改进YOLO轻量化网络的口罩检测算法

王 兵[1],乐红霞[1],李文璟[2],张孟涵[3]
1.西南石油大学 计算机科学学院,成都 610500
2.中国电信股份有限公司 成都分公司,成都 610051
3.电子科技大学 信息与软件工程学院,成都 610500

摘 要:针对目前YOLO轻量网络在口罩佩戴检测任务中出现的特征提取不足和特征利用率不高的问题,提出了一种基于改进YOLOv4-tiny的轻量化网络算法。增加Max Module结构以获取更多目标的主要特征,提高检测准确率。提出自下而上的多尺度融合,结合低层信息丰富网络的特征层次,提高特征利用率。使用CIoU作为边框回归损失函数,加快模型收敛速度。相较于原算法,在公开数据集PASCAL VOC和口罩佩戴检测任务中,mAP分别提高4.9个百分点和3.3个百分点,检测速率分别达到74 frame/s和64 frame/s,满足口罩佩戴检测任务的准确率和实时性。

关键词:口罩佩戴检测;YOLOv4-tiny;Max Module结构;多尺度融合;CIoU
文献标志码:A **中图分类号:**TP391.4 doi:10.3778/j.issn.1002-8331.2009-0356

Mask Detection Algorithm Based on Improved YOLO Lightweight Network

WANG Bing[1], LE Hongxia[1], LI Wenjing[2], ZHANG Menghan[3]
1.School of Computer Science, Southwest Petroleum University, Chengdu 610500, China
2.Chengdu Branch of China Telecom Corporation Limited, Chengdu 610051, China
3.Scholl of Information and Software Engineering, University of Electronic Science and Technology, Chengdu 610500, China

Abstract: Aiming at the problem of insufficient feature extraction and low feature utilization in mask wearing detection tasks in the current YOLO lightweight network, a lightweight network algorithm based on improved YOLOv4-tiny is proposed. It increases the Max Module structure to obtain more main features of the target and improves the detection accuracy. A bottom-up multi-scale fusion is proposed, which combines low-level information to enrich the feature level of the network to improve feature utilization. It uses CIoU as the bounding box regression loss function to speed up model convergence. Compared with the original algorithm, in the public data set PASCAL VOC and mask wearing detection tasks, mAP is increased by 4.9 percentage points and 3.3 percentage points, respectively, and the detection rate reaches 74 frame/s and 64 frame/s, respectively, which meets the accuracy and real-time performance of mask wearing detection tasks.

Key words: mask wearing detection; YOLOv4-tiny; Max Module structure; multi-scale fusion; CIoU

一些大型病毒可以通过飞沫和其他介质传播,在公共场所佩戴口罩对于减少疾病的传播至关重要。在人群密集的区域(例如社区、超市和车站)通过人工方式检查口罩佩戴情况会消耗大量的人力资源且容易漏检,因此实现口罩佩戴检测算法具有重要现实意义。

近年来,由于深度学习的快速性、可扩展性和端到端学习等优点,一系列基于深度学习的目标检测算法被提出。YOLO系列[1-4]可能是实际应用中最流行的目标检测算法,在YOLO的基础上对网络进行改进也容易取得需要的效果。例如杨晋升等人研究对基于YOLO轻量化网络的交通标志检测[5]通过在骨干网络中使用深度可分离卷积更好地提取中小型目标。施辉等人对基于YOLO轻量化网络的安全帽佩戴检测[6]采用图像金字塔结构并构建安全帽数据集获取更具工业应用的模型。这些改进的算法在目标检测领域上有着重要的意义,但是这些方法仍没有很好地说明特征提取和利用率的问题。

YOLOv4为最近的开源目标检测网络,在速度和精度上与同时期目标检测网络有着明显的优势。YOLOv4采用具有深层结构的CSPDarknet53作为骨干网络,使用PANet[7]代替FPN[8]进行参数聚合,检测准确率高但对硬件配置要求较高,在小型硬件平台中检测速度慢,因

作者简介:王兵(1977—),男,硕士,副教授,CCF会员,研究领域为机器学习、模式识别、数据挖掘;乐红霞(1995—),女,硕士研究生,CCF会员,研究领域为机器学习、目标检测、模型压缩,E-mail:honsia803@163.com。

收稿日期:2020-09-21 **修回日期:**2020-12-25 **文章编号:**1002-8331(2021)08-0062-08

此在嵌入式平台上普遍使用YOLOv4tiny进行检测,虽然检测速度快但由于网络层次简单,特征提取能力不足,检测效果低于YOLOv4。

为解决以上不足,本文提出以YOLOv4-tiny为基础的一种改进YOLO轻量化网络的检测算法,主要贡献如下:

(1)针对YOLOv4-tiny网络层次较简单,无法提取更多主要特征的问题,提出了增加Max Module结构获取更多有效局部特征,提升检测准确率。

(2)针对YOLOv4-tiny较深层网络丢失浅层边缘信息的问题,构建自下而上的多尺度特征融合网络,提升数据利用效率。

(3)针对模型收敛速度慢的问题,采用CIoU作为边框回归损失函数,使预测框更接近真实框,加快模型收敛速度。

1　相关工作

1.1　目标检测算法

采用深度学习进行目标检测算法大致分为两类:一种是使用区域候选网络(RPN)来提取候选目标信息的两阶段检测算法,如AlexNet[9]、R-CNN[10]、Faster R-CNN[11]、Mask R-CNN[12]等。两阶段检测器主要由三部分组成:骨干网、区域建议模块和检测头。首先,区域建议模块用于区域建议。它可以生成可能包含感兴趣对象的许多候选区域。通过判断前景和背景,它使用边界框回归来校正锚点的位置。通过区域推荐网络[13]、区域改进方法[14-15]、区域建议深度特征计算方法[16-17]和骨干网络结构[18]可以生成高质量区域,但是,使用这些区域来推断两阶段目标检测器需要消耗大量计算资源,并且需要依赖更高的硬件平台。于是端到端的YOLO、SSD[19]和Retinanet[20]等一阶段目标检测器被提出,该目标检测算法通过获取输入图像并学习相对于预定义锚点的类别概率和边界框坐标,直接将目标检测视为回归问题。基于一阶段目标检测器的Anchor-free的网络模型,如CenterNet[21]和CornerNet[22]也取得巨大发展,这些模型不再使用锚框机制而是直接使用预测模型输出的中心点或者边角点与真实检测框的偏移进行回归。目前,越来越多研究者也开始重视轻量网络结构[23]以及改进特征图在网络中的特征提取和融合算法[24]。NAS-FPN[25]针对不同尺度特征图的神经架构来搜索最优的跨尺度特征网络拓扑结构,但是需要多GPU花费大量时间搜索最优结构。BiFPN[26]将简化的横向扩展并在各层融合,以实现最大效率特征图的融合效果,但是这种横向扩展多层次的融合对原始图的位置描述信息可能有较大的损失。

1.2　YOLOv4-tiny网络

一阶段检测算法通常具有更快的检测速度,它的检测速度和检测精度相对平衡,但由于YOLO的深层次结构和巨大的参数量,各种轻量级网络(YOLO-tiny[27-28]、YOLO Nano[29]等)被提出。YOLOv4-tiny是最新的YOLO轻量级网络,相较于之前的轻量网络,在mAP和检测速率上都有巨大的提升。其骨干网络主要包括下采样CBL结构和CSP结构,下采样CBL结构中,每个卷积核大小为3×3,步长为2,主要对图像进行下采样处理。CSP结构[30]将基础层的特征映射划分为两部分,通过跨层连接将它们合并,增强卷积神经网络的学习能力,在减少了计算量的同时可以保证准确率。跨层连接与残差网络的结果类似,这样有两个好处:(1)形成特征映射,实现特征的重用以获得更多的语义信息,提高检测准确率;(2)降低计算瓶颈,减少内存开销。其具体网络结构参数如图1所示。

Type	Filters	Size/Stride	Output	
Convolutional	32	3×3/2	208×208×32	下采样CBL
Convolutional	64	3×3/2	104×104×64	
Convolutional	64	3×3/1	104×104×64	
Residual 2			104×104×64	
Convolutional	32	3×3/1	104×104×64	
Convolutional	32	3×3/1	104×104×64	CSP结构
Residual 5,4			104×104×64	
Convolutional	64	1×1/1	104×104×64	
Residual 2,7			104×104×128	
Max		2×2/2	52×52×128	
Convolutional	128	3×3/1	52×52×128	
Residual 10			52×52×64	
Convolutional	64	3×3/1	52×52×64	
Convolutional	64	3×3/1	52×52×64	CSP结构
Residual 13,12			52×52×128	
Convolutional	128	1×1/1	52×52×128	
Residual 10,15			52×52×256	
Max		2×2/2	52×52×128	
Convolutional	256	3×3/1	26×26×256	
Residual 18			26×26×128	
Convolutional	128	3×3/1	26×26×128	
Convolutional	128	3×3/1	26×26×128	CSP结构
Residual 21,20			26×26×256	
Convolutional	256	1×1/1	26×26×256	
Residual 18,23			26×26×512	
Max		2×2/2	13×13×512	
Convolutional	512	3×3/1	13×13×512	下采样CBL
Convolutional	256	1×1/1	13×13×256	
Convolutional	512	3×3/1	13×13×512	
Convolutional	255	1×1/1	13×13×255	
yolo				
Residual 27			13×13×256	
Convolutional	128	1×1/1	13×13×128	
Upsample		2×	26×26×128	
Residual 33×23			26×26×384	
Convolutional	256	3×3/1	26×26×256	
Convolutional	255	1×1/1	26×26×255	
yolo				

图1　YOLOv4-tiny网络结构图

图1中的Convolutional由一个卷积层、批标准化BN层[31]以及LeakyRelu激活函数构成。BN层可降低不同样本间值域的差异性,避免梯度消失和梯度爆炸的问题,同时减少参数或其初始值尺度的依赖性,提高网络范化能力。Leaky ReLU给所有负值赋予一个非零斜率,避免神经元的失活现象。

在YOLO中,将整个图片划分为$S\times S$个格子,每个格子作为先验锚框的局部坐标,在格子内训练的网络预测的坐标偏移量、物体置信度和类别置信度对每个锚框分别进行拟合,最后经过非极大值抑制筛选后得到检测的边界框坐标和类别。其损失函数如式(1)所示:

$$\begin{cases} Loss = L_{xywh} + L_{\text{conficence}} + L_{\text{classes}} \\ L_{xywh} = \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} [(x_i - \tilde{x}_i)^2 + (y_i - \tilde{y}_i)^2] + \\ \qquad \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} [(w_i - \tilde{w}_i)^2 + (h_i - \tilde{h}_i)^2] \\ L_{\text{confidence}} = -\lambda_{\text{obj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} [\tilde{c}_i \lg C_i + (1 - \tilde{c}_i) \lg(1 - C_i)] - \\ \qquad \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{noobj}} [\tilde{c}_i \lg C_i + (1 - \tilde{c}_i) \lg(1 - C_i)] \\ L_{\text{classes}} = \sum_{i=0}^{S^2} I_{ij}^{\text{obj}} \sum_{c \in classes} [\tilde{p}_i(c) \lg P_i(c) + \\ \qquad (1 - \tilde{p}_i(c)) \lg(1 - P_i(c))] \end{cases} \quad (1)$$

其中，L_{xywh} 为预测框与真实框的中心点和宽高误差之和，λ_{coord} 为坐标系数；$L_{\text{confidence}}$ 为目标置信度误差，分为有物体和无物体的两项置信度误差，λ_{obj} 和 λ_{noobj} 分别为有物体和无物体的置信度系数；L_{classes} 为目标分类损失，I_{ij}^{obj} 表示为第 i 个网格的第 j 个锚框的匹配情况。

2　网络模型

本文提出了一种改进YOLOv4-tiny的新网络结构（以下简称YOLOv4-tiny Max）。低层特征可以提供更加准确的位置信息，使用最大池化层能降低图像尺寸并提取关键信息，但由于最大池化层只和前层部分神经元连接，一个池化神经元没有权重，仅通过最大聚合函数对输入特价进行聚合可能会丢失重要的位置信息，因此使用大小为3×3，步长为2的卷积层代替网络结构中的最大池化操作，带参数的卷积层会保留更多特征图信息。经过一系列下采样CBL结构和多次卷积操作会使得深层网络的目标定位存在误差，因此构建一个MaxModule结构提取中小型目标的主要特征，其中一条分支网络再经过自下而上的多尺度特征融合结构，最终获得两种不同尺度的检测头输出，其网络结构如图2所示。

2.1　Max Module

He[32]等人的研究表明卷积神经网络全连接层的输入必须是固定的特征向量，直接将图片进行拉伸会导致图片信息的丢失从而影响识别的精度。SPP作为一个优秀的网络组件，其不需指定输入图像的尺寸或比例，就能够产生固定大小的特征表示再送进全连接层，这样就可以很好地解决该问题。

基于以上研究，提出Max Module结构，添加在多尺度融合过程中以获得更多有效局部特征信息，Max Module结构如图3所示。

使用大（13×13）、中（9×9）、小（5×5）三种不同尺度的最大池化窗口分别作用于传入的上层卷积特征，选取特征图区域的最大值作为该区域池化后的值，为保持特征图大小不变，设置步长为1，最后把输入特征图和经过最大池化后的局部尺寸特征图进行通道融合再传入瓶颈层，增加网络深度的同时又保留前层特征，提升网络性能。图4从左往右依次是原图、采用Max Module结构和采用对应层数卷积操作后的特征图。

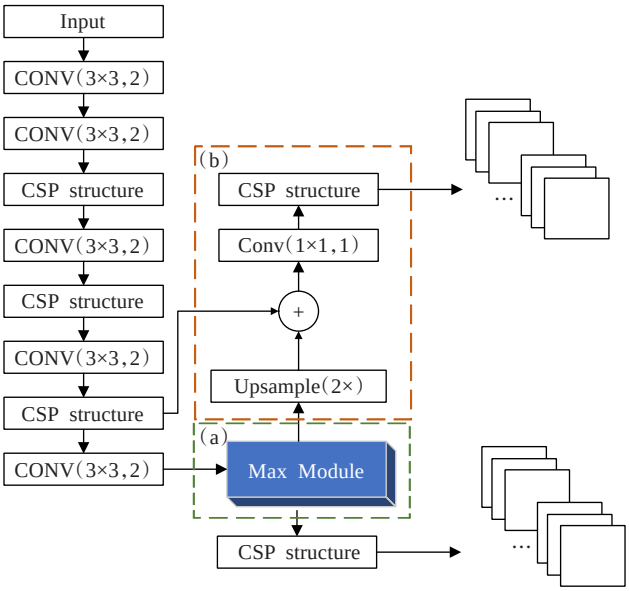


图2　改进的网络结构图

（(a)Max Module结构；(b)自下而上的多尺度特征融合）

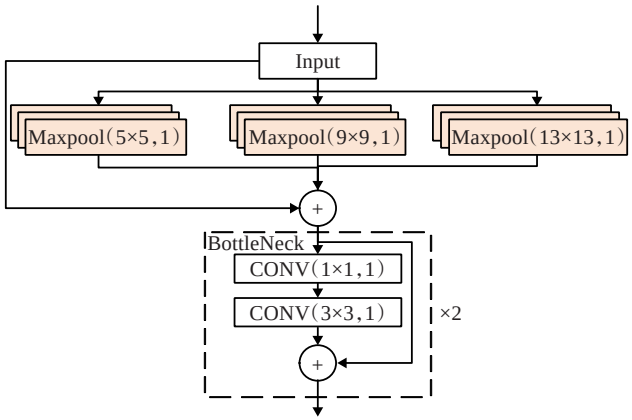


图3　Max Module结构

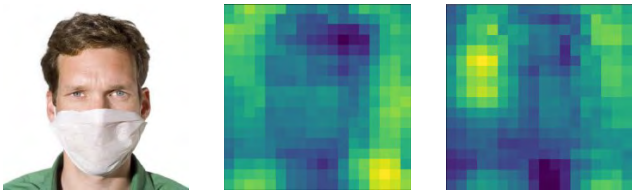


图4　特征图对比

从图4可以看出，采用Max Module结构的特征图边缘信息和主要特征更清晰丰富，有助于提升检测准确率。Max Module中类似空间金字塔的结构不需要对输入特征图进行等分就能提取多尺度的局部特征图像，瓶颈结构使得网络仍然能有效进行反向传播。

2.2　自下而上的多尺度特征融合结构

浅层特征到深层特征到传递路径较长,其边缘信息和定位信息容易丢失,导致数据利用率低、检测精度不理想等问题,为充分使用特征信息,对输出较大检测头的分支网络使用自下而上对多尺度特征融合结构。

不同于原始YOLOv4-tiny直接使用卷积和上采样操作,该结构(如图2(b))首先进行二倍上采样,与主干网络第三个CSP结构进行特征图融合后再经过一个1×1卷积传入CSP结构,其前馈传递方程如式(2)所示:

$$\begin{cases} X_k = W_k * [x_0, x_1, \cdots, x_{k-1}] \\ X_T = W_T * [x_0, x_1, \cdots, x_k] \\ X_U = W_U * [x_0, x_T] \end{cases} \tag{2}$$

其中,$*$ 表示卷积算子,$[x_0, x_1, \cdots]$ 表示连接 $(x_0, x_1, \cdots)$ 的各个分量,X_i 和 W_i 分别是第 i 个连接的输出和权重。权重更新方式如式(3)所示:

$$\begin{cases} W_K' = f(W_K, g_0, g_1, g_2, \cdots, g_{k-1}) \\ W_T' = f(W_T, g_0, g_1, g_2, \cdots, g_k) \\ W_U' = f(W_U, g_0, g_T) \end{cases} \tag{3}$$

其中,f 是权重更新的函数,g_i 表示传播到第 i 个连接的梯度。可以看出更新的权重信息 W_T' 和 W_U' 是由不同梯度信息分开整合的,这样既保留了特征重复使用的特点,又通过截断梯度防止了过多的重复梯度信息,提升数据利用效率。类似残差结构的特征图融合可以让网络获取到深层结构信息也不会导致梯度消失,同时又能传递浅层的强定位信息和边缘特征,在不同图像细粒度上聚合并形成更全面的图像特征,提高目标检测效果。

2.3　CIoU在改进网络中的使用

通过骨干网络和特征融合结构后,最终产生两个检测头,分别负责检测不同尺度的目标。每个检测头中的特征图被分配了三个不同的锚框,以预测由四个边框坐标生成预测框。在以前的工作中,IoU[33]用于测量所生成的预测框与真实框之间的重叠率,计算公式如式(4)所示:

$$IoU_{(A,B)} = \frac{A \cap B}{A \cup B} \tag{4}$$

其中,A 为预测框的面积,B 为真实框的面积,$IoU_{(A,B)}$ 为 A 与 B 的交并比,也就是预测框的面积与真实框的面积的交集除以其并集。由公式可以看出,对于两个IoU相同的物体,无法表示它们的对齐方式,若预测框和真实框没有重叠(没有交集),IoU始终为0,无法优化,为避免这些问题,本文采用CIoU[34]作为边框回归损失函数,损失函数如式(5)所示:

$$L_{\mathrm{CIoU}} = 1 - IoU + \frac{\rho(b, b^{gt})}{c^2} + \alpha v \tag{5}$$

其中,α 是用于做协调比例的参数,v 是用来衡量长宽比一致性的参数,b 和 b^{gt} 分别表示预测框和真实框的中心点,$\rho()$ 表示欧式距离,c 表示预测框和真实框的最小外界矩形的对角线距离。α 和 v 的计算方法如下:

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{6}$$

$$v = \frac{4}{\pi^2}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h})^2 \tag{7}$$

CIoU直接最小化预测框与目标框之间的归一化距离以达到更快的收敛速度,且对尺度具有不变形,使回归在与目标框有重叠甚至包含时更准确、更快。

从表1和图5可知,CIoU在损失收敛效果和mAP上均优于IoU,因此使用CIoU作为边框回归损失函数对网络性能的提升是有很大意义的。

表1　不同边框回归损失函数方法对比

方法	mAP/%
IoU	79.80
CIoU	82.36

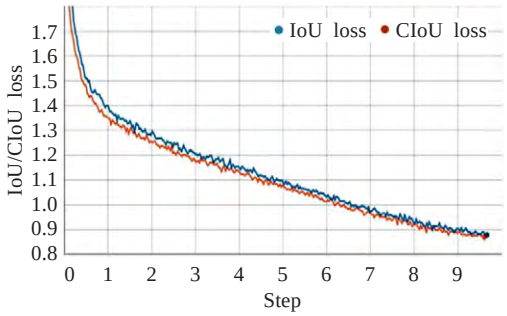


图5　损失收敛对比

3　实验与结果分析

根据默认配置训练本文算法,训练期间的初始学习率为0.001,衰减系数为0.000 5,最小批量大小为64,采用半精度加速训练。在平台方面,操作系统为Ubuntu 64位,CPU为 Intel i7-7700 4.2 GHz;内存大小为32 GB;GPU采用NVIDIA GeForce GTX1080ti*4的32 GB显卡;编译环境为Pycharm/python语言。分别在公开数据集PASCAL VOC和自制口罩数据集对该算法进行实验对比与分析。

3.1　口罩数据集的制作

口罩数据集来自于公开数据集MAFA和Wilder Face中所有佩戴口罩的数据集和部分未佩戴口罩的人脸数据集。对数据进行筛选并删除标签和特征不对应的图片后,总共保留了6 757张图像,包括3 893张脸部和2 864张被口罩遮挡的脸部,使用两个预定义类别:face(脸部)和face_mask(被口罩遮挡的脸部)标记数据集中的图像。部分图像如图6所示,其中,A类对应标签为face_mask,B类对应标签为face。

3.1.1　口罩数据集的预处理

仿照PASCAL VOC格式处理口罩数据集,将标注信息进行归一化处理,归一化公式如式(8)所示:

图6　部分Wilder MAFA Face数据集示例

$$\begin{cases} x=\dfrac{x_{\max}+x_{\min}}{2width}, y=\dfrac{y_{\max}+y_{\min}}{2height} \\ w=\dfrac{x_{\max}-x_{\min}}{width}, h=\dfrac{y_{\max}-y_{\min}}{height} \end{cases} \tag{8}$$

其中，$(weight, height)$ 为原始图片的宽度和高度，$(x_{\min}, y_{\min})$、$(x_{\max}, y_{\max})$ 分别为原始样本真实边界框的左上角位置信息和右下角位置信息，(x,y)、(w,h) 分别为目标进行归一化后的中心点坐标和宽高。图片归一化后，边界框信息总共包含5个参数：即 (x,y,w,h) 和类别对应的标签编号。

3.1.2　重置口罩数据集的锚框

在基于锚框的目标检测网络中，锚框设置的合理性对于最终模型的性能至关重要，若锚框的大小与被测物体的尺度不一致，那么锚框的正样本数可能会非常少，这将导致大量漏检和误检情况。大部分目标检测网络使用默认的通用锚框参数以适应通用的公开数据集，例如 YOLOv4-tiny 使用的6组适用于通用场景的通用锚框参数：[(10,14),(23,27),(37,58),(81,82),(135,169),(344,319)]。为了避免在口罩数据集上使用通用锚框造成正负样本的不平衡问题，本文使用 k-means++聚类算法[23]根据聚类中心和数据框分布重新生成6组新的锚框参数[(12,16),(23,30),(41,53),(70,94),(124,168),(251,338)]用于本算法的口罩算法训练。数据的聚类中心分布结果如图7所示，其中，灰点是对象框大小的分布，红色三角形是聚类的结果。数据框分布统计如图8所示。

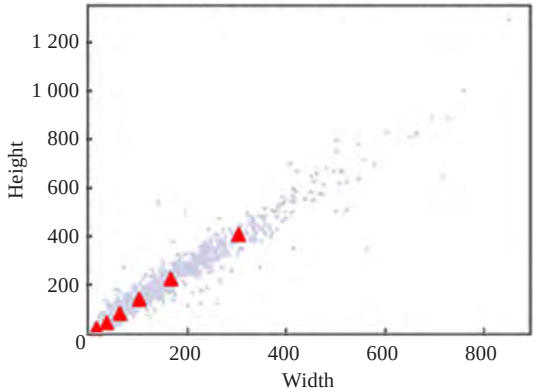


图7　聚类中心分布结果

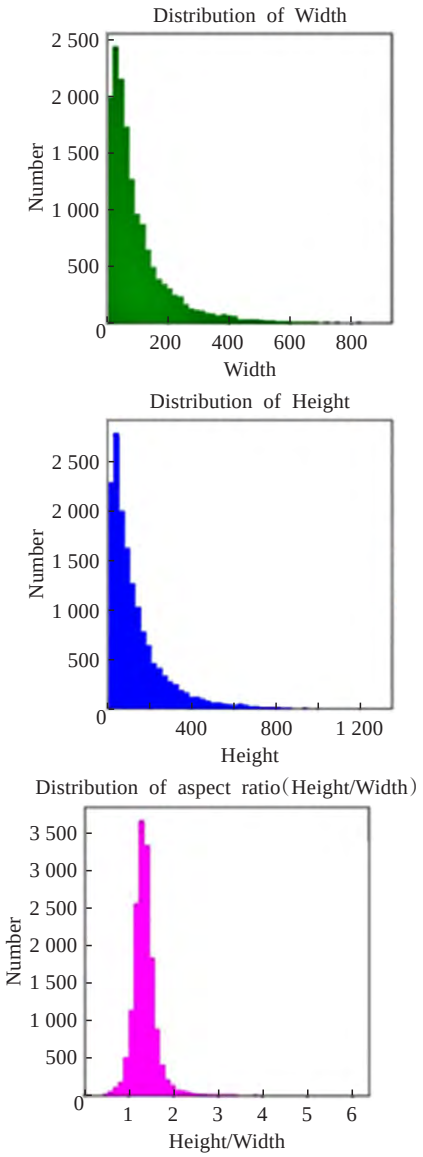


图8　数据框的分布统计

可以看到人脸高宽比例多数都在1.4:1左右。所以，对于口罩数据集，可以将锚框比例设置为1:1、1.4:1、1.7:1，而没必要设置为通用锚框比例。

3.2　Mosaic数据增强

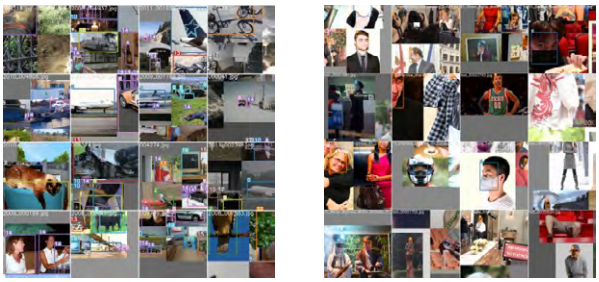
随机改变训练样本可以降低模型对物体出现位置的依赖，提高模型的泛化能力，因此本文算法在训练过程中对训练数据进行Mosaic数据增强训练技巧，即随机读取4张训练图像，进行翻转和旋转等操作后，按一定比例组合成1张图片。部分训练图片如图9所示。

3.3　PASCAL VOC数据集实验结果分析

选取VOC2007训练验证集和VOC2012训练验证集作为训练数据(总共包含16 551张图片)，VOC2007测试集作为测试数据(总共包含4 952张图片)。将本文算法与Faster RCNN、SSDLite、SSD、YOLOv3、YOLOv3-tiny、YOLOv4和YOLOv4-tiny进行对比，所有算法在PASCAL VOC数据集中均采用通用锚框比例，实验对比结果如表2所示。

表2　不同算法结果对比

算法	基础网络	mAP/%	检测速率/(frame·s^{-1})	模型体积/MB
Faster RCNN	VGG16	73.2	28	528
SSD	MobileNet	77.4	51	101
YOLOv3	Darknet53	78.3	59	248
YOLOv3-tiny	—	57.1	133	36
YOLOv4	CSPDarknet53	82.3	64	258
YOLOv4-tiny	—	65.3	99	24
本文算法	—	70.2	74	14



PASCAL VOC数据集　　Wilder MAFA Face数据集

图9　经Mosaic处理的训练数据集示例

由表2可知,大型网络检测准确率高,但是检测速度较慢,轻量网络检测速度快但是检测准确率较低。YOLOv4-tiny在轻量级网络中检测速度和检测准确率较为均衡,但由于网络结构简单,存在特征提取能力不足等问题。本文所提算法虽然mAP不及表2中的大型网络,但模型体积最小,更适合部署于移动端。在模型体积相差不大的同等轻量级网络YOLOv3-tiny和YOLOv4-tiny中,mAP分别提高13.1个百分点和4.9个百分点,其主要原因是Max Module能更好提取图像特征,自下而上的多尺度融合增强模型对特征的利用率,提高准确率,使用CIoU更好地描述预测框和真实框的距离,加快模型收敛速度,同时对训练集采取Mosaic处理,丰富检测物体的背景,获得更好的泛化能力。检测速率略低是由于随着mAP的提高,会检测出更多目标框,因此时间开销增加,但该检测速度仍符合实际检测场景的实时性要求。

3.4　口罩数据集实验结果分析

按7:3随机将口罩数据集划分为训练集和测试集,将本文算法与同等轻量级网络YOLOv3-tiny和YOLOv4-tiny进行对比,为进一步说明Max Module结构的有效性,在YOLOv3-tiny对应的检测头网络前相同位置加入Max Module结构(以下简称YOLOv3-tiny Max)。所有算法在口罩数据集中均采用k-means++聚类生成的锚框比。以平均精度均值(mAP)、每秒识别帧数、精确率(Precision)和召回率(Recall)作为评价指标,不同算法的实验结果如图10和表3所示。

从图10和表3可以看出,本文算法在mAP和检测速率上表现最好。YOLOv3-tiny采用卷积层和最大池化层组成的7层网络较浅,无法提取更多特征,故mAP最低,从精确率和召回率可知,YOLOv3-tiny中,正负样本不平衡,误检情况可能较高,增加Max Module结构的YOLOv3-tiny Max相较于原算法,mAP提升明显,但由于未改变基础网络和特征融合方式,正负样本优化情况仍有待改善。相较于YOLOv4-tiny,检测速率相差不大,但mAP提高3.3个百分点。这是由于改进的算法结构增加Max Module更好地提取主要特征,采用自下而上的多尺度特征融合,提升浅层网络边缘信息利用率,使低层定位信号增强整个特征层次。因此,从实验对比结果可知,对于实际的口罩佩戴检测场景而言,本文提出的改进点是有效的,改进的YOLO网络同时兼顾了检测准确率和检测速率,能较好完成口罩佩戴检测任务。

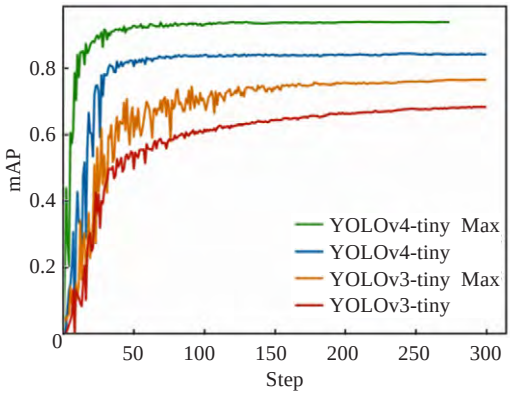


图10　轻量算法在口罩数据训练集的mAP对比

表3　不同轻量网络的口罩检测结果对比

算法	类别	R	P	mAP/%	检测速率/(frame·s^{-1})
YOLOv3-tiny	face	88.6	26.9	78.3	53
	face_mask	90.6	32.5	86.2	
	all	89.6	29.7	82.3	
YOLOv3-tiny Max	face	93.7	27.1	84.8	49
	face_mask	93.0	21.1	87.3	
	all	93.4	24.1	86.0	
YOLOv4-tiny	face	92.3	84.3	92.4	63
	face_mask	92.6	81.4	93.0	
	all	92.4	82.8	92.7	
本文算法	face	92.4	84.7	96.7	64
	face_mask	93.0	81.2	95.8	
	all	92.7	83.0	96.2	

图11 各种算法的检测效果对比图

为了更加直观地说明不同检测算法之间的区别,选取了一些检测图像进行对比分析,从左到右依次是:原始图像、本文算法检测结果、YOLOv4-tiny检测结果、YOLOv3-tiny Max检测结果和YOLOv3-tiny检测结果。

从图11可以看出,YOLOv3-tiny漏检情况严重且检测框位置偏差严重,YOLOv3-tiny Max和YOLOv4-tiny检测效果相差不大,但均未识别出远处的人物,改进YOLO轻量化网络则弥补了这一缺陷。因此,在以上轻量网络算法中,本文提出的改进YOLO轻量化网络方法更适合口罩佩戴检查任务。

4 结束语

本文提出了一种改进YOLO轻量化网络的口罩检测算法。提出Max Module结构能获取更主要的特征,自下而上的特征融合结构保留浅层网络的边缘信息和定位信息,提升特征利用率,引用CIoU预测框与真实框的位置,加快损失收敛速度,构建口罩佩戴数据集并使用k-means++重构锚框比例,采用Mosaic方法处理训练集,提高模型在实际检测场景中的泛化能力,使模型更加适用于口罩佩戴检测场景。实验结果表明,相比于原算法YOLOv4-tiny,在VOC数据集和口罩检测任务中,mAP分别提升4.9个百分点和3.3个百分点,检测速率分别达到74 frame/s和64 frame/s,其检测准确率和检测速率更为均衡,适用于口罩佩戴检测任务。但是在其他检测场景和通用场景中,检测准确率仍不及大型检测网络,如何使模型适用于更多检测场景,这依然是一个有待解决的问题。

参考文献:

[1] BOCHKOVSKIY A,WANG C Y,LIAO H Y M.Yolov4:optimal speed and accuracy of object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2020.

[2] REDMON J,FARHADI A.Yolov3:an incremental improvement[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2018.

[3] REDMON J,FARHADI A.Yolo9000:better,faster,stronger[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2016.

[4] REDMON J,DIVVALA S,GIRSHICK R.You only look once:unified,real-time object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2016.

[5] 杨晋生,杨雁南,李天骄.基于深度可分离卷积的交通标志识别算法[J].液晶与显示,2019,34(12):1191-1201.

[6] 施辉,陈先桥,杨英.改进YOLO v3的安全帽佩戴检测方法[J].计算机工程与应用,2019,55(11):213-220.

[7] LIU Shu,QI Lu,QIN Haifang,et al.Path aggregation network for instance segmentation[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2018.

[8] LIN T Y,DOLLÁR P,GIRSHICK R,et al.Feature pyramid networks for object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2016.

[9] KRIZHEVSKY A,SUTSKEVER I,HINTON G.Imagenet classification with deep convolutional neural networks[C]//Advances in Neural Information Processing Systems,2012.

[10] GIRSHICK R,DONAHUE J,DARRELL T.Rich feature hierarchies for accurate object detection and semantic segmentation[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2014.

[11] 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.

[12] HE Kaiming,GKIOXARI G,DOLLÁR P,et al.Mask R-CNN[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2018.

[13] KONG T,YAO A,CHEN Y.Hypernet:towards accurate region proposal generation and joint object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2016.

[14] CHO M A,CHUNG T Y,LEE H.N-RPN:hard example learning for region proposal networks[C]//IEEE International Conference on Image Processing(ICIP),2019.

[15] RAO Y,CHENG Y,XUE J.FPSiamRPN:feature pyramid siamese network with region proposal network for target tracking[J].IEEE Access,2020,8:176158-176169.

[16] ZHONG Qiaoyong,LI Chao,ZHANG Yingying,et al. Cascade region proposal and global context for deep object detection[J].Neurocomputing,2020,395:170-177.

[17] CAI C,CHEN L,ZHANG X,et al.End-to-end optimized ROI image compression[J].IEEE Transactions on Image Processing,2019,29:3442-3457.

[18] SEFERBEKOV S,IGLOVIKOV V,BUSLAEV A,et al. Feature pyramid network for multi-class land segmentation[C]//IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops(CVPRW),2018.

[19] LIU W,ANGUELOV D,ERHAN D.SSD:single shot multibox detector[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2016.

[20] PEI D,JING M,LIU H.A fast RetinaNet fusion framework for multi-spectral pedestrian detection[J].Infrared Physics & Technology,2020,105:103178.

[21] DUAN K,BAI S,XIE L.Centernet:keypoint triplets for object detection[C]//Proceedings of the IEEE International Conference on Computer Vision,2019:6569-6578.

[22] LAW H,DENG J.Cornernet:detecting objects as paired keypoints[C]//Proceedings of the European Conference on Computer Vision(ECCV),2018:734-750.

[23] HOWARD A,SANDLER M,CHEN B,et al.Searching for mobileNetV3[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019.

[24] XIONG S Q,WU X H,CHEN H G,et al.Bi-directional skip connection feature pyramid network and sub-pixel convolution for high-quality object detection[J].Neurocomputing,2021,440:185-196.

[25] GHIASI G,LIN T Y,PANG R,et al.NAS-FPN:learning scalable feature pyramid architecture for object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019.

[26] HAN Kai,WANG Yunhe,TIAN Qi,et al.GhostNet:more features from cheap operations[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2020.

[27] Wai Y J,YUSSOF Z B M,SALIM S I B,et al.Fixed-point implementation of Tiny-Yolo-v2 using OpenCL on FPGA[J].International Journal of Advanced Computer Science & Applications,2018,9(10):506-512.

[28] ZHANG Yi,SHEN Yongliang,ZHANG Jun.An improved tiny-yolov3 pedestrian detection algorithm[J].Optik,2019,183:17-23.

[29] WONG A,FAMUORI M,SHAFIEE M J,et al.YOLO Nano:a highly compact you only look once convolutional neural network for object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019.

[30] WANG C Y,LIAO H Y M,WU Y H,et al.CSPNet:a new backbone that can enhance learning capability of CNN[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019.

[31] IOFFE S,SZEGEDY C.Batch normalization:accelerating deep network training by reducing internal covariate shift[C]//The 32nd International Conference on Machine Learning,2015:448-456.

[32] HE Kaiming,ZHANG Xiangyu,REN Shaoqing.Spatial pyramid pooling in deep convolutional networks for visual recognition[J].IEEE Transactions on Pattern Analysis and Machine Intelligence,2015,37(9):1904-1916.

[33] AHMED F,TARLOW D,BATRA D.Optimizing expected intersection-over-unionwith candidate-constrained CRFs[C]//IEEE International Conference on Computer Vision,2016.

[34] ZHENG Zhaohui,WANG Ping,LIU Wei,et al.Distance-IOU loss:faster and better learning for bounding box regression[C]//IEEE Conference on Computer Vision and Pattern Recognition(CVPR),2019.