

doi: 10.7690/bgzd.2021.02.009

基于 SSD 的小目标特征强化检测算法

李炳臻, 姜文志, 顾佼佼, 刘 克

(海军航空大学岸防兵学院, 山东 烟台 264001)

摘要: 为解决原始单次多框目标检测 (single shot multibox detector, SSD) 目标检测算法中对小目标物体检测能力不足的问题, 提出一种改进的 SSD 目标检测算法。采用 VGG19 作为特征提取网络, 在低层特征图部分引入 Conv3_3 卷积特征图, 对 Conv4_4 进行转置卷积操作, 将转置卷积后得到的 Conv4_3 同 Conv3_3 的特征图进行特征拼接, 实验部分使用 VOC 数据集对模型进行训练与测试。结果表明: 该算法可提高检测能力, 目标检测精度能比原始 SSD 算法提高 3.6%, 小目标检测效果比改进前也有明显提升。

关键词: 深度学习; 目标检测; 卷积神经网络; 单次多框目标检测 (SSD) 模型

中图分类号: TP391.41 **文献标志码:** A

Enhanced Detection Algorithm of Small Target Based on SSD

Li Bingzhen, Jiang Wenzhi, Gu Jiaojiao, Liu Ke

(School of Coast Guard, Navy Aviation University, Yantai 264001, China)

Abstract: In order to solve the problem of insufficient ability of small target detection in the original single shot multibox detector (SSD) target detection algorithm, an improved SSD target detection algorithm is proposed. VGG19 is used as the feature extraction network, Conv3_3 convolution feature graph is introduced into the low-level feature map, the transpose convolution operation is carried out on Conv4_4, and the Conv4_3 obtained by transpose convolution is spliced with the feature map of Conv3_3. In the test, the VOC data set is used to train and test the model. The results show that the algorithm can improve the detection ability, the target detection accuracy can be improved by 3.6% compared with the original SSD algorithm, and the effect of small target detection is also significantly improved.

Keywords: deep learning; target detection; convolutional neural network; single multi frame target detection (SSD) model

0 引言

近年来, 人工智能技术发展越来越成熟, 并广泛应用于各个领域。机器学习是实现人工智能的一个必经途径。深度学习是机器学习领域的一个研究方向, 通过大量的数据, 自动提取特征, 训练特定算法模型, 从而达到输出任务的结果。如今深度学习技术在数据挖掘、机器翻译、自然语言处理、人脸识别和目标识别等方面都应用广泛, 使得人工智能的相关技术取得了飞速进步。

目标检测是一种计算机视觉任务, 用于在图像上检测特定类别的图像, 在生产生活中的各个场景都有广泛应用, 随着计算机技术的发展和视觉技术的广泛应用, 目标检测也发展迅速, 对军事目标检测、生物医疗、工业制造等领域都有应用研究价值。

目前主流的目标检测算法可以分成 One-stage 算法与 Two-stage 算法 2 大类^[1], 最大区别在于是否包含了候选区域目标推荐的过程。

SSD (single shot multiBox detector) 算法是 Wei

Liu 在 ECCV 2016 提出的一种目标检测算法^[2], 能够直接回归目标类别和位置, 属于 One-stage 算法的一种, 是继 Faster RCNN^[3-4]和 YOLO^[5]发展之后的另一个主流目标检测算法。与 Faster RCNN 和 YOLO 相比, 它的识别速度和性能都得到了显著提高; 但是对于小目标的检测效果却并不是很好。小目标分为绝对小目标和相对小目标 2 种, 绝对小目标指尺寸小于 32×32 像素的目标, 相对小目标指相对于整个图像的尺寸大小, 一般小于原图像尺寸的 0.1 就被认为是小目标。小目标检测在诸多应用中有着重要意义, 比如医学方面早期肿瘤检测诊断、工业方面对材料表面小缺陷的检测、卫星图像中的目标检测等; 因此, 笔者提出一种基于 SSD 的小目标强化检测算法。

1 SSD 算法原理

1.1 设计思想

SSD 的设计思想同时借鉴了 Faster RCNN 与 YOLO 检测算法。类似于 YOLO, SSD 也使用基于

收稿日期: 2020-08-30; 修回日期: 2020-09-25

作者简介: 李炳臻(1996—), 男, 山西人, 硕士, 从事人工智能深度学习, 图像识别研究。E-mail: libingzhen123456@163.com。

回归的模式，在网络模型中直接回归出物体的类别和位置^[6]，检测速度快；类似于 Faster RCNN，SSD 同时也使用了基于区域的概念，提取不同尺度的特征图来做检测，大尺度特征图(网络模型结构中较靠前)用于检测小目标，而小尺度特征图(网络模型结构中较靠后)用于检测大目标；并且采用了不同尺度和长宽比的先验框(prior boxes, default boxes, 在 Faster R-CNN 中叫作锚, Anchors)。YOLO 算法中对小物体检测不足以及定位不准的问题在 SSD 算法中都得到了相应改进。

1.2 网络结构

SSD 在网络结构上利用了多尺度特征图用以检测目标。模型的输入图片大小是 300×300 ，或者是 512×512 (增加一个卷积层)，无论哪种输入大小，网络结构都没有本质的差别。该算法的主干网络结构是 VGG16，也可采用 VGG、ResNet、MobileNets 等代替原来的 VGG16，将最后 2 个用于处理图像分类问题的全连接层及输出层改成卷积层，并额外增加 4 个卷积层来构造网络结构。其中 VGG16 中的卷积层 Conv4_3 是用于目标检测的第 1 个特征图，从后面网络中提取出的卷积层 Conv7、Conv8_2、Conv9_2、Conv10_2、Conv11_2 作为检测所用的其他特征图，在 SSD 网络模型中，共提取了 6 个特征图，其大小分别是 38×38 、 19×19 、 10×10 、 5×5 、 3×3 、 1×1 。不同特征图设置的先验框数目不同，先验框的设置包括尺度(大小)和长宽比 2 方面。尺度的设置遵守一个线性递增规则：随着特征图大小降低，先验框尺度线性增加，可用下式表示：

$$S_k = S_{\min} + (S_{\max} - S_{\min}) / (m - 1) (k - 1), \quad k \in [1, m]. \quad (1)$$

式中： m 为特征图个数，由于第 1 层(Conv4_3 层)是单独设置，所以 $m=5$ ； S_k 为在 300×300 输入中的比例，表示先验框大小相对于图片的比例； S_{\min} 和 S_{\max} 分别为第 1 层和最后一层所占比的最小值与最大值，SSD300 中为 $0.2 \sim 0.9$ 。

第 1 个特征图 conv4_3，其先验框的尺度比例默认设置为 $S_{\min}/2=0$ ，此时 $k=1$ ，尺度为 $300 \times 0.1=30$ ；除此之外，其他的特征图先验框尺度经过计算分别为 30, 60, 111, 162, 213, 264。

先验框的长宽比，则按照下式分别计算出先验框的宽度与高度：

$$\omega_k^\alpha = s_k \sqrt{a_r}, \quad h_k^\alpha = s_k / \sqrt{a_r}. \quad (2)$$

式中： S_k 为先验框实际尺度； a_r 为先验框宽高比例，一般选取 $a_r \in \{1, 2, 3, 1/2, 1/3\}$ ，默认情况下，每个特

征图会有一个 $a_r=1$ 且尺度为 S_k 的先验框^[7]。除此之外，还会设置一个 $a_r=1$ 且尺度为 $s'_k = \sqrt{s_k s_{k+1}}$ 的先验框，即 2 个大小不同但长宽比为 1 的正方形先验框；因此，每个特征图一共有 6 个不同尺度的先验框 $\{1, 2, 3, 1/2, 1/3, 1\}$ ，但在实现中，大小为 38×38 、 3×3 、 1×1 的特征图仅使用 4 个先验框，不使用长宽比为 $3, 1/3$ 的先验框。每个单元先验框的中心点分布在各个单元的中心，中心点坐标为 $((i+0.5)/|f_k|, (j+0.5)/|f_k|)$ ， $i, j \in [0, |f_k|]$ ，其中， i, j 分别代表在特征图中的水平和垂直的第几格， $|f_k|$ 代表特征图大小。

得到特征图后，对特征图进行卷积操作，得到进一步的检测结果。图 1 所示为 5×5 大小特征图的检测过程，其中 Priorbox 是得到的先验框。检测值包含类别置信度与边界框位置 2 部分，分别采用 3×3 卷积来进行检测。假设 n_k 代表该特征图所采用的先验框数目，类别置信度需要 $n_k \times c$ 个卷积核，而边界框位置需要 $n_k \times 4$ 个卷积核数量。由于每个先验框都会预测一个边界框，所以 SSD300 一共可以预测 $38 \times 38 \times 4 + 19 \times 19 \times 6 + 10 \times 10 \times 6 + 5 \times 5 \times 6 + 3 \times 3 \times 4 + 1 \times 1 \times 4 = 87\ 323$ 个边界框，最后再使用非极大抑制值(NMS)得到边界框的最优解^[8-11]。

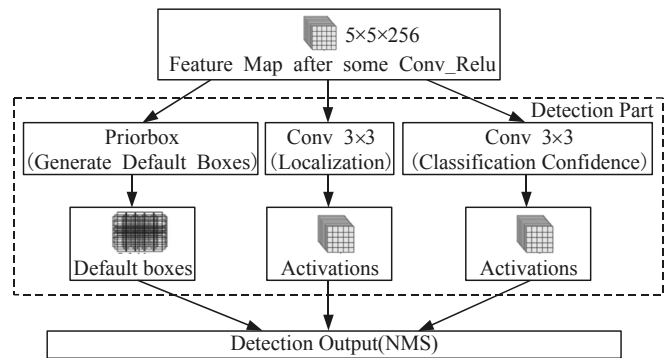


图 1 基于卷积得到检测结果流程

2 改进的 SSD 目标检测算法

从更改基础网络、增强特征区域提取能力 2 方面出发，对原始 SSD 算法作出改进，其中，增强特征区域提取能力具体包含 3 个方面，分别是额外引入一层低层特征层 Conv3_3，在 Conv4_3 卷积层中加入转置卷积的操作增加特征图分辨

率以及将上述 2 层的特征图进行特征融合，增加低层特征图对于目标的特征表达能力。

2.1 SSD 模型的不足之处

SSD 网络模型结构中，低层特征图主要对应细节信息，高层特征图对应抽象的语义信息^[12]。低层

细节特征用于检测小占比目标，高层抽象特征用于检测中等占比目标和大目标。然而在网络结构中，高层卷积层包含 5 层，用于进行小目标检测的低层卷积层仅有一层 Conv4_3，小占比目标物体的特征表达能力显然不够，细节信息不够充足，导致 SSD 模型对小目标检测效果远弱于大目标。

针对 SSD 上述存在的对小目标检测能力不足的问题，提出一种改进 SSD 算法；针对 SSD 模型多尺度的特性对低层特征区域进行强化，增加对小目标的特征表达能力，从而增强检测能力^[13]。若要提高小占比物体的检测能力，网络结构必须具备更强大地提取小目标特征信息的能力，才能更好地学习小目标的分类及检测信息。基于此，从以下 2 点对 SSD 算法进行改进：1) 使用网络层数更深的 VGG19 作为基础网络架构；2) 增强低层特征区域的特征提取能力。

2.2 采用 VGG-19 作为基础网络架构

若要提高模型的检测性能，最简单的办法是增加卷积层数，从而增强特征提取的效果，但是考虑到卷积层数的增加，势必会带来大量数据参数，使得模型的训练及检测时间增加，这与 SSD 设计之初提升速度的思想相违背。笔者没有选择单纯的提升卷积层数的办法，而是选择 VGG19 作为基础网络。网络列表如图 2 所示。

		Input Image (512,512,3)		
		Size	Number	Output
Conv1	Conv1-1	3×3	64	
	Conv1-2	3×3	64	(512,512,64)
	Maxpool			(256,256,64)
Conv2	Conv2-1	3×3	128	
	Conv2-2	3×3	128	(256,256,64)
	Maxpool			(128,128,128)
Conv3	Conv3-1	3×3	256	
	Conv3-2	3×3	256	
	Conv3-3	3×3	256	
	Conv3-4	3×3	256	(128,128,128)
	Maxpool			(64,64,256)
Conv4	Conv4-1	3×3	512	
	Conv4-2	3×3	512	
	Conv4-3	3×3	512	
	Conv4-4	3×3	512	(64,64,256)
	Maxpool			(32,32,512)
Conv5	Conv5-1	3×3	512	
	Conv5-2	3×3	512	
	Conv5-3	3×3	512	
	Conv5-4	3×3	512	(32,32,512)
	Maxpool			(16,16,512)
FC		FC_4096		
		FC_4096		
		FC_4096		
		Softmax		

图 2 VGG-19 网络结构参数列表

相比于 VGG16, VGG19 拥有更深的网络结构，更多的卷积层 (VGG16 拥有 13 个，VGG19 拥有 16 个)；因此，具有更好的特征提取能力，并且随着网络的加深，可在一定程度上提高目标检测的精度。网络模型其余部分与原 SSD 模型相同，将 VGG19 的 2 个全连接层以及输出层转换成普通的卷积层，用以加强特征提取的能力。

2.3 增强低层特征区域特征提取能力

考虑到 SSD 目标检测网络架构采用多尺度特征图进行检测的特点，通过以下方面增强对低层特征区域的特征提取能力：1) 引入更低一层卷积特征图，即 Conv3_3 卷积后的特征图；2) 在原有低层卷积特征层 Conv4_3 中加入转置卷积的操作，提升该层卷积特征图的分辨率；3) 将引入的 Conv3_3 特征图与转置卷积操作过后的 Conv4_3 特征图进行特征拼接，形成具备更丰富特征信息的低层卷积特征图，用于进行后续的目标检测，增加网络对目标物体的特征提取能力。

2.3.1 引入低层特征层

在卷积神经网络中，低层卷积提取到的是目标物体的纹理等细节信息，通过引入低层卷积特征图的方法，提高特征图对目标物体的特征表达能力，从而提高整个网络的检测性能。对应到网络模型结构中，即在原始 SSD 模型的基础上，额外引入更低一层特征层 Conv3_3，增加用于小目标物体检测特征层中所包含的细节信息，其特征层分辨率为 75×75，相比于特征层 Conv4_3 的特征图分辨率更大，细节信息也更为丰富。不同层级特征图对比如图 3 所示。

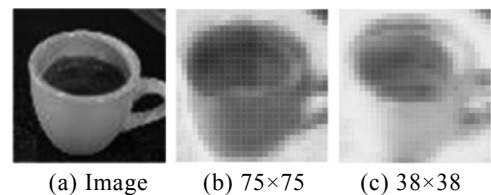


图 3 不同层级特征图对比

图 3(a)为输入图像，图 3(b)为 SSD 网络中 Conv3_3 可视化特征图，图 3(c)为 SSD 网络中 Conv4_3 可视化特征图。从图 3 中可以明显看出：特征层 Conv3_3 包含的细节信息比 Conv4_3 更丰富，特征也更明显；因此，在浅层卷积网络部分引入 Conv3_3 层卷积后得到的特征图，用于提高浅层卷积网络特征图的特征表达能力。

为了保证模型检测的实时性，避免因引入过多

参数导致计算量过大而影响到检测速度，选择直接对 Conv3_3 进行特征层提取，将提取出的特征图与下一步骤的转置卷积一并进行下一步操作。

2.3.2 低层特征图中引入转置卷积

采用转置卷积的方法对特征层 Conv4_3 进行上采样，放大图像的分辨率，将小尺度的特征映射成高维输入。转置卷积示意图如图 4 所示。

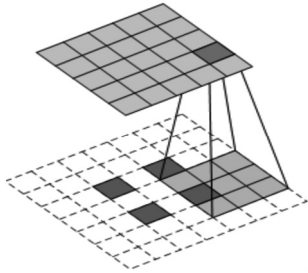


图 4 转置卷积

转置卷积是卷积的反向过程。将特征图展开为 1 维向量，卷积核变为卷积核的转置，转置卷积运算即为矩阵乘法^[14]，计算公式如下：

$$Z^T \times B = A. \quad (3)$$

式中：A 和 B 表示特征图展开得到的一维向量；Z 表示原卷积核展成的稀疏矩阵。

与上采样相同，转置卷积的作用也是放大图像的分辨率，将特征图放回原来的尺寸。不同于上采样中基于插值的方法扩大特征图的分辨率，转置卷积从网络的角度出发，提供了一种最优的“上采样”方法，因为转置卷积是卷积的反向过程，即卷积操作的输入作为转置卷积的输出，卷积操作的输出作为转置卷积的输入；因此，可以在训练过程中经过反向传播学习来调整参数，使上采样参数更加合理，从而获得更高分辨率的特征图，相应的细节信息也就更为丰富，具体如图 5 所示。

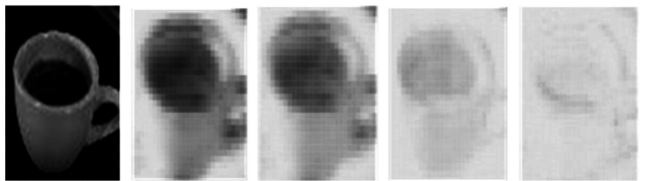


图 5 转置卷积特征

图 5 中给出了经过转置卷积操作后不同尺度的特征图，图 5(a)为输入图像，图 5(b)为 SSD 网络中 Conv4_3 可视化特征图，图 5(c)、(d)、(e)为将 Conv4_3 特征图经过转置卷积操作后得到的不同尺寸的转置卷积特征图，转置卷积操作特征图的选择需要能够获得丰富的细节信息。从图 5 可看出，图

5(c)的细节信息更为丰富，75×75 大小的转置卷积特征图计算公式如下：

$$Q = S \times (M - 1) + P - 2N. \quad (4)$$

式中：Q 为转置卷积输出特征图分辨率；S 为步长；M 为输入特征图分辨率；P 为卷积核尺寸；N 为边缘补充尺寸。设定卷积核尺寸大小 P=3，步长 S=2，边缘扩充 N=1，38×38 特征图经转置卷积操作后分辨率大小计算可得 75×75，与上一步 Conv3_3 特征层提取后的 75×75 特征图大小一致，便于进行后续的特征池化，并且在不同尺寸大小的转置卷积操作特征图中，75×75 特征图细节更为丰富，语义信息更为清晰；因此，选择将 Conv4_3 特征图通过转置卷积增大为 75×75 的特征图。

2.3.3 特征拼接形成新的特征图

Conv4_3 卷积层经过转置卷积操作后，Conv4_3 特征图分辨率将与 Conv3_3 特征图分辨率相同，因而可以将 2 种特征图进行拼接，并对特征区域进行池化，去除冗余信息，具体如图 6 所示。

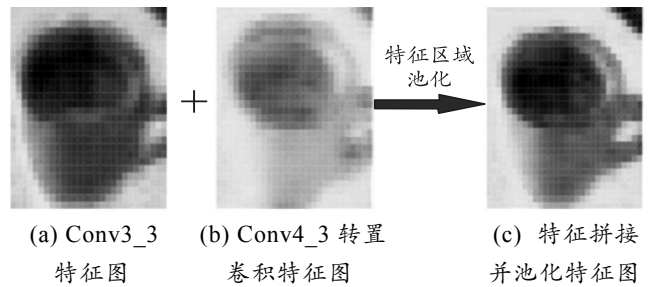


图 6 特征拼接及池化

图 6(a)为 Conv3_3 的特征图像，图 6(b)为 SSD 网络中 Conv4_3 转置卷积后的特征图，图 6(c)是将前两者进行特征拼接与特征池化操作后得到的可视化特征图，再次池化操作选择的是最大池化。可以看出：相比于图 6(a)和图 6(b)，图 6(c)细节信息更丰富、分辨率更清晰。改进后的 SSD 模型网络结构如图 7 所示。

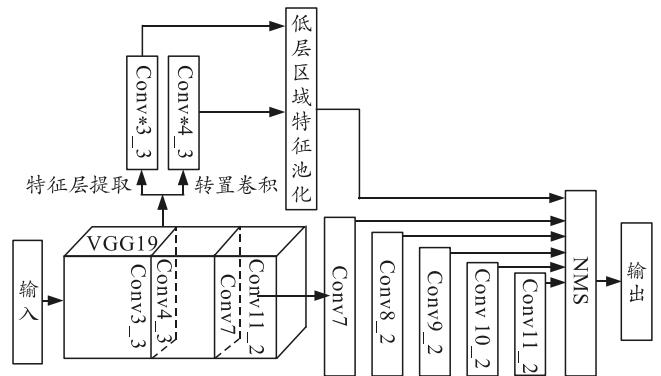


图 7 改进的 SSD 模型

改进后的 SSD 网络结构中低层特征图将具有更高的分辨率、更加全面的信息及更强的拟合能力。同时网络结构中靠后的特征层拟合能力不变，不会产生过拟合的问题；并且只针对原始网络模型结构中的低层特征图即特征层 Conv4_3 进行改进，避免引入过多转置卷积导致参数过多而造成检测速度过低的问题。

3 实验与分析

为了检验 SSD 改进算法的检测效果，在 PASCAL VOC 数据集上进行检测，将 VOC 2007 训练集部分与 VOC 2012 训练集部分合并为此次实验训练数据集，测试集为 VOC 2007 测试集。

在目标检测算法中，评价一个算法模型的优劣，最客观的评价指标有召回率(recall)、准确率(precision)、平均准确率 AP、均值平均准确率 mAP(mean average precision) 以及速度评价指标 FPS(frame per second)，通常使用 mAP 来评估一个检测算法的检测性能，使用 FPS 评价该算法的速度性能，一般来说数值越大，证明检测速度越快。

虽然现有的目标检测算法处理的是多标签的分类问题，但是对于某一个类别，更关注的是该类别对错与否，相当于二分类问题，对于某一类问题，将分类的目标计为正例(positive)和负例(negative)，就有如下的指标：

- 1) True positives(TP): 被正确地划分为正例的个数，即实际为正例且被分类器判别出正例的数目；
- 2) False positives(FP): 被错误地划分为正例的个数，即实际为负例但被分类器判别为正例的数目；
- 3) False negatives(FN): 被错误地划分为负例的个数，即实际为正例但被分类器判别为负例的数目；
- 4) True negatives(TN): 被正确地划分为负例的个数，即实际为负例并被分类器判别为负例的数目。

这些指标间的关系可以用混淆矩阵来表示。如表 1 所示。

表 1 分类结果混淆矩阵

真实情况	判别结果	
	正例	反例
正例	TP(真正例)	FN(假反例)
反例	FP(假正例)	TN(真反例)

召回率 R 以及准确率 P 的定义公式分别为：

$$R=TP/(TP+FN), P=TP/(TP+FP)。(5)$$

式中：AP 值为召回率与准确率所围成的曲线面积；mAP 值是通过对各类的 AP 值求均值得到的。

本次实验环境的操作系统为 Windows10，深度学习框架为 tensorflow。实验硬件环境的 CPU 为 Intel (R) Core(R) i7 9700KF，内存为 32 G，GPU 为 NVIDIA(R) GTX(R) 1080TI^[15]。

3.1 结果分析

VOC200 数据集包含了 20 类目标，包括 person、bird、horse 等；VOC 2012 数据集同样也包含 20 类目标，但与 VOC 2007 数据集互斥；因此，可以将两者的训练数据集相结合使用。表 2 为 PASCAL VOC 数据集的具体类别。

表 2 PASCAL VOC 数据集目标类别

目录	类别
人类	人
动物	鸟, 猫, 牛, 狗, 马, 羊
交通工具	飞机, 自行车, 船, 公共汽车, 轿车, 摩托车, 火车
室内物品	瓶子, 椅子, 餐桌, 盆栽, 沙发, 电视

在 PASCAL VOC 数据集上进行训练，由于目前官方只公布了 VOC2007 的训练集与测试集与 VOC2012 的训练集，VOC2012 的测试集并未公布。因此本次实验训练集为 VOC2007 与 VOC2012 的训练集，预训练模型为 VGG-19，batchsize 设置为 64，并进行了 60 000 次迭代，训练结束后通过 tensorflow 的可视化工具 tensorboard 查看，可得训练过程中的 mAP 值曲线如图 8 所示。

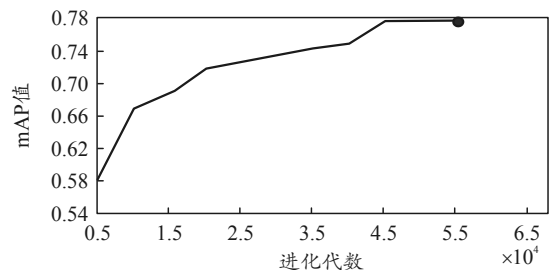


图 8 训练过程中 mAP 曲线

从图中可以看出，mAP 值最终达到了 77.9%，表明模型的训练已经达到了不错的效果。

利用 VOC2007 数据集进行测试，并对比一些性能比较优异的算法。性能对比如表 3 所示。

表 3 PASCAL VOC 检测结果对比

方法	基础网络	mAP/%	FPS/(帧/s)
Faster R-CNN	VGG-16	73.2	6.0
YOLO	GoogLeNet	66.8	20.0
YOLO V2	Darknet-19	73.7	81.0
SSD300	VGG-16	74.3	28.0
DSSD	ResNet-101	78.6	9.5
SSD300*	VGG-19	77.9	24.0

表中，SSD300*代表改进后的 SSD 算法，可以看出 SSD300 的检测精度明显高于 YOLO，而改进

的 SSD300 模型在原始 SSD300 的基础上检测精度提高了 3.6%。

笔者统计了不同目标检测算法的检测速度和精度的分布。如图 9 所示，与原算法 SSD300 相比，改进后的 SSD 目标检测算法在精度和速度方面具有明显优势。

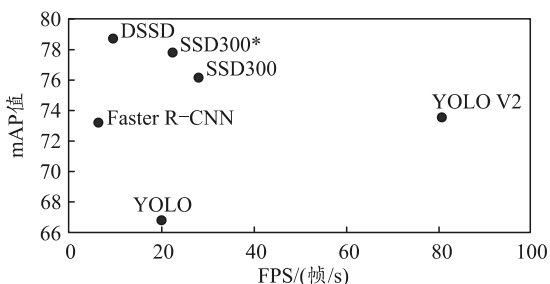


图 9 不同检测算法在速度和精度指标上的分布

从图中可以看出：检测精度方面虽然 DSSD 精度最高，但 DSSD 速度损耗十分严重。这是因为 DSSD 的基础网络采用 ResNet-101^[16]网络深度比 VGG19 要深，结构设计也更为复杂，网络中存在大

量计算参数，性能损耗严重，造成了 DSSD 的检测速度仅有 9.5 帧/s，而且检测精度仅比文中算法高 0.7%。文中采取的改进方式，在网络结构中增加了一层特征图和计算参数，但并没有对检测速度造成太大影响，FPS 达到了 24 帧/s，仅比原 SSD 算法慢 4 帧/s，已满足实时性检测的要求；因此，相比于 DSSD，改进算法在速度没有受太大影响的基础上，精度却提高了 3.6%，仅用很小的速度损耗就达到了接近 DSSD 的检测效果，其性能比原算法也有极大提升。

检测速度方面，虽然 YOLO V2 检测速度最快，但其检测精度过低，这是因为 YOLO V2 网络结构比较简洁，并且其设计最初目的就是速度为主的一个快速检测算法，所以检测的综合效能并不是十分优秀。

对于在每一类目标检测的精度，将改进后的算法与原 SSD 算法进行了比较，统计结果如表 4 所示。

表 4 PASCAL VOC 测试集单个类别 AP 值 %

算法	人	鸟	猫	狗	牛	羊	飞机	自行车	船	摩托车	mAP
SSD300	78.2	75.3	86.3	84.2	81.8	76.1	75.4	81.3	70.4	83.5	74.3
SSD300*	80.0	77.8	88.8	86.2	81.2	78.0	79.1	84.1	72.2	84.4	77.9
算法	小轿车	公共汽车	火车	椅子	电视	马	植物	餐桌	沙发	瓶子	mAP
SSD300	83.7	84.3	87.9	59.3	76.2	86	50.1	77.6	78.3	48.6	74.3
SSD300*	86.2	85.7	87.8	61.2	77.6	87.3	52.7	76.8	80.4	51.2	77.9

改进后的算法在诸如飞机、鸟、瓶子等小目标的检测效果上明显优于原 SSD 目标检测算法，AP 值有明显提升。

图 10 为在 VOC 2007 测试数据集上的部分检测图片。

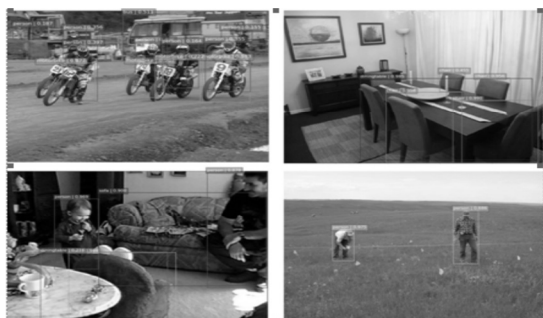


图 10 PASCAL VOC 数据集检测结果

3.2 拓展试验

除了在标准数据集上进行试验和测试对比算法的性能外，针对改进后算法的小目标检测能力，对原 SSD 算法与改进后的算法进行对比试验，收集一些图片进行测试，检测结果对比如图 11 所示。可以直观地看出：SSD 算法在经过改进后，对于小目标

物体的检测效果有了明显提升。这是因为在网络结构中增加了一层更底层的特征层并融合转置卷积以加强对小目标物体的特征感知能力。



(a) SSD300 检测图像 (b) SSD300*检测图像

图 11 改进前后算法对小目标的检测结果