

回回收敛缩放混合的深度迭代复合缩放CNN目标检测算法

王国刚, 吴艳, 刘一博

引用本文

王国刚, 吴艳, 刘一博. 回回收敛缩放混合的深度迭代复合缩放CNN目标检测算法[J]. 计算机科学, 2023, 50(6A): 220500230-9.

WANG Guogang, WU Yan, LIU Yibo. Target Detection Algorithm Based on Compound Scaling Deep Iterative CNN by RegressionConverging and Scaling Mixture [J]. Computer Science, 2023, 50(6A): 220500230-9.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于改进Yolov4-tiny的轻量型目标检测算法](#)

Lightweight Target Detection Algorithm Based on Improved Yolov4-tiny

计算机科学, 2023, 50(6A): 220700006-7. <https://doi.org/10.11896/jsjcx.220700006>

[CT影像阶段化目标检测方法研究](#)

Study on Phased Target Detection in CT Image

计算机科学, 2023, 50(6A): 220200063-10. <https://doi.org/10.11896/jsjcx.220200063>

[基于区域注意力机制和多尺度特征融合的输电线路螺栓缺陷检测](#)

Defect Detection of Transmission Line Bolt Based on Region Attention Mechanism and Multi-scale Feature Fusion

计算机科学, 2023, 50(6A): 220200096-7. <https://doi.org/10.11896/jsjcx.220200096>

[基于改进YOLOv5的电动车头盔佩戴检测算法](#)

Electric Bike Helmet Wearing Detection Algorithm Based on Improved YOLOv5

计算机科学, 2023, 50(6A): 220500005-6. <https://doi.org/10.11896/jsjcx.220500005>

[基于主动学习和U-Net++分割的芯片封装空洞率的研究](#)

Study on BGA Packaging Void Rate Detection Based on Active Learning and U-Net++ Segmentation

计算机科学, 2023, 50(6A): 220200092-6. <https://doi.org/10.11896/jsjcx.220200092>

回归收敛缩放混合的深度迭代复合缩放 CNN 目标检测算法

王国刚 吴艳 刘一博

山西大学物理与电子工程学院 太原 030006

摘要 针对 EfficientDet 算法鲁棒性低、回归损失函数收敛性能差、标签边缘化问题,提出了回归收敛缩放混合的深度迭代复合缩放 CNN 目标检测算法。该算法采用 2×2 缩放混合正则化方法,增强训练样本,避免训练过拟合,提高模型泛化能力;利用完全交并比损失抑制冗余预测框,将中心点距离和纵横比作为边界框坐标预测的损失函数惩罚项,使卷积神经网络回归更准确,提高了收敛速度和定位精度;设置平滑参数,对边缘化标签分布和均匀分布加权求和生成标签平滑正则化分布,建立类标签平滑交叉熵损失,提高模型的标签容错率。实验结果表明,所提算法的均值平均精度为 88.31%,网络模型参数个数为 8.10×10^6 ,相比 EfficientDet-D2 算法,均值平均精度提高了 3.29%,网络模型参数个数没有增加,相比 YOLOv4, YOLOv3, SSD, Faster R-CNN 和 Fast R-CNN 算法,均值平均精度分别提升了 5.2%, 10.71%, 14.01%, 15.11% 和 18.30%,网络模型参数个数分别减少了 55.94×10^6 , 52.91×10^6 , 16.09×10^6 , 55.18×10^6 和 53.11×10^6 。所提目标检测模型,提高了检测准确度和 F1 得分;检测每张测试图片仅需 0.73 s,满足实时性要求。

关键词: 目标检测; EfficientDet; IOU; 标签平滑

中图法分类号 TP391.4

Target Detection Algorithm Based on Compound Scaling Deep Iterative CNN by Regression Converging and Scaling Mixture

WANG Guogang, WU Yan and LIU Yibo

College of Physics and Electronic Engineering, Taiyuan 030006, China

Abstract A novel algorithm named as target detection algorithm based on compound scaling deep iterative CNN by regression converging and scaling mixture is proposed to avoid the disadvantages of low robustness, label marginalization and poor convergence performance of the regression loss function in the EfficientDet algorithm. After utilizing the 2×2 scaling mixture regularization strategy to enhance the training samples, the proposed method avoids the over fitting and improves the generalization ability of the model. The convergence speed, the positioning accuracy and the CNN regression accuracy are improved, since the aspect ratio and the center distance are taken into account in the penalty items of the CIOU loss function that can predict the bounding frame coordinate and suppress the redundant boxes. The proposed method improves the label fault tolerance rate because the cross entropy loss with label smoothing for class is established after generating the label smoothing regularization distribution, which is a weighted sum of the marginal label distribution and the uniform distribution by setting the smoothing parameter. Experiments are performed on the PASCAL VOC 2007 and 2012 datasets, and the results show that while the number of the network model parameters remain unchanged, the mean average precision of the proposed algorithm reaches 88.31%, which is 3.29% higher than that of the original network (EfficientDet-D2, 84.12%). Compared with YOLOv4, YOLOv3, SSD, Faster R-CNN and Fast R-CNN, the mean average precision increases by 5.2%, 10.71%, 14.01%, 15.11% and 18.30%, respectively, and the number of network model parameters is reduced by 55.94×10^6 , 52.91×10^6 , 16.09×10^6 , 55.18×10^6 and 53.11×10^6 , respectively. Not only the algorithm improves the detection accuracy and the F1 score, but also it takes 0.73 s to detect each test image, which meets the real-time requirements during the detecting phase.

Keywords Object detection, EfficientDet, IOU, Label smoothing

1 引言

作为 CV 领域重要的研究热点,目标检测在人脸识别^[1-2]、智能视频监控^[3-4]、无人驾驶系统^[5]以及遥感图像检测^[6-7]等领域有着重要的研究意义。

传统目标检测方法基于人工经验提取样本的数据特征,

结合特征分类模型和机器学习算法判断目标类别^[8]。因为存在重叠、遮挡、光照不均等问题,传统机器学习算法在目标检测任务中难以取得令人满意的检测效果。

相比传统的机器学习算法,基于深度学习^[9]的目标检测算法往往能获得更好的性能。该类算法大致分为以 YOLO^[10-12], SSD^[13-14]为代表的基于回归分析的目标检测算法和

基金项目:山西省自然科学基金(201901D111031)

This work was supported by the Natural Science Foundation of Shanxi Province, China(201901D111031).

通信作者:王国刚(kingguogang@sxu.edu.cn)

以 R-CNN^[15], Fast R-CNN, Faster R-CNN^[16] 为代表的基于候选区域的目标检测算法两种。尽管基于深度学习的方法取得了一定成功,但其检测性能的提升是以网络深度加深、计算量大、运行速度慢为代价的。

EfficientDet^[17] 算法通过优化比例系数,实现了各模块之间按比例联合缩放,不仅取得了更好的检测性能,而且提高了运算速度。但该算法的特征提取网络只关注了样本本身,没有进行数据增强^[18],易导致特征提取不充分,检测鲁棒性低;EfficientDet 模型把 IOU 作为目标定位损失函数^[19],收敛速度慢,无法准确反映预测框和真实框间的重叠情况,定位精度不高;另外,该模型的损失函数^[20]仅考虑了正确分类标签损失,未关注错误分类标签损失,从而使模型的标签容错率低,泛化能力差。

针对以上问题,本文提出了回归收敛缩放混合的深度迭代复合缩放 CNN 目标检测算法(SCS 算法)。该算法利用 2×2 缩放混合正则化方法,增强训练样本,避免训练过拟合,提高模型的泛化能力;利用完全交并比(Complete Intersection Over Union, CIOU)损失,将纵横比和中心点距离作为边界框坐标预测损失函数的惩罚项,使卷积神经网络回归更准确;设置平滑参数,生成标签平滑正则化分布,建立类标签平滑交叉熵损失,提高模型的标签容错率。实验结果表明,所提目标检测算法提高了检测准确度和 F1 得分;检测每张测试图片仅需 0.73s,满足实时性要求。

2 SCS 算法的网络架构

2.1 基准模型

EfficientDet 的特征提取网络、特征融合网络和预测网络由复合系数 Φ 一致缩放获得,共有 8 种网络模型。SCS 算法以 EfficientDet-D2 为基准模型,其网络配置如表 1 所列。

表 1 SCS 算法基准模型的网络配置

Table 1 Network configuration of SCS algorithm baseline model

网络模型	输入尺寸	特征提取网络	特征融合网络		预测网络
			通道	深度	深度
SCS 算法基准模型	512×512	EfficientNet-B2	112	5	3

2.1.1 特征提取网络

SCS 算法将 EfficientNet-B2^[21] 作为特征提取网络,网络结构如表 2 所列。

表 2 EfficientNet-B2 的网络结构

Table 2 Network structure of EfficientNet-B2

阶段	操作算子	分辨率(H×W)	通道	层
1	Conv3×3	512×512	32	1
2	MBConv1, k3×3	256×256	16	2
3	MBConv6, k3×3	256×256	24	3
4	MBConv6, k5×5	128×128	48	3
5	MBConv6, k3×3	64×64	88	4
6	MBConv6, k5×5	32×32	120	4
7	MBConv6, k5×5	16×16	208	5
8	MBConv6, k3×3	8×8	352	2
9	Conv1×1&Pooling&FC	4×4	1408	1

2.1.2 特征融合与预测网络

SCS 算法特征融合网络由 5 个加权双向特征金字塔网络(Efficient Bidirectional Cross-scale Connections and Weighted Feature Fusion, BiFPN)^[17] 组成。BiFPN 模型形成了包含上采样路径和下采样路径的特征融合过程;采用跳跃连接,去掉只有一个输入的节点,使网络轻量化;在特征融合过程中添加可自主学习特征信息的权重。在减少参数计算量的同时,提升了目标检测效果。

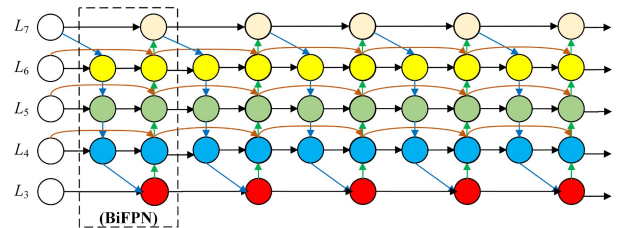


图 1 特征融合网络结构

Fig. 1 Structure of feature fusion network

以加权双向特征金字塔网络结构中的第 4 层为例,其特征信息的传递如式(1)、式(2)所示:

$$L_4^{id} = Conv \left(\frac{w_1 * L_4^{in} + w_2 * Resize(L_3^{id})}{w_1 + w_2 + \epsilon} \right) \quad (1)$$

$$L_4^{out} = Conv \left(\frac{w_1' * L_4^{in} + w_2' * L_4^{id} + w_3' * Resize(L_3^{out})}{w_1' + w_2' + w_3' + \epsilon} \right) \quad (2)$$

其中, L_4^{id} 代表第四层隐藏层, L_4^{out} 代表第四层输出, L_4^{in} 代表第四层输入。

预测网络由 Class prediction net 和 Box prediction net 组成。将特征融合网络输出的 5 个有效特征层传输到预测网络,获得预测结果。

2.2 整体框架

SCS 算法的整体框架如图 2 所示。

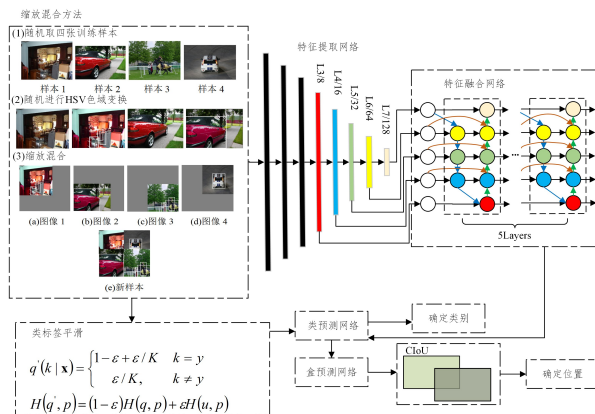


图 2 SCS 整体框架示意图

Fig. 2 Diagram of SCS network model

SCS 算法在目标检测过程中,首先对不同分辨率、不同尺寸的样本进行缩放混合,增强训练样本,将训练样本输入特征提取网络和特征融合网络,生成预测标签;将训练样本的真实标签利用标签平滑正则化策略,生成平滑后的真实标签,将平滑后的真实标签和预测标签输入预测网络,经过 Softmax 激活层确定目标类别,经过 CIOU 函数层衡量真实框与预测框的重合程度,确定目标位置。

3 回归收敛缩放混合的深度迭代复合缩放 CNN 目标检测算法

3.1 2×2 缩放混合正则化方法

为提升算法的鲁棒性,提出了 2×2 缩放混合正则化方法。随机取 4 张训练图像,随机进行 HSV 色域变换和尺度缩放,对缩放后的图像进行随机混合,获得新的训练样本。

设原图像、缩放后的图像分别为 $X_{M \times N}$ 和 $Y_{I \times J}$,则有 $Y_{I \times J} = g(X_{M \times N})$,其中, $I = \lceil wM \rceil$, $J = \lceil wN \rceil$, g 表示将 $X_{M \times N}$ 的宽和高分别缩放 w 倍。 w 是服从均匀分布的随机变量,其概率密度如式(3)所示:

$$f(x) = \begin{cases} 10, & 0.5 < x < 0.6 \\ 0, & \text{其他} \end{cases} \quad (3)$$

简记为 $w \sim U(0.5, 0.6)$ 。

设 $Y_{I \times J} = g(\mathbf{X}_{M \times N}) = (y_{ij})_{I \times J}$,其中 $\mathbf{X}_{M \times N} = (x_{m',n'})_{M \times N}$, y_{ij} 可由式(4)~式(7)确定。

$$y_{ij} = \begin{pmatrix} x_{\lfloor m \rfloor, \lfloor n \rfloor} & x_{\lfloor m \rfloor, \lfloor n \rfloor + 1} \\ x_{\lfloor m \rfloor + 1, \lfloor n \rfloor} & x_{\lfloor m \rfloor + 1, \lfloor n \rfloor + 1} \end{pmatrix} \otimes \begin{pmatrix} p_{11}^{ij} & p_{12}^{ij} \\ p_{21}^{ij} & p_{22}^{ij} \end{pmatrix} \quad (4)$$

$$\begin{pmatrix} p_{11}^{ij} & p_{12}^{ij} \\ p_{21}^{ij} & p_{22}^{ij} \end{pmatrix} = \begin{pmatrix} s_{ij} t_{ij} & (1-s_{ij}) t_{ij} \\ s_{ij} (1-t_{ij}) & (1-s_{ij})(1-t_{ij}) \end{pmatrix} \quad (5)$$

$$s_{ij} = \lfloor m \rfloor - m \quad (6)$$

$$t_{ij} = \lfloor n \rfloor - n \quad (7)$$

其中, $m = i/w$, $n = j/w$, \otimes 表示矩阵卷积。

随机取 4 张 $M \times N$ 训练图像,设为 $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$,将它们宽和高分别缩放 w 倍得到的新图像记为 $\mathbf{A}_{IJ}, \mathbf{B}_{IJ}, \mathbf{C}_{IJ}, \mathbf{D}_{IJ}$ 。设 $M \times N$ 混合图像 Z 行或列归一化混合分界点 μ 为随机变量,均匀分布在 $(1-w, w)$ 上,简记为 $\mu \sim U(1-w, w)$ 。将缩放后生成的训练样本混合成一个新样本的过程,混合操作如式(8)所示:

$$\begin{aligned} \mathbf{Z} &= \begin{pmatrix} \mathbf{A}_{IJ} & 0 \\ 0 & 0 \end{pmatrix} \odot \begin{pmatrix} \mathbf{I}_{\mu M, \mu N} & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & \mathbf{B}_{IJ} \\ 0 & 0 \end{pmatrix} \odot \begin{pmatrix} 0 & \mathbf{I}_{\mu M, (1-\mu)N} \\ 0 & 0 \end{pmatrix} + \\ &\begin{pmatrix} 0 & 0 \\ \mathbf{C}_{IJ} & 0 \end{pmatrix} \odot \begin{pmatrix} 0 & 0 \\ \mathbf{I}_{(1-\mu)M, \mu N} & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{D}_{IJ} \end{pmatrix} \odot \\ &\begin{pmatrix} 0 & 0 \\ 0 & \mathbf{I}_{(1-\mu)M, (1-\mu)N} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{A}_{\mu M, \mu N} & \mathbf{B}_{\mu M, (1-\mu)N} \\ \mathbf{C}_{(1-\mu)M, \mu N} & \mathbf{D}_{(1-\mu)M, (1-\mu)N} \end{pmatrix} \end{aligned} \quad (8)$$

其中, \mathbf{I} 是全 1 矩阵, \odot 表示矩阵的点乘运算。

设训练样本 $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ 的标签分别为 l_A, l_B, l_C, l_D ,则缩放混合得到的新样本标签 \tilde{l} 如式(9)所示:

$$\tilde{l} = \mu^2 l_A + \mu(1-\mu)l_B + (1-\mu)\mu l_C + (1-\mu)^2 l_D \quad (9)$$

其中, $\mu \sim U(1-w, w)$ 。

3.2 类标签平滑

针对标签边缘化造成的过拟合问题,SCS 算法采用了标签平滑正则化 (Label Smoothing Regularization, LSR) 机制,通过估计标签缺失的边缘效应来规范分类器层,平滑数据集的真实标签,提高容错率,增强模型的泛化能力。

3.2.1 标签平滑正则化

设 \mathbf{x}, \mathbf{y} 分别为样本和真实标签, $q(k|\mathbf{x})$ 为标签边缘化概率分布。 $q(k|\mathbf{x})$ 可表示为式(10)。

$$q(k|\mathbf{x}) = \delta_{k,y} = \begin{cases} 1, & k=y \\ 0, & k \neq y \end{cases} \quad (10)$$

设 $u(k)$ 为标签先验概率分布, $q'(k|\mathbf{x})$ 为标签平滑正则化概率分布。 $q'(k|\mathbf{x})$ 可表示为式(11)。

$$q'(k|\mathbf{x}) = (1-\varepsilon)q(k|\mathbf{x}) + \varepsilon u(k) \quad (11)$$

其中, ε 为平滑参数。为简便,设 $q(k|\mathbf{x}) = q(k)$, $q'(k|\mathbf{x}) = q'(k)$ 。 $u(k)$ 一般取均匀分布,则式(11)可简化为式(12)。

$$q'(k) = (1-\varepsilon)q(k) + \frac{\varepsilon}{K} = \begin{cases} 1-\varepsilon+\varepsilon/K, & k=y \\ \varepsilon/K, & k \neq y \end{cases} \quad (12)$$

3.2.2 LSR 交叉熵损失函数

设 \mathbf{x} 和 $p(k|\mathbf{x})$ 分别为样本和分类概率,则 $p(k|\mathbf{x})$ 可表示为式(13)。

$$p(k|\mathbf{x}) = \frac{e^{\ln p(k|\mathbf{x})}}{\sum_{k=1}^K e^{\ln p(k|\mathbf{x})}} \quad (13)$$

令 $z_k = \ln p(k|\mathbf{x})$,则式(13)可简化为:

$$p(k|\mathbf{x}) = \frac{\exp(z_k)}{\sum_{k=1}^K \exp(z_k)} \quad (14)$$

其中, z_k 表示网络模型 softmax 层第 k 个神经元的输入。

目标检测模型中,分类概率和真实标签概率的差异一般采用交叉熵损失函数来度量。

设 $H(q, p)$ 为标签边缘化交叉熵损失函数,参见式(15)。

$$H(q, p) = - \sum_{k=1}^K \log p(k|\mathbf{x}) \cdot q(k|\mathbf{x}) \quad (15)$$

为简单,设 $p(k|\mathbf{x}) = p(k)$, $q(k|\mathbf{x}) = q(k)$ 。综合式(14)、式(15),可得式(16):

$$H(q, p) = - \sum_{k=1}^K q(k) z_k + \log \left(\sum_{k=1}^K \exp(z_k) \right) \quad (16)$$

$H(q, p)$ 对 z_k 可导,导数如式(17)所示:

$$\frac{\partial H(q, p)}{\partial z_k} = p(k) - q(k) \quad (17)$$

其中, $\frac{\partial H(q, p)}{\partial z_k} \in [-1, 1]$ 。

设 $H(q', p)$ 为标签平滑正则化交叉熵损失函数,参见式(18)。

$$\begin{aligned} H(q', p) &= - \sum_{k=1}^K \log p(k) \cdot q'(k) \\ &= (1-\varepsilon)H(q, p) + \varepsilon H(u, p) \end{aligned} \quad (18)$$

其中, $H(u, p)$ 表示分类分布与均匀分布的交叉熵。

将式(14)代入 $H(q', p)$,可得式(19)。

$$\begin{aligned} H(q', p) &= (1-\varepsilon)H(q, p) - \frac{\varepsilon}{K} \sum_{k=1}^K z_k + \\ &\frac{\varepsilon}{K} \sum_{k=1}^K \log \left\{ \sum_{k=1}^K \exp(z_k) \right\} \end{aligned} \quad (19)$$

$H(q', p)$ 对 z_k 可导, 导数如式(20)所示:

$$\begin{aligned} \frac{\partial H(q', p)}{\partial z_k} &= (1-\varepsilon) \frac{\partial H(q, p)}{\partial z_k} + \varepsilon \frac{\partial H(u, p)}{\partial z_k} \\ &= (1-\varepsilon) \{p(k) - q(k)\} + \varepsilon \left\{p(k) - \frac{1}{K}\right\} \quad (20) \end{aligned}$$

标签平滑正则化交叉熵损失函数将均匀分布与预测分布的交叉熵考虑在内, 优化了模型更新的重估公式, 增加了真实标签的容错率, 提高了算法性能。

3.3 完全交并比

为提高目标检测损失函数的准确度, 使训练的收敛速度加快, SCS算法把完全交并比作为目标定位的损失函数。作为交并比损失函数的一种, 完全交并比通过损失函数的收敛性表征神经网络预测结果的准确性。

交并比损失函数主要包括雅卡尔系数、广义交并比、距离交并比和完全交并比。雅卡尔系数也称交并比, 与两个框(预测框与真实框)的重叠面积有关, 不随边界框空间尺度的变化而变化, 但两框外离时, 雅卡尔系数为 0, 不能准确辨别两框之间不同的对齐方式。广义交并比将两框并集相对于最小外接矩形的补集考虑在内, 有效解决了两框外离时雅卡尔系数为 0 的问题, 但若两框内含时, 广义交并比退化为交并比, 不能有效区分两框的相对位置, 且收敛速度更慢。

距离交并比和完全交并比, 把两框中心点距离与最小外接矩形对角线距离之比的平方作为惩罚项, 解决了广义交并比两框内含问题, 收敛速度有所提高, 同时完全交并比又把两框纵横比考虑在内, 使损失函数获得更好的收敛性能。因此 SCS 算法的边界框回归预测网络采用完全交并比损失函数。

完全交并比损失计算如式(21)~式(24)所示。

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(B, B^{gt})}{c^2} + \alpha\beta \quad (21)$$

$$\beta = \frac{4}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2 \quad (22)$$

$$\alpha = \frac{\beta}{(1 - IoU) + \beta} \quad (23)$$

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \quad (24)$$

其中, c 和 $\rho(B, B^{gt})$ 分别表示两框最小外接矩形对角线距离和两框中心点距离; w 和 h 分别表示预测框的宽和高; w^{gt} 和 h^{gt} 分别表示真实框的宽和高; β 是两框纵横比反正切差值平方的归一化, 衡量了纵横比意义下两框的相似程度。

4 实验结果与分析

4.1 实验细节

实验采用的处理器为 Intel(R) Core(TM) i7-10750H CPU@2.60 GHz 2.59 GHz, GPU 为 NVIDIA GeForce RTX 2060, 深度学习框架为 Tensorflow 1.13.1 和 Keras 2.1.5。

预处理阶段, 对训练图像实施了随机平移、翻转等数据增强策略, 然后对样本进行缩放混合, 并调整样本尺寸为 512×512 , 将其输入到网络进行训练。

训练阶段, 采用 SGD 优化器更新网络参数, 初始学习率设置为 5×10^{-5} , 学习率按照余弦退火规则进行更新, 矩 Momentum 和衰减权重 weight decay 分别设置为 0.9 和 $4 \times$

10^{-5} 。每个 batch 随机选取 8 个训练样本, 在一块 RTX 2060 显卡上训练 25 个 epoch 后停止。

LSR 平滑参数设置为 0.01。当 CIOU 和类别置信度均超过 0.5 时, 判定检测结果为正确检测。

4.2 数据集与评价指标

实验采用的训练数据集为 PASCAL VOC 2007 和 2012 的训练、验证集, 共有 16551 个训练样本、40058 个目标物体; 算法评估采用的测试数据集为 PASCAL VOC 2007 测试集, 共有 4952 个测试样本、12032 个目标物体。实验采用的训练集及测试集的划分情况如表 3 所列。

表 3 实验数据集的划分情况

Table 3 Statistics of experimental data sets

类别	训练集		测试集	
	样本个数	目标个数	样本个数	目标个数
Aero	908	1171	204	285
Bicycle	795	1064	239	337
Bird	1095	1605	282	459
Boat	689	1140	172	263
Bottle	950	1764	212	469
Bus	607	822	174	213
Car	1874	3267	721	1201
Cat	1417	1593	322	358
Chair	1564	3152	417	756
Cow	444	847	127	244
D-table	738	824	190	206
Dog	1707	2025	418	489
Horse	769	1072	274	348
M-bike	771	1052	222	325
Person	6095	13256	2007	4528
P-plant	772	1487	224	480
Sheep	421	1070	97	242
Sofa	736	814	223	239
Train	805	925	259	282
Tv	831	1108	229	308
Total	16551	40058	4952	12032

实验采用平均精度(Average Precision, AP)、均值平均精度(Mean Average Precision, mAP)、F1 得分、查准率(Precision, P)、查全率(Recall, R)和参数量作为评价指标, 以衡量模型的目标分类能力、目标检测能力和算法的复杂度。此外, 为直观地比较模型的检测性能, 通过设置不同的置信度阈值, 绘制模型的 P-R 曲线和 F1 曲线。

4.3 定量分析

表 4 列出了 SCS 算法与其他 7 种算法的精度对比。由表 4 可知, SCS 算法 14 类目标的 AP 均高于对比算法, 且 SCS 算法的均值平均精度达到了 88.31%, 分别比 Efficient-Det-D2, EfficientDet-D0, YOLO v4, YOLO v3, SSD, Faster R-CNN, Fast R-CNN 算法的 mAP 提升了 3.29%, 4.19%, 5.20%, 10.71%, 14.01%, 15.11%, 18.30%。为综合比较算法性能, 实验采用 P-R 曲线和 F1 曲线进行对比。由表 3 的训练集可以看出, 包含类别 bus 的训练样本有 607 个, 占全部训练样本的 3.67%; 训练样本中, 类别 bus 的目标个数有 822 个, 占训练集总目标个数的 2.05%。由于 bus 是最能体现样本不均衡问题的类别之一, 因此实验选用类别 bus 为代表, 给出了此类别 8 种算法的 P-R 曲线和 F1 曲线, 如图 3、图 4 所示。

表 4 8 种算法的精度对比
Table 4 Accuracy comparison of eight algorithms

算法	mAP/%	各类别的平均精度(AP/%)							
SSD	74.30	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		75.50	80.21	72.29	66.32	47.57	83.01	84.20	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		86.12	54.70	78.31	<u>73.90</u>	84.49	85.31	82.60	
		Person	P-plant	Sheep	Sofa	Train	Tv		
76.18	48.60	73.89	76.01	83.40	73.39				
Fast R-CNN	70.01	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		77.01	78.10	69.31	59.41	38.32	81.60	78.60	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		86.71	42.61	78.80	68.91	84.72	82.01	76.61	
		Person	P-plant	Sheep	Sofa	Train	Tv		
69.92	31.81	70.12	74.80	80.41	70.42				
Faster R-CNN	73.20	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		73.31	83.90	73.30	60.59	53.20	83.21	83.40	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		85.31	55.50	78.70	70.49	83.70	83.10	78.31	
		Person	P-plant	Sheep	Sofa	Train	Tv		
79.59	42.40	71.57	70.70	84.43	69.29				
YOLO v3	77.60	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		81.60	82.39	74.40	68.51	51.02	87.79	83.90	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		90.11	59.60	83.18	74.10	87.90	85.91	84.20	
		Person	P-plant	Sheep	Sofa	Train	Tv		
86.29	50.20	80.21	73.60	87.90	79.19				
YOLO v4	83.11	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		91.05	89.93	<u>87.10</u>	78.15	75.69	90.89	<u>93.01</u>	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		91.12	73.21	<u>85.03</u>	71.03	84.11	88.56	92.07	
		Person	P-plant	Sheep	Sofa	Train	Tv		
90.01	54.21	83.97	69.12	86.95	87.02				
Efficient Det-D0	84.12	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		91.52	95.29	84.92	<u>82.01</u>	54.73	98.60	91.97	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		93.61	66.71	76.33	71.56	89.78	<u>93.27</u>	93.13	
		Person	P-plant	Sheep	Sofa	Train	Tv		
89.03	74.79	81.75	69.54	91.38	<u>92.55</u>				
Efficient Det-D2	85.02	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		92.42	95.89	85.88	82.81	55.65	98.89	92.78	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		<u>94.82</u>	68.91	77.48	72.78	<u>90.76</u>	94.25	<u>93.85</u>	
		Person	P-plant	Sheep	Sofa	Train	Tv		
89.99	75.63	82.57	69.56	<u>92.13</u>	93.35				
SCS	88.31	Aero	Bicycle	Bird	Boat	Bottle	Bus	Car	
		94.02	98.54	92.13	80.13	<u>74.38</u>	100	96.14	
		Cat	Chair	Cow	D-table	Dog	Horse	M-bike	
		100	<u>72.70</u>	94.05	57.49	94.68	92.86	100	
		Person	P-plant	Sheep	Sofa	Train	Tv		
90.58	77.03	91.65	82.81	93.83	83.18				

注:参数个数越小越好,加粗字体为最优结果,加下划线字体为次优结果

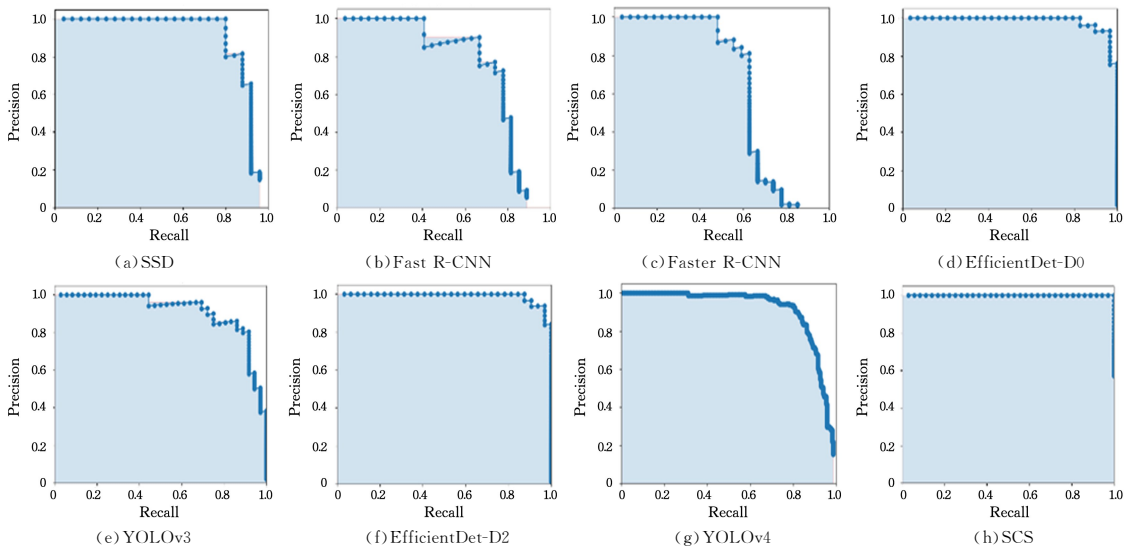


图 3 bus 类别的 P-R 曲线比较
Fig. 3 P-R curve comparison of bus category

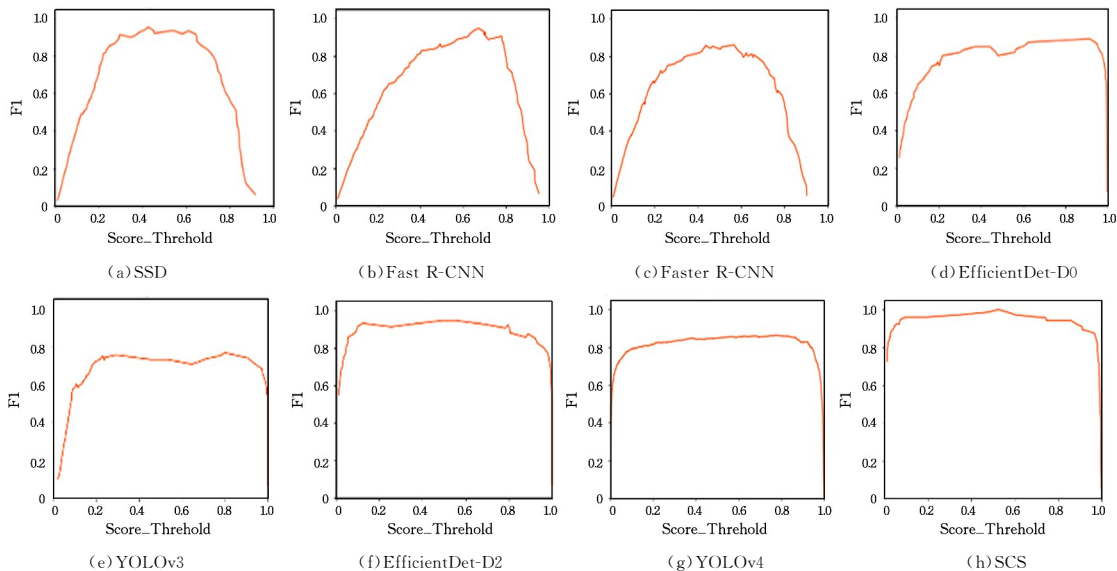


图4 bus类别的F1曲线比较

Fig. 4 Comparison of F1 curves of bus category

每个类别的检测可分解为两个任务,首先确定是否有此类别,则若有此类别,再确定有几个包含此类别的目标。设测试集中包含某个类别的测试样本和不包含此类别的测试样本分别为 x 个和 $4952 - x$ 个,那么在测试集中确定是否有此类别相当于在测试集信息源中随机抽取一个测试样本,猜测其是否包含此类别。类别信源熵为 $H_{\text{class}}\left(\frac{x}{4952}, 1 - \frac{x}{4952}\right)$, 满足式(25)。

$$H_{\text{sheep}}\left(\frac{97}{4952}, 1 - \frac{97}{4952}\right) \leq H_{\text{class}}\left(\frac{x}{4952}, 1 - \frac{x}{4952}\right) \leq H_{\text{person}}\left(\frac{2007}{4952}, 1 - \frac{2007}{4952}\right) \quad (25)$$

由表4的测试集还可以看出,包含类别 person 的测试样本有2007个,类别 person 的目标有4528个,平均每个包含类别 person 的测试样本中,有2.26个人物目标个数。综合以上两点可知, person 是测试集中最难检测的类别之一,故实验选用类别 person 为代表,给出此类别8种算法的P-R

曲线和F1曲线,如图5、图6所示。PR曲线和横轴间的面积越大,算法性能越好。由图3和图5可以看出,SCS算法的PR曲线和横轴间的面积最大,SCS算法的性能优于7种对比算法。样本不均衡情况下,PR曲线越平稳,检测性能越好。由图3和图5可以看出,SCS算法的PR曲线最平稳,检测性能最好。F1曲线越平滑,越趋近于1,算法性能越好。由图4和图6可知,相比7种对比算法,SCS算法的F1曲线最平滑,最接近于1,目标检测性能最好。

表5列出了8种算法在客观性评价指标上的结果。由表5可以看出,SCS算法检测结果的5项指标中有3个最优。EfficientDet-D0的参数量为 3.90×10^6 , 小于SCS的参数量,但SCS算法的其余4项指标均优于EfficientDet-D0算法。EfficientDet-D2算法的 precision 为93.08%, 但 recall 仅为70.70%, 这说明EfficientDet-D2算法在测试集上的错检率比SCS算法低0.95%, 但EfficientDet-D2算法的漏检率比SCS算法高19.2%。

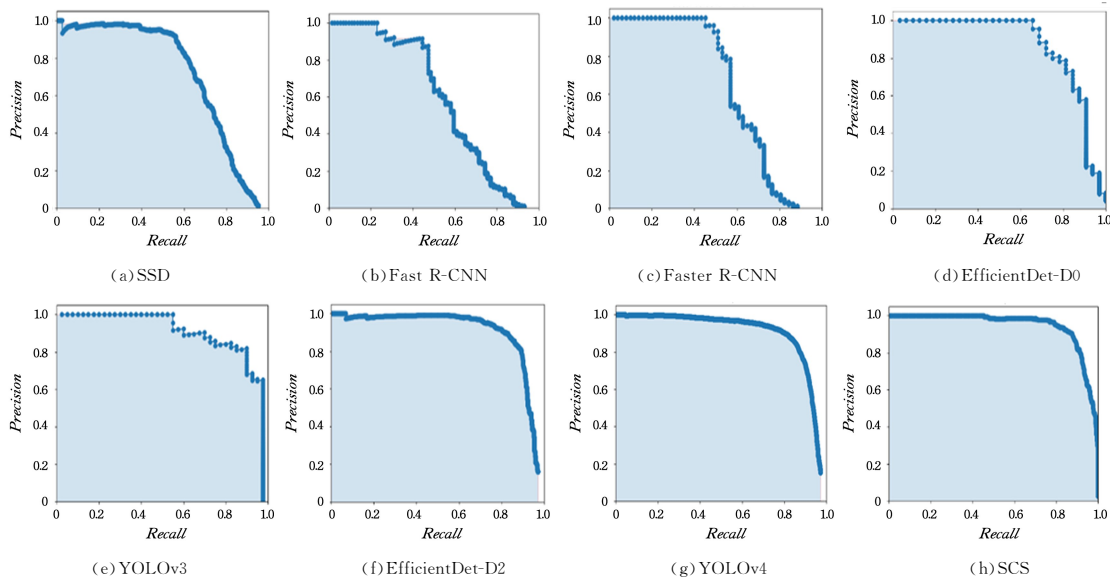


图5 person类别的P-R曲线比较

Fig. 5 P-R curve comparison of person category

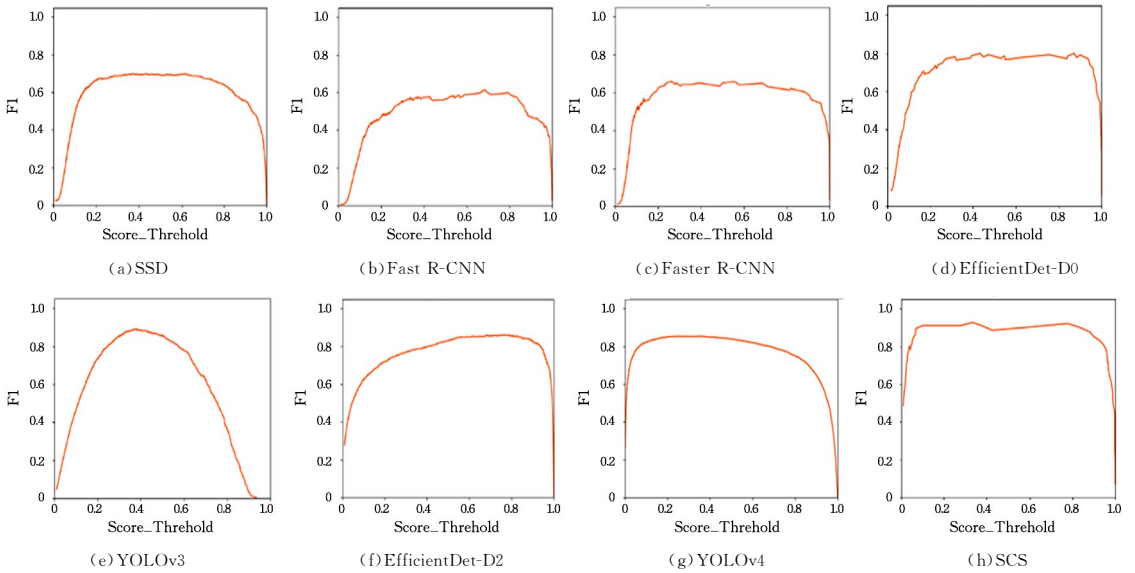


图6 person类别的F1曲线比较

Fig. 6 Comparison of F1 curves of person category

表5 8种算法的客观评价指标对比

Table 5 Comparison of objective evaluation indices of eight algorithms

算法	mAP/%	参数个数	F1	P/%	R/%
SCS	88.31	8.10×10^6	0.91	<u>92.13</u>	89.90
EfficientDet-D0	84.12	3.90×10^6	0.78	87.93	70.36
EfficientDet-D2	<u>85.02</u>	8.10×10^6	<u>0.80</u>	93.08	<u>70.70</u>
YOLOv4	83.11	64.04×10^6	0.79	89.74	70.56
SSD	74.30	24.19×10^6	0.63	86.52	50.14
FastR-CNN	70.01	61.21×10^6	0.62	88.86	47.61
FasterR-CNN	73.20	63.28×10^6	0.68	81.37	58.40
YOLO v3	77.60	61.01×10^6	0.80	88.58	72.94

在运行时间方面,虽然 SCS 算法的训练时间达到了 25 h,耗时较长,但训练过程为线下操作,并不影响测试过程。测试阶段,SCS 算法检测完 4952 张图片,耗时 1 h。检测每张测试图片仅需 0.73 s 满足实时性要求。

4.4 定性分析

为评估算法性能,把 7 种算法在 PASCAL VOC 2007 测试集上的检测结果进行对比。图 7—图 11 是对 5 张待检图像采用不同目标检测模型得到的实验结果。

待检图像 1 为测试集中 004307 编号的图片,包含 2 个 horse 标签,场景中背景阴暗,存在遮挡,实验结果如图 7 所示。由图 7 可看出,SSD 算法将 horse 类别错测为 sheep 类别;Fast-RCNN, Faster-RCNN, YOLOv3, EfficientDet-D0 和 EfficientDet-D2 算法对目标定位精度不高;只有 SCS 算法没有出现错检、定位精度不高的问题,检测效果最好。

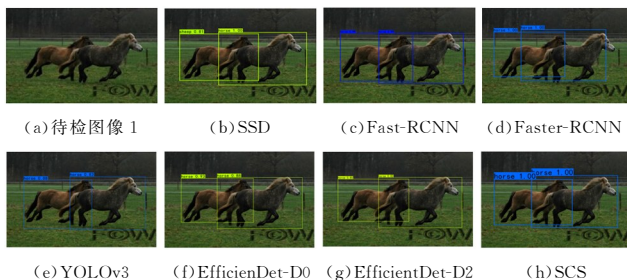


图7 (a)待检图像1的检测结果

Fig. 7 Detected results of image 1

待检图像 2 为测试集中 004304 编号的图片,包含 1 个

car 和 1 个 person 标签,场景中光照不均,背景复杂,实例目标尺度变化较大,实验结果如图 8 所示。由图 8 可知,Faster-RCNN 算法对人物目标有漏检现象;SSD, EfficientDet-D0、EfficientDet-D2 算法把 car 目标错检成 truck 目标;Fast-RCNN 算法出现了重复检测问题;只有 YOLOv3 和 SCS 算法没有出现漏检、错检和重复检测的问题。

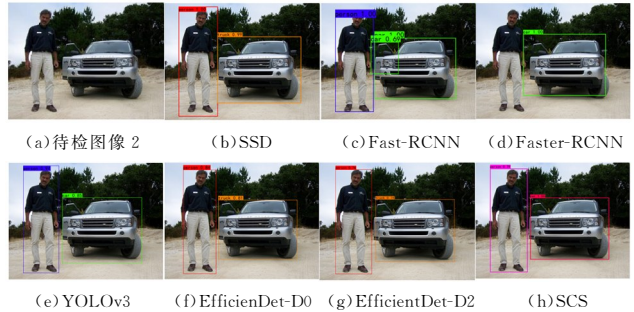


图8 (a)待检图像2的检测结果

Fig. 8 Detected results of image 2

待检图像 3 为测试集中 000282 编号的图片,包含 4 个 person 和 1 个 chair 标签,场景中目标重叠,存在遮挡,实验结果如图 9 所示。由图 9 可知,SSD 算法漏检了椅子,Fast-RCNN, EfficientDet-D0 和 EfficientDet-D2 算法漏检了人;Faster-RCNN 算法有定位精度不高的问题;YOLOv3 算法检测出了 sofa 类别,属于错检;只有 SCS 算法没有出现漏检、错检和定位精度不高的问题,检测效果最好。

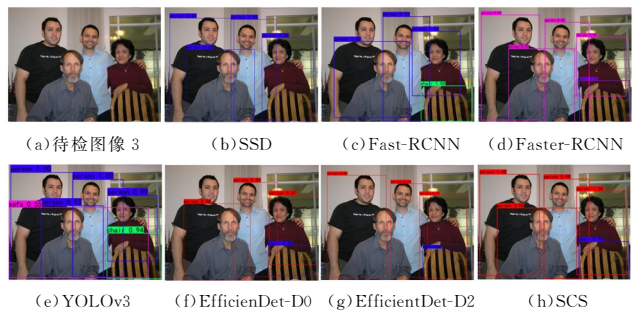


图9 (a)待检图像3的检测结果

Fig. 9 Detected results of image 3

待检图像 4 为测试集中 001028 编号的图片, 包含 5 个人物 和 1 个 chair 标签, 场景中实例尺度变化较大, 背景复杂, 存在遮挡, 实验结果如图 10 所示。由图 10 可以看出, SSD, Fast-RCNN, Faster-RCNN 算法漏检了小女孩和椅子, 且 Fast-RCNN, Faster-RCNN 算法对较小的人物目标定位精度不高; YOLOv3 和 EfficientDet-D0 算法漏检了椅子; EfficientDet-D2 算法漏检了小女孩, 且把 chair 错检成了 dining-table; 只有 SCS 算法没有出现漏检、错检、定位精度不高的问题, 检测效果最好。



图 10 (a)待检图像 4 的检测结果
Fig. 10 Detected results of Image 4

待检图像 5 为测试集中 001821 编号的图片, 包含 2 个 car 标签, 场景中背景阴暗, 光照不足, 实例尺度变化较大, 存在小目标, 实验结果如图 11 所示。由图 11 可知, Fast-RCNN, YOLOv3 和 EfficientDet-D0 算法出现了漏检; Faster-RCNN 算法对较大的 car 目标定位精度不高; SSD, EfficientDet-D2 算法把较小的 car 目标错检成 truck, 且 EfficientDet-D2 算法还出现了重复检测问题; 只有 SCS 算法没有出现漏检、错检、重复检测和定位精度不高的问题, 检测效果最好。

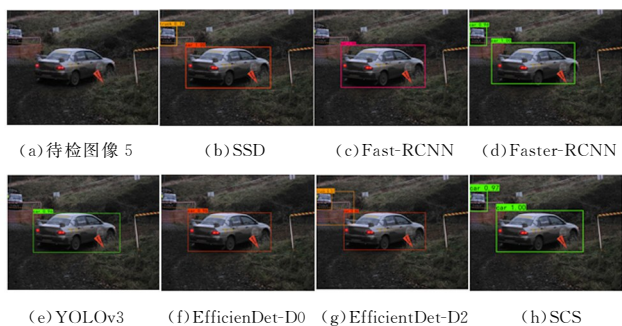


图 11 (a)待检图像 5 的检测结果
Fig. 11 Detected results of Image 5

4.5 消融实验

针对所提 CIUO、类标签平滑和缩放混合正则化方法, 进行了消融实验。以 EfficientDet-D2 网络为对比基准, 分别把 CIUO、类标签平滑和缩放混合正则化方法添加到 EfficientDet-D2 网络, 对比在 PASCAL VOC 2007 测试集上的均值平均精度, 如表 6 所列。由表 6 可知, EfficientDet-D2 网络在 PASCAL VOC 2007 测试集上的均值平均精度为 85.02%, 融合 CIUO、类标签平滑和缩放混合正则化方法的 SCS 算法的均值平均精度为 88.31%, 提高了 3.29%。

表 6 消融实验

Table 6 Ablation experiment

缩放混合方法	类标签平滑	CIUO	mAP/%
			85.02
✓			87.27
	✓		85.12
		✓	85.13
✓	✓		87.40
✓		✓	87.60
	✓	✓	85.58
✓	✓	✓	88.31

由表 6 中第 5 行和第 8 行可以看出, 缺少 CIUO 时, mAP 下降了 0.91%, 这说明利用 CIUO 损失抑制了冗余预测框, 使网络回归更准确, 提高了定位精度。由表 6 中第 6、8 行可以看出, 缺少类标签平滑时, mAP 下降了 0.71%, 这说明类标签平滑模块避免了训练过拟合, 提高了模型泛化能力。由表 6 中第 7—8 行可以看出, 缺少缩放混合正则化方法时, mAP 下降了 2.73%, 这说明缩放混合数据增强方法, 增强了训练样本, 提升了检测性能。由表 6 中第 1—4 行还可以看出, 把 CIUO、类标签平滑和缩放混合正则化方法分别引入 EfficientDet-D2 网络, 网络模型的 mAP 分别提升了 0.11%, 0.10%, 2.25%, 这说明缩放混合正则化方法对网络性能的改善效果最好。

结束语 本文提出了一种回归收敛缩放混合的深度迭代复合缩放 CNN 目标检测算法。该算法采用 2×2 缩放混合正则化方法来增强训练样本, 避免训练过拟合, 提高模型泛化能力; 利用 CIUO 损失抑制冗余预测框, 将中心点距离和纵横比作为边界框坐标预测的损失函数惩罚项, 使卷积神经网络回归更准确, 提高了收敛速度和定位精度; 设置平滑参数, 生成标签平滑正则化分布, 建立类标签平滑交叉熵损失, 提高了模型的容错率。实验结果表明, 所提目标检测算法提高了检测准确度和 F1 得分; 检测每张测试图片仅需 0.73 s, 满足实时性要求。

参考文献

- [1] ARRINGTON M, ELBICH D, DAI J, et al. Introducing the female Cambridge face memory test-long form (F-CFMT+) [J]. Behavior Research Methods, 2022; 1-14.
- [2] YU J, HAO X, CUI Z, et al. Boosting Fairness for Masked Face Recognition [C] // Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021; 1531-1540.
- [3] SUNG C S, PARK J Y. Design of an intelligent video surveillance system for crime prevention: applying deep learning technology [J]. Multimedia Tools and Applications, 2021, 80 (26): 34297-34309.
- [4] KIM J S, KIM M G, PAN S B. A study on implementation of real-time intelligent video surveillance system based on embedded module [J]. EURASIP Journal on Image and Video Processing, 2021, 2021 (1): 1-22.
- [5] XIAOFENG T. Ecological driving on multiphase trajectories and multiobjective optimization for autonomous electric vehicle platoon [J]. Scientific Reports, 2022, 12 (1): 1-16.
- [6] TIAN X, LIU J, MALLICK M, et al. Simultaneous detection and tracking of moving-target shadows in ViSAR imagery [J]. IEEE Transactions on Geoscience and Remote Sensing, 2020, 59 (2):

- 1182-1199.
- [7] LIU S, WANG S, LIU X, et al. Fuzzy detection aided real-time and robust visual tracking under complex environments[J]. IEEE Transactions on Fuzzy Systems, 2020, 29(1): 90-102.
- [8] FELZENSZWALB P F, GIRSHICK R B, MCALLESTER D, et al. Object detection with discriminatively trained part-based models[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 32(9): 1627-1645.
- [9] FRANCOIS C. Xception: Deep learning with depth wise separable convolutions[C]// CVPR, 2017: 1800-1807.
- [10] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]// IEEE Conference on Computer Vision and Pattern Recognition. New York, USA: IEEE, 2016. 779-788.
- [11] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, USA: IEEE, 2017. 6517-6525.
- [12] REDMON JOSEPH, FARHADI A. Yolov3: An incremental improvement[J]. arXiv: 1804. 02767, 2018.
- [13] FU C Y, LIU W, RANGA A, et al. DSSD: Deconvolutional single shot detector[J]. arXiv: 1701. 06659, 2017.
- [14] SHEN Z Q, LIU Z H, LI J G, et al. DSOD: Learning deeply supervised object detectors from scratch[C]// Proceedings of the IEEE International Conference on Computer Vision. New York, USA: IEEE, 2017: 1919-1927.
- [15] GIRSHICK R, DONAHUE J, DARRELL T. Rich feature hierarchies for accurate object detection and semantic segmentation [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Columbus, USA: IEEE, 2014: 580-587.
- [16] REN S, HE K M, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[C]// International Conference on Neural Information Processing Systems. Cambridge USA: MIT Press, 2015: 91-99.
- [17] MINGXING TAN, RUOMING PANG, QUOC V L E. Efficient-Det: Scalable and efficient object detection[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, 2020: 10778-10787.
- [18] SANGDOO Y, DONGYOON H, SEONG J O, et al. Cut Mix: Regularization strategy to train strong classifiers with localizable features[C]// Proceedings of the IEEE International Conference on Computer Vision. ICCV, 2019: 6023-6032.
- [19] ZHENG Z H, WANG P, LIU W, et al. Distance-IoU Loss: Faster and better learning for bounding box regression[C]// Proceedings of the AAAI Conference on Artificial Intelligence. AAAI, 2020: 12993-13000.
- [20] CHRISTIAN S, VINCENT V, SERGEY I, et al. Rethinking the inception architecture for computer vision[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, 2016: 2818-2826.
- [21] TAN M, LE Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]// International Conference on Machine Learning. PMLR, 2019: 6105-6114.



WANG Guogang, born in 1977, Ph. D., associate professor, is a member of China Computer Federation. His main research interests include the image processing, computer vision, machine learning and artificial intelligence.