计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(2):128−138 [doi: 10.15888/j.cnki.csa.008942]

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

# 基于轻量化神经网络的社交距离检测①

王　林, 张江涛

(西安理工大学 自动化与信息学院, 西安 710048)
通信作者: 张江涛, E-mail: 1307510426@qq.com

**摘　要**: 保持安全社交距离是有效防止病毒传播的重要手段之一, 不仅可以减少感染者数量和医疗负担, 同时也极大降低死亡率. 在 YOLOv4 框架基础上使用轻量化网络 E-GhostNet 代替原网络中的 CSPDarknet-53, E-GhostNet 网络在输入数据和原始 Ghost 模块生成的输出特征之间建立关系, 使网络能够捕获上下文特征. 然后, 在 E-GhostNet 中引入坐标注意力机制 (CA) 增强模型对有效特征的关注. 另外, 使用 *SIoU* 损失函数更换 *CIoU* 损失获得更快的收敛速度和优化效果. 最后, 结合 DeepSORT 多目标跟踪算法来检测和标记行人, 并使用仿射变换 (IPM) 判定行人间距离的违规行为. 实验结果显示, 该网络检测速度为 40 FPS, 精度值达到 85.71%, 相比原始 GhostNet 算法提升 2.57%, 达到实时行人距离检测的效果.

**关键词**: YOLOv4; DeepSORT; 社交距离; E-GhostNet; 轻量化网络; 目标检测

引用格式: 王林,张江涛.基于轻量化神经网络的社交距离检测.计算机系统应用,2023,32(2):128–138. http://www.c-s-a.org.cn/1003-3254/8942.html

## Social Distance Detection Based on Lightweight Neural Network

WANG Lin, ZHANG Jiang-Tao

(School of Automation and Information, Xi'an University of Technology, Xi'an 710048, China)

**Abstract**: Maintaining a safe social distance is one of the important means to effectively prevent the spread of the virus. Moreover, it can not only reduce the number of infected people and ease the medical burden but also greatly lower the mortality rate. On the basis of the you only look once version 4 (YOLOv4) framework, the lightweight network E-GhostNet is used to replace the CSPDarknet-53 in the original network. The E-GhostNet network establishes a relationship between the input data and the output features generated by the original Ghost module, thereby enabling the network to capture contextual features. Then, the coordinate attention (CA) mechanism is introduced to E-GhostNet to enhance the model's attention on effective features. In addition, the complete intersection over union (*CIoU*) loss function is replaced by the soft intersection over union (*SIoU*) loss function to obtain a faster convergence speed and optimization effect. Finally, the DeepSORT multi-target tracking algorithm is utilized to detect and label pedestrians, and affine transformation (IPM) is employed to determine the violation of the required distance between pedestrians. The experimental results show that the network achieves real-time pedestrian distance detection with a detection speed of 40 FPS and an accuracy of 85.71%, which is 2.57% higher than that of the original GhostNet algorithm.

**Key words**: YOLOv4; DeepSORT; social distance; E-GhostNet; lightweight network; object detection

新型冠状病毒于 2019 年被发现, 面对疫情所带来的巨大风险挑战, 动态清零政策一直是我们坚持的主要方针之一. 随着疫情进入稳定阶段, 全国各地偶尔出现疫情的反弹, 保障人民生命安全给防疫工作带来了

① 基金项目: 陕西省科技计划重点项目 (2017ZDCXL-GY-05-03)
收稿时间: 2022-06-23; 修改时间: 2022-07-25; 采用时间: 2022-08-15; csa 在线出版时间: 2022-10-28
CNKI 网络首发时间: 2022-11-16

严峻的考验. 定期的核酸检测、密集场所佩戴口罩以及保持安全距离是防止病毒传染的重要手段. 在火车站、购物中心和大学校园等人员密集场所保持安全距离对于预防或减缓病毒蔓延尤为重要. 为减少疫情的影响, 保持安全距离与自我隔离被认为是重启经济、打破感染链的最有效途径. 目前人流密集场所多数使用人工监测, 不仅需要大量人力, 且存在监测人员感染风险. 近年, 深度学习被广泛用于各种检测任务中, 并取得显著的效果. 因此, 基于深度学习的视频监控下社交距离检监测对常态化疫情防控具有重要意义和实践价值.

目标检测算法分为两阶段目标检测算法和一阶段目标检测算法. 两阶段目检测算法又叫做基于感兴趣区域的目标检测算法, 其第 1 个过程目的在于找到目标物体出现的位置生成预选框, 第 2 个过程对每个预选框进行分类和位置的修正, 主要的特点是精度高, 但速度较慢. 常见的双阶段目标检测算法有 Faster R-CNN[1]、R-FCN[2] 和 FPN[3] 等. 一阶段目标检测算法又叫基于回归的目标检测算法, 这类算法不直接生成感兴趣区域而将目标检测任务看做是对整幅图像的回归任, 直接产生目标物体的概率和位置信息. 主要的特点是速度相比两阶段的速度快, 但是精度有所损失. 常见的一阶段目标检测算法 YOLO[4–7]、SSD[8] 等.

随着深度学习技术的飞速发展, 国内外许多学者将此技术应用于社交距离检测任务中. 2020 年, 赵嘉晴[9] 用 YOLOv3 模型对行人社交距离判断, 将违反信息通过无线信号传送 OpenMv 模块, 提醒行人保持安全距离. Ramadass 等[10] 将训练好 YOLOv3 算法嵌入无人机摄像头中, 提醒公共场合人保持安全距离和口罩监测. Yadav[11] 设计通过树莓 pi4 与计算机结合的方法实现公共场所自动监控口罩佩戴和社交距离, 违反信息通过树莓 pi4 发送警局中. Rezaei 等[12] 开发了一个混合的计算机视觉和 YOLOv4 的深度神经网络模型, 用于室内和室外环境中使用普通闭路电视安全摄像头的人群自动检测. 该模型结合自适应逆透视映射 (IPM) 技术和排序跟踪算法, 实现了一种鲁棒的人群检测和社会距离监测.

2021 年, Ahmed 等[13] 使用 Faster-RCNN 在俯视图人类数据集上训练, 同时利用迁移学习, 将新训练层和预训练结构融合, 利用距离对像素信息的影响, 确定两个人是否违反社会距离. Saponara 等[14] 利用热图像结合 YOLOv2 算法对行人在室内和室外场景中进行社交距离分类. 杨森泉等[15] 设计一辆利用 PyramidBox 和 YOLOv3 模块进行口罩和社交距离的自主巡查车, 从而保障安全的社交距离. Li 等[16] 使用 YOLOv4 和 DeepSORT 多目标跟踪算法对行人进行检测, 在分析行人运动的基础上, 提出了一种新的行人聚类算法, 以避免同伴对监测结果的影响. 最后, 选取 3 个指标对行人进行分类, 分析和评价某一场所的病毒感染风险.

目前大多数算法模型相对较大且检测的速度比较慢, 大多数的检测都忽视了同伴的识别的判定. 因此, 设计出轻量化、实时性以及精确度之间相对平衡的模型框架检测行人之间社交距离, 且能够分辨出同伴的功能. 本文对公共场所行人安全距离进行检测, 针对模型大的问题, 在特征提取阶段使用更加轻量的 E-Ghost 瓶颈模块进行骨干的特征提取. 通过在骨干特征提取阶段引入坐标注意力模块, 实现对于候选区域特征中重要特征的突出以及噪声特征的抑制, 进而提升了特征的表达能力, 实现了更高的检测精度. 此外, 替代原有 YOLOv4 网络 SPP 模块为 SPPF 模块实现更加高效的检测效果. 在损失函数方面使用 *SIoU* 代替 *CIoU* 获得更快的收敛效果. 最终结合 DeepSORT 算法和 IPM 进行目标跟踪与像素距离的估计, 实现社交距离的防控和检测.

## 1 相关工作

### 1.1 YOLOv4 网络

YOLOv4 网络[7] 是单阶段目标检测算法整体包括 4 个主体部分. 输入端包括 Mosaic 数据增强、SAT 自对抗训练. 主干特征提取网络包括 CSPDarknet、Mish 激活函数等. 特征融合网络 Neck 在骨干和输出层之间的模块, 如 SPP 模块、FPN+PAN 结构. Head 用于目标检测的输出, 检测头分别检测大中小目标, 每个检测头预设 3 个先验框. 其中损失函数使用 *CIoU* 损失函数. 并使用 *DIoU_nms* 替代 NMS 提高算法的检测精度.

### 1.2 GhostNet 网络模型

Ghost 模块[17] 如图 1 所示, 先通过普通 1×1 卷积获得输入特征的充分浓缩, 接着对生成的特征图进行深度可分离卷积操作获得浓缩的相似特征图. 最后将 1×1 卷积与深度可分离卷积操作生成的特征图进行拼接操作, 得到新的输出.

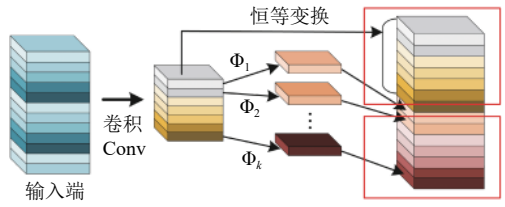



图 1 Ghost 模块

Ghost 瓶颈如图 2 所示, 第 1 层 Ghost 模块充当扩展层用于增加通的数目, 第 2 层 Ghost 模块减少通道的数目来满足残差路径 Add 操作. Ghost 瓶颈两种结构, 当需要对特征层的宽高进行压缩的时候, 设置这个 Ghost 瓶颈的 Stride=2, 即步长为 2.

## 2 改进的 YOLOv4 网络

与原本 YOLOv4 网络相比主要 3 部分改进: 一是采用轻量化 E-GhostNet 网络结构, 该模型增加计算量, 因此用 SPPF 结构替换 SPP 结构, 提高检测速度的同时获取更多的细节特征信息, 二是引入 CA 模块增强特征的关注度, 三是在 *CIoU* 损失函数基础增加面积尺度调节因子来调整边界框的损失值, 从而获得更好的优化效果. 改进后的 EC-YOLOv4 网络如图 3 所示. 其中, *K* 代表卷积核的大小, *s* 表示步长, *p* 表示填充, *c* 表示通道的个数.

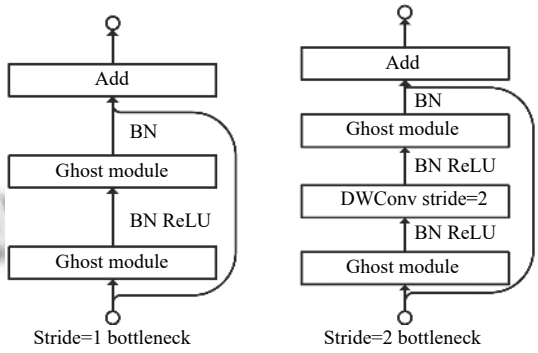



图 2 Ghost 瓶颈

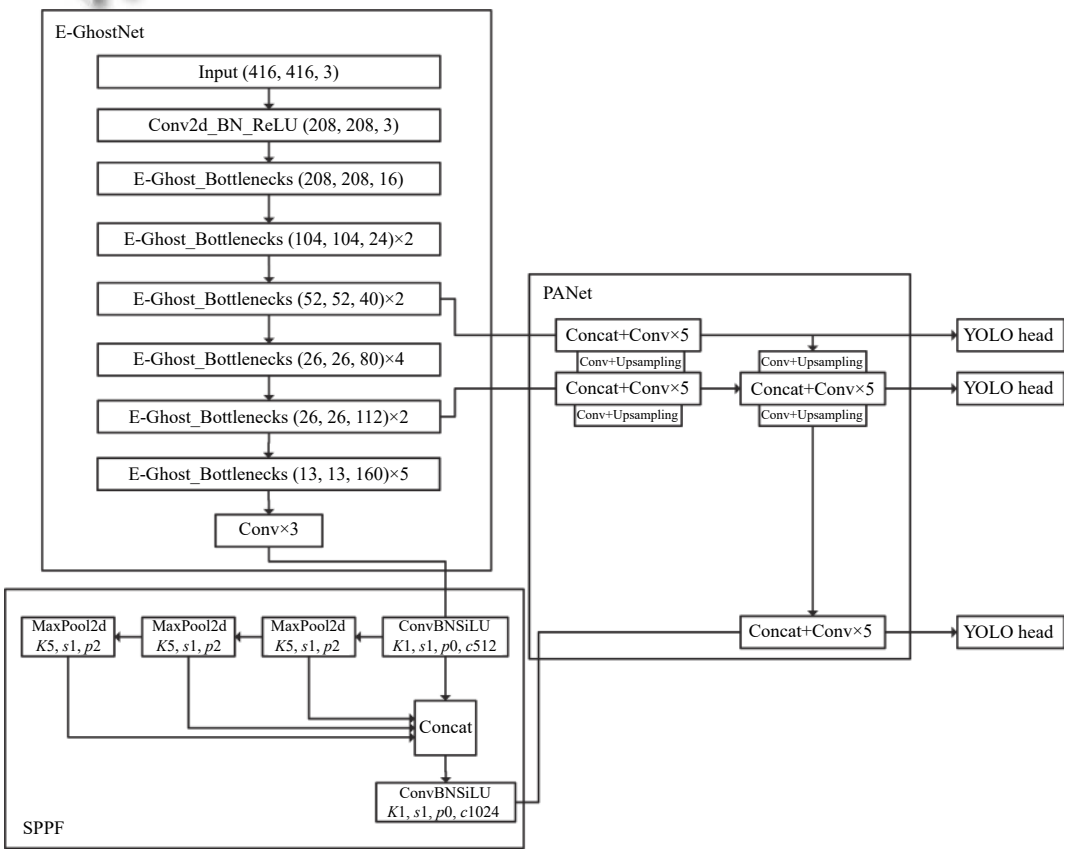



图 3 EC-YOLOv4 网络结构

### 2.1 Ghost 模块改进

E-Ghost 模块生成的输出特征融合了输入数据和原始 Ghost 模块生成的输出特征, 如图 4 所示. 由于 E-Ghost 模块能够捕获多层次的空间上下文特征, 增强了

网络的特征表达能力. 另一方面, 由于两个模块生成的输出特征数相等, 因此本文不向网络引入额外的计算. 但是将输入数据拼接到 Ghost 模块生成的输出特征中, 内存量增加, 从而影响网络的识别速度.

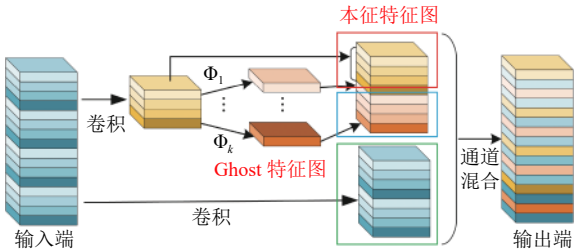



图 4 E-Ghost 模块

E-Ghost 瓶颈由两个 E-Ghost 模块组成的网络结构, 其余与原 Ghost 瓶颈网络结构一致, 如图 5 所示.

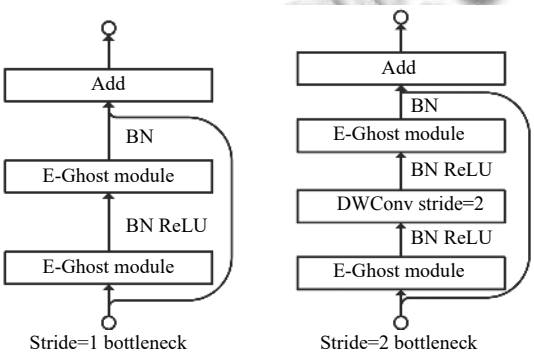



图 5 E-Ghost 瓶颈

### 2.2 引入坐标注意力网络模型

坐标注意力机制 (coordinate attention)[18] 是一种新颖且高效的注意力机制如图 6 所示, 通过嵌入位置信息到通道注意力, 从而使移动网络获取更大区域的信息而避免引入大的开销. 为了避免 2D 全局池化引入位置信息损失, 提出分解通道注意为两个并行的 1D 特征编码来高效地整合空间坐标信息到生成的注意特征图中.

具体而言, 用两个池化核 $(H, 1)$ 和 $(1, W)$ 沿着特征图的两个方向池化, 得到两个嵌入式的信息特征图, 沿空间维度拼接激活, 卷积激活后沿着空间维度进行 split 获得两个分离的特征图, 对其进行转换和激活, 最后得到注意力向量.

### 2.3 引入 SPPF 结构

SPP 网络[19] 如图 7 所示, 将输入并行通过多个不同大小的 MaxPool, 进一步融合一定程度上解决多目标尺度问题. SPPF 网络如图 8 所示, 将输入串行通过多个 5×5 大小的 MaxPool 层, 其中两个串行 5×5 大小的 MaxPool 层和一个 9×9 大小的 Maxpool 层计算结果是一样的, 串行 3 个 5×5 大小的 MaxPool 层是和一个 13×13 大小的 MaxPool 层计算结果是一样. 通过表 1, 可看出同数据同迭代次下, SPPF 的效率更高.

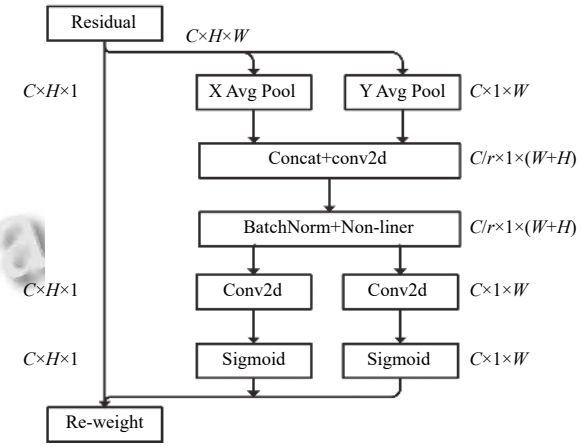



图 6 坐标注意力网络模型

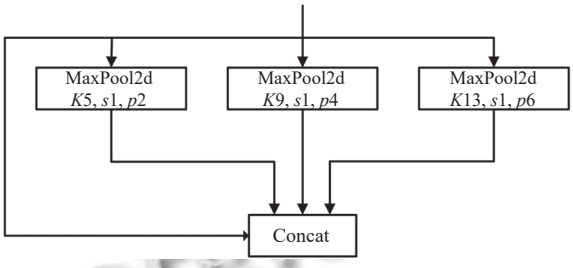



图 7 SPP 模块

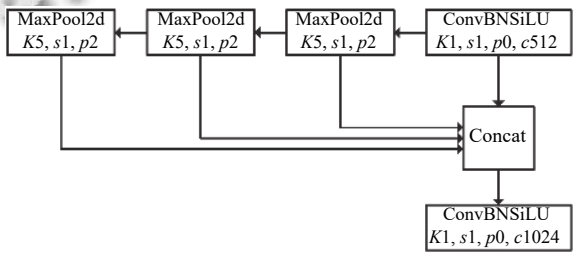



图 8 SPPF 模块

表 1 SPP 和 SPPF 网络结构运行时间对比 (s)

| 网络结构 | 运行时间 (100次) | 数据结果 |
|---|---|---|
| SPP | 0.537 305 116 653 442 4 | 相同 |
| SPPF | 0.207 807 064 056 396 48 | 相同 |

### 2.4 改进损失函数

*CIoU* 是目前比较优秀的回归定位损失函数, 它主

要考虑 3 种几何参数: 重叠面积、中心点距离、长宽比. *CIoU* 即在 *DIoU* 的基础上增加了检测框尺度的 *loss*, 增加了长和宽的 *loss*, 这样预测框就会更加的符合真实框, 如式 (1) 所示:

$$CIoU = IoU - \left(\frac{\rho^2(B,B^{gt})}{c^2} + \alpha v\right) \tag{1}$$

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

$$\alpha = \frac{v}{(1-IoU)+v} \tag{3}$$

其中, $B$表示预测边界框的中心坐标, $B^{gt}$表示真实边界框的中心坐标. $\rho^2(B,B^{gt})$代表$B$和$B^{gt}$之间欧式距离的平方. 两个边界框最小外接的对角线的长度用$c$表示. $\alpha v$代表目标边界的长宽比.

为了进一步修正 *CIoU* 损失, 从面积差的角度对 *CIoU* 损失进行修正, *CIoU* 在理论上无法区分所有的边界框信息并在大损失值点的周围的梯度趋于平滑, 因此在 *CIoU* 的基础上添加面积调节因子, 即 *SIoU* 损失[20], 如式 (4) 所示:

$$\begin{aligned} SIoU_{loss} &= (\gamma+1)(1-IoU) + \frac{\rho^2(B,B^{gt})}{c^2} + \alpha v \\ &= 1 - IoU + \frac{\rho^2(B,B^{gt})}{c^2} + \alpha v + \gamma(1-IoU) \\ &= CIoU_{loss} + \gamma(1-IoU) \end{aligned} \tag{4}$$

$$\gamma = \begin{cases} \left(\tanh\left(k\times\dfrac{s-s_{gt}}{s_{gt}} - 2.3\right) + \tanh(2.3)\right)\Big/2, & IoU > 0 \\ 0, & IoU = 0 \end{cases} \tag{5}$$

$$k = \begin{cases} 1.25, & s - s_{gt} \geqslant 0 \\ -1.25, & s - s_{gt} < 0 \end{cases} \tag{6}$$

其中, $\gamma$是引入新的几何因子代表面积差, 面积差与重叠面积并不相同. 当边界框完全覆盖目标或被完全覆盖时, 其面积差可描述为式 (7) 所示:

$$\begin{aligned} 面积差 &= \frac{|s-s_{gt}|}{s_{gt}} = |s/s_{gt} - 1| \\ &= \begin{cases} 1 - IoU, & s < s_{gt} \\ 1/IoU, & s \geqslant s_{gt} \end{cases} \end{aligned} \tag{7}$$

如图 9 所示, 两个边界框对同一个目标框 *IoU* 相同, 但是面积之间存在差别. 当图形中目标框的面积比例变化比较大时, 边界框和目标框之间会出现更多情况, 当每对边界框满足式 (8) 时, 现有的 3 个几何因子失去效果, 无法区分它们.

$$IoU_1 = IoU_2,\ w_1/h_1 = w_2/h_2,\ d_1/c_1 = d_2/c_2 \tag{8}$$

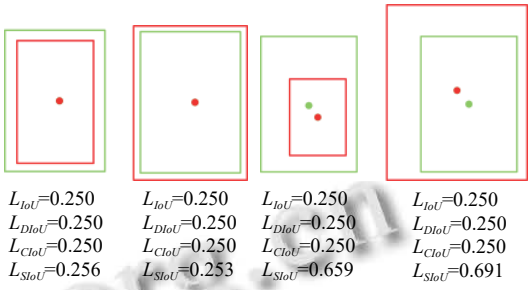



图 9 不同 *IoU* 损失对比

## 3 DeepSORT 多目标跟踪算法

DeepSORT 算法[21] 在 SORT 算法[22] 的基础上增加了级联匹配机制和新轨迹的确认. 轨迹分为确认态和不确认态, 新产生的轨迹是不确认态. 不确认态的轨迹必须要和检测连续匹配一定的次数 (默认 3) 才可以转化成确认态. 确认态的轨迹必须和检测连续失配一定次数 (默认 30 次), 才会被删除. 为解决 ID 切换和遮挡问题, DeepSORT 在 SORT 的基础上加入外观度量信息. 因此, 使用改进的网络同 DeepSORT 结合实现行人目标的检测和跟踪.

### 3.1 匈牙利算法

匈牙利算法是解决指派问题简单且新颖的一种方法. 主要是将检测和预测目标之间进行匹配. 具体是将融合运动特征和外观特征级联匹配形成代价矩阵求解得到最优解, 实现上下两帧检测和预测目标的匹配. 流程如算法 1.

算法1. 匈牙利算法

1) 发现每行最小元素, 每一行元素减去最小值;
2) 发现每列最小元素, 每一列元素减去最小值;
3) 用最少的行线和列线将新矩阵中的零全部穿起来, 检查目前是否为最优分配. 如果行线和列线没有将矩阵所有元素都穿起来, 进入第4步, 否则进入步骤5);
4) 将行线和列线没有穿起来的元素中找到最小元素, 将剩余元素减去最小元素, 对应行线和列线的交叉点的元素加上最小元素;
5) 找到每一行对应0元素和列对应的0元素. 根据0元素找到最优解.

### 3.2 卡尔曼滤波算法

最初, 跟踪的场景定义在八维状态空间上$(u,v,\gamma,h,\dot{u},\dot{v},\dot{\gamma},\dot{h})$, 其包含边界框的中心$(u,v)$, 长宽比例$\gamma$及高$h$它们各自在图像坐标系中的速度, 并取边界坐标

$(u,v,\gamma,h)$为目标的观测值. 通过卡尔曼滤波器进行预测和更新, 预测是利用上一帧检测框和运动速度等预测当前帧相应信息, 预测方程和协方差方程, 如式 (9) 和式 (10) 所示:

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_{k-1} \tag{9}$$

$$P_k^- = AP_{k-1}A^{\mathrm{T}} + Q \tag{10}$$

其中, $\hat{x}_k^-$为$k-1$时预测$k$时的状态向量; $\hat{x}_{k-1}$为$k-1$时刻的最优状态向量; $P_k^-$和$P_{k-1}$分别为与的协方差矩阵; $Q$为噪声的协方差.

测量更新, 计算卡尔曼增益, 如式 (11) 所示:

$$K_k = P_k^- H^{\mathrm{T}}\left(HP_k^- H^{\mathrm{T}} + R\right)^{-1} \tag{11}$$

其中, 增益$K_k$为$n\times m$阶矩阵, 目的是使后验估计误差协方差最小, $R$为观测噪声的协方差, $P_k^-$表示先验估计误差的协方差.

多目标跟踪使用一般忽略$u$控制输入得到最终更新结果, 如式 (12) 和式 (13) 所示:

$$\hat{x}_k = \hat{x}_k^- + K_k\left(z_k - H\hat{x}_k^-\right) \tag{12}$$

$$p_k = (I - K_k H)P_k^- \tag{13}$$

## 4 社交距离估计

通过使用单目摄像头, 使得三维世界场景投影到二维平面图像会导致对象之间存在不切实际的像素距离, 称为透视效应. 在透视效应中, 无法感知整个图像中距离的均匀分布, 如平行线在地平线相交, 距离相机远的人比距离相机坐标中心的人要小得多.

在三维空间中, 每个边界框的中心与$(x,y,z)$这 3 个参数有关, 但在相机成像中, 原始三维空间变为两个维度$(x,y)$深度参数$z$消失. 为了应用 IPM 转换, 需要设$z=0$来进行相机矫正, 达到消除透视效果[23]. 同时还需要摄像机的给固定参数. 例如摄像机位置、高度、视角等. 通过 IPM, 将 2D 像素点$(u,v)$映射到世界坐标$(X_w,Y_w,Z_w)$如式 (14) 所示:

$$[u,v,1] = KRT[X_w,Y_w,Z_w,1] \tag{14}$$

其中, $R$为旋转矩阵, $T$为转移矩阵, $K$摄像机固有参数矩阵.

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\dfrac{h}{\sin\theta} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}$$

$$K = \begin{bmatrix} f\times ku & s & c_x & 0 \\ 0 & f\times kv & c_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

其中, $h$是相机高度, $f$是焦距, $ku$和$kv$分别是以水平和垂直像素单位测量的校准系数值. $(c_x,c_y)$是校正图像平面光轴的主点偏移.

相机通过世界坐标上的三维点投影在视网膜平面上形成图像. 利用齐次坐标, 三维点与投影得到的图像点之间的关系如式 (18) 所示:

$$\begin{bmatrix} u \\ (v) \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{18}$$

其中, $M\in R^{3\times 4}$为式 (18) 中含有$m_{ij}$元素的变换矩阵, 由相机位置和参照系将世界坐标点映射到图像点, 由相机固由矩阵$K$、旋转矩阵$R$和平移矩阵$T$提供.

摄像机图像所在平面垂直于世界坐标系中的$Z$通道, 即$z=0$, 上述方程的尺寸可简化为式 (19) 所示:

$$\begin{bmatrix} u \\ (v) \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} X_w \\ (Y_w) \\ 1 \end{bmatrix} \tag{19}$$

透视空间到逆透视空间的转换可用以下的标量形式, 如式 (20) 所示:

$$(u,v) = \left(\frac{m_{11}\times x_w + m_{12}\times y_w + m_{13}}{m_{31}\times x_w + m_{32}\times y_w + m_{33}}, \frac{m_{21}\times x_w + m_{22}\times y_w + m_{23}}{m_{31}\times x_w + m_{32}\times y_w + m_{33}}\right) \tag{20}$$

### 4.1 同伴识别

移动行人主要是判断两者之间的距离. 距离太近会增加病毒感染的风险. 然而, 现实的生活中家人、朋友之间的陪伴是不可或缺的, 例如有孩子或者需要照顾的老人. 如果将他们判定为存在违规社交距离显示是不正确的, 他们之间不会增加的感染风险. 如果这类人群被视为危险人群, 感染风险更大, 这将增加人们对公共场所危险的恐慌. 因此, 将此看作为一个同组质点去计算行人距离, 满足如下条件.

(1) 两个行人之间的距离小于阈值, 不仅是同伴判断的前提, 同时也是社会距离检测的判断条件.

$$d\left(centroid^{(1)},centroid^{(2)}\right)\leqslant threshold^{(1)} \tag{21}$$

(2) 在最大记忆帧中, 若两行人速度方向$\vec{f}r_v$基本相同, 并且速度的大小$|fr_v|$也是基本一样的, 则可表示两个行人具有相同的运动状态.

$$\theta\left(\vec{f}r_v^{(1)},\vec{f}r_v^{(2)}\right)\leqslant threshold^{(2)} \tag{22}$$

$$d\left(\left|fr_v^{(1)}\right|,\left|fr_v^{(2)}\right|\right)\leqslant threshold^{(3)} \tag{23}$$

(3) 在前面条件的前提下, 如果两个行人的速度方向之和与距离的夹角大约是 90°. 可认为两行人是肩并肩行走.

$$\theta\left(\vec{f}r_v^{(1)}+\vec{f}r_v^{(1)},\vec{d}_v\right)\in[90°-\varepsilon,\ 90°+\varepsilon] \tag{24}$$

其中, $fr_v$表示帧速向量, 其大小是单位帧中像素的变化, 方向是从初始状态到最终状态. 它抽象地描述了行人的运动. $\vec{d}_v$表示距离向量, 其大小是两点之间的欧氏距离, 方向是一点指向另一点. 它描述了某一时刻两点之间的位置关系. $d(x,y)$欧氏距离是一个反映数据之间差异的标量. $\theta$矢量角是一个标量$\theta\in(0,180°)$.

## 5 社交距离检测系统

社交检测系统图主要由 3 部分构成: 一是目标检测模块本文采用 EC-YOLOv4 模型对视频帧进行行人检测, 获取边界坐标信息并计算坐标中心, 二是跟踪模块结合 DeepSORT 算法对行人跟踪并绘制行动轨迹, 三是行人距离模块使用 IPM 技术对行人的距离进行估计且达到能识别同伴的效果, 最终将违规信息显示出来便于之后相关人员的分析, 其系统框图如图 10.

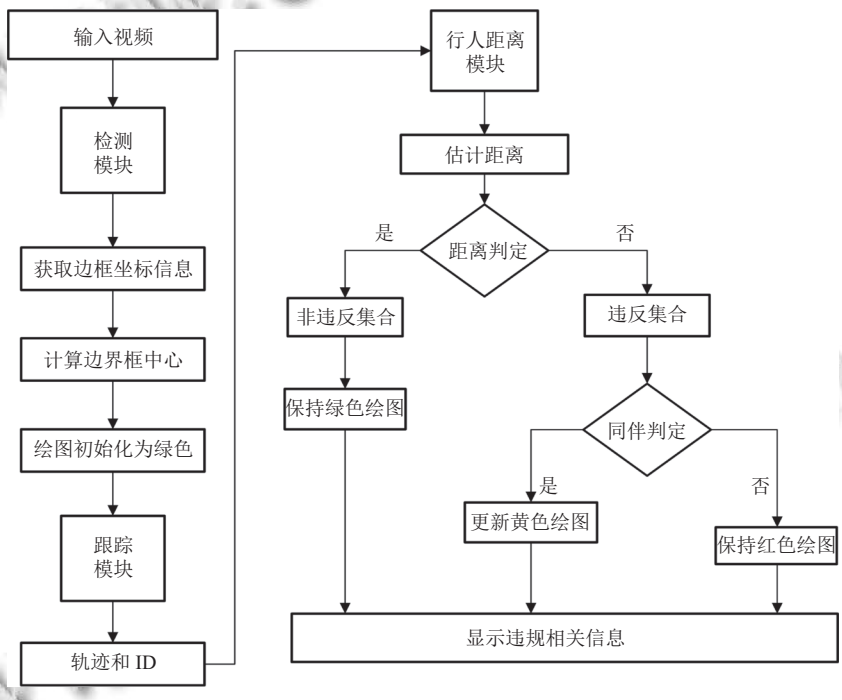



图 10 社交距离检测系统

## 6 实验结果与分析

### 6.1 实验环境

实验的环境配置如表 2 所示.

### 6.2 实验数据集和 Anchor 的重新聚类

本文从 UNN 数据集、ShanghaiTechCampus 数据集以及通过网络爬虫等, 总共选取 2 820 张图片, 其中训练使用 2 538 张图片, 测试使用 282 张图片. 如表 3 所示, 对数据集使用 K-means 重新聚类, 在 VOC07+12 的训练的基础之上进行迁移学习, 训练过程中参数设置如表 4 所示, 训练损失曲线结果如图 11 所示. 同时根据召回率 (Recall) 和精确度 (Precision) 绘制 P-R 曲线, 如图 12 所示, 行人的 AP 为 85.71% (class: 85.71%= person AP), AP 表示曲线包围的面积大小可衡量类别的好坏.

实验的评价指标采用 mAP (mean average precision)、视频每秒传输的帧数 (frame per second, FPS).

其中 mAP 的值高说明模型性能好, FPS 数值越大模型的实时检测能力越强.

表 2 实验环境及其配置

| 名称 | 环境配置 |
|---|---|
| 操作系统 | CentOS 7.0 |
| 处理器 | Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.20 GHz |
| 显卡 | GeForce GTX 1080Ti (×4), 11 GB |
| 深度学习框架 | PyTorch 1.7.0 |
| 集成开发环境 | Anaconda, CUDA 10.2 |

表 3 K-means 聚类

| 特征层 | 原始Anchor值 | 修正Anchor值 |
|---|---|---|
| 52×52 | (12, 16), (19, 36), (40, 28) | (7, 26), (11, 45), (14, 66) |
| 26×26 | (36, 75), (76, 55), (72, 146) | (19, 83), (30, 65), (26, 112) |
| 13×13 | (142, 110), (192, 243), (459, 401) | (44, 150), (82, 234), (175, 323) |

表 4 训练参数

| 参数 | 数值 | 参数 | 数值 |
|---|---|---|---|
| 总训练epoch | 300 | 最小学习率 | 1E–2 |
| 冻结骨干 | 50 | 优化器 | SGD |
| Batch_size | 16 | 权重衰减 | 5E–4 |

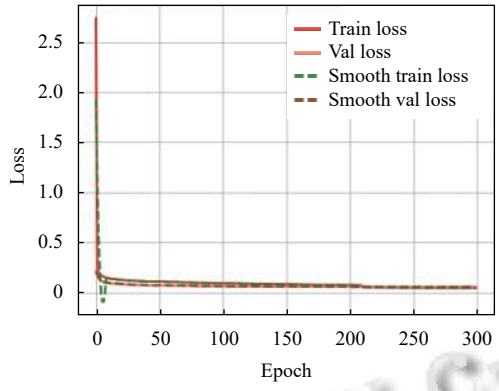



图 11 训练损失

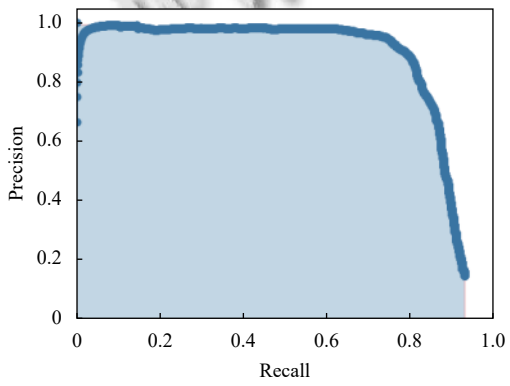



图 12 P-R 曲线

### 6.3 消融实验

为了验证本文算法的有效性和可靠性采用 mAP 和 FPS 作为消融实验的评价指标, 以 GhostNet-YOLOv4 算法为基准, 并将改进分为 A、B、C、D 四组进行消融实验. 其中, “√”表示在网络使用该改进方法, “×”表示在网络不使用该改进方法.

通过表 5 可看出, A 组使用 E-Ghost 模块作为主干网络后, 模型精度提升了 1.38%, 检测速度下降了 2 FPS, 这是由于将输入数据拼接到 Ghost 模块生成的输出特征, 使得特征浓缩更高 mAP 提升, 同时此模型增加计算量导致检测速度下降; B 组在 A 组的基础引入 CA 模块, 精度提升了 0.53%, 检测速度下降 1 FPS; C 组在 B 的基础上使用 SPPF 模块, 模型精度提升 0.04%, 检测速度保持稳定; D 组在 C 组的基础上使用 *SIoU* 损失函数, 模型精度提升 0.62%, 检测速度下降 1 FPS; 总体上, D 组相比 A 组模型精度提升 2.57%, 检测速度下降 4 FPS, 满足实时性的基本需求.

表 5 消融实验结果对比

| 算法 | E-Ghost | CA | SPPF | *SIoU* | mAP@0.5 (%) | 检测速度 (FPS) |
|---|---|---|---|---|---|---|
| GhostNet-YOLOv4 | × | × | × | × | 83.14 | 44 |
| A | √ | × | × | × | 84.52 | 42 |
| B | √ | √ | × | × | 85.05 | 41 |
| C | √ | √ | √ | × | 85.09 | 41 |
| D | √ | √ | √ | √ | 85.71 | 40 |

### 6.4 EC-YOLOv4 与其他算法对比

通过表 6 分析可得, 本文提出算法在行人检测方面比 MobileNetv3-YOLOv4、YOLOv4-tiny、GhostNet-YOLOv4 算法精度分别提升 3.12%, 7.93%, 2.57%. 模型大小和参数量大小相对较小, 但检测速度相比 GhosNet-YOLOv4 减少 4 FPS, 与 YOLOv4 算法相比检测精度低 6.81%, 但是检测速度比其高 23 FPS. 如图 13 所示, 对比发现在 YOLOv4-tiny、MobileNetv3-YOLOv4、GhostNet-YOLOv4 中都存在不同程度漏检, 同时 YOLOv4-tiny 中存在少量误检, 本文算法几乎正确检测到所有行人. 实验表明, 改后网络在轻量化网络中检测精度和检测速度方面取得不错的效果.

通过校园拍摄数据分别对 3 种算法结合 DeepSORT 进行测试, 将视频集转为帧统计跟踪结果, 如表 7 所示, 分析得 C 算法在跟踪方面相比 A 和 B 算法有一定的提升, 跟踪的准确率达到 91.34%, 分别提升了 8.65% 和 6.73%, 基本达到对视频行人的检测和跟踪. 其中 A、B、C 分别为 MobileNetv3-YOLOv4、Ghost-

Net-YOLOv4 及 EC-YOLOv4 算法. 人员数量是视频出现的人数, 跟踪准确率是跟踪人数占人员数量的比例.

本文算法通过结合使用 DeepSORT 对行人在校园和公开牛津市中心数据集进行跟踪测试, 并对其进行行动轨迹绘制和 ID 标记. 通过如图 14 所示, 可以看出本文算法在检测和跟踪都取得不错的表现.

表 6 不同网络数据对比

| 算法 | mAP@0.5 (%) | 参数量 | 模型大小 (MB) | 检测速度 (FPS) |
|---|---|---|---|---|
| MobileNetv3-YOLOv4 | 82.59 | 11 729 069 | 44.74 | 48 |
| YOLOv4 | 92.52 | 64 363 101 | 245.53 | 17 |
| YOLOv4-tiny | 77.78 | 6 056 606 | 23.10 | 105 |
| GhostNet-YOLOv4 | 83.14 | 11 428 545 | 43.60 | 44 |
| EC-YOLOv4 | 85.71 | 10 216 628 | 38.97 | 40 |

(a) YOLOv4-tiny (b) MobileNetv3-YOLOv4

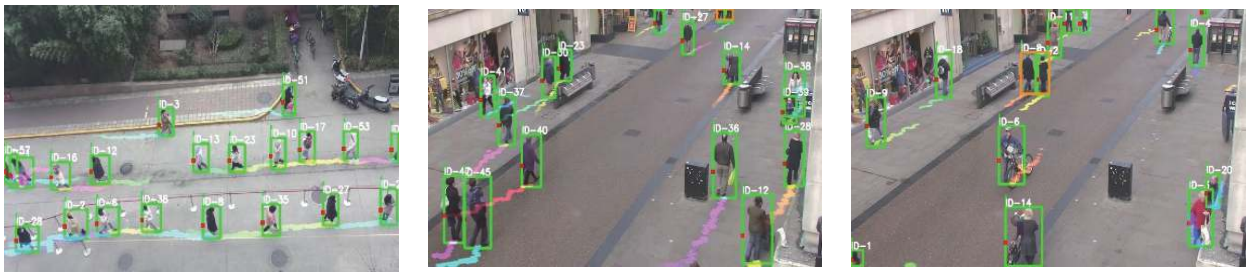
(c) GhostNet-YOLOv4 (d) 改进算法

图 13 不同算法检测对比

表 7 不同检测器跟踪结果对比

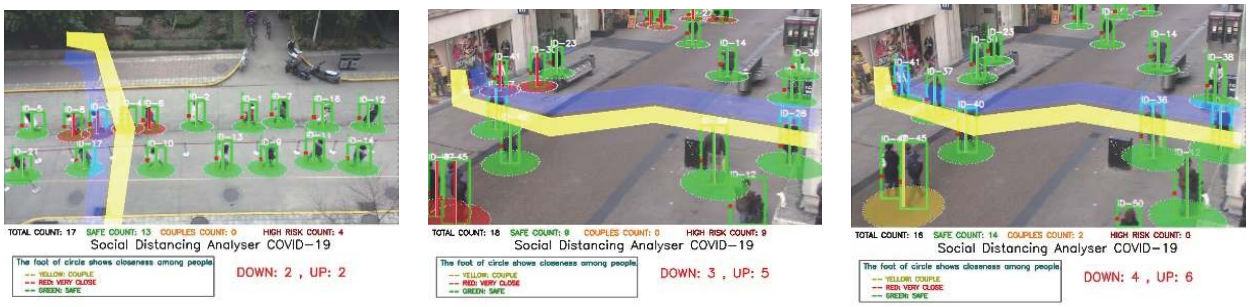
| 算法 | 人员数量 | 跟踪人数 | 跟踪准确率 (%) | ID切换数量 |
|---|---|---|---|---|
| A | 104 | 86 | 82.69 | 12 |
| B | 104 | 88 | 84.61 | 11 |
| C | 104 | 95 | 91.34 | 9 |

距离检测在跟踪的基础上通过 IPM 实现, 假设社交的安全距离为 1.2 m, 成人身高为 1.7 m, 通过行人身高所占图片的像素值, 结合 IPM 实现距离的估计. 如图 15 所示, 其中图 15(a) 是校园环境中做核酸检测的场景, 共检测到 17 个行人, 安全 13 个行人, 4 个行人处于高风险, 此帧上行和下行通过紫色与黄色撞线分别为 2 个行人. 图 15(b) 和图 15(c) 在公开牛津市中心数据集测试, 在满足距离检测判定的同时, 最大的不同是可以识别出同伴且看作一个整体去估计距离. 这些数据的统计结合跟踪轨迹便于分析此区域感染风险的程度, 同时也可以提醒行人保持安全的社交距离. 其中红、绿、黄圈分别表示违反社交距离、安全距离以及同伴组.



(a) 校园 (b) 牛津市中心 (c) 同伴识别

图 14 跟踪结果




(a) 校园 (b) 牛津市中心 (c) 同伴识别

图 15 距离检测结果

## 7 结论与展望

本文提出一种基于深度学习的社交距离检测和跟踪方法. 检测模型中在 YOLOv4 的基础上替换轻量化骨干网络 GhostNet 并对其改进为 E-GhostNet 便于捕获多层次的空间上下文特征, 增强了网络的特征表达能力. 同时, 在骨干中引入坐标注意力模块, 提升检测性能. 然后, 引入 SPPF 模块减少 E-GhostNet 中 Concat 对网络的识别速度影响. 其次是将损失函数有 *CIoU* 改为 *SIoU* 加快网络的收敛速度, 从而提升检测性能. 最后结合 DeepSORT 跟踪算法和 IPM 将 2D 像素点映射到 3D 世界, 最终估计出行人间的距离. 本文提出的算法可有效地平衡检测精度、速度以及网络模型的大小, 平均精度为 85.71%, 检测速度为 40 FPS, 模型总参数量为 10 216 628. 该算法对视频流测试发现, 其在行人的检测、跟踪及距离检测 3 方面都取得良好的效果, 对减少病毒传播发挥着重要作用. 在未来的工作中可以结合行人的姿态特征, 如咳嗽、打喷嚏等与感染病毒相关的行为综合分析社交距离存在的危险等级, 以便更好地保障人民生命以及减轻医疗负担.

## 参考文献

1 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]

2 Dai JF, Li Y, He KM, *et al*. R-FCN: Object detection via region-based fully convolutional networks. Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona: ACM, 2016. 379–387.

3 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. 2117–2125.

4 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.

5 Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6517–6525.

6 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv: 1804.02767, 2018.

7 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv: 2004.10934, 2020. [doi: 10.48550/arXiv.2004.10934.]

8 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.

9 赵嘉晴, 易映萍, 黄松. 基于 YOLOv3 的无人机智能社交距离监测系统. 软件, 2020, 41(12): 107–112. [doi: 10.3969/j.issn.1003-6970.2020.12.025]

10 Ramadass L, Arunachalam S, Sagayasree Z. Applying deep learning algorithm to maintain social distance in public place through drone technology. International Journal of Pervasive Computing and Communications, 2020, 16(3): 223–234. [doi: 10.1108/IJPCC-05-2020-0046]

11 Yadav S. Deep learning based safe social distancing and face mask detection in public areas for COVID-19 safety guidelines adherence. International Journal for Research in Applied Science and Engineering Technology, 2020, 8(7): 1368–1375. [doi: 10.22214/ijraset.2020.30560]

12 Rezaei M, Azarmi M. DeepSOCIAL: Social distancing monitoring and infection risk assessment in COVID-19 pandemic. Applied Sciences, 2020, 10(21): 7514. [doi:

10.3390/app10217514]

13 Ahmed I, Ahmad M, Jeon G. Social distance monitoring framework using deep learning architecture to control infection transmission of COVID-19 pandemic. Sustainable Cities and Society, 2021, 69: 102777. [doi: 10.1016/j.scs.2021.102777]

14 Saponara S, Elhanashi A, Gagliardi A. Implementing a real-time, AI-based, people detection and social distancing measuring system for COVID-19. Journal of Real-time Image Processing, 2021, 18(6): 1937–1947. [doi: 10.1007/s11554-021-01070-6]

15 杨森泉, 陈泳豪, 张镇宇, 等. 口罩佩戴检测和社交安全距离预警巡查车设计. 单片机与嵌入式系统应用, 2021, 21(7): 79–81.

16 Li JX, Wu ZA. The application of YOLOv4 and a new pedestrian clustering algorithm to implement social distance monitoring during the COVID-19 pandemic. Journal of Physics: Conference Series, 2021, 1865: 042019. [doi: 10.1088/1742-6596/1865/4/042019]

17 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.

18 Hou QB, Zhou DQ, Feng JS. Coordinate attention for efficient mobile network design. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 13708–13717.

19 He KM, Zhang XY, Ren SQ, *et al*. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904–1916. [doi: 10.1109/TPAMI.2015.2389824]

20 Du SJ, Zhang BF, Zhang P, *et al*. An improved bounding box regression loss function based on CIOU loss for multi-scale object detection. Proceedings of 2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning. Chengdu: IEEE, 2021. 92–98. [doi: 10.1109/PRML52754.2021.9520717]

21 Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. Proceedings of 2017 IEEE International Conference on Image Processing (ICIP). Beijing: IEEE, 2017. 3645–3649. [doi: 10.1109/ICIP.2017.8296962]

22 Bewley A, Ge ZY, Ott L, *et al*. Simple online and realtime tracking. Proceedings of 2016 IEEE International Conference on Image Processing (ICIP). Phoenix: IEEE, 2016. 3464–3468. [doi: 10.1109/ICIP.2016.7533003]

23 Rezaei M, Klette R. Vision-based driver-assistance systems. In: Rezaei M, Klette R, eds. Computer Vision for Driver Assistance. Cham: Springer, 2017. 1–18. [doi: 10.1007/978-3-319-50551-0_1]

(校对责编: 孙君艳)