Journal of Computer Applications
计算机应用,2020,40(8):2472－2478
ISSN 1001-9081
CODEN JYIIDU
2020-08-10
http://www.joca.cn

文章编号:1001-9081(2020)08-2472-07
DOI:10.11772/j.issn.1001-9081.2020010062

基于YOLO v3算法改进的交通标志识别算法

江金洪[1,2*],鲍胜利[1,2],史文旭[1,2],韦振坤[1,2]
(1.中国科学院 成都计算机应用研究所,成都 610041; 2.中国科学院大学,北京 100049)
(*通信作者电子邮箱 1127515524@qq.com)

摘 要:针对目前交通标志识别任务在使用深度学习算法时存在模型参数量大、实时性较差和准确率较低的问题,提出了基于YOLO v3改进的交通标志识别算法。该算法首先将深度可分离卷积引入YOLO v3算法的特征提取层,将卷积过程分解为深度卷积、逐点卷积两部分,实现通道内卷积与通道间卷积之间的分离,从而保证了在较高识别准确率的基础上极大地减少了算法模型参数数量以及计算量。其次,在损失函数设计上使用广义交并比(GIoU)损失替换均方误差(MSE)损失,将评测标准量化为损失,解决了MSE损失存在的优化不一致和尺度敏感的问题,同时将Focal损失加入到损失函数以解决正负样本严重不均衡的问题,通过降低大量简单背景类的权重使得算法更专注于检测前景类。将该算法应用于交通标志任务中的结果表明,在TT100K数据集上,该算法的平均精度均值(mAP)指标达到了89%,相较于YOLO v3算法提升了6.6个百分点,且其参数量仅为原始YOLO v3算法的1/5左右,每秒帧数(FPS)亦比YOLO v3算法提升了60%。该算法在极大地减少模型参数量和计算量的同时,提高了检测速度和检测精度。

关键词:交通标志识别;YOLO v3算法;广义交并比;深度可分离卷积;损失函数;Focal损失
中图分类号:TP391.4 **文献标志码**:A

Improved traffic sign recognition algorithm based on YOLO v3 algorithm

JIANG Jinhong[1,2*], BAO Shengli[1,2], SHI Wenxu[1,2], WEI Zhenkun[1,2]
(1. *Chengdu Institute of Computer Application, Chinese Academy of Sciences, Chengdu Sichuan* 610041, *China*;
2. *University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: Concerning the problems of large number of parameters, poor real-time performance and low accuracy of traffic sign recognition algorithms based on deep learning, an improved traffic sign recognition algorithm based on YOLO v3 was proposed. First, the depthwise separable convolution was introduced into the feature extraction layer of YOLO v3, as a result, the convolution process was decomposed into depthwise convolution and pointwise convolution to separate intra-channel convolution and inter-channel convolution, thus greatly reducing the number of parameters and the calculation of the algorithm while ensuring a high accuracy. Second, the Mean Square Error (MSE) loss was replaced by the GIoU (Generalized Intersection over Union) loss, which quantified the evaluation criteria as a loss. As a result, the problems of MSE loss such as optimization inconsistency and scale sensitivity were solved. At the same time, the Focal loss was also added to the loss function to solve the problem of severe imbalance between positive and negative samples. By reducing the weight of simple background classes, the new algorithm was more likely to focus on detecting foreground classes. The results of applying the new algorithm to the traffic sign recognition task show that, on the TT100K (Tsinghua-Tencent 100K) dataset, the mean Average Precision (mAP) of the algorithm reaches 89%, which is 6.6 percentage points higher than that of the YOLO v3 algorithm; the number of parameters is only about 1/5 of the original YOLO v3 algorithm, and the Frames Per Second (FPS) is 60% higher than YOLO v3 algorithm. The proposed algorithm improves detection speed and accuracy while reducing the number of model parameters and calculation.

Key words: traffic sign recognition; YOLO v3 algorithm; Generalized Intersection over Union (GIoU); depthwise separable convolution; loss function; Focal loss

0 引言

目前,交通标志在现实生活中随处可见,道路上的减速限行、安全警示、车辆引流等交通标志为人们安全便捷出行提供了强有力的保障。针对理想情况下的交通标志识别算法研究已取得较高的成就,但由于车辆在实际道路上获取的图片容易受到光照强度、天气状况等因素的影响,且交通标志目标往往只占整张图片的极小部分,这使得交通标志识别在车辆真实行驶过程中的应用存在诸多挑战[1]。因此,真实自然条件下交通标志识别的研究具有重要价值。

收稿日期:2020-02-04;**修回日期**:2020-03-31;**录用日期**:2020-03-31。
基金项目:四川省科技厅重点研发项目(2018SZ0040);四川省新一代人工智能重大专项(2018GZDZX0036)。
作者简介:江金洪(1994—),男,四川绵阳人,硕士研究生,主要研究方向:深度学习、数据挖掘; 鲍胜利(1973—),男,安徽黄山人,研究员级高级工程师,博士研究生,主要研究方向:智能信息处理、深度学习; 史文旭(1995—),男,河南焦作人,硕士研究生,主要研究方向:深度学习、智能算法; 韦振坤(1995—),男,安徽阜阳人,博士研究生,主要研究方向:强化学习、机器学习。

传统交通标志识别算法主要利用图像处理技术对图像的颜色、形状、边缘等进行提取特征和分类。文献[2]中提出了在HSV(Hue, Saturation, Value)空间训练自适应增强(Adaptive boosting, Adaboost)分类器的检测算法,该方法具有较好的鲁棒性和较高的准确率,但检测速度较低;文献[3]中基于CIELab和YCbCr空间的方向梯度直方图(Histogram of Oriented Gradient, HOG)特征训练支持向量机(Support Vector Machine, SVM)分类器,但该方法泛化能力较弱;文献[4]中根据交通限速标志的颜色和形状特征,提出了一种基于车载视频的交通限速标志的检测和识别算法;文献[5]中则提出了基于深度森林的交通标志识别算法。上述算法虽然在准确率上不断提高,但它们在实时性和准确率的平衡性上依然难以达到卷积神经网络(Convolutional Neural Network, CNN)所能达到的效果。

自2012年AlexNet[6]在ImageNet[7]图像分类比赛中获得巨大成功后,CNN便广泛应用于计算机视觉领域。近几年由于各种新型CNN结构不断地被提出,使得目标检测算法得以迅猛发展。当前,深度学习目标检测算法可以分为两类,以Faster R-CNN(Faster Region-CNN)[8]为代表的双阶段目标检测算法和以YOLO(You Only Look Once)[9]、单次多框检测(Single Shot multiBox Detector, SSD)算法[10]为代表的单阶段目标检测算法。由于CNN在计算机视觉领域存在速度快、准确度高的优势,使得它在交通标志识别任务中得到广泛应用。2011年,Sermanet等[11]在德国交通标志(German Traffic Sign Recognition Benchmark, GTSRB)数据集[12]上实现了神经网络识别交通标志首次超过人工的效果,仅有0.56%的错误率;2016年,腾讯公司联合清华大学创建了一个接近真实驾驶环境的数据集TT100K(Tsinghua-Tencent 100K)[13],并训练了两个卷积网络用于识别与分类,其准确率能达到88%,召回率能达91%;2018年,Wang等[14]提出了一个级联掩码生成框架来解决分辨率与小目标检测之间的矛盾,通过多次对感兴趣区域(Region Of Interest, ROI)的回归,得到了定位更准确的目标框及更高的精度。

深度CNN虽然能提升识别算法的准确率和实时性,但其计算量和参数量都相对比较大,对硬件需求较高,且目标框交并比(Intersection over Union, IoU)计算与边框回归损失函数的优化方向并不完全等价,会使得目标框定位存在误差。为减少算法的计算量和提高目标框的定位精度,本文提出了一种深度可分离的YOLO v3改进算法IYOLO(Improved YOLO v3),主要工作如下:

1)在YOLO v3[15]的网络结构基础上引入深度可分离卷积(Depthwise Separable Convolution, DSC)[16],使其在不损失准确率的基础上,减少了模型参数数量和计算量;

2)为提高算法的准确率和目标框定位精度,在原YOLO v3损失函数的基础上引入了广义IoU(Generalized IoU, GIoU)损失[17]和Focal损失(Focal Loss)[18],使设计的损失函数优化方向与目标框最大IoU计算方向一致,同时在一定程度上解决了类别之间的不均衡问题,提高了检测准确率。

1　IYOLO网络结构

1.1　YOLO v3

YOLO v3算法是基于YOLO、YOLOv2[19]算法的改进算法,它在检测速度和精度上均有很大的提高。YOLO算法最早是由Redmon等[9,15,19]提出,其思想是将整张图片作为神经网络的输入,并在最后输出层直接输出回归的目标框位置和类别信息。不同于Faster R-CNN算法需要在中间层生成候选区域,YOLO算法采用直接回归的思路,实现了端到端的结构,这使得算法在输入图片大小为448×448时每秒帧数(Frames Per Second, FPS)能达到45,其精简版本Fast YOLO的FPS甚至可达到155,检测速度远远快于Faster R-CNN。针对YOLO算法存在对小目标和密集目标检测效果差以及泛化能力较弱的问题,作者在之后又逐渐提出了YOLO v2和YOLO v3两种升级版本算法,其中YOLO v3算法由于其速度快、准确率高,现已广泛应用于工业检测。

YOLO v3算法使用一种残差神经网络(Darknet-53)作为特征提取层,在花费更少浮点运算和时间的情况下达到与ResNet-152[20]相似的效果。在预测输出模块,YOLO v3借鉴FPN(Feature Pyramid Network)[21]算法思想,对多尺度的特征图进行预测,即在三种不同尺度上,每个尺度上的每个单元格都会预测出三个边界框,其结构示意图如图1所示。

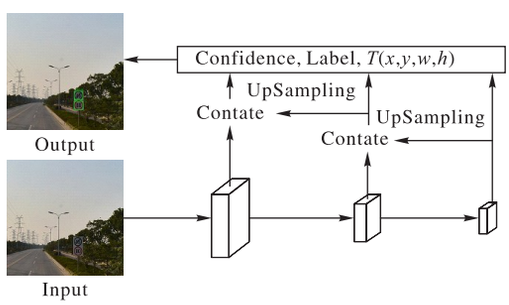

图1　YOLO v3结构示意图

Fig. 1　Schematic diagram of YOLO v3 structure

自YOLO v2算法起,YOLO算法引入anchor box,初始9个anchor box的大小由K-Means算法对所有真实目标框的长宽聚类得到,网络预测输出相对于anchor box偏移量分别为t_x, t_y, t_w, t_h,则边界框真实位置如式(1)所示:

$$\begin{cases} b_x = \sigma(t_x) + c_x \\ b_y = \sigma(t_y) + c_y \\ b_w = p_w e^{t_w} \\ b_h = p_h e^{t_h} \end{cases} \tag{1}$$

其中:(c_x, c_y)为当前单元格相对于图像左上角的偏移值,(p_w, p_h)为对应尺度anchor box的长和宽。

1.2　深度可分离卷积

在使用传统卷积计算时,每一步计算都会考虑到所有通道的对应区域的计算,这使得卷积过程需要大量的参数和计算。深度可分离卷积则是将分组卷积思路做到极致(每一通道作为一组),先对每一通道的区域进行卷积计算,然后进行通道间的信息交互,实现了将通道内卷积和通道间卷积完全分离。

在传统卷积算法中,输入为$H \times W \times N$特征图与C个尺度为$k \times k \times N$的卷积核进行卷积计算时,会得到输出特征图大小为$H \times W \times C$($padding = \lfloor k/2 \rfloor$、$stride = 1$),在不考虑偏置情况下,参数量为$N \times k \times k \times C$,计算量为$H \times W \times k \times k \times N \times C$,其卷积过程如图2所示。

在深度可分离卷积中,将卷积过程分为深度卷积(Depthwise Convolution)和逐点卷积(Pointwise Convolution)两部分。深度卷积是对输入的同一通道类进行尺寸为$k \times k$的卷积,通道间并没有信息交互,提取到的是一个通道内的特征

信息，其参数量为 $N \times k \times k$，计算量为 $H \times W \times k \times k \times N$。逐点卷积则是利用 C 个尺寸大小为 $1 \times 1 \times N$ 的卷积对通道间的信息进行融合，在实现通道间通信的同时可调控通道数量，其参数量为 $N \times 1 \times 1 \times C$，计算量为 $H \times W \times 1 \times 1 \times N \times C$，其卷积过程如图 3 所示。

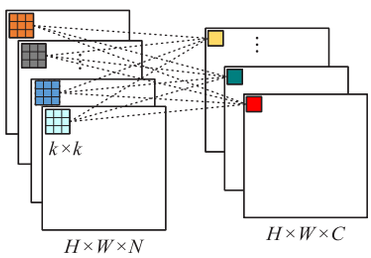

图 2　标准卷积过程

Fig. 2　Standard convolution process

相对于传统卷积计算，深度可分离卷积参数量压缩 $P_{pr} = \dfrac{N \times k \times k + N \times 1 \times 1 \times C}{N \times k \times k \times C} = \dfrac{1}{C} + \dfrac{1}{k^2}$，计算量压缩 $P_{cr} = \dfrac{H \times W \times k \times k \times N + H \times W \times 1 \times 1 \times N \times C}{H \times W \times k \times k \times N \times C} = \dfrac{1}{C} + \dfrac{1}{k^2}$。通常情况下，$C$ 远大于 k^2，则 $P_{rp} \approx 1/k^2$、$P_{rc} \approx 1/k^2$。以常见的 3 × 3 卷积为例，深度可分离卷积所需的参数约为传统卷积的 1/9，而计算量也仅为原来的 1/9，极大地降低了计算成本，提高了算法效率。

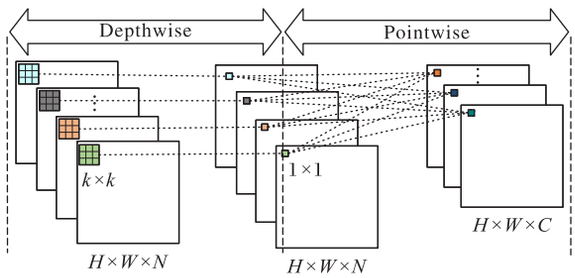

图 3　深度可分离卷积过程

Fig. 3　Depthwise separable convolution process

1.3　IYOLO 网络结构

IYOLO 整个网络结构如图 4 所示。

为解决 YOLO v3 算法在高分辨率交通标志图片上参数量较大、实时性较差的问题，提出利用深度可分离卷积重新构建网络结构。相对于标准卷积模块，深度可分离卷积模块（Depthwise Separable Convolution Module，DSC Module）如图 5 所示。

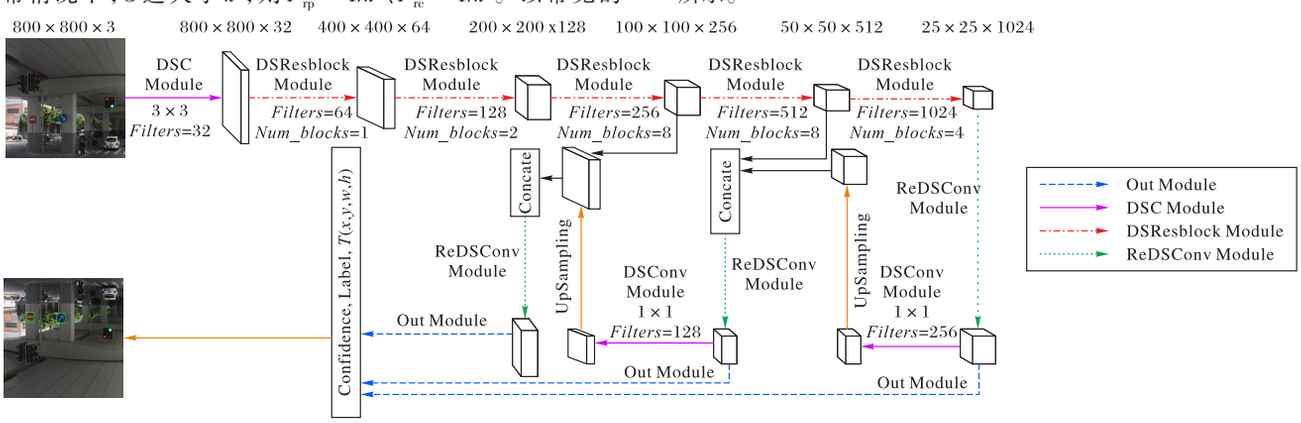

图 4　IYOLO 结构示意图

Fig. 4　Schematic diagram of IYOLO structure

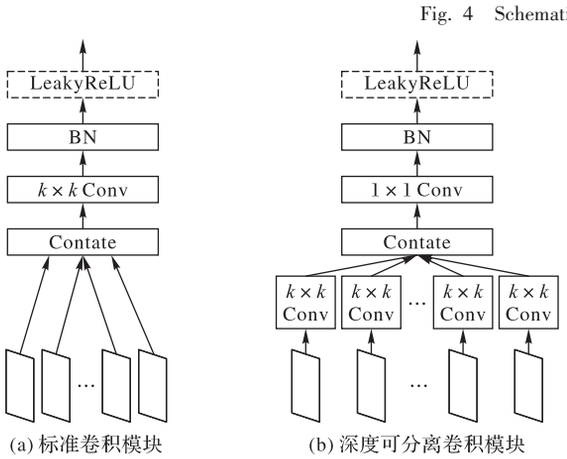

图 5　标准卷积模块和深度可分离卷积模块

Fig. 5　Standard convolution module and depthwise separable convolution module

网络结构的设计上依然借鉴了 ResNet 网络残差的思想，主体网络由多个 DSResblock 模块组成。DSResblock 模块结构如图 6 所示，其中虚线框内的结构会被重复 *Num_blocks* − 1 次。

IYOLO 的最后三层输出部分主要由 ReDSConv 模块和 Out 模块两部分组成，其结构如图 7 所示。

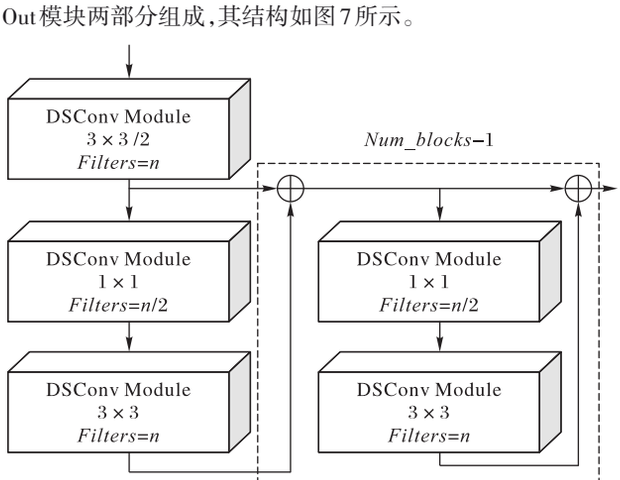

图 6　深度可分离残差模块

Fig. 6　Depthwise separable residual module

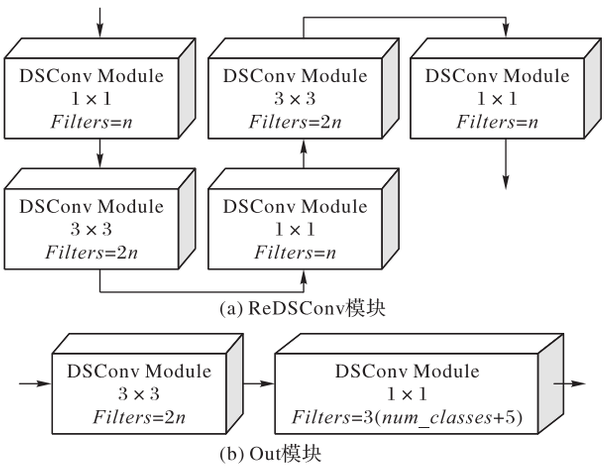

图7　输出部分示意图

Fig. 7　Schematic diagram of output section

2　IYOLO损失函数

YOLO v3算法在目标框坐标回归过程中采用的是均方误差(Mean Square Error, MSE)损失函数,在类别和置信度上使用了交叉熵作为损失函数,其损失函数如式(2)所示。

$$\begin{aligned}Loss = {} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[\left(x_i - \hat{x}_i^j\right)^2 + \left(y_i - \hat{y}_i^j\right)^2 \right] + \\ & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[\left(\sqrt{w_i^j} - \sqrt{\hat{w}_i^j}\right)^2 + \left(\sqrt{h_i^j} - \sqrt{\hat{h}_i^j}\right)^2 \right] - \\ & \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[\hat{C}_i^j \log\left(C_i^j\right) + \left(1 - \hat{C}_i^j\right) \log\left(1 - C_i^j\right) \right] - \\ & \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{noobj}} \left[\hat{C}_i^j \log\left(C_i^j\right) + \left(1 - \hat{C}_i^j\right) \log\left(1 - C_i^j\right) \right] - \\ & \sum_{i=0}^{S^2} I_{ij}^{\text{obj}} \sum_{c \in classes} \left[\hat{P}_i^j \log\left(P_i^j\right) + \left(1 - \hat{P}_i^j\right) \log\left(1 - P_i^j\right) \right] \end{aligned} \tag{2}$$

其中:λ_{coord}、λ_{noobj}分别表示坐标损失权重和不包含object的置信度损失权重;I_{ij}^{obj}表示第i个单元格的第j个box是否负责该object(1或0);$(x_i, y_i, w_i^j, h_i^j, C_i^j, P_i^j)$表示预测目标框坐标、置信度和类别;$(\hat{x}_i^j, \hat{y}_i^j, \hat{w}_i^j, \hat{h}_i^j, \hat{C}_i^j, \hat{P}_i^j)$表示真实目标框坐标、置信度和类别。

但以MSE为目标框坐标的损失函数会存在两个缺点:

1)L_2损失(即MSE损失)值越低并不等价于IoU值越高,如图8所示,三对目标框具有相同的L_2损失值,但IoU值却不一样;

2)L_2损失值对目标框尺度比较敏感,不具有尺度不变性,如在式(2)中,对w、h值开方处理就是为缓解目标框尺度对L_2损失值的影响。

IoU是在目标检测算法常用的距离测量标准,其值的计算如式(3)所示,其中A、B分别为两目标框面积。

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{3}$$

针对于MSE损失函数存在的缺陷,提出利用GIoU损失作为目标框坐标回归的损失。与IoU相似,GIoU也是一种距离度量标准,其值的计算如式(4)所示,其中A^c为两目标框的最小闭包区域面积,U为两目标框的相交面积。

$$GIoU = IoU - \frac{A^c - U}{A^c} \tag{4}$$

GIoU损失的计算如下所示:

$$L_{\text{GIoU}} = 1 - GIoU \tag{5}$$

GIoU作为距离度量标准,满足非负性、不可分的同一性、对称性和三角不等性;GIoU值是比值,因此对目标框尺度并不敏感,具有尺度不变性;由式(4)所知,GIoU的上限是IoU,当两目标框越接近且形状相似时,GIoU越接近IoU;即有当GIoU值越高时,IoU值越高。

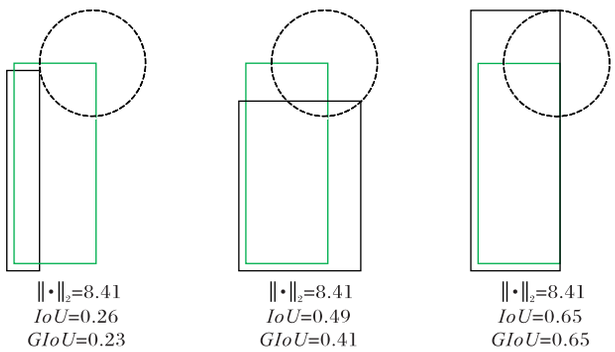

图8　三种指标关系示意图

Fig. 8　Schematic diagram of relationship among three indicators

为进一步提高识别的准确率,在对置信度设计损失函数时采用了Focal损失替换交叉熵损失。Focal损失是基于交叉熵损失的改进,主要是解决了one-stage目标检测算法中前景类与背景类比例严重不均衡的问题。Focal损失通过降低大量简单背景类在训练过程中所占的权重使得训练的算法模型更专注于前景类的检测。Focal损失如式(6)所示:

$$FL(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \tag{6}$$

其中:$p_t = \begin{cases} p, & y = 1 \\ 1 - p, & \text{其他} \end{cases}$;$\alpha_t$是控制正负样本的权重的超参;$\gamma$是控制难易分类样本的超参。

类别损失仍使用交叉熵损失如式(7)所示,其中$\hat{c}$是真实类别,c是预测类别。

$$CE(c, \hat{c}) = -\hat{c} \log(c) - (1 - \hat{c}) \log(1 - c) \tag{7}$$

改进后的算法损失函数$GFLoss$如式(8)所示:

$$GFLoss = \sum L_{\text{GIoU}} + \sum FL(p_t) + \sum CE(c_t) \tag{8}$$

IYOLO损失函数将GIoU损失作为目标框坐标回归的损失,量化评测指标GIoU为损失,这解决了原MSE损失存在的损失优化与最大IoU值计算方向不一致和对尺度敏感的问题。同时引入Focal损失,缓解了数据类别不均衡对检测算法的影响,并提高了算法的检测准确率。

3　实验与结果分析

3.1　实验数据集

为评估本文所提的IYOLO算法在真实自然环境下对交通标志的检测性能,采用了清华大学与腾讯公司公开发布的TT100K数据集。TT100K数据集数据是在腾讯街景地图上截取并进行人工标注,其图像的分辨率为2 048 × 2 048,标注类别数为221,其中包含6 107张图像的训练集和3 073张图像的测试集,覆盖了不同天气条件和不同光照下的交通标志图像。由于原始图像分辨率较大,因此在本次实验中对原图像进行了裁剪处理,裁剪后的图像尺度为800 × 800。由于数据集中各个类别之间的数据量存在严重不平衡的问题,因此本次实验只选择了数据量较多的45类交通标志进行识别,并对训练集中数据量较少的类别进行数据扩充,随机采用了加入随机

高斯噪声、亮度调整、镜像三种数据增强策略，最终使得每个类别的数据量均达3 000以上。经裁剪和扩充后，训练集包含212 384张图片，测试集包含52 413张图片，其中45类交通标志类别分别是：pn、pne、i5、p11、pl40、po、pl50、pl80、io、pl60、p26、i4、pl100、pl30、il60、pl5、i2、w57、p5、p10、ip、pl120、il80、p23、pr40、ph4.5、w59、p12、p3、w55、pm20、pl20、pg、pl70、pm55、il100、p27、w13、p19、ph4、ph5、wo、p6、pm30、w32。

3.2 评测指标

本文采用平均精度均值(mean Average Precision，mAP)和FPS两个指标对算法模型进行评估。

mAP指标通过首先计算每个类别的平均精度(Average Precision，AP)，再对所有类别的平均精度求取均值得到，计算如式(9)所示。其中：TP(True Positive)为真正例，FP(False Positive)为假正例，FN(False Negative)为假负例，N_c表示第c类划分精确率P(Precision)和召回率R(Recall)的数量，$p(r_c)$表示在c类召回率为r_c时的p值。

$$\begin{cases} P = \dfrac{TP}{TP + FP} \\ R = \dfrac{TP}{TP + FN} \\ AP_c = \dfrac{1}{N_c} \sum\limits_{r_c \in R_c} p(r_c) \\ mAP = \dfrac{1}{N} \sum AP_c \end{cases} \tag{9}$$

在实时检测任务中，FPS值是极其重要的指标，是检测速度的直接体现，对任务的应用场景有直接的影响。

3.3 结果与分析

本文实验是在Ubuntu16.04系统下进行，深度学习框架为Keras 2.1.5，所使用的显卡配置为：4块Nvidia GeForce RTX 2080 Ti，显存为44 GB。

仅引入了深度可分离卷积后，改进的YOLO v3算法明显优于原始YOLO v3算法，其对比结果如表1所示。由表1可知，引入深度可分离卷积后的YOLO v3算法相较于原始YOLO v3算法在参数量和模型大小上有了较明显的优势，只占原始算法的1/5左右，同时在mAP指标上，改进的算法也有0.3个百分点的提升。对比实验表明，YOLO v3算法结构中大部分参数是冗余的，且将深度可分离卷积引入到YOLO v3算法中以减少参数量的方法是可行的。

表1 引入DSC前后YOLO v3算法性能对比

Tab. 1 Performance comparison of YOLO v3 algorithm before and after introducing DSC

算法	参数量	模型大小/MB	*mAP*/%
YOLO v3	61 760 674	236	82.4
YOLO v3+DSC	12 425 917	47.7	82.7

将IYOLO算法与YOLO v3、SSD300、Faster R-CNN三种典型的多尺度目标检测算法对每个类别的AP值进行对比，结果如表2所示。同时，四种算法的检测精度、检测速度和模型大小整体性能对比结果如表3所示。对表2和表3数据进行分析可知，IYOLO算法mAP能达到89%，相较于YOLO v3[15]、SSD300[10]、Faster R-CNN[8]算法分别提升了6.6个百分点、25.29个百分点、2.1个百分点，且它在每个类别上的检测效果均优于YOLO v3、SSD300两种算法。从检测速度上看，IYOLO算法远远优于Faster R-CNN算法，且相较于YOLO v3算法FPS提升了60%，但与SSD300算法之间还有一定的差距。而在模型大小方面，IYOLO算法仅有原始YOLO v3算法模型大小的1/5左右，其参数量亦远小于SSD300和Faster R-CNN，得到极大的压缩。

表2 四种算法的AP值对比 单位：%

Tab. 2 Comparison of AP values of four algorithms unit：%

交通标志	SSD300	YOLO v3	Faster R-CNN	IYOLO
pn	76.15	91.52	91.40	**95.07**
pne	80.85	94.33	**96.95**	95.95
i5	77.95	93.35	95.49	**96.06**
p11	57.68	88.73	92.93	**93.49**
pl40	59.15	91.12	90.55	**94.39**
po	45.12	70.89	78.45	**80.50**
pl50	47.04	88.97	90.01	**92.69**
pl80	54.36	90.62	92.91	**94.11**
io	65.33	85.26	86.50	**88.65**
pl60	58.43	82.57	85.05	**88.72**
p26	71.53	88.99	92.19	**92.81**
i4	71.16	91.80	**95.71**	94.07
pl100	79.01	94.11	**98.23**	97.35
pl30	61.58	82.95	86.19	**88.09**
il60	78.61	89.97	94.75	**92.09**
pl5	63.57	84.92	88.39	**88.62**
i2	53.32	85.32	82.91	**83.80**
w57	70.49	89.52	88.11	**91.65**
p5	77.20	91.18	92.53	**95.73**
p10	63.39	83.36	88.75	**90.70**
ip	74.43	88.22	**91.97**	90.23
pl120	79.50	92.16	94.24	**94.70**
il80	71.25	89.53	90.64	**95.26**
p23	80.60	89.38	**89.13**	86.67
pr40	87.59	92.93	96.94	**98.89**
ph4.5	75.45	75.40	**82.66**	81.79
w59	57.04	60.95	**73.47**	71.65
p12	53.03	82.72	**92.85**	89.28
p3	63.03	79.58	**94.64**	90.55
w55	65.77	76.79	82.59	**91.49**
pm20	60.31	79.28	81.67	**81.88**
pl20	40.71	77.58	68.24	**84.44**
pg	81.54	87.54	**94.07**	93.84
pl70	50.30	81.16	**87.07**	86.40
pm55	65.50	78.02	86.20	**88.65**
il100	71.22	96.26	**99.42**	99.18
p27	70.30	90.28	91.44	**94.61**
w13	41.49	69.93	76.14	**86.40**
p19	68.46	83.37	**89.60**	87.59
ph4	53.66	60.24	72.40	**76.40**
ph5	40.71	58.20	69.24	**74.67**
wo	38.59	35.74	58.65	**59.61**
p6	48.94	62.22	66.09	**85.33**
pm30	55.12	80.57	85.43	**89.06**
w32	60.72	80.64	87.54	**91.87**

此外，本文设置了在IoU分别为0.5、0.6、0.7、0.75时，IYOLO算法与SSD300、YOLO v3、Faster R-CNN三种算法在检测精度上的对比，其对比结果如表4所示。

IYOLO算法与其他三种算法检测目标框对比效果如图9所示。

表3　四种算法整体检测性能对比

Tab. 3　Comparison of overall detection performance of four algorithms

算法	*mAP*/%(*IoU*=0.5)	FPS	模型大小/MB
SSD300	63.71	**42**	97.3
YOLO v3	82.40	7.5	236
Faster R-CNN	86.90	0.6	181
IYOLO	**89.00**	12	**48.1**

从表4中可以看出,随着IoU阈值的提高,IYOLO算法较其他三种算法在检测精度上的优势越发明显,其在高IoU阈值的情况下仍能保持高mAP值,而其他三种算法随着IoU阈值的增大mAP急剧下降。在IoU阈值为0.5时,IYOLO算法较SSD300、YOLO v3、Faster R-CNN算法的mAP值提升分别为25.29个百分点、6.6个百分点、2.1个百分点,而其在IoU阈值为0.75时的mAP值提升分别为30.84个百分点、13.52个百分点、11.39个百分点,即在高IoU阈值下提升越明显,这说明了IYOLO算法得到的预测框与真实目标框重合度更高,目标框定位更准确,这使得其应用场景更广阔。且从图9中可以看出,IYOLO算法比其他三种算法对目标框的定位更精确,并且解决了SSD300、YOLO v3算法中存在漏检、误检的问题。

表4　不同IoU阈值下检测精度的对比　　**单位:%**

Tab. 4　Comparison of detection accuracy under different IoU thresholds　　unit: %

算法	*IoU*=0.5	*IoU*=0.6	*IoU*=0.7	*IoU*=0.75
SSD300	63.71	60.85	55.67	46.82
YOLO v3	82.40	80.39	74.75	64.14
Faster R-CNN	86.90	82.84	74.36	66.27
IYOLO	89.00	88.17	84.21	77.66

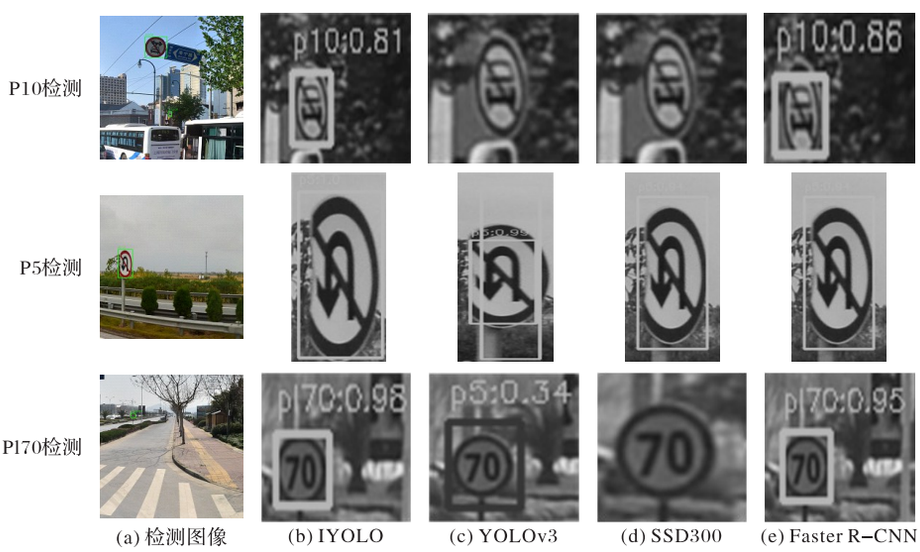

图9　不同算法检测交通标志效果对比图

Fig. 9　Comparison of different algorithms for detecting traffic signs

4　结语

本文提出了一种基于YOLO v3的改进算法,旨在解决交通标志识别任务中存在检测精度较低、算法模型参数量巨大以及实时性较差的问题。其中:引入深度可分离卷积实现了在不降低检测准确率的条件下极大地降低算法模型参数量的目标;在对目标框坐标回归损失的设计上采用了GIoU损失,这使得算法的检测精度大幅提升,且定位的目标框也更加精准;同时将Focal损失加入到置信度损失中,缓解了数据类别之间不均衡问题对算法模型的影响。通过对实验数据的分析可知,IYOLO算法在其参数只有YOLO v3算法的1/5时,mAP指标提高了6.6个百分点,FPS也提高了4.5。因此,该算法在模型大小、检测精度、检测速度上均优于YOLO v3算法。为提高检测精度,IYOLO算法采用的输入图片尺度为800 × 800,这使得其检测速度FPS只能达到12,与达到实时检测还存在一定距离,未来可以在检测速度上做进一步的提升以达到实时检测的效果。

参考文献 (References)

[1]　于硕. 交通标志识别技术综述[J]. 科技资讯, 2019, 17(6):15-16.(YU S. Overview of traffic sign recognition technology [J]. Science and Technology Information, 2019, 17(6): 15-16.)

[2]　FLEYEH H, BISWAS R, DAVAMI E. Traffic sign detection based on AdaBoost color segmentation and SVM classification [C]// Proceedings of the 2013 Eurocon. Piscataway: IEEE, 2013: 2005-2010.

[3]　CREUSEN I M, WIJNHOVEN R G J, HERBSCHLEB E, et al. Color exploitation in hog-based traffic sign detection [C]// Proceedings of the 2010 IEEE International Conference on Image Processing. Piscataway: IEEE, 2010: 2669-2672.

[4]　杜影丽,贾永红,韩静敏. 自然场景车载视频道路交通限速标志的检测与识别方法[J]. 测绘地理信息, 2018, 43(2): 32-34, 37.(DU Y L, JIA Y H, HAN J M. A detection and recognition method for traffic speed limit signs based on vehicle videos [J]. Journal of Geomatics, 2018, 43(2): 32-34, 37.)

[5]　李志军,崔利娟. 基于深度森林的交通标志识别方法研究[J]. 工业控制计算机, 2019, 32(5): 114-115, 120.(LI Z J, CUI L J. Research on traffic sign recognition algorithm based on deep forest [J]. Industrial Control Computer, 2019, 32(5): 114-115, 120.)

[6]　KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet

classification with deep convolutional neural networks [C]// Proceedings of the 25th International Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press, 2012, 1: 1097-1105.

[7] RUSSAKOVSKY O, DENG J, SU H, et al. ImageNet large scale visual recognition challenge[J]. International Journal of Computer Vision, 2015, 115(3): 211-252.

[8] REN S, HE K, 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.

[9] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2016: 779-788.

[10] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot multibox detector [C]// Proceedings of the 2016 European Conference on Computer Vision, LNCS 9905. Cham: Springer, 2016: 21-37.

[11] SERMANET P, LECUN Y. Traffic sign recognition with multi-scale convolutional networks [C]// Proceedings of the 2011 International Joint Conference on Neural Networks. Piscataway: IEEE, 2011: 2809-2813.

[12] STALLKAMP J, SCHLIPSING M, SALMEN J, et al. Man vs. computer: benchmarking machine learning algorithms for traffic sign recognition[J]. Neural Networks, 2012, 32: 323-332.

[13] ZHU Z, LIANG D, ZHANG S, et al. Traffic-sign detection and classification in the wild [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2016: 2110-2118.

[14] WANG G, XIONG Z, LIU D, et al. Cascade mask generation framework for fast small object detection[C]// Proceedings of the 2018 IEEE International Conference on Multimedia and Expo. Piscataway: IEEE, 2018: 1-6.

[15] REDMON J, FARHADI A. YOLO v3: an incremental improvement [EB/OL]. [2019-04-08]. https://arxiv. org/pdf/1804. 02767. pdf.

[16] CHOLLET F. Xception: deep learning with depthwise separable convolutions[C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 1800-1807.

[17] REZATOFIGHI H, TSOI N, GWAK J, et al. Generalized intersection over union: a metric and a loss for bounding box regression[C]// Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2019: 658-666.

[18] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 318-327.

[19] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 6517-6525.

[20] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2016: 770-778.

[21] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 936-944.

This work is partially supported by the Key Research and Development Program of Science and Technology Commission of Sichuan Province (2018SZ0040), the Major Project of New Generation Artificial Intelligence in Sichuan Province (2018GZDZX0036).

JIANG Jinhong, born in 1994, M. S. candidate. His research interests include deep learning, data mining.

BAO Shengli, born in 1973, Ph. D., senior engineer. His research interests include intelligent information processing, deep learning.

SHI Wenxu, born in 1995, M. S. candidate. His research interests include deep learning, intelligent algorithm.

WEI Zhenkun, born in 1995, Ph. D. candidate. His research interests include reinforcement learning, machine learning.