计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(3):195−201 [doi: 10.15888/j.cnki.csa.009021]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

YOLOv5 改进的轻量级口罩人脸检测①

葛云飞, 祁云嵩, 孟祥宇

(江苏科技大学 计算机学院, 镇江 212100)
通信作者: 葛云飞, E-mail: 442714602@qq.com

摘　要: 针对疫情防控下人脸识别应用出现人脸漏检、移动端平台的计算能力不足和硬件资源受限等问题, 提出一种 YOLOv5 改进的轻量级口罩人脸检测模型. 设计轻量化的 C3Ghost 模块替换原网络中的 C3 模块以压缩卷积过程的计算量和模型大小, 在主干网络中添加注意力机制以提高网络的特征提取能力, 并改进边框回归损失函数以提高检测速度和精度. 实验结果表明, 改进后的模型计算量和参数量分别降低了 29.79% 和 33.33%, 模型权重文件大小仅有 2.8 M, 减轻了对硬件条件的依赖, 同时模型的检测率达到了 96.6%, 相比现有轻量级模型优势突出, 能够有效地应用于人脸识别之中.

关键词: 人脸检测; YOLOv5; 注意力机制; C3Ghost; α-CIoU

引用格式: 葛云飞,祁云嵩,孟祥宇.YOLOv5 改进的轻量级口罩人脸检测.计算机系统应用,2023,32(3):195–201. http://www.c-s-a.org.cn/1003-3254/9021.html

Improved Lightweight Masked Face Detection Based on YOLOv5

GE Yun-Fei, QI Yun-Song, MENG Xiang-Yu

(School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212100, China)

Abstract: To address the problems of missed detection of faces, the insufficient computing power of mobile platforms, and the limited hardware resources of face recognition applications under epidemic prevention and control, this study proposes an improved lightweight detection model for faces with masks based on YOLOv5. In this model, the C3 module in the original network is replaced with a lightweight C3Ghost module to compress the computations of the convolution process and the size of the model. Moreover, an attention mechanism is added to the backbone network to improve the feature extraction capability of the network, and the border regression loss function is improved to improve the speed and accuracy of detection. The experimental results indicate that the amount of calculation and parameters of the improved model are decreased by 29.79% and 33.33%, respectively, with the weight file size of only 2.8 M. The improved model reduces the dependence on the hardware environment, and its detection rate reaches 96.6%. Compared with the existing models, it has outstanding advantages and can be effectively applied to face recognition.

Key words: face detection; YOLOv5; attention mechanism; C3Ghost; α-CIoU

2019 年 12 月底在湖北武汉爆发的新型冠状病毒肺炎肆虐至今, 对世界各国人民造成了巨大损失. 人们佩戴口罩出行已经成为常态, 而佩戴口罩所导致的面部遮挡问题干扰了人脸检测, 导致人脸识别算法出现漏检情况, 给如今众多人脸识别应用带来巨大的挑战, 如在通过火车站、机场安检通道时进行人脸认证就需要摘下口罩, 在某种程度上这就会带来一定的安全隐患. 在人脸识别算法中增强人脸检测网络, 高效准确识

① 基金项目: 国家自然科学基金 (61471182)
收稿时间: 2022-08-18; 修改时间: 2022-09-22, 2022-09-30; 采用时间: 2022-10-14; csa 在线出版时间: 2022-12-16
CNKI 网络首发时间: 2022-12-20

别口罩遮挡的人脸, 此举能够减少人员交叉感染的风险, 有效地抑制病毒的传播.

口罩人脸检测[1] 从本质上来说是属于目标检测范畴, 传统的目标检测方法利用滑动窗口遍历目标区域进行选择, 然后使用 SIFT、Harr 等特征对滑框选中的候选区域进行特征提取, 最后通过训练好的分类器[2] 如 AdaBoost、SVM 对特征进行分类, 滑框的选择没有针对性且特征提取的好坏直接影响后续的分类效果, 因而该方法效率低且鲁棒性差; 近年来, 深度学习的快速发展给目标检测带来了新思路, 通过卷积神经网络[3] 能够提取高层特征, 显著地提升了分类的准确率, 加快模型检测速度. 基于深度学习的目标检测算法主要分为“两阶段”和“单阶段”, “两阶段”指的是将提取特征和检测分为两个步骤, 以 R-CNN[4]、Fast R-CNN[5]、Faster R-CNN[6] 为代表, 首先为先进性区域选取, 再进行分类; 而“单阶段”则是将这两个步骤合并为一步, 以 SSD[7]、YOLO[8] 系列为代表, 将区域选取和分类融合到同一个网络结构中, 构建一个“分类+回归”的多任务学习模型结构.

在人脸检测模型的实际应用场景中, 大多数是部署在移动端或嵌入式平台上, 考虑到移动端平台的计算能力不足和无法使用 GPU 加速处理数据的问题, 实现轻量级的网络模型显得尤为重要. 针对人脸口罩佩戴检测的精度、速度和模型大小的问题, 多位学者进行了研究, 比如文献 [9] 基于原始的 Faster R-CNN 框架, 引入基于空间-通道注意力结构改进的 Res2Net 分组残差结构, 在 AIZOO 和 FMDD 两个人脸数据集上对佩戴口罩的人脸检测准确率分别达到 90.37% 和 90.11%. 文献 [10] 在 YOLOv4 算法中引入轻量级骨干网络 L-CSPDarkNet 以提高模型的检测速度, 同时提出轻量级特征增强模块 Light-FEB 和多尺度注意力机制 Multi-Scale-Sam 增强轻量级主干网络的特征提取能力, 该算法精度可达 91.94%.

本文选取 YOLO 系列最新的 YOLOv5 模型为基础, 设计更加轻量化的网络结构以减少参数量和计算量并提高检测速度, 添加注意力机制以增强特征表达能力, 并改进损失函数, 在模型轻量化的同时提升检测精度.

1 YOLOv5 算法介绍

YOLOv5 是 YOLO 系列当前最新的实时目标检测算法, 自 Ultralytics 公司于 2020 年 5 月提出迭代更新至今, 最新版 v6.0 在之前版本的 YOLOv5s、YOLOv5m、YOLOv5l、YOLOv5x 这 4 个模型基础上提出了更小的 YOLOv5n 模型, 新提出的 YOLOv5n 模型保持了 YOLOv5s 的深度即 Bottleneck 的数量, channel 数降为一半, 模型总参数减少了 75%, 非常适用移动端平台.

YOLOv5 现版本网络主要由输入端、Backbone、Neck、Prediction 四部分组成, 在输入端用了 Mosaic 数据增强、自适应锚框计算、自适应图片缩放等策略对数据进行预处理, Backbone 结构主要由 C3、Conv 和 SPPF 模块组合而成, Neck 结构采用了 PANet[11] 结构, Detect 结构对 3 个不同尺寸特征图进行预测. Mosaic 数据增强对 4 张图片采取随机缩放、随机裁剪、随机排布的方式进行拼接, 在提升数据集多样性的同时增加许多小目标, 训练得到的模型鲁棒性更好; C3 模块参照 CSPNet[12] 的设计将输入的特征图分成了两部分, 然后分别进行各自的阶段操作后合并以实现更丰富的梯度组合, 在保证准确率的前提下降低计算成本, 并且通过 shortcut 的 true 和 false 值控制有无残差网络[13]; SPPF 模块通过 3 次递进的池化操作, 最后拼接得到了 2 倍通道数的特征图, 极大地提升了模型的特征提取能力, 有利于检测图像中不同大小的目标对象. PANet 结构如图 1 所示, 它是在特征金字塔网络 FPN[14] 结构的基础上引入了 Bottom-up path augmentation 结构, FPN 将高层特征信息通过上采样的方式和低层特征融合达到高层语义特征与低层细节特征融合互补的目的, PAN 再进行自底向上的特征融合, 强化特征提取能力.

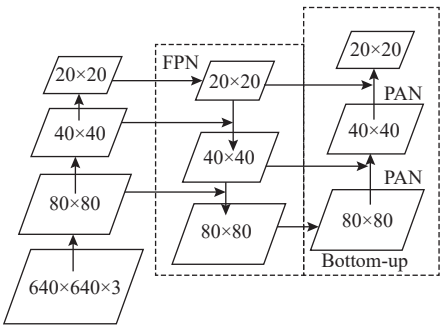

图 1 PANet 结构

YOLOv5n 和 YOLOv5s 对于 640×640 和 1280×1280 不同输入尺寸的性能表现, 1280×1280 尺寸精度虽高但输入图片的高分辨率对设备资源增加了更大的负担, 640×640 尺寸下, YOLOv5n 的参数量和计算量

优于 YOLOv5s, 更易部署于移动端平台, 最终选择 YOLOv5n 为基准进行模型优化.

2 模型优化

2.1 改进的 C3Ghost

GhostNet 轻量级网络能够在保持原有卷积输出特征图的尺寸和通道大小的前提下, 大幅降低网络的计算量和参数量, 实现原理是将传统的卷积分成两步操作进行, 分别是普通卷积和廉价的线性计算, 首先利用较少的卷积核生成一部分特征图, 接着对这部分特征图进行通道卷积生成更多特征图, 最后拼接两组特征图生成 GhostNet 特征图. 传统卷积与 GhostNet 卷积过程如图 2 所示.

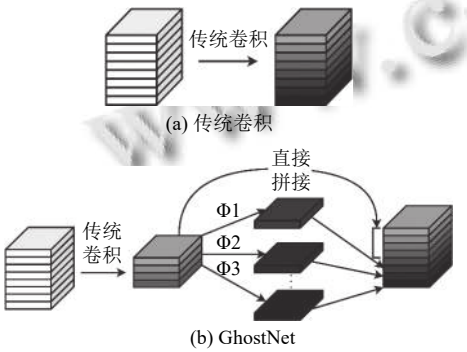

图 2 传统卷积与 GhostNet 卷积过程

传统卷积计算方式可表示为:

$$Y = F * f + b \tag{1}$$

其中, $F \in R^{H\times W\times C}$表示输入特征图, H、W表示特征图的高和宽, C表示通道数, $f \in R^{N\times K\times K\times C}$表示$N$个$C$通道$K\times K$大小的卷积核, b表示偏置项, $Y \in R^{M\times H'\times W'}$表示通过卷积运算得到的输出特征图. 由式 (1) 可以得出传统卷积运算的 FLOPs 高达$N\times H'\times W'\times K\times K\times C$, 参数量为$N\times K\times K\times C$.

GhostNet 卷积的计算方式[15] 可表示为:

$$Y' = F * f' \tag{2}$$

$$y_{ij} = \Phi_{i,j}(y_i), i \in [1, M], j \in [1, S] \tag{3}$$

其中, $f' \in R^{M\times K\times K\times C}$表示$M$个$C$通道$K\times K$大小的卷积核, 其中$M < N$, 这里的传统卷积减少了卷积核的数量并省略了偏置项, y_i表示$Y' \in R^{M\times H'\times W'}$中的第$i$个通道特征图, $\Phi_{i,j}$表示第j个线性计算 (除最后一个$\Phi_{i,S}$), 用于生成第j个 Ghost 特征图, y_i可以生成一个或多个 Ghost 特征图, 使用$\Phi_{i,S}$表示对Y'特征图的 identify 映射, 即图 2(b) 的直接拼接, 最终得到$N = M\times S$个与传统卷积相同的输出特征图. 理想情况下线性运算可以使用不同的参数和形状, 但考虑到实际效用, 采用相同大小的线性运算 (3×3或5×5), 这里我们设为$D\times D$, 由式 (2) 和式 (3) 得出 GhostNet 运算的 FLOPs 为$M\times H'\times W'\times K\times K\times C + (N-M)\times H'\times W'\times D\times D$, 参数量为$M\times K\times K\times C + (N-M)\times D\times D$.

通过对比传统卷积和 GhostNet 卷积的计算量和参数量, 如式 (4):

$$\begin{aligned}&\frac{N\times H'\times W'\times K\times K\times C}{M\times H'\times W'\times K\times K\times C + (N-M)\times H'\times W'\times D\times D}\\ =&\frac{N\times K\times K\times C}{\frac{N}{S}\times K\times K\times C + \frac{N}{S}(S-1)\times D\times D}\\ =&\frac{K\times K\times C}{\frac{1}{S}\times K\times K\times C + \frac{1}{S}(S-1)\times D\times D}\\ \approx&\frac{S\times C}{C+S-1}\approx S\end{aligned} \tag{4}$$

采用同样大小的$K\times K$与$D\times D$时, 因为S远小于C, GhostNet 计算量和参数量可近似为传统卷积的S分之一, 从理论上证实了 GhostNet 的轻量化效力, 故而将 GhostNet 网络融入 C3 模块形成新的 C3Ghost 模块, 具体结构如图 3 所示.

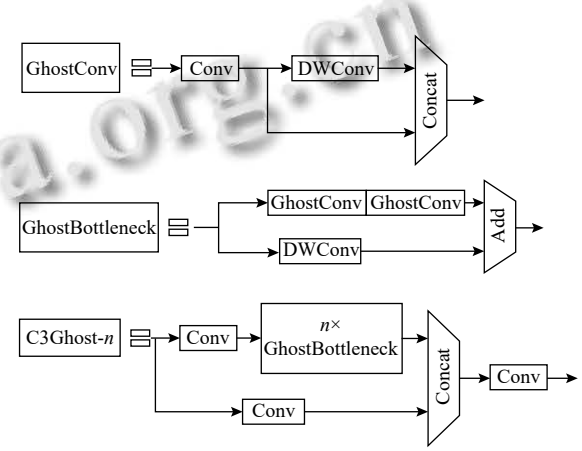

图 3 C3Ghost 模块

GhostConv 第 1 个 Conv 采用步长为 1 的1×1卷积核将输入特征图通道数减半, 再将上一步得到的特征图进行5×5卷积核的深度卷积[16], 最后进行拼接. 改造后的 GhostBottleneck 通过第 1 个 GhostConv 将输入特征图通道数减半, 再由第 2 个 GhostConv 将特征图通道数恢复如初, 最后与经过3×3深度卷积的残差边相

加融合特征. GhostBottleneck 替换原 C3 模块中的 Bottleneck 得到 C3Ghost 模块, 减少了原结构中大多数 3×3传统卷积, 压缩了模型并降低了计算量, 提高了运行速度, 部署在移动端也能达到不错的效果.

2.2 注意力机制

注意力在人类感知中起着重要作用, 使人们能够有选择地关注重要部分, 获取感兴趣的信息, 将注意力纳入机器学习中就形成了注意力机制, 在模型的学习过程中, 将有限的精力集中在重要信息上, 减小计算量的同时节约了成本. 卷积注意力模块 (CBAM)[17] 是一个简单而有效的前馈卷积神经网络, 融合通道注意力机制和空间注意力机制, 先通道后空间的串行方式组合在保持较小开销的情况下实现了相当大的性能提升. 特征图$F \in R^{H\times W\times C}$作为输入, CBAM 依次计算得到一个一维通道注意图$M_C \in R^{1\times 1\times C}$和一个二维空间注意图$M_S \in R^{H\times W\times 1}$, 整个注意过程如下:

$$F' = M_C(F)\otimes F \tag{5}$$

$$F'' = M_S(F')\otimes F' \tag{6}$$

其中, ⊗表示元素相乘.

2.2.1 通道注意力机制

通道注意力机制关注的是输入特征图的通道信息, 给予每条通道不同的权重, 权重参数代表了该通道特征信息对特征图的关键信息的影响程度, 合理的权重分配强化了特征图的特征信息表达. 通道注意力模块的网络结构如图 4, 对输入特征图同时进行最大池化和平均池化以聚合空间维度信息, 然后依次送入一个权重共享的多层感知机器 (MLP)[18], 最后通过 Sigmoid 激活函数得到通道注意力图$M_C(F)=\sigma(W_1(W_0(F^C_{avg}))+W_1(W_0(F^C_{max})))$, W_0, W_1表示多层感知器中的参数.

图 4 通道注意力结构

2.2.2 空间注意力机制

空间注意力[19] 关注的是输入特征图的位置信息, 按照对特征图像素影响的重要性, 分配权重, 在一定程度上弥补了通道注意力的不足. 空间注意力模块的网络结构如图 5 所示, 对输入特征图先后进行最大池化和平均池化得到两组特征图, 接着在通道维度上进行拼接并通过 7×7 的卷积核处理, 最后通过 Sigmoid 激活函数得到空间注意力图$M_S(F)=\sigma\left(f^{7\times 7}\left(\left[F^C_{avg};F^C_{max}\right]\right)\right)$.

在 Backbone 结构中的所有 Conv 模块加入 CBAM, 先进行标准卷积再通过 CBAM 模块削弱网络中无关特征的权重, 提高对脸部特征关注, 会减少网络模型收敛的时间, 理论上可以提升小目标检测性能.

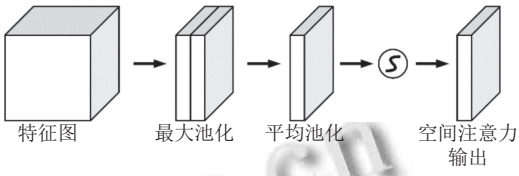

图 5 空间注意力结构

2.3 α-CIoU 损失函数

损失函数用来计算模型输出的预测值与输入的实际值之间的差距, 在模型训练优化过程中至关重要. 目标检测通过 Bounding box 回归来预测定位图像中的目标, 早期的目标检测使用IoU作为损失函数, 当预测框与真实框不重叠时, IoU损失会出现梯度消失问题, 导致收敛速度减慢、检测器不准确. 随着GIoU[20]、DIoU、CIoU[21] 的提出, GIoU在IoU损失中引入惩罚项以缓解梯度消失的问题, 而DIoU和CIoU在惩罚项中考虑了预测框与真实框中心点之间的距离和宽高比, 使得 Bounding box 回归效果更佳. YOLOv5 中选择了CIoU损失函数, 公式如下:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b,b^{gt})}{c^2} + \beta\nu \tag{7}$$

其中, $\rho(b,b^{gt})$表示预测框与真实框中心点之间的欧式距离, c 表示能够同时覆盖预测框与真实框的最小矩形的对角线距离, $\beta=\dfrac{\nu}{(1-IoU)+\nu}$表示权重系数, $\nu=\dfrac{4}{\pi^2}\left(\arctan\dfrac{w^{gt}}{h^{gt}}-\arctan\dfrac{w}{h}\right)^2$是衡量预测框与真实框长宽比的一致性.

在现有的IoULoss中引入 power 变换, 提出了一个新的IoU损失函数α-IoU[22], 公式如下:

$$L_{IoU} = 1 - IoU^{\alpha} \tag{8}$$

其中, power 参数可作为调节α-IoU损失的超参数以满足不同水平的 Bounding box 回归精度, $\alpha=3$训练会取得比较好的效果. 对原有CIoU改造得到α-CIoU, 理论上收敛速度更快, 精度更高, 公式如下:

$$L_{\alpha\text{-CIoU}} = 1 - IoU^{\alpha} + \frac{\rho^{2\alpha}(b,b^{gt})}{c^{2\alpha}} + (\beta\nu)^{\alpha} \tag{9}$$

3 实验结果及分析

3.1 实验环境及数据集

本文实验机器操作系统版本为 Windows 10 专业版, CPU 型号为 Intel(R) Core(TM) i9-10900F CPU @ 2.8 GHz, 显卡型号为 GeForce RTX 3090, 24 GB 显存, 32 GB 内存, 模型基于 PyTorch 1.10 深度学习框架, 并使用 cuda 11.3 对 GPU 进行加速.

本文的实验数据集收集于公开数据集 WIDER Face[23], 对这些图片进行了筛选, 删除错误不合理标签并添加缺失标签, 然后通过爬虫在百度图片爬取了少量图片扩充数据集, 最终得到 9 240 张图片, 分为人脸类 (face) 和口罩人脸类 (mask), 训练前将数据集按 6:2:2 比例随机划分为训练集、验证集和测试集, 使用 Python 的 random 函数随机生成一个 0–100 之间的数字, 0–60 划分为训练集, 60–80 划分为验证集, 80–100 划分为测试集, 共得到训练集 5 449 张、验证集 1 903 张和测试集 1 888 张.

3.2 评价指标

考虑到人脸检测模型的主要部署在移动端或嵌入式设备上, 并且疫情防控影响深远, 本文选取精确率 (precision, P)、召回率 (recall, R)、平均精度均值 (mean average precision, mAP)、参数量 (parameters)、计算量 (GFLOPs)、模型大小、平均检测时间等作为评价指标.

精确率表示预测为正样本中实际正样品的概率, 召回率表示实际正样本中预测为正样品的概率, TP、FP、FN 分别是实际正样品预测为正样品数、实际负样品预测为正样品数、实际正样品预测为负样品数, 平均精度均值表示所有类别平均精度 (AP) 的均值, 模型大小指训练结束得到权重文件大小, 平均检测时间指模型检测一张图片所耗费的时间. 具体计算公式如下.

$$P = \frac{TP}{TP+FP} \tag{10}$$

$$R = \frac{TP}{TP+FN} \tag{11}$$

$$AP = \int_0^1 PdR \tag{12}$$

$$mAP = \frac{\sum_{i=1}^{n} AP_i}{n} \tag{13}$$

3.3 实验结果及分析

实验中采用 Warmup[24] 学习率优化策略, 在训练开始时采用一维线性插值方式更新学习率, 使学习率从 0 增加到初始学习率 (0.01), 这样可以规避过高初始学习率引起模型震荡的风险, 之后采用余弦退火算法[25] 更新学习率. 所有训练图像输入尺度统一为 640×640, 一共训练 400 轮, batch-size 设置为 128 即每轮中一个批次 GPU 处理的图片数量, 充分利用显存.

3.3.1 实验结果

通过消融实验验证本文的改进对口罩人脸检测性能的影响, 分别进行了 YOLOv5n、C3Ghost、CBAM、α-CIoU、YOLOv5n-face 共 5 次训练, 改进方式与名称对应, YOLOv5n-face 是最终改进模型. 训练得到的模型参数及测试集测试结果如表 1 所示.

表 1 消融实验结果

模型	P (%)	R (%)	mAP@0.5 (%)	Params	Speed-GPU (ms)	GFLOPs	Weight (M)
YOLOv5n	94.8	93.0	96.2	1 761 871	1.7	4.2	3.8
C3Ghost	95.2	92.9	96.1	1 236 955	1.4	2.8	2.8
CBAM	95.1	93.1	96.4	1 773 623	1.6	4.2	3.8
α-CIoU	95.7	93.0	96.5	1 761 871	1.6	4.2	3.8
YOLOv5n-face	95.8	93.2	96.6	1 248 707	1.5	2.9	2.8

如表 1 所示, 替换了融合 GhostNet 网络的 C3Ghost 模块后, 相对于原模型, 计算量和参数量分别减少了 33.33% 和 29.79%, 模型权重大小减少了 26.31%, 模型得到了理想压缩, 并且在 GPU 加速情况下模型检测速度提升了 17.64%, 这些都归功于廉价的 GhostNet 卷积, 鉴于移动端平台的计算能力不足的情况, 以上性能的提升显得尤为突出. 此外精确率提升了 0.4%, 召回率和平均精度均值仅降低了 1%, 不难想到, 模型计算量和参数量的压缩势必会给模型检测精度带来一定的影响; 原模型加入注意力机制 (CBAM) 后, 精确率、召回率和平均精度均值分别提升 0.3%、0.1% 和 0.2%, 参数量仅增加了 0.67%, 因为原模型对人脸特征的提取不够明确, 不能准确地检测出所有人脸, 造成漏检, CBAM 更加关注人脸位置和特征并且提高了模型的特征提取能力, 能够更加快速准确地检测到人脸位置, 检测速度 5.88% 提升也是理所当然. CIoU 损失函数优化

后的 α-CIoU 提升了 0.9% 的精确率和 0.3% 的平均精度均值, 正如提出者所说, 它通过增加高交并比对象的损失和梯度的权重来提高 Bounding box 回归精度, 尤其是在高平均精度均值的情况下, 并且在轻量级模型中表现更优; 最终模型 YOLOv5n-face 相比原模型, 权重文件大小减少了 23.68% 而平均精度均值提升了 0.4%, 检测速度也略有提升, 实现了模型轻量化的目标.

3.3.2 不同网络的对比实验

为了进一步验证本文改进后轻量级模型 YOLOv5n-face 在口罩人脸检测中的检测性能, 与 YOLO 同系列下的 YOLOv3-tiny[26]、YOLOv4-tiny[27] 展开对比实验, YOLOv3-tiny 和 YOLOv4-tiny 均属于单阶段轻量型目标检测算法且具有不错的检测速度和精度, 采用平均精度均值、参数量、计算量等多个指标进行对比评估, 对比实验结果如表 2 所示.

表 2 不同模型对比

模型	*mAP*@0.5 (%)	*mAP*@0.5:0.95 (%)	Params	GFLOPs	Speed-GPU (ms)	Speed-CPU (ms)	Weight (M)
YOLOv3-tiny	94.9	65.4	8 669 002	12.9	1.8	56.6	17.4
YOLOv4-tiny	95.2	66.7	6 267 590	17.1	2.0	69.4	12.6
YOLOv5n-face	96.6	69	1 248 707	2.9	1.5	41.8	2.8

不难看出, 本文模型的参数量和计算量为 YOLOv3-tiny 模型的 14.4%、22.48% 和 YOLOv4-tiny 模型的 19.92%、16.96%, 并且训练得到的模型大小仅有 2.8 M, 不到 YOLOv3-tiny 和 YOLOv4-tiny 模型的 16.67%、23.02%. 此外, 参数量和计算量的降低并没有导致平均精度均值的降低, 相比 YOLOv3-tiny 和 YOLOv4-tiny 模型 *mAP*@0.5(*IoU* 阈值为 0.5) 分别提高了 1.7% 和 1.4%, *mAP*@0.5:0.95 (*IoU* 阈值取 0.5 到 0.95, 步长 0.05) 分别提高了 3.6%、2.3%. 本文模型在减少模型大小的同时兼顾检测精度和速度, 无论是 CPU 还是 GPU 环境, 其检测精度和速度都优于 YOLOv3-tiny 和 YOLOv4-tiny, 仅 2.8 M 的模型大小部署在移动端或嵌入式设备的效果不言而喻, 整体性能相比 YOLOv3-tiny 和 YOLOv4-tiny 具有明显的优势.

4 结束语

针对疫情防控下移动端或嵌入式设备的人脸检测模型检测精度降低的问题, 考虑到移动端或嵌入式平台计算能力不足, 本文通过融合 GhostNet 网络, 添加注意力机制, 改进目标框 CIoU 损失函数, 提出了 YOLOv5 改进的轻量级口罩人脸检测模型. 该模型极大压缩了计算量和参数量并降低检测时间, 模型精度得到一定程度的优化, 且最终得到的模型权重大小仅 2.8 M, 减少对硬件条件的依赖. 后续将实现移动端平台的模型部署, 进一步实验证实该模型的有效性, 真正的投入疫情防控工作中以满足社会的实际需求.

参考文献

1 Hasan K, Ahsan S, Abdullah-Al-Mamun, *et al*. Human face detection techniques: A comprehensive review and future research directions. Electronics, 2021, 10(19): 2354. [doi: 10.3390/electronics10192354]

2 Verschae R, Ruiz-Del-Solar J, Correa M. A unified learning framework for object detection and classification using nested cascades of boosted classifiers. Machine Vision and Applications, 2008, 19(2): 85–103. [doi: 10.1007/s00138-007-0084-0]

3 Deng B, Lv H. Survey of target detection based on neural network. Journal of Physics: Conference Series, 2021, 1952: 022055. [doi: 10.1088/1742-6596/1952/2/022055]

4 Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 580–587.

5 Girshick R. Fast R-CNN. Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 1440–1448.

6 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

7 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot multibox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37.

8 Redmon J, Divvala S, Girshick R, *et al*. You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 779–788.

9 李泽琛, 李恒超, 胡文帅, 等. 多尺度注意力学习的 Faster R-CNN 口罩人脸检测模型. 西南交通大学学报, 2021, 56(5): 1002–1010. [doi: 10.3969/j.issn.0258-2724.20210017]
10 丁培, 阿里甫·库尔班, 耿丽婷, 等. 自然环境下实时人脸口罩检测与规范佩戴识别. 计算机工程与应用, 2021, 57(24): 268–275. [doi: 10.3778/j.issn.1002-8331.2106-0363]
11 Liu S, Qi L, Qin HF, *et al*. Path aggregation network for instance segmentation. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8759–8768.
12 Wang CY, Liao HYM, Wu YH, *et al*. CSPNet: A new backbone that can enhance learning capability of CNN. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Seattle: IEEE, 2020. 1571–1580.
13 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778.
14 Lin TY, Dollár P, Girshick R, *et al*. Feature pyramid networks for object detection. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 936–944.
15 Han K, Wang YH, Tian Q, *et al*. GhostNet: More features from cheap operations. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 1577–1586.
16 Chollet F. Xception: Deep learning with depthwise separable convolutions. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2016. 1800–1807.
17 Woo S, Park J, Lee JY, *et al*. CBAM: Convolutional block attention module. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 3–19.
18 Tolstikhin IO, Houlsby N, Kolesnikov A, *et al*. MLP-Mixer: An all-MLP architecture for vision. Proceedings of the 35th Advances in Neural Information Processing Systems. 2021. 24261–24272.
19 Zhao L, Yang F, Bu LG, *et al*. Driver behavior detection via adaptive spatial attention mechanism. Advanced Engineering Informatics, 2021, 48: 101280. [doi: 10.1016/j.aei.2021.101280]
20 Rezatofighi H, Tsoi N, Gwak JY, *et al*. Generalized intersection over union: A metric and a loss for bounding box regression. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 658–666.
21 Zheng ZH, Wang P, Liu W, *et al*. Distance-IoU loss: Faster and better learning for bounding box regression. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(7): 12993–13000. [doi: 10.1609/aaai.v34i07.6999]
22 He JB, Erfani S, Ma XJ, *et al*. Alpha-IoU: A family of power intersection over union losses for bounding box regression. arXiv:2110.13675, 2021.
23 Yang S, Luo P, Loy CC, *et al*. WIDER FACE: A face detection benchmark. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 5525–5533.
24 Xiong RB, Yang YC, He D, *et al*. On layer normalization in the transformer architecture. Proceedings of the 37th International Conference on Machine Learning. JMLR.org, 2020. 975.
25 Loshchilov I, Hutter F. SGDR: Stochastic gradient descent with warm restarts. Proceedings of the 5th International Conference on Learning Representations. Toulon: ICLR, 2017.
26 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv:1804.02767, 2018.
27 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934, 2020.

(校对责编: 孙君艳)