文章编号:1001-5078(2019)08-1019-07 ·图像与信号处理·

基于SSD改进的目标检测方法研究

张俊蓉,徐长彬,唐明周,鹿 玮,卞紫阳
(华北光电技术研究所,北京 100015)

摘 要:为了满足目标检测任务实时性的要求,基于轻量级深度学习目标检测网络SSD_Mobilenetv1,通过改进其网络结构,以及增加更细粒特征图参与位置回归和分类来综合网络的上下文信息及引入反残差模块提升网络提取特征的能力,实验表明在保证实时检测速度的同时提高了检测精度,并在KITTI数据集上进行训练验证,取得了良好的效果。

关键词:深度学习;KITTI数据集;目标检测
中图分类号:TP391.41 **文献标识码:**A **DOI:**10.3969/j.issn.1001-5078.2019.08.020

The improved target detection methods based on SSD network

ZHANG Jun-rong,XU Chang-bin,TANG Ming-zhou,LU Wei,BIAN Zi-yang
(North China Research Institute of Electro-Optics,Beijing 100015,China)

Abstract: In order to use the video image information to detect and track the target in real time,based on the light-weight deep learning target detection network SSD_Mobilenetv1,by improving its network structure,using the more fine-grained feature map to participate in position regression and classification to integrate the context information of the network and introduce the inverse,the residual module improves the ability of the network to extract features. The experiment shows that the real-time detection speed is guaranteed and the detection accuracy is improved,and the training and verification on KITTI data set have achieved good results.

Key words:deep learning;KITTI dataset;object detection

1 引 言

目标检测是数字图像处理和计算机视觉的重点研究方向,广泛应用于航天事业、新零售行业、智能家居和无人驾驶等诸多领域,通过计算机视觉可以有效的减少人力资源损耗和加速工业化进程,具有十分重要的应用价值[1]。因此,目标检测技术成为了工业界和学术界近年来关注的重点方向,它是新时代侦察系统的重点部分,也是计算机视觉的主要应用。与此同时,随着计算机运算能力的不断加速和互联网共享资源不断积累使得基于深度学习的目标检测方法得到了广泛应用,这也为目标检测技术开辟了一条新的道路,促使其更加蓬勃的发展。

2 SSD_Mobilenetv1 网络结构

SSD_Mobilenetv1是由目标检测网络SSD利用目标分类网络Mobilnetv1作为主干网络。该网络结构是为了利用SSD的高检测精度的优势,并结合可在移动端运行的Mobilenetv1运行速度快的特点打造的轻量级目标检测网络结构。

3 SSD 目标检测网络

SSD(Single Shot MultiBox Detector)是Wei Liu[2]在ECCV 2016上提出的一种目标检测算法,截至目前是主要的检测框架之一,也是one-stage类

作者简介:张俊蓉(1983-),女,工程师,从事边海防技术研究,产品开发和系统建设工作。E-mail:canghai831118@163.com
收稿日期:2019-03-14

目标检测算法的典型代表。其相比与 two-stage 的 RCNN 系列少了 RPN 层的前景和背景的辨别，有明显的速度优势，相比 YOLO 少了全连接层和采用了空间金字塔式的多尺度训练，又有明显的 mAP 优势，SSD 与其他算法的平均精度对比如图 1 所示。

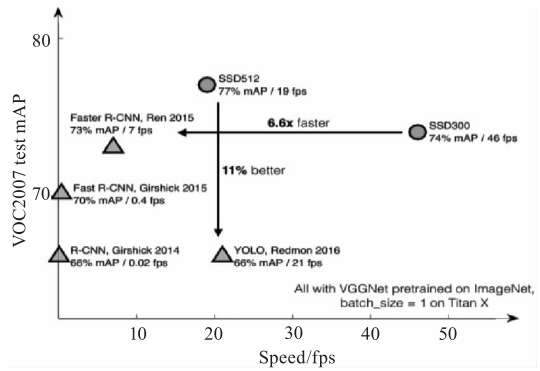

图 1 SSD 与其他目标检测算法对比

Fig. 1 Comparison of SSD and other target detection algorithms

从图 1 可以看出 SSD512 的速度虽然较慢，但其检测精度却相比于 YOLO 系列有较大的提升，而 SSD300 相对精度降低，但其速度却远超其他算法。

SSD 方法采用了端到端检测的思想，其产生若干大小、比例的初始框集合和框中目标类别的分数，接着是非最极大抑制步骤以产生最终检测[3]，SSD 为了更好地结合上下文信息，采用金字塔结构结合不同感受野大小的特征图，在多个尺度特征图上同时进行位置回归和 softmax 分类。SSD 提取图像特征的主干网络为 VGG－16，并利用了 conv4_3/conv_7/conv6_2/conv7_2/conv8_2/conv9_2 这些不同大小的特征图来进行位置回归和分类，进一步提高了模型的精度和对小目标的检测能力。

SSD 采用 Fast RCNN 生成锚框的相同思想来生成先验框，但不同的是 SSD 采用的是多尺度的训练方式，所以会有不同尺度特征图参与位置回归和分类，在每个特征图上都要生成相应的先验框，SSD 生成先验框的方式如图 2 所示。

如图 2 所示，SSD 以每个像素点的中点为中心根据网络输入图大小和设置的比例因子（aspect_ration）生成一系列同心的先验框，首先需要根据特征图的先后顺序计算出一个缩放因子 s_k，然后根据该缩放因子和网络输入图的大小计算出先验框的 min_size 和 max_size。

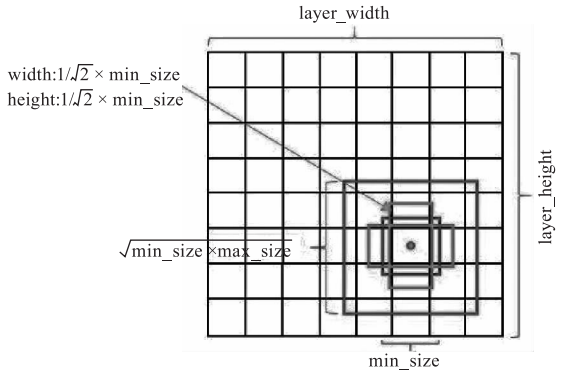

图 2 SSD 生成先验框

Fig. 2 SSD generation a priori box

缩放因子 s_k 和 min_size 及 max_size 的计算公式如下：

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m - 1}(k - 1), k \in [1, m) \quad (1)$$

$$\text{min\_size}_k = s_k \times \text{input\_size} \quad (2)$$

$$\text{max\_size}_k = s_k \times \text{input\_size} \quad (3)$$

式中，m 是指参与位置回归和分类的特征图的数量。在得到 min_size 和 max_size 值之后，会生成两个边长分别为 min_size 和 $\sqrt{\text{min\_size} \times \text{max\_size}}$ 的正方形。最后，每增加一个比例因子生成两个高和宽分别为 $\sqrt{\text{aspect\_ratio}} \times \text{min\_size}$ 和 $\frac{1}{\sqrt{\text{aspect\_ratio}} \times \text{min\_size}}$ 的矩形。

在网络运行的时候不能一个特征图单独计算一次分类和位置回归（虽然原理如此，但是不能如此实现），图 3 以 conv4_3 和 fc7 层展示 SSD 数据流动和其分类及回归过程。

在图 3 中，假设待检测的目标共有 21 类（包括背景），以 conv_4 的数据维度（1，512，h_1，w_1）为例说明各符号的含义，1 代表一次迭代的样本数据批量数，512 代表特征图的通道数，h_i 和 w_i 分别代表该层特征图的高和宽。conv_4 的数据流动首先分为两条线路，第一条经过 conv4_3_priorbox 层将数据维度转换为（1，2，$4w_1h_1$num_priorbox$_1$），其表示共生成 $4w_1h_1$ 个先验框，因为在这一层每个像素点都有 4 个先验框。第二条首先经过批量归一化层，然后在分为两条线路，其中一条经过 conv4_3_norm_mbox_conf 层将数据维度转化为（1，84，h_1，w_1），这一层表示对先验框的分类数据维度，84 表示每个像素点都有 4 个先验框，而每个先验框的类别都有 21

种可能,最后的两次维度变化只是为了计算方便。同理,fc7 层也经过相同的变化来进行先验框分类,最后将两层的最后一个维度拼接在一起同时计算。第二条经过 conv4_3_norm_mbox_loc 层将数据维度转化为 $(1,16,h_1,w_1)$,这一层表示对先验框的位置回归,16 表示每个像素点都有 4 先验框,而每个先验框的位置维度为 4 维,最后的两次维度变化也只是为了计算方便。同理,fc7 层也经过相同的变化来进行位置回归,最后将两层的最后一个维度拼接在一起同时计算。

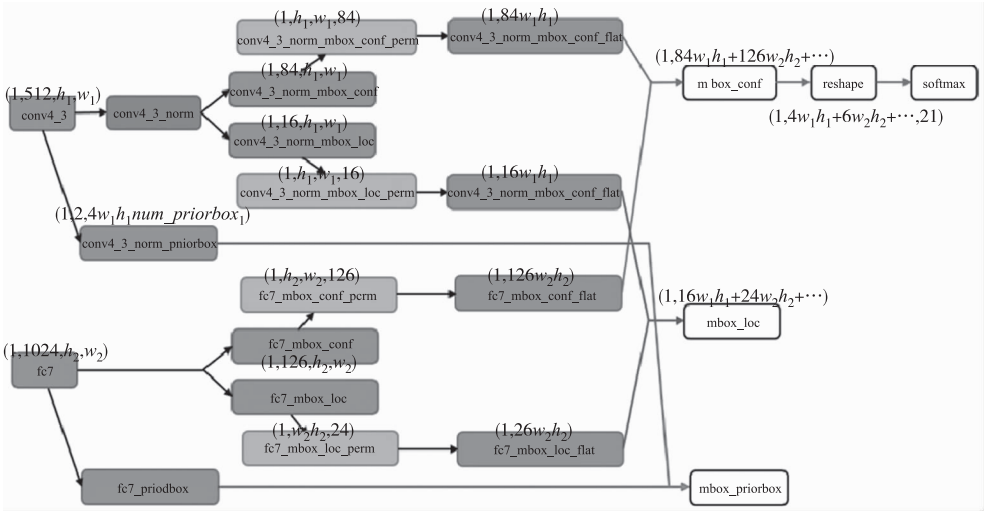

图 3　SSD 网络数据流动示意

Fig. 3 SSD network data flow indication

4　Mobilenetv1 分类算法

MobileNet v1 是谷歌在 2017 年提出的手机端的网络架构[4],其目的是从结构上减少网络参数,加速网络运行。MobileNetv1 模型基于如图所示深度可分解的卷积,它可以将标准卷积分解成一个深度卷积和一个点卷积(1 × 1 卷积核)。深度卷积相当于做切片卷积其输出通道与输入通道数相同,而 1 × 1卷积相当于对特征图的每一个像素点位置的通道做全连接操作,其示意图如图 4 所示。

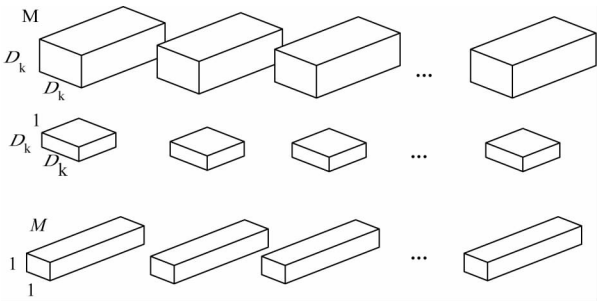

图 4　深度分离卷积示意图

Fig. 4 Deep separation convolution schematic

图 4 中,$D_k \times D_k$ 是卷积核的大小,M 代表了卷积核的通道数,N 代表了卷积核的个数。

Mobilenetv1 通过 Depthwise + Pointwise 的拆分,相当于将普通卷积的计算量压缩为:

$$\frac{D_k \times D_k \times M \times D_F \times D_F + M \times N \times D_F \times D_F}{D_k \times D_k \times M \times N \times D_F \times D_F} = \frac{1}{N} + \frac{1}{D_k^2} \tag{1}$$

以上列出的是 MobileNetv1 的基准模型,但是有时候需要更小的模型,引入了两个超参数:宽度因子和分辨率因子。宽度因子的主要功能是依照一定比例缩减通道数,将其记为 α,其取值范围为(0,1],那么在其作用下输入与输出通道数将变为 αM 和 αN。第二个超参数是分辨率因子 ρ,其取值范围与宽度因子相同为(0,1]比如原来输入特征图是 512 × 512,可以减少为 300 × 300。分辨率因子用来改变输入数据层的分辨率,同样也能减少参数。在 α 和 ρ 共同作用下,MobileNets 某一层的计算量为:

$$D_k \times D_k \times \alpha M \times \rho D_F \times \rho D_F + \alpha M \times \alpha N \times \rho D_F \times \rho D_F \tag{2}$$

默认情况下这两个调节因子的值都为 1,通过自由调节宽度因子和分辨率因子的大小,可以自由地控制模型的大小,但是如果模型的参数个数变少的同时也会影响模型的效果。SSD_Mobilenetv1 就是以 Mobilenetv1 替换 SSD 的 VGG 主干网络,利用其深度可分离卷积网络从而达到模型压

缩的效果。

5 算法改进及结果分析

本文对原始的 SSD_Mobilenetv1 算法的网络结构进行了改进,并做了对比实验。对比实验环境为 Linux 16.6 操作系统,NVIDIA 1080 型号显卡,采用 Caffe 深度学习框架。实验结果的评估指标为召回率、精度和 IOU 为 0.5 情况下的平均精度值。在实验过程中使用 K 折交叉验证将 KITTI 数据集[5]等分为 11 份,用第 11 份作为测试集,在分别以剩余每一份作为验证集,其余 9 份为相应的训练集进行训练。

第一组实验为基础实验,使用了 KITTI 数据集的所有训练图片和类别信息,共包括车辆、行人、自行车和背景 4 类目标。这组实验主要是为了测试 SSD_Mobilenetv1 的基本检测效果用来做对比实验。将 7840 张图片划分为训练集,交叉验证集和测试集。使用在 ImageNet 数据集上迭代训练 73000 次得到的模型参数进行迁移学习来加快网络的收敛速度。初始学习率设定为 0.0005,批量数为 24,输入图片大小为 300×300,共迭代 60000 次,total_loss 稳定在 2.5 左右,车辆检测的平均精度值为 0.67,召回率为 0.55,精度值为 0.9,其精度 - 召回率曲线如图 5 所示。其中,$C=4$,表示训练时包含的类别为 4 类(车辆、行人、自行车、背景)。

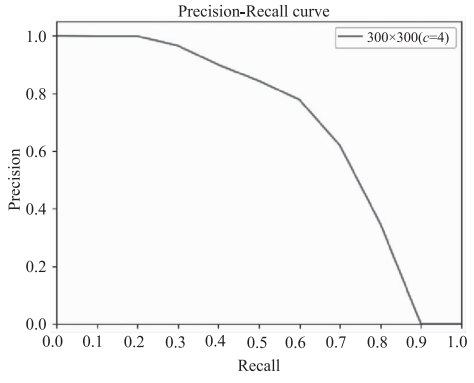

图 5 车辆目标 300×300 精度 - 召回率曲线

Fig. 5 Vehicle target 300×300 accuracy-recall rate curve

第二组实验简单地希望通过增加输入图像的分辨率来改善目标的提取能力,其采用的方法是直接将网络输入图片的大小增加到 512×512,在 SSD_Mobilenetv1 网络中对输入图片共进行了 9 次降采样,并利用最后 6 次降采样的特征图进行多尺度的训练,300×300 和 512×512 的降采样特征图尺寸对比如表 1 所示。

表 1 300×300 和 512×512 降采样尺寸对比

Tab. 1 Comparison of 300×300 and 512×512 downsampling sizes

	输入图片大小	输入图片大小
Data	300×300	512×512
cov0	150×150	256×256
cov2	75×75	128×128
cov4	38×38	64×64
conv11	19×19	32×32
conv13	10×10	16×16
conv14	5×5	8×8
conv15	2×2	4×4
conv16	1×1	2×2
conv17	1×1	1×1

从表 1 中可以看出,输入大小为 300×300 和 512×512 的网络都经过了 9 次降采样,最终的特征图大小都为 1×1,同时 SSD_Mobilenetv1 使用最后 6 个降采样的特征图进行位置回归和分类。

将输入改为 512×512,同时为防止内存溢出,在训练时将批量数超参数改为 4,在其他网络参数不变的情况下,训练后测试得到的精度 - 召回率曲线如图 6 所示。

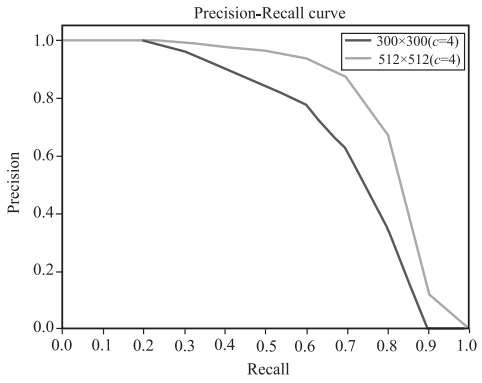

图 6 车辆目标 512×512 时精度 - 召回率对比曲线

Fig. 6 Accuracy-recall ratio comparison curve for vehicle target 512×512

由图 6 可知,通过提高网络输入图片的大小可有效改善网络模型的检测效果,但与此同时网络的运行速度也会降低。最终测得 512×512 时模型的平均精度为 0.775,相比于输入为 300×300 的基础模型提高了 10 % 左右。

第三组实验通过增加 Conv11 层特征图上生成的先验框个数尝试提升模型的目标检测能力,因为

在 SSD_Mobilenetv1 中每个尺寸的特征图像素点上生成的先验框的大小和个数都不尽相同,每层都会通过计算 min_size 和 max_size,并且通过给定的比例因子生成固定大小和个数的先验框。在 SSD_Moilenetv1 中的每个尺寸上的 min_size 和 max_size 大小和比例因子如表 2 所示。

表 2 不同尺寸上的先验框对比图

Tab. 2 Comparison of a priori boxes on different sizes

	min_size	max_size	比例因子	个数
conv11	60	0	2	3
conv13	105	150	[2,3]	6
conv14	150	195	[2,3]	6
conv15	195	240	[2,3]	6
conv16	240	285	[2,3]	6
conv17	285	300	[2,3]	6

从表 2 中可以看出 conv11 层每个像素点只生成 3 个先验框,而且只设定了 mine_size = 60,可计算出 conv11 的特征图大小为 60 × 60, 84.85 × 42.42,42.42 × 84.85,而由表车辆大小分布统计可以看出目标大部分分布在 32 × 32 ~ 96 × 96 范围内,所以考虑增加 conv11 在这个范围内的先验框从而提升目标检测中等目标的能力。

首先按照 SSD 论文中生成先验框的方法,按照公式计算可得 conv11 的 max_size 为 105,并且增加比例因子为 3,因此可在 conv11 的特征图上增加大小为 79.37 × 79.37,103.92 × 34.64,34.64 × 103.92 的先验框。在输入为 512 × 512,批量数为 4 的条件下,其他参数不变,按上述方法更改网络结构,训练后测试结果如图 7 所示。

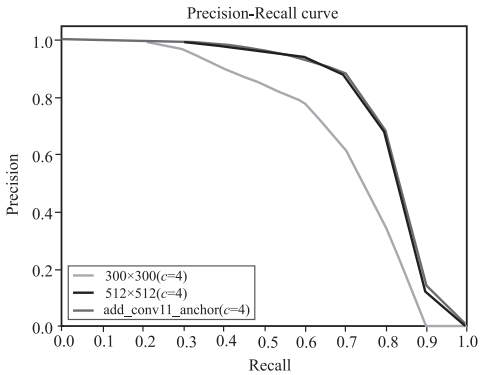

图 7 conv11 增加先验框后的精度 - 召回率对比曲线

Fig. 7 Accuracy after adding a prior frame to conv11-recall ratio comparison curve

从图 7 可以看出,通过增加 conv11 层的先验框后的平均精度为 0.780,相比 512 × 512 模型提高 0.03 %,提升效果不明显。

第四组对比实验通过增加 conv5 层的特征图参加位置回归和分类来改善模型的目标检测能力。在车辆目标大小分布中可知小于 32 × 32 的目标占总目标的 19.8 %,而现有 conv11 最小的先验框大小为 60 × 60,所以期望通过增加 conv5 的 64 × 64 的尺寸提高小目标检测能力。

通过增加 conv5 的输入特征图参与目标框回归和分类,并设定该尺寸的生成框的参数为 min_size 为 30,max_size = 60,比例因子为[2,3]。其生成的先验框大小分别为 30 × 30,42.42 × 42.42,42.42 × 21.21, 21.21 × 42.42, 51.96 × 17.32, 17.32 × 51.96。在输入为 512 × 512,批量数为 4 条件下,其他参数不变,按上述方法更改网络结构,训练后测试结果如图 8 所示。

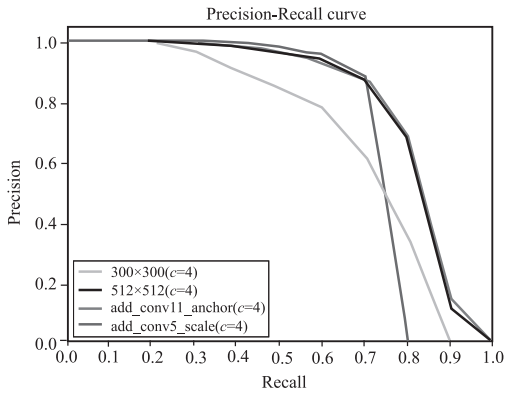

图 8 车辆目标增加 conv5 尺寸的精度 - 召回率对比曲线

Fig. 8 Accuracy of conv5 size increase for vehicle target-recall ratio comparison curve

从图 8 可以看出,通过增加 conv5 的特征图参与位置回归和分类后,平均精度为 0.709,相比于上一组实验模型精度降低 7 %。

第五、六、七组实验将车辆、行人和自行车 3 类目标改为车辆目标,在之前的训练中同时训练 3 类检测目标,在网络的反向传播其他类别的信息对于车辆目标检测相当于噪声,所以将训练的目标改为车辆 1 类。将标注文件中的自行车和行人删去,并只保留有车辆目标的图片,重新生成训练数据集,重新训练。按照 3 类时三种改进方法的参数设置在新构造的数据集训练后的结果如图 9 所示。

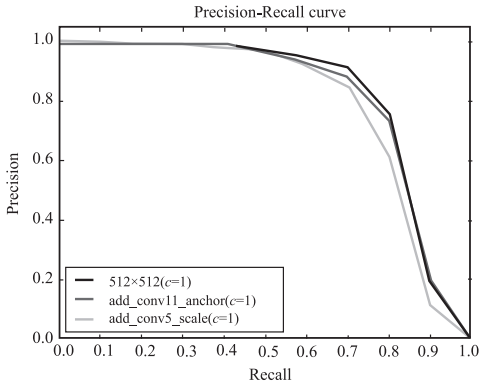

图9 车辆单目标检测的精度-召回率对比曲线

Fig. 9 Accuracy of vehicle single target detection-recall ratio comparison curve

从图9可以看出,通过将车辆、行人和自行车3类目标改为车辆当目标重新训练之后,512×512(class=1)时平均精度为0.798,同3类相比增长,add_conv11_anchor($c=1$)时平均精度为0.791,同4类相比增长,add_conv5_scale($c=1$)时平均精度为0.769,同4类相比增长。

第八组实验在conv11之前添加反残差模块,因为SSD_Mobilenetv1的特征提取能力较弱,在conv11之前新增反残差模块,提升网络的特征提取能力。

原始的conv11层和添加反残差后的网络结构分别如图10和图11所示。将conv11/dw/relu的输出结果先经过1×1×512×1024的卷积核将原先的512通道数扩张为1024,再经过一个3×3×1024的深度卷积层,接着通过1×1×1024×512的点卷积层将通道数压缩为512,最后通过sum层按元素相加。

在输入大小为512×512,批量数为4,其他超参数不变,按照上述改进网络结构,得到的结果与前几组实验对比结果如图12所示。

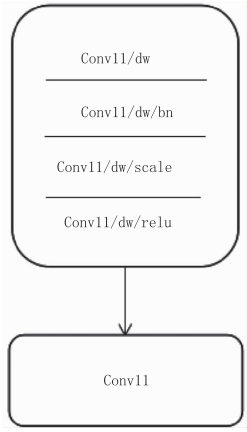

图10 conv11原始网络结构图

Fig. 10 conv11 original network structure diagram

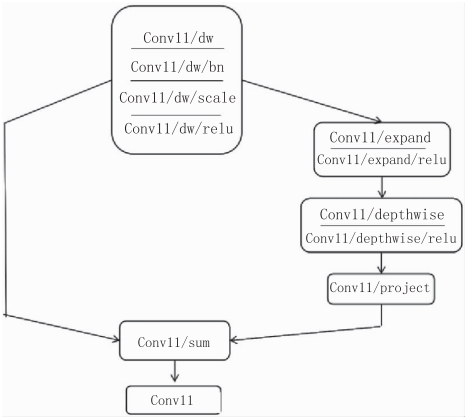

图11 添加反残差之后的conv11结构图

Fig. 11 conv11 structure diagram after adding anti-residual

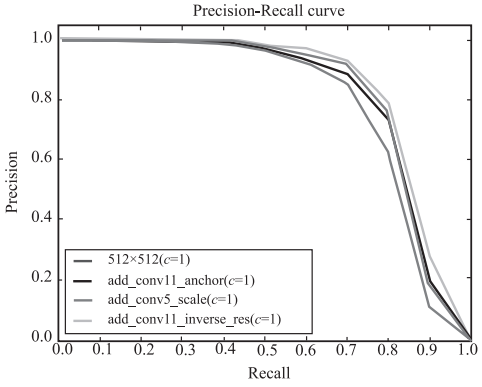

图12 车辆目标增加反残差的精度-召回率对比曲线

Fig. 12 Accuracy of increasing anti-residuality of vehicle target-recall ratio comparison curve

从图12可以看出,通过在conv11层增加反残差模块,在车辆单类目标数据集上训练测试,平均精度为0.811。

在以上的分析中,为了在不显著降低SSD_Mobilenetv1的检测速度前提下提升模型的检测精度共进行了八组对比实验,最后一组实验相对于第一组基础网络实验的平均精度从0.677提升到了0.811,表3对各组实验的各项评估做出了对比。

从表3可以看出深度分离卷积可大幅度提高检测速度,以单类数据集进行训练相对于以多类数据集进行训练可获得更大的精度,并且以提高图像的分辨率来提升目标检测精度的方式效果最明显(平均精度提升约10%),但是这也会相应的增加网络的运行时间,其次通过在conv11加入反残差模块也可提高模型检测精度,这也反映了SSD_Mobilenetv1的特征提取能力较弱,还有很大的提升空间,但其检测速度也会大幅下降。在增加conv5的尺度训练时精度下降是因为在KITTI训练数据集中许多小目标未标出,在计算损失函数的时候造成较大的干扰。

表 3 各组实验评价指标结果对比

Tab. 3 Comparison of results of each group of experimental evaluation indicators

	召回率	准确率	平均精度	速度(组方式)/ms	速度(分离方式)/ms	模型大小/MB
第一组	0. 374	0. 916	0. 677	27. 341	15. 517	22. 2
第二组	0. 553	0. 949	0. 775	29. 063	18. 119	22. 2
第三组	0. 489	0. 964	0. 780	30. 407	17. 334	22. 2
第四组	0. 540	0. 975	0. 709	30. 504	16. 424	22. 3
第五组	0. 567	0. 960	0. 798	33. 031	18. 162	22. 0
第六组	0. 496	0. 960	0. 791	32. 971	18. 210	22. 1
第七组	0. 481	0. 971	0. 769	32. 061	17. 392	22. 1
第八组	0. 602	0. 965	0. 811	36. 031	20. 742	26. 3

6 结 论

本文基于 SSD_Mobilenetv1 网络模型完成了车辆目标的检测任务。基于 KITTI 数据集和 SSD_Mobilenetv1 共做了八组对比实验,第一组、第二组、第三组和第四组利用车辆、行人和自行车三类目标的数据集分别采取了原结构网络不变、增大输入图片尺寸、增加 conv11 层先验框个数、添加 conv5 的特征图大小参与位置回归和分类的方法提升网络检测效果,第五组、第六组、第七组和第八组利用车辆单目标的数据集分别采取了增大输入图片尺寸、增加 conv11 层先验框个数、添加 conv5 的特征图大小参与位置回归、在 conv11 层添加反残差模块的方式提升网络检测效果。最终,将目标检测的平均精度由基础网络的 0. 667 提升到了 0. 811,但其检测速度也有所降低。

参考文献:

[1] Liu Y, Wang D. Application of deep learning in genomic selection[C]//IEEE International Conference on Bioinformatics and Biomedicine, IEEE Computer Society, 2017: 2280 – 2280.

[2] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multiBox detector[C]//European Conference on Computer Vision, Springer International Publishing, 2016: 21 – 37.

[3] Neubeck A, Gool L V. Efficient non-maximum suppression [C]//International Conference on Pattern Recognition, IEEE, 2006: 850 – 855.

[4] Howard A G, Zhu M, Chen B, et al. MobileNets: efficient convolutional neural networks for mobile vision applications[J]. arxir: Computer Vision and pattern Recognition. 2017, 12(5): 114 – 116.

[5] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The KITTI dataset[J]. The International Journal of Robotics Research, 2013, 32(11): 1231 – 1237.