. International Journal of Computer Vision, 2010, 88(2): 303-338.
- [14] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: common objects in context [M] // Fleet D, Pajdla T, Schiele B, et al. Computer vision-ECCV 2014. Lecture notes in computer science. Cham: Springer, 2014, 8693: 740-755.
- [15] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[EB/OL]. [2019-08-14]. <https://arxiv.org/abs/1409.1556>.
- [16] Deng J, Dong W, Socher R, et al. ImageNet: a large-scale hierarchical image database [C] // 2009 IEEE Conference on Computer Vision and Pattern Recognition, June 20-25, 2009, Miami, FL. New York: IEEE, 2009: 248-255.
- [17] Chen L L, Zhang Z D, Peng L. Real-time detection based on improved single shot MultiBox detector[J]. Laser & Optoelectronics Progress, 2019, 56(1): 011002.
陈立里, 张正道, 彭力. 基于改进 SSD 的实时检测方法[J]. 激光与光电子学进展, 2019, 56(1): 011002.
- [18] Bell S, Zitnick C L, Bala K, et al. Inside-outside net: detecting objects in context with skip pooling and recurrent neural networks [C] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE, 2016, 3:850-855.
- [19] Redmon J, Farhadi A. YOLO9000: better, faster, stronger. [EB/OL]. [2019-02-10]. <https://arxiv.org/abs/1612.08242>.