文章编号:1007-2780(2021)11-1525-10

基于 YOLOv4-tiny 改进的轻量级口罩检测算法

朱 杰[1,2],王建立[1*],王 斌[1]
(1. 中国科学院 长春光学精密机械与物理研究所,吉林 长春 130033;
2. 中国科学院大学,北京 100049)

摘要:新冠疫情防控期间,为防止病毒扩散,需要在机场、车站等公共场所对人流的口罩佩戴情况实施管控。为了高效地监测人流的口罩佩戴情况,提出了一种基于 YOLOv4-tiny 改进的轻量级口罩检测算法。在 YOLOv4-tiny 的骨干网络之后引入空间金字塔池化结构,对输入特征层进行多尺度池化并融合,同时大幅增强网络的感受野。结合路径聚合网络,分两条路径将不同尺度的特征层相互融合、反复增强,以提升特征层对目标的表达能力。使用标签平滑策略优化了网络损失函数,以抑制网络训练过程中的过拟合问题。实验结果表明,该算法在口罩目标和人脸目标上的检测精度分别达到了 94.7%和 85.7%,相比于 YOLOv4-tiny 分别提高了 4.3%和 7.1%,实时检测速度达到 76.8 FPS(on GeForce GTX 1050Ti),满足了多种场景下口罩检测任务的检测精度与实时性要求。

关 键 词:口罩检测;YOLOv4-tiny;空间金字塔池化;特征融合
中图分类号:TP391.41 **文献标识码**:A **doi**:10.37188/CJLCD.2021-0059

Lightweight mask detection algorithm based on improved YOLOv4-tiny

ZHU Jie[1,2], WANG Jian-li[1*], WANG Bin[1]

(1. *Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun* 130033, *China*;
2. *University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: During the period of 2019-nCoV controlling, to prevent the spread of the virus, it is necessary to regulate the coverage of mask wearing in densely populated places such as airports and stations. In order to effectively monitor the coverage of mask wearing of crowd, this paper proposes a lightweight mask detection algorithm based on improved YOLOv4-tiny. Following the backbone network of YOLOv4-tiny, a spatial pyramid pooling structure is introduced to pool and fuse the input features at multi-scale, which makes the receptive field of the network enhanced. Then, combined with the path aggregation network, multi-scale features are fused and enhanced repeatedly in two paths to improve the expressive ability of feature maps. Finally, label smoothing is utilized to optimize the loss function for modifying the over-fitting problem in the training process. The experimental re-

收稿日期:2021-03-01;**修订日期**:2021-03-26.
基金项目:国家自然科学基金(No. 11703024)
Supported by National Natural Science Foundation of China(No. 11703024)
*通信联系人,E-mail:wangjianli@ciomp.ac.cn

sults show that the proposed algorithm achieves 94.7% AP and 85.7% AP on mask target and face target respectively (at real-time speed of 76.8 FPS on GeForce GTX 1050ti), which is 4.3% and 7.1% higher than that of YOLOv4-tiny. The proposed algorithm meets the accuracy and real-time requirements of mask detection tasks in various scenes.

Key words: mask detection; YOLOv4-tiny; Spatial Pyramid Pooling; feature fusion

1 引 言

新冠病毒席卷全球的当下,在机场、车站等公共场所佩戴口罩可以有效降低病毒的传播概率,防止疫情的扩散。目前国内外的公共场所对人流的口罩佩戴情况的监测手段主要以人工监督为主,广播呼吁为辅,这种手段效率低下的同时也浪费了大量公共资源。而近几年人工智能技术日新月异,尤其是目标检测领域大量优秀算法的涌现,例如 YOLO[1-5] 系列为代表的单阶段算法以及 RCNN[6-8] 系列为代表的两阶段算法,让我们可以通过计算机搭配监控摄像头就能实现人流口罩佩戴情况的自动化监测。

目前,专门应用于口罩检测的数据集和算法都相对较少。Cabani 等人[9] 提出了一个包含 13 万张图像的大规模口罩检测数据集,但是该数据集的图像目标和人物背景都相对单一,对现实场景的覆盖面严重不足。Wang 等人[10] 提出了一个口罩人脸数据集 (Real-World Masked Face Dataset,RMFD),不过该数据集的场景覆盖面依旧不充分,更重要的是公开的数据标注并不完善,难以直接应用于口罩检测任务。牛作东等人[11] 在人脸检测算法 RetinaFace 中引入注意力进行机制优化,然后应用于口罩检测任务,取得了不错的平均精度均值 (mean Average Precision, mAP),但是算法的每秒传输帧数 (Frames Per Second,FPS) 相对较低,很难满足口罩检测任务的实时性要求。王艺皓等人[12] 在 YOLOv3[3] 算法中引入空间金字塔结构改进后用于口罩检测任务,实现了 mAP 和 FPS 指标的小幅提升,不过算法数据集的背景相对单一,难以扩展到复杂的多场景口罩检测任务中去,而且实时检测速度依旧不理想。

YOLOv4-tiny 由 Wang 等人于 2020 年 6 月发布,在 COCO[13] 数据集上以 443 FPS(on GeForce RTX 2080Ti) 的实时检测速度实现了 42.0% 的检测精度[5]。本文对 YOLOv4-tiny 进行改进和优化,以应用于复杂场景下的口罩检测任务。保持 YOLOv4-tiny 的骨干网络不变,在其后引入空间金字塔池化[14] (Spatial Pyramid Pooling,SPP) 模块,对输入特征层进行多尺度池化并融合,同时大幅增强网络的感受野。利用路径聚合网络[15] (Path Aggregation Network,PAN) 取代 YOLOv4-tiny 的特征金字塔网络[16] (Feature Pyramid Networks,FPN),分两条路径将输入特征层上采样和下采样而来的信息相互融合,以增强特征层对目标的表达能力。使用标签平滑[17] (Label smoothing) 策略优化了网络损失函数,结合 Mosaic 数据增强[4] 和学习率余弦衰退思路,提高了模型的训练效率。实验结果表明,本文算法在普通性能的硬件 (GeForce GTX 1050Ti) 上达到了 76.8 FPS 的实时检测速度,在口罩目标和人脸目标上的检测精度分别达到了 94.7% 和 85.7%,相比 YOLOv4-tiny 分别提高了 4.3% 和 7.1%,满足了多种复杂场景下口罩检测任务的检测精度与实时性要求。

2 YOLOv4-tiny 算法原理

YOLOv4-tiny[5] 由 YOLOv4[4] 进行尺度缩放而来。相比于 YOLOv4,YOLOv4-tiny 牺牲了一定的检测精度,但是它的参数量不到前者的 10%,推理速度则是前者的 6~8 倍。

2.1 YOLOv4-tiny 的骨干网络

如图 1 所示,骨干网络 CSPDarknet53-tiny 主要由 DarknetConv2D_BN_Leaky 模块和 Resblock_body 模块堆叠而成。DarknetConv2D_BN_Leaky 模块结合了二维卷积层、归一化处理层和激活函数 Leaky ReLU。Resblock_body 模块引入了如图 2 所示的 CSPNet 结构[18]:模块主干部分依旧是残差块的常规堆叠,但在分支部分引入了一条大跨度的残差边,该残差边经少量卷积处理之后直接连接到模块的最后,与主干部分的输

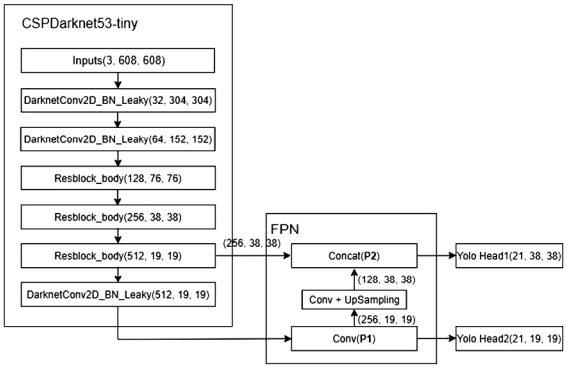


图 1　YOLOv4-tiny **网络结构**

Fig.1　YOLOv4-tiny network structure

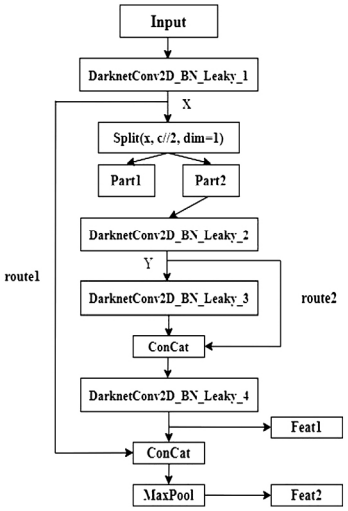


图 2　Resblock_body **结构**

Fig.2　Resblock_body structure

出在通道维度进行堆叠，最后经 2×2 **的最大池化层处理得到模块的输出。**CSPNet **结构在减少** 10%～30% **的网络参数量的同时，能够保证模型准确率基本不变或者稍有提高。图** 2 **中的** Feat1 **和** Feat2 **为** Resblock_body **模块输出的初始特征层。骨干网络** CSPDarknet53-tiny **中的前两个** Resblock_body **模块输出的初始特征层** Feat1 **会被直接丢弃，**Feat2 **则作为后一个** Resblock_body **模块的输入。对于第三个** Resblock_body **模块的输出** Feat1 **和** Feat2：Feat1 **直接作为特征增强网络** FPN **的第一个输入，**Feat2 **经** DarknetConv2D_BN_Leaky **模块处理后作为** FPN **的第二个输入。第三个** Resblock_body **模块的输出** Feat1 **和** Feat2 **的尺寸分别是** (256,38,38) **和** (512,19,19)，**其中，第一维的** 256 **或** 512 **代表的是特征层的通道数，后面两个维度** 38×38 **或** 19×19 **代表的是特征层的高和宽。**

2.2　YOLOv4-tiny **的特征增强网络**

如图 1 **所示，**YOLOv4-tiny **使用** FPN **作为特征增强网络。**FPN **的两个输入** Feat1 **和** Feat2 **的尺寸分别为** (256,38,38) **和** (512,19,19)。FPN **分两个步骤构建特征金字塔：**

1. Feat2 **经** 1×1 **卷积(**Conv**)处理之后得到** FPN **的第一个输出** P1(256,19,19)；

2. P1 **再经** 1×1 **的卷积处理和上采样处理之后与** Feat1 **在通道维度进行堆叠，得到** FPN **的第二个输出** P2(384,38,38)。

最后，FPN **模块的两个输出** P1 **和** P2 **再经少量卷积处理即可得到用于边界框预测的两个预测特征层** Yolo Head1(21,38,38) **和** Yolo Head2(21,19,19)。YOLOv4-tiny **的** FPN **模块构造非常简单，甚至过于简单，特征层的融合仅仅体现在单一的上采样之后的特征层堆叠，这为** YOLOv4-tiny **带来了非常优秀的实时检测速度(**COCO：443 FPS on RTX 2080Ti)，**但也为** YOLOv4-tiny **带来了网络整体感受野过弱、特征融合不充分以及对骨干网络提取到的特征信息利用率过低等缺点，使其难以适应复杂场景下包含小目标或者遮挡目标的检测任务。**

3　改进的 YOLOv4-tiny 算法

本文对 YOLOv4-tiny **进行了一系列改进：在骨干网络之后引入了** SPP **模块，利用** PAN **结构代替** FPN **作为特征增强网络，使用标签平滑策略对网络损失函数进行了优化，并结合** Mosaic **数据增强和学习率余弦衰退策略进行模型训练。**

3.1　空间金字塔池化

一般来说，可以使用剪裁或扭曲等方法改变输入图像的尺寸以满足卷积神经网络中分类器层(全连接层)的固定尺寸输入要求，但是这也会带来剪裁区域未覆盖完整目标或者图像畸变等问题。为了应对上述问题，He **等人**[14] **提出了空间金字塔池化(**SPP**)网络，可以对多尺度提取的特征层进行池化和融合，并得到固定尺寸的输出。本文在** YOLOv4-tiny **的骨干网络之后引入** SPP **模块以实现从细粒度到粗粒度的多尺度特征融合，增强特征层对目标的表达能力。本文改进后**

的算法网络结构如图 3 所示，可以看到，骨干网络 CSPDarknet53-tiny 的第二个输出特征层会先经 SPP 模块处理后再输入改进的特征增强网络 PAN。

如图 3 所示，SPP 模块处理步骤为：首先，输入特征层经过 3 次卷积处理（卷积核大小分别为 1×1、3×3 和 1×1）后通道数由 512 降至 256；然后，使用 3 个池化层（核大小分别为 3×3、7×7 和 11×11）进行最大池化处理，每个池化层的输出通道数均为 256；再将上一步的输入与 3 个池化层处理后的输出在通道维度进行堆叠，通道数变为 1 024；最后，经过又一次 3 次卷积处理，SPP 模块输出的通道数由 1 024 恢复至 256。SPP 模块在对输入特征层进行多尺度池化并融合的同时，也大幅增强了网络的感受野，对输入特征层的信息提取更为充分。

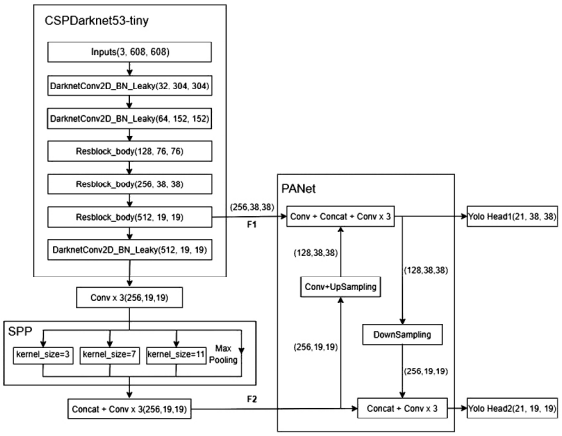


图 3 改进的 YOLOv4-tiny 网络结构

Fig.3 Improved YOLOv4-tiny network structure

3.2 特征增强网络 PAN

如图 1 所示，CSPDarknet53-tiny 的两个输出特征层在转化为预测特征层 YOLO Head 之前，会先经过 FPN 结构进行特征信息的融合。FPN 结构对语义信息更丰富的高层次特征层进行上采样之后，与细节信息更为丰富的低层次特征层在通道维度进行堆叠，以达成特征融合的目的。但是 YOLOv4-tiny 的 FPN 结构过于“简陋”，导致网络整体感受野过弱，对细节信息的利用率过低，在面对复杂场景下的小目标或遮挡目标时的表现并不理想。

为了应对上述问题，本文算法使用路径聚合网络（PAN）作为特征增强网络。如图 3 所示，相比 FPN，PAN 的特征反复提取、相互融合的策略更加复杂，包含了自下而上和自上而下两条特征融合的路径。可以看到，PAN 一样有两个输入特征层，F1 来自骨干网络 CSPDarknet53-tiny 的第三个 Resblock_body 模块，F2 则来自 SPP 模块。在自下而上的融合路径里，输入 F2 经 1＊1 的卷积处理和上采样处理之后，与同样经过卷积处理的输入 F1 在通道维度进行堆叠，特征层的通道数由 128 上升至 256；该特征层经过 3 次卷积处理（卷积核大小分别为 1×1、3×3 和 1×1）之后，得到 PAN 的第一个输出。类似地，在自上而下的路径里，第一条路径的输出在经过下采样完成高和宽的压缩之后与输入 F2 在通道维度进行堆叠，再经过 3 次卷积处理之后得到 PAN 的第二个输出。预测特征层 YOLO Head 由一个卷积核大小为 3×3 的 DarknetConv2D_BN_Leaky 模块和一个卷积核大小为 1×1 的普通二维卷积层相连而成，其输出数据主要用于先验框的参数调整。PAN 的特征融合策略里，来自不同尺度的特征相互融合、反复增强，充分利用骨干网络提取到的细节信息，大幅提升了预测特征层对目标的表达能力。

3.3 算法的损失函数

目标检测的损失函数一般包括 3 个部分，边界框定位损失 L_{loc}、预测类别损失 L_{cls} 和置信度损失 L_{conf}。整体的网络损失 L 的计算公式如下：

$$L=L_{\mathrm{loc}}+L_{\mathrm{cls}}+L_{\mathrm{conf}} \quad , \tag{1}$$

边界框定位损失 L_{loc} 用于衡量预测框与真实框的位置（包括高、宽和中心坐标等）误差，交并比（Intersection over Union，IoU）是目前最常用的评估指标，计算公式如下：

$$\mathrm{IoU}=\frac{|B\cap B^{\mathrm{gt}}|}{|B\cup B^{\mathrm{gt}}|} \quad , \tag{2}$$

其中，$B=(x,y,w,h)$ 表示预测框的位置，$B^{\mathrm{gt}}=(x^{\mathrm{gt}},y^{\mathrm{gt}},w^{\mathrm{gt}},h^{\mathrm{gt}})$ 表示真实框的位置。但是 IoU 损失仅在两边界框之间有重叠部分时生效，而对于非重叠的情形，IoU 损失不会提供任何可供传递的梯度。Zheng 等人[19]提出了 Complete IoU（CIoU）损失函数，综合考虑边界框之间的重叠面积、中心点距离以及长宽比等几何因素，使边界框的回归相比 IoU 更加稳定。本文算法将 CIoU 引入边框回归的损失函数 L_{loc}，计算公式如下：

$$\mathrm{CIoU}=\mathrm{IoU}-\frac{\rho^2(B,B^{\mathrm{gt}})}{d^2}-\alpha v \quad , \tag{3}$$

其中，$\rho^2(B,B^{gt})$表示预测框和真实框的中心点之间的欧氏距离，d 表示包含预测框和真实框的最小闭包区域的对角线距离，α 是权重参数，v 是衡量长宽比一致性的参数，α 和 v 的计算公式如下：

$$\alpha=\frac{v}{1-\mathrm{IoU}+v}\quad , \tag{4}$$

$$v=\frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}}-\arctan\frac{w}{h}\right)^2\quad , \tag{5}$$

则边界框定位损失函数 L_{loc} 为：

$$L_{loc}=1-\mathrm{IoU}+\frac{\rho^2(B,B^{gt})}{d^2}+\alpha v\quad , \tag{6}$$

预测类别损失 L_{cls} 用于衡量预测框与真实框的类别误差，采用交叉熵损失函数进行衡量：

$$L_{cls}=-\sum_{i=0}^{K\times K}I_{ij}^{obj}\sum_{c\in\ classes}\left[\hat{p}_i(c)\log(p_i(c))+(1-\hat{p}_i(c))\log(1-p_i(c))\right], \tag{7}$$

其中，$K\times K$ 表示不同尺度（19×19 或 38×38）的特征图上的网格数量；如果第 i 个网格的第 j 个先验框有需要负责预测的物体，那么 $I_{ij}^{obj}=1$，否则为 0；c 表示某一目标类别，$\hat{p}_i(c)$和 $p_i(c)$分别表示第 i 个网格的第 j 个先验框属于类别 c 的概率的真实值和预测值。为了在网络训练时抑制过拟合问题，本文算法引入了标签平滑策略，标签平滑前后的真实概率$\hat{p}_i(c)$分布改变如下：

$$\hat{p}_i(c)=\begin{cases}1, & i=y\\ 0, & i\neq y\end{cases}\Rightarrow\hat{p}_i(c)=\begin{cases}1-\delta, & i=y\\ \delta, & i\neq y\end{cases}, \tag{8}$$

其中，δ 为标签平滑超参数，取 0.05。

置信度损失 L_{conf} 同样采用交叉熵损失函数进行衡量，计算公式如下：

$$\begin{aligned}L_{conf}=&\sum_{i=0}^{K\times K}\sum_{j=0}^{M}I_{ij}^{obj}\left[\hat{C}_i\log(C_i)+(1-\hat{C}_i)\log(1-C_i)\right]-\\&\sum_{i=0}^{K\times K}\sum_{j=0}^{M}I_{ij}^{noobj}\left[\hat{C}_i\log(C_i)+(1-\hat{C}_i)\log(1-C_i)\right],\end{aligned} \tag{9}$$

其中，$K\times K$、I_{ij}^{obj} 的含义与式（8）一致，M 表示先验框的个数，$\hat{C}_i$ 和 C_i 表示置信度的真实值和预测值；如果第 i 个网格的第 j 个先验框没有需要负责预测的物体，那么 $I_{ij}^{noobj}=1$，否则为 0。

4　实验结果与分析

4.1　数据集

目前专门应用于口罩检测的公开数据集相对较少。本文从 CMFD[9] 和 RMFD[10] 中分别选取了 3 800 和 4 200 张图像，并结合互联网图像爬取，建立了包含 13 324 张多场景图像及标注的口罩检测数据集。考虑到口罩目标的特殊性，数据集中还特意加入了约 250 张佩戴面巾、头巾等遮挡物或者手部捂脸的图像，以增强模型的泛化能力与实用性。模型训练之前，随机选取 20% 的数据（2 665 张图像）作为测试集；训练过程中，以 9∶1 的比例划分训练集与验证集，并采用随机的剪裁、图像翻转、色域变换和 Mosaic 等数据增强方式增加训练样本的多样性，提高模型的鲁棒性。如图 4(d) 所示，Mosaic 拼接方式随机选取 4 张图片进行拼接，然后输入网络进行训练，可以有效丰富检测目标的背景。

4.2　评价指标

本文采用平均精度均值（mAP）和每秒传输帧数（FPS）分别作为算法检测精度和检测速度的评价指标。

在口罩检测任务中，mAP 即为口罩目标和人脸目标的平均精度（AP）的均值，如式（10）所示：

$$\mathrm{mAP}=\frac{1}{N}\sum_{i=1}^{N}\mathrm{AP}_i\quad , \tag{10}$$

其中，N 为类别数，i 表示某一类别。某一类别 i 的 AP 计算公式如下：

$$\mathrm{AP}_i=\int_0^1P(R)\mathrm{d}R\quad , \tag{11}$$

其中，$P(R)$ 是精确率（Precision，P）与召回率（Recall，R）之间的映射关系，常用 P-R 曲线表示，曲线下方的区域面积即为该类别 AP 值。精确率和召回率的计算方式如下：

$$P_i=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}},\tag{12}$$

$$R_i=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}},\tag{13}$$

其中，TP 表示检测类别与真实标签均为 i 的样本数量，FP 表示检测类别为 i 与真实标签不为 i 的样本数量，FN 表示检测类别不为 i 但真实标签为 i 的样本数量。

(a) 原图
(a) Origin image

(b) 剪裁+翻转
(b) Image cropping and flipping

(c) 色域变换
(c) Gamut transformation

(d) Mosaic
(d) Mosaic

图 4 数据增强示例

Fig.4 Examples of data augmentation

FPS 指的是网络模型每秒能够检测的图像帧数，常用于衡量模型的推理速度。FPS 的计算过程主要考虑了模型前向计算、阈值筛选和非极大值抑制几个步骤。

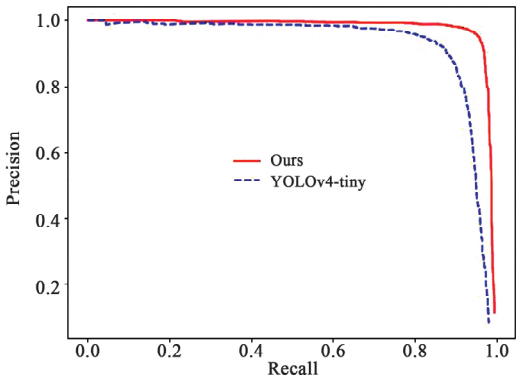
4.3 实验设置

本文算法在基于 Python 3.7 的 Ubuntu 18.04 环境下使用 PyTorch1.7.0 框架实现，在桌面工作站（AMD Ryzen 3600x CPU @ 3.60 GHz，GeForce RTX 2080Ti）上进行训练，并在个人电脑（Intel Core i5-8300H CPU @ 2.30 GHz，GeForce GTX 1050Ti）上进行测试。训练时，使用 COCO 数据集上预训练的 YOLOv4-tiny 模型对骨干网络的参数进行初始化，以获得更好的模型初始性能。网络输入尺寸为 608×608，优化器选择 Adam，训练过程分两个阶段：第一阶段冻结骨干网络，训练剩余网络参数，初始学习率为 1e−3，批量大小为 128，学习率调整策略为余弦衰退策略 CosineAnnealingLR（$T\_max=5$，eta_min=1e−5），其中 T_max 表示余弦函数的周期，eta_min 表示学习率的最小值，训练 50 轮次；第二阶段解冻骨干网络，训练所有网络参数，初始学习率为 1e−4，批量大小为 32，学习率调整策略与第一阶段一致，训练 100 轮次。

4.4 结果分析

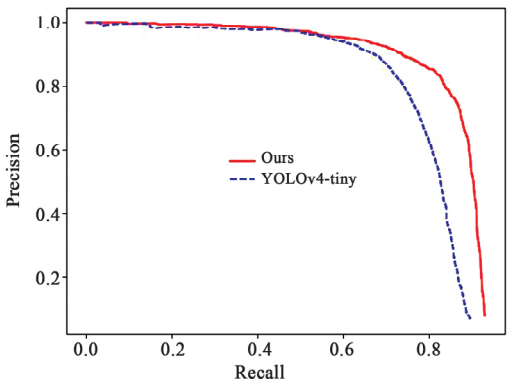
本文算法和 YOLOv4-tiny（在相同环境下以相同方式训练）对口罩目标和人脸目标检测结果的 P-R 曲线如图 5 所示。可以看到，不管是对口罩目标还是人脸目标，本文算法的检测性能均优于 YOLOv4-tiny（P-R 曲线越靠右，代表检测效果越好）。

图 6 则列出了 YOLOv4-tiny 和本文算法的检测结果对比，可以看到，YOLOv4-tiny 对于小



(a) 口罩检测 P-R 曲线

(a) P-R curves of mask detection



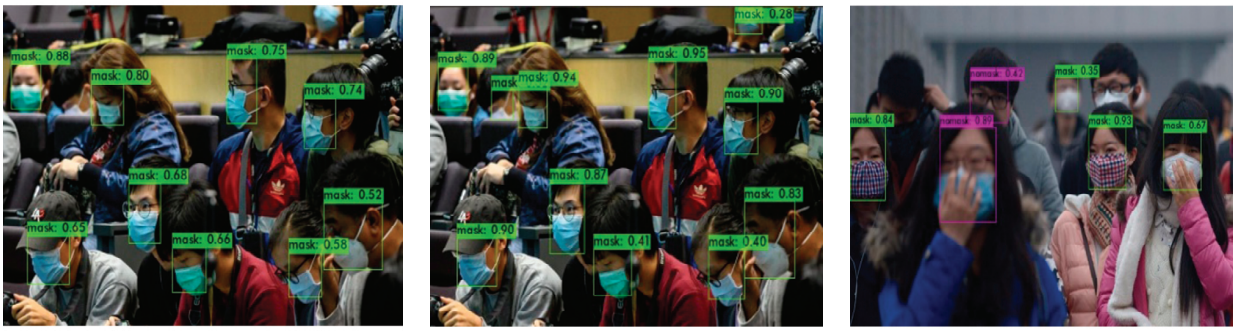
(b) 人脸检测 P-R 曲线

(b) P-R curves of face detection

图 5 YOLOv4-tiny 和本文算法的 P-R 曲线对比

Fig.5 Comparison of P-R curves between YOLOv4-tiny and the proposed algorithm

目标或者遮挡目标存在一定的漏检或错检情况，尤其是对遮挡目标，YOLOv4-tiny 的适应性明显较弱，而本文算法不管是检测准确率还是检测结果的置信度都明显高于 YOLOv4-tiny。



(a) YOLOv4-tiny 算法 1
(a) YOLOv4-tiny algorithm 1

(b) 本文算法 1
(b) Proposed algorithm in this paper 1

(c) YOLOv4-tiny 算法 2
(c) YOLOv4-tiny algorithm 2

(d) 本文算法 2
(d) Proposed algorithm in this paper 2

(e) 本文算法 3
(e) YOLOv4-tiny algorithm 3

(f) 本文算法 3
(f) Proposed algorithm in this paper 3

图 6　YOLOv4-tiny 和本文算法的检测结果对比
Fig.6　Comparison of detection results between YOLOv4-tiny and the proposed algorithm

表 1 列出了本文算法和包括 YOLOv4-tiny 在内的其他几种主流目标检测算法的 mAP 和 FPS 指标对比。对于口罩目标和人脸目标，本文算法的检测 AP 值分别达到了 94.7%和 85.7%，相比 YOLOv4-tiny 分别提高了 4.3%和 7.1%，具备更高的检测精度。在检测速度上，由于本文算法引入了 SPP 模块并且使用了更复杂的特征融合策略 PAN，增加了网络的参数量，所以检测速度略低于 YOLOv4-tiny，不过仍然取得了 76.8 FPS 的实时检测速度（on GeForce GTX 1050Ti）。相比 YOLOv3-tiny，本文算法的 mAP 则有着 11.5%的大幅提升，不管是口罩目标还是人脸目标，本文算法的检测性能都明显更优。至于其他非轻量级的单阶段目标检测算法，例如 YOLOv3、YOLOv4 和 SSD[20]，YOLOv3 和 YOLOv4 的检测精度与本文算法相当或稍有提升，SSD（输入尺寸为 512×512）的检测精度低于本文算法，不过三者的检测速度均低于 18 FPS，不到本文算法的 25%，难以满足口罩检测任务的实时性要求。而两阶段目标检测算法的检测速度则更低：FasterR-CNN[8]（以 VGG16[21] 为骨干网络）的检测速度不到 5 FPS，检测精度也低于本文算法 4.7%。综上所述，本文算法以 76.8 FPS 的实时检测速度取得了 90.2%的平均检测精度，很好地满足了口罩检测任务的精度与实时性要求。

表 1 不同检测算法性能比较

Tab.1 Performance comparison of different detection algorithms

算法	AP/%		mAP/%	FPS
	mask	nomask		
SSD-512	90.2	82.6	86.4	13.12
Faster R-CNN	87.4	83.6	85.5	4.52
YOLOv3	94.8	85.8	90.3	17.35
YOLOv4	95.8	88.2	92.0	13.96
YOLOv3-tiny	84.1	73.2	78.7	93.50
YOLOv4-tiny	90.4	78.6	84.5	89.71
本文算法	94.7	85.7	90.2	76.80

4.5 消融实验

为了研究各个模块或者训练策略对模型带来的影响，设计了如下几组消融实验，实验设置如表 2 所示。实验中骨干网络均为 CSPDartnet53-tiny，表中的“√”代表包含该结构或者使用该策略，“×”代表未包含该结构或未使用该策略。可以看到，第 1 组实验设置代表了 YOLOv4-tiny，mAP 值为 84.5%；第 2 组实验在第 1 组实验的基础上引入了 SPP 模块，增强了网络的感受野，提取多尺度特征并进行融合，对输入特征层的信息利用更充分，mAP 相比第 1 组实验提高了 2.2%；第 3 组实验使用 PAN 模块代替第 1 组(YOLOv4-tiny)中的 FPN 模块作为特征增强网络，分两条路径将不同尺度的特征层相互融合、反复增强，mAP 相比第 1 组实验提高了 2.6%；第 4 组实验在第 3 组实验的基础上结合了 SPP 模块和 PAN 模块，mAP 相比第 3 组实验提高了 2.4%，相比第 1 组实验提高了 5.0%；第 5 组实验在第 4 组实验的基础上使用了标签平滑策略对网络损失函数进行优化，该策略同样提高了模型的检测精度，mAP 相比第 4 组实验提高了 0.7%，相比第 1 组实验提高了 5.7%。综上所述，本文算法对 YOLOv4-tiny 的改进和优化具备合理性和有效性，在口罩检测任务中提升了原算法的检测性能。

表 2 消融实验结果比较

Tab.2 Comparison of ablation experiment results

ID	FPN	SPP	PANet	标签平滑	mAP/%
1	√	×	×	×	84.5
2	√	√	×	×	86.7
3	×	×	√	×	87.1
4	×	√	√	×	89.5
5	×	√	√	√	90.2

5 结 论

为了在公共场所高效监测人流的口罩佩戴情况，本文提出了一种基于 YOLOv4-tiny 改进的轻量级口罩检测算法。通过 SPP 模块和 PAN 模块的引入，以及标签平滑和 Mosaic 数据增强等策略的应用，提高了算法在复杂场景下对小目标和遮挡目标的适应能力，在口罩目标和人脸目标上的检测精度分别达到了 94.7% 和 85.7%，相比 YOLOv4-tiny 分别提高了 4.3% 和 7.1%，实时检测速度达到了 76.8 FPS(on

GeForce GTX 1050Ti），满足了多种复杂场景下口罩检测任务的检测精度与实时性要求。本文算法对人脸目标的检测精度（85.7%）低于口罩目标的检测精度（94.7%），主要原因是人脸目标检测面临的场景更为复杂多变，需要提取更多的细节信息，而 YOLOv4-tiny 作为通用目标检测算法，它的骨干网络对复杂场景下的人脸目标的适应能力略显不足，在未来的研究中可以引入注意力机制或者结合专门的人脸检测算法进行针对性优化。

参 考 文 献：

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

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

[3] REDMON J, FARHADI A. Yolov3: an incremental improvement [J]. *arXiv*: 1804.02767, 2018.

[4] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. Yolov4: optimal speed and accuracy of object detection [J]. *arXiv*: 2004.10934, 2020.

[5] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. Scaled-YOLOv4: scaling cross stage partial network [J]. *arXiv*: 2011.08036,2020.

[6] GIRSHICK R, DONAHUE J, DARRELL T, *et al*. Region-based convolutional networks for accurate object detection and segmentation [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 38(1): 142-158.

[7] GIRSHICK R. Fast R-CNN [C]//*Proceedings of the IEEE International Conference on Computer Vision*. Santiago: IEEE, 2015: 1440-1448.

[8] REN S Q, HE K M, 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] CABANI A, HAMMOUDI K, BENHABILES H, *et al*. MaskedFace-Net – a dataset of correctly/incorrectly masked face images in the context of COVID-19 [J]. *Smart Health*, 2021, 19: 100144.

[10] WANG Z Y, WANG G C, HUANG B J, *et al*. Masked face recognition dataset and application [J]. *arXiv*: 2003.09093, 2020.

[11] 牛作东，覃涛，李捍东，等.改进 RetinaFace 的自然场景口罩佩戴检测算法 [J].计算机工程与应用，2020，56(12)：1-7.
NIU Z D, QIN T, LI H D, *et al*. Improved algorithm of RetinaFace for natural scene mask wear detection [J]. *Computer Engineering and Applications*, 2020, 56(12): 1-7. (in Chinese)

[12] 王艺皓，丁洪伟，李波，等.复杂场景下基于改进 YOLOv3 的口罩佩戴检测算法 [J].计算机工程，2020，46(11)：12-22.
WANG Y H, DING H W, LI B, *et al*. Mask wearing detection algorithm based on improved YOLOv3 in complex scenes [J]. *Computer Engineering*, 2020, 46(11): 12-22. (in Chinese)

[13] LIN T Y, MAIRE M, BELONGIE S, *et al*. Microsoft COCO: common objects in context [C]//*13th European Conference on Computer Vision*. Zurich: Springer, 2014: 740-755.

[14] HE K M, ZHANG X Y, REN S Q, *et al*. Spatial pyramid pooling in deep convolutional networks for visual recognition [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(9): 1904-1916.

[15] LIU S, QI L, QIN H F, *et al*. Path aggregation network for instance segmentation [C]//*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Salt Lake City: IEEE, 2018: 8759-8768.

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

[17] MÜLLER R, KORNBLITH S, HINTON G. When does label smoothing help?[J]. *arXiv*: 1906.02629, 2019.

[18] WANG C Y, LIAO H Y M, WU Y H, *et al*. CSPNet: a new backbone that can enhance learning capability of CNN [C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. Seattle: IEEE, 2020: 1571-1580.

[19] ZHENG Z H, WANG P, LIU W, *et al*. Distance-IoU loss: faster and better learning for bounding box regression [J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, 34(7): 12993-13000.

[20] LIU W, ANGUELOV D, ERHAN D, *et al*. SSD: single shot multibox detector [C]//*14th European Conference on Computer Vision*. Amsterdam: Springer, 2016: 21-37.

[21] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition [J]. *arXiv*: 1409.1556, 2014.

作者简介:

朱　杰(1996—),男,贵州毕节人,硕士研究生,2018 年于中国科学技术大学获得学士学位,主要从事计算机视觉方面的研究。E-mail: zhujie181@mails.ucas.ac.cn

王建立(1971—),男,山东曲阜人,博士,研究员,2002 年于中国科学院长春光学精密机械与物理研究所获得博士学位,主要从事地基空间目标光电探测技术、光电精密跟踪控制技术、光电精密测量误差补偿技术、大口径望远镜总体技术、大型军用光电系统总体集成技术等方面的研究。E-mail: wangjianli@ciomp.ac.cn