DOI:10.19651/j.cnki.emt.2108650

基于改进 YOLO 和 DeepSORT 的实时多目标跟踪算法*

黄凯文[1] 凌六一[1,2] 王成军[2] 吴 起[2] 李学松[1]
(1.安徽理工大学电气与信息工程学院 淮南 232001; 2.安徽理工大学人工智能学院 淮南 232001)

摘 要: 针对基于检测的两步多目标跟踪算法模型结构复杂、实时性差的问题,提出一种基于改进 YOLOv4-Tiny 和 DeepSORT 的实时多目标跟踪算法。在 YOLOv4-Tiny 算法中引入深度可分离卷积,压缩模型计算量;将检测分支增加至 3 个,并使用多尺度特征融合结构以降低对小目标的漏检率;利用改进的 GC 注意力模块,加强网络对全局上下文特征的提取能力。跟踪部分使用 DeepSORT 算法,使用匀加速卡尔曼滤波优化其行人运动模型,利用浅层分类网络重构其外观模型,最后在 MOT16 测试序列中实验。结果表明,改进算法的总参数量为 4.2 M,较原算法减少 52%且 MOTA 增加 5.2%,GPU 下处理时间加快,单 CPU 时能达到平均每秒 11 帧的跟踪速度,能满足低算力设备对跟踪任务精度和速度的要求。

关键词: 图像处理;多目标跟踪;YOLOv4-Tiny;DeepSORT;深度可分离卷积
中图分类号: TP391 **文献标识码:** A **国家标准学科分类代码:** 520.60

Real-time multiple object tracking algorithm based on improved YOLO and DeepSORT

Huang Kaiwen[1] Ling Liuyi[1,2] Wang Chengjun[2] Wu Qi[2] Li Xuesong[1]
(1. School of Electrical and Information Engineering, Anhui University of Science and Technology, Huainan 232001, China;
2. School of Artificial Intelligence, Anhui University of Science and Technology, Huainan 232001, China)

Abstract: To solve the issue of complicated structure and low real-time performance of the two-step multiple object tracking by detection algorithm, a real-time multiple object tracking algorithm founded on modified YOLOv4-Tiny and DeepSORT algorithm is proposed. The depthwise separable convolution is employed in YOLOv4-Tiny, to reduce the calculation of the model. The detection branches are increased to 3, and multi-scale feature fusion structure is built to decrease the missed ratio of tiny objects. The modified GC attention module is used to extract the global context features of the model. In the tracking part, the pedestrian motion model of DeepSORT is optimized and the appearance model is reconstructed, the detection and tracking algorithms are combined and experimented in MOT16 test sequences finally. The results show that the total parameters of the improved algorithm are 4.2 M, 52% less than the original algorithm and 5.2% more MOTA, faster processing speed under GPU, and the tracking speed of an average of 11 frames per second can be achieved under a single CPU, which can fulfill the requests of precision and speed for multiple object tracking mission in low-calculation devices.

Keywords: image processing; multiple object tracking; YOLOv4-Tiny; DeepSORT; depthwise separable convolution

0 引 言

多目标跟踪(MOT)是计算机视觉领域的研究热点,广泛应用于视频监控、安防、智能交通等领域[1-3]。多目标跟踪算法可分为基于检测跟踪和无检测跟踪两种类型,二者的主要区别是前者依赖于检测器的输出,后者需要在第一帧中手动初始化跟踪器的数量。在目标跟踪中,相继出现以 Faster R-CNN[4]、Mask R-CNN[5]等 two-stage 检测算法为代表的跟踪框架,和以 SSD[6]、YOLO[7-9]等 one-stage 检测算法为代表的跟踪框架,与 two-stage 需要预产生候选区域

收稿日期:2021-12-22
* 基金项目:安徽省高校协同创新项目(GXXT2019-018)、安徽省重点研发计划项目(201904a05020092)资助

不同，one-stage 直接利用 CNN 反馈目标的位置和种类信息，检测速度更快。相较于 YOLOv3 等 YOLO 系列算法，YOLOv4 的各项检测指标都有一定的提升，但这些模型参数和计算量较大，在算力和功耗有限的移动和嵌入式设备上使用是一个巨大的挑战。目前，应对该问题的方法主要有两类：1）对训练好的大模型进行处理，如剪枝、量化、知识蒸馏等；2）设计轻量化网络结构，如 ShuffleNet[10]、MobileNet[11]、GhostNet[12]等。

在多目标跟踪领域，文献[13]提出 SORT 算法，跟踪部分使用卡尔曼滤波器进行状态预测并联合 IOU 构建代价矩阵，然后使用匈牙利算法检测框和轨迹关联，跟踪速度快，但没有考虑框内目标特征，容易发生身份切换。文献[14]提出 POI 算法，使用改进后的 Faster R-CNN 进行检测，跟踪部分引入行人重识别网络以减少行人身份切换，使用卡尔曼滤波和重识别网络的输出向量构建相似矩阵，最后结合 KM 算法将检测框和轨迹关联，跟踪精度高，但实时性略差。文献[15]提出 JDE 算法，将行人重识别模型和检测器相结合，直接利用检测网络提取特征，之后再结合卡尔曼滤波和数据关联算法进行匹配，整体算法实时性有所增强，但在行人相互遮挡的情况下，检测精度降低且模型难以端到端训练优化。文献[16]提出 TubeTK 算法，使用 3 维卷积来提取多目标跟踪任务中的空间维度与时间维度信息，并进行回归，能有效解决遮挡、无法利用运动信息和端到端训练困难等问题，但在 MOT16 上的速度仅在 1 Hz 左右。DeepSORT[17]算法在 SORT 的基础上，加入了浅层的深度表观特征提取网络，大大降低了身份切换，提升了跟踪精度。

针对目前多目标跟踪算法难以实时运行在一般低算力设备上的问题，提出一种轻量型多目标跟踪算法。在检测部分选用高效的 YOLOv4-Tiny 算法，并引入深度可分离卷积进一步减少其计算量；为了降低对小目标的漏检率，在主干部分 52×52 特征层增加一条检测分支；并使用改进的 GC 注意力结构来校准通道注意力权重。在跟踪部分选用 DeepSORT 算法，使用匀加速代替匀速卡尔曼滤波模型，降低行人运动不确定性带来的影响；使用轻量级的重识别模型替换原模型，减少跟踪器数量增加对速度的影响。最后将两种算法相结合，完成对多目标的跟踪任务。与改进前相比，算法的模型参数量少、精度高、推理速度快，适合部署在参与构建物联网的移动终端设备中。

1　算法原理及改进

1.1　YOLOv4-Tiny 算法

YOLOv4-Tiny 是 YOLOv4 的简化版，模型计算量更小，能够在移动端或嵌入式设备中实时运行。主干部分 CSPDarknet53-Tiny 负责提取输入图像的位置信息，与主干部分相连的两个检测分支获取图像的深层语义信息。每个分支将图像分割成大小不同的单元格，每个单元格都会预测 B 个检测框的位置和置信度信息，可分别用 (x,y,w,h,c) 来表示，其中 (x,y) 为目标中心相对于左上角的坐标，(w,h) 为目标的宽高，c 代表置信度。若数据集包含 k 个种类，最后网络将会输出 $B\times(5+k)$ 个值，用来生成检测结果。

1.2　YOLOv4-Tiny 算法改进

1）深度可分离卷积

深度可分离卷积将标准卷积分成两个模块，第 1 个模块为深度卷积，其中每一个卷积核只负责一个通道；第 2 个模块为逐点卷积，使用 1×1 卷积计算不同通道的线性加权信息。在保证输出特征图大小不变的情况下，将空间信息和通道信息分开表示，降低了卷积操作的参数量。

假设输入的特征图的宽、高、维数为 $W\times H\times M$，输出特征图的宽、高、维数为 $W\times H\times N$，卷积核大小为 $K\times K$，标准卷积的计算量为：

$$W\times H\times M\times N\times K\times K \tag{1}$$

深度可分离卷积的计算量为：

$$W\times H\times M\times K\times K+M\times N\times K\times K \tag{2}$$

二者计算量之比为：

$$\frac{W\times H\times M\times K\times K+M\times N\times K\times K}{W\times H\times M\times N\times K\times K}=\frac{1}{N}+\frac{1}{W\times H} \tag{3}$$

在检测网络中，使用深度可分离卷积替代标准卷积可降低网络的计算量，提升推理速度。

2）注意力机制

在 GCNet[18]和 ECANet[19]的基础上，本文构建了新的注意力模块 GCA(global context attention)。GCA 使用一维卷积替换原 GC 模块中的两层 1×1 二维卷积，通过显式建模卷积特征层各通道之间的相互依赖性，重新校准特征。

如图 1 所示，其中 C、H、W 分别为输入特征图 $\boldsymbol{x}=\{x_i\}_{i=1}^{N_p}$ 的通道数、长、宽，$N_p=H\cdot W$。当输入为 $\boldsymbol{x}$ 时，上下文模块先使用 1×1 卷积 $\boldsymbol{w}_k$ 和 $softmax$ 标准化函数来获取全局线性加权信息 $HW\times1\times1$，再将 $C\times HW$ 特征向量与其矩阵乘操作，进行全局注意力池化，得到初步全局上下文特征 $C\times1\times1$；转换模块通过一维卷积 $\boldsymbol{w}_v$ 捕获每个通道及其 y 个相邻通道的交互信息来重新校准注意力权重，最后将权重信息与输入 $\boldsymbol{x}$ 加权得到输出 $\boldsymbol{z}$。表达式为：

$$z_i=x_i+\boldsymbol{w}_v\sum_{j=1}^{N_p}\frac{\exp(\boldsymbol{w}_k x_j)}{\sum_{m=1}^{N_p}\exp(\boldsymbol{w}_k x_m)}x_j \tag{4}$$

式中：i 是查询位置的索引，j 代表特征图中的某点，$\boldsymbol{w}_k$、$\boldsymbol{w}_v$ 为线性转换矩阵。$\boldsymbol{w}_v$ 由一维卷积构成，考虑到通过交叉验证或手动调优的方式计算 y 的值，会耗费计算资源，y 由式(5)方程自适应决定。

$$y=\left|\frac{\log_2(C)}{\gamma}+\frac{b}{\gamma}\right|_{odd} \tag{5}$$

式中：$|t|_{odd}$ 表示向上取离 t 最近的奇数；b、γ 分别设置为 1、2。

为了不增加模型的推理负担,只在每个分支与主干部分相连的位置插入 GCA,以增加 YOLOv4-Tiny 对深层语义特征的捕捉与调整,弥补自身网络特征提取能力的不足。

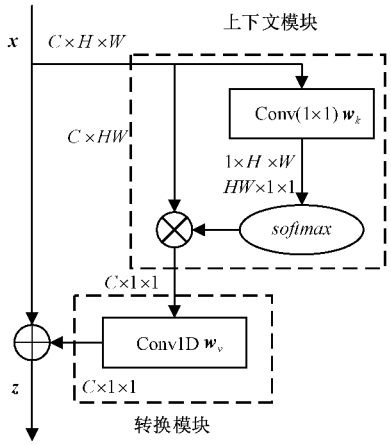

图 1　GCA 注意力结构

3)多尺度特征融合

由于构建的数据集中包含不同尺度的行人标注框,部分图片中行人的密度高,目标小。YOLOv4-Tiny 两个预测分支的特征层相当于由输入压缩 32 倍和 16 倍得到,直接在这两个特征层上利用预定义锚框回归目标位置并重映射到原图中,会损失坐标信息。为了缩短信息的传播路径,并充分利用主干部分提取的位置信息,如图 2 所示,在主干部分的 52×52 特征层引出另一分支,采用多尺度特征融合(multi-scale feature fusion, MFF)策略,将 3 个分支提取的特征进行上、下路径聚合,以解决模型由多尺度、多角度行人目标引起精度下降的问题。为了减少引入的参数,将 3 个分支和下采样中的卷积都替换成深度可分离卷积。

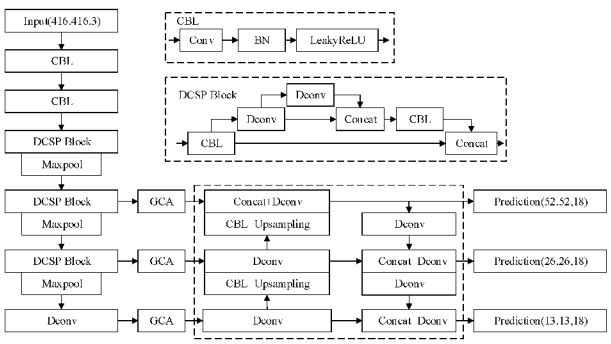

图 2　改进的 YOLOv4-Tiny 模型结构

4)改进的 YOLOv4-Tiny 结构

如图 2 所示,其中 Dconv 表示深度可分离卷积,在主干部分将 CSP 结构主分支中卷积核大小为 3×3 的标准卷积 Conv 替换为 Dconv,构成 DCSP Block;并引出一个分支,在 3 个分支与主干相连的地方插入 3 个 GCA 注意力模块;分支之间采用自上而下的上采样特征聚合路径和自下而上的下采样特征聚合路径互相连接,除上采样时采用的 1×1 卷积,其他卷积都替换成 Dconv。

1.3　DeepSORT 算法

跟踪目标时,DeepSORT 建立的运动模型会计算检测框与滤波器预测框之间的马氏距离,表达式为:

$$\boldsymbol{d}^{(1)}(i,j)=(\boldsymbol{d}_j-\boldsymbol{y}_i)^{\mathrm{T}}\boldsymbol{S}_i^{-1}(\boldsymbol{d}_j-\boldsymbol{y}_i) \tag{6}$$

式中:$\boldsymbol{d}_j$ 表示第 j 个检测框,$\boldsymbol{y}_i$ 表示为第 i 条轨迹的滤波器预测框,$\boldsymbol{S}_i$ 表示 i 条轨迹之间的标准差矩阵。

当相机运动或目标长时间被遮挡时,外观模型发挥作用。对于每个检测框 $\boldsymbol{d}_j$,特征提取网络会计算一个 128 维且 $\|\boldsymbol{r}_j\|=1$ 特征向量,并对每个目标构建一个成功关联最近 $L_k=100$ 帧的向量 $\boldsymbol{R}_k=\{\boldsymbol{r}_k^{(i)}\}_{k-1}^{L_k}$,计算这些向量与当前帧第 i 个检测框特征向量的最小余弦距离,表达式为:

$$\boldsymbol{d}^{(2)}(i,j)=\min\{1-\boldsymbol{r}_j^{\mathrm{T}}\boldsymbol{r}_k^{(i)}\mid \boldsymbol{r}_k^{(i)}\in\boldsymbol{R}_k\} \tag{7}$$

分别对两种模型设置阈值,通过调整式(8)中的超参数 λ 来控制组合指标对关联代价的影响,最后通过匈牙利算法进行帧与帧之间多个目标的匹配,完成跟踪任务。

$$\boldsymbol{c}_{i,j}=\lambda\boldsymbol{d}^{(1)}(i,j)+(1-\lambda)\boldsymbol{d}^{(2)}(i,j) \tag{8}$$

1.4　DeepSORT 算法优化

1)匀加速卡尔曼滤波模型

DeepSORT 算法使用一个恒速线性(constant velocity, CV)的卡尔曼滤波器来对行人轨迹状态进行预测和更新,状态向量 $\boldsymbol{X}$ 由八维空间向量 $(u,v,r,h,\dot{u},\dot{v},\dot{r},\dot{h})^{\mathrm{T}}$ 所组成,其中 (u,v) 为预测框的中心坐标,r 表示预测框的长宽比,h 表示预测框的高度,其他的 4 个向量表示相应的速度分量。CV 模型建立在行人匀速直线运动的基础上,但现实中行人可能会受不确定因素的影响,有停顿或加速等运动趋势,CV 模型并不可取。本文引入了加速度参数分量构成匀加速(constant accceleration, CA)模型来替代 CV 模型,在原 $\boldsymbol{X}$ 中增加四维 $(\ddot{u},\ddot{v},\ddot{r},\ddot{h})^{\mathrm{T}}$,与其他滤波模型相比,CA 模型能满足实时跟踪的需要。令 $\boldsymbol{C}=(u,v,r,h)^{\mathrm{T}}$,$\boldsymbol{v}=(\dot{u},\dot{v},\dot{r},\dot{h})^{\mathrm{T}}$,$\boldsymbol{a}=(\ddot{u},\ddot{v},\ddot{r},\ddot{h})^{\mathrm{T}}$,则有表达式如下:

$$\begin{cases}\boldsymbol{X}_t=\begin{pmatrix}\boldsymbol{C}\\ \boldsymbol{v}\\ \boldsymbol{a}\end{pmatrix}_t=\begin{pmatrix}1 & \Delta t & \dfrac{\Delta t^2}{2}\\ 0 & 1 & \Delta t\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\boldsymbol{C}\\ \boldsymbol{v}\\ \boldsymbol{a}\end{pmatrix}_{t-1}+\boldsymbol{w}_t\\ \boldsymbol{Z}_t=(1\quad 0\quad 0)(\boldsymbol{C}\quad \boldsymbol{v}\quad \boldsymbol{a})_t^{\mathrm{T}}+\boldsymbol{v}_t\end{cases} \tag{9}$$

式中:$\boldsymbol{w}_t$ 为过程噪声,$\boldsymbol{v}_t$ 为观测噪声,$\boldsymbol{Z}_t$ 为观测状态。

卡尔曼滤波通过过去的状态和当前的测量值来不断更新矫正行人的运动轨迹,从而获得好的跟踪效果。卡尔曼滤波算法的时间更新方程如下:

$$\begin{cases}\boldsymbol{X}_t'=\boldsymbol{F}\boldsymbol{X}_{t-1}+\boldsymbol{B}\boldsymbol{u}_{t-1}\\ \boldsymbol{P}_t'=\boldsymbol{F}\boldsymbol{P}_{t-1}\boldsymbol{F}^{\mathrm{T}}+\boldsymbol{Q}\end{cases} \tag{10}$$

式中:$\boldsymbol{F}$ 为行人的状态转移矩阵,$\boldsymbol{P}$ 为预测状态协方差矩阵,$\boldsymbol{Q}$ 为状态转移协方差矩阵,$\boldsymbol{B}$ 为控制矩阵。

卡尔曼滤波算法的状态更新方程如下：

$$\begin{cases} \boldsymbol{K}_t = \boldsymbol{P}'_t\boldsymbol{H}^{\mathrm{T}}(\boldsymbol{H}\boldsymbol{P}_t\boldsymbol{H}^{\mathrm{T}}+\boldsymbol{R})^{-1} \\ \boldsymbol{X}_t = \boldsymbol{X}'_t+\boldsymbol{K}_t(\boldsymbol{Z}_t-\boldsymbol{H}\boldsymbol{X}') \\ \boldsymbol{P}_t = (\boldsymbol{I}-\boldsymbol{K}_t\boldsymbol{H})\boldsymbol{P}'_t \end{cases} \tag{11}$$

式中：$\boldsymbol{H}$ 为观测矩阵，$\boldsymbol{R}$ 为观测噪声的协方差矩阵，$\boldsymbol{I}$ 为单位矩阵，$\boldsymbol{X}$ 为最佳估计值，$\boldsymbol{K}_t$ 为卡尔曼系数。

2)行人重识别模型

在 DeepSORT 外观模型中，构建了一个离线的神经网络模型 WRN(wide residual network)来提取行人的特征向量，但 WRN 的参数多、计算量大，当跟踪器的数量增加时，跟踪速度会受到影响。为了更好平衡算法的推理速度和精度，受 VOVNet[20] 启发，本文使用 OSA (onc-shot aggregation)架构代替 WRN 中的堆叠残差块，参考 WRN 中残差块输出特征图的大小对其结构进行调整，如图 3 所示，先对输入特征图进行最大池化处理，将其长、宽压缩为原来的 1/2，维数保持不变，接着进行 3 次 Dconv 操作，使用的激活函数为 ReLU6，然后将上述操作产生的特征图进行拼接，最后使用 1×1 Conv 进行特征聚合并降维输出。OSA 模块包含多个感受野的多样化特征，通过拼接的方式，在输出前聚合所有特征来达到特征复用的目的。

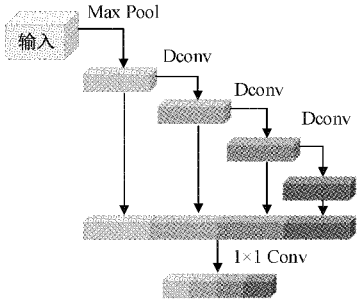

图 3　OSA 模块结构

VOVNetV1 行人重识别模型如表 1 所示，网络的输入尺寸为 120×60，使用堆叠 OSA 模块提取特征，最后经过平均池化处理和 L_2 标准化生成 128 维特征向量，用于计算最小余弦距离。

表 1　VOVNetV1 网络结构组成

名称	输出尺寸
Conv 1	32×120×60
Conv 2	32×120×60
OSA 1	64×60×30
OSA 2	128×30×15
OSA 3	128×15×8
Average Pool	128×1×1
L_2 Normalization	128

2　实验与分析

2.1　实验准备

实验环境：在内存大小为 32 G，GPU 为 NVIDIA GeForce RTX 3060(6 G)，CPU 为 AMD Ryzen 7 5800H 的笔记本电脑上进行实验，使用的深度学习框架为 Pytorch。

数据集和预训练：在目标检测部分，分别在 CrowdHuman、Caltech Pedestrian Dataset、MOT2020、MOT2017 数据集中筛选共 15 017 张图片作为数据集，按 8∶1∶1 将数据集分为训练、验证和测试集。训练改进的 YOLOv4-Tiny 时，选用 Adam 优化器和余弦退火衰减法来调整学习率，学习率由 1×10^{-3} 降至 1×10^{-5}，批量大小为 16，共训练 150 epochs。在跟踪部分，选用 Market-1501 数据集训练重识别模型，并进行数据增强处理，采用 SGD 优化器，批量大小为 64，学习率设为 1×10^{-1}，每隔 20 个 epoch 降低 0.1，共训练 60 epochs。

2.2　评价指标

选取平均精度 AP(IOU=0.5)作为检测模型的指标。目标跟踪部分，选取 MOTChallenge 定义的评价指标，MOTA(↑)：多目标跟踪精度；IDF1(↑)：正确识别的检测和计算的检测数的比值；MT(↑)：主要跟踪目标百分比；ML(↓)：主要丢失目标百分比；IDS(↓)：目标身份发生切换的次数；其中(↑)表示数值越高跟踪效果越好，(↓)则相反。选取每秒传输帧数 FPS、神经网络模型参数量 Paras 和模型的浮点运算数 FLOPs 作为共同指标。

2.3　实验结果与分析

1)注意力结构对比试验

为了验证 GCA 模块的有效性，将其与 SE[21] 和 GC 注意力结构加入到检测算法中，进行对比试验。如表 2 所示，其中 V4Tiny 是加入深度卷积和多尺度融合后的 YOLOv4-Tiny。3 个注意力模块都能提升模型的检测精度，但 GCA 比 SE 和 GC 模块拥有更少的参数和计算量，而且对 AP 的提升更大。在对输入特征图进行处理时，GCA 和 GC 都使用上下文模块的方式进行降维，而不是和 SE 一样使用全局池化来获取通道注意力，避免了信息的丢失。在捕捉非线性的跨通道交互时，GC 和 SE 都使用全连接的方式，而 GCA 使用一维卷积，效果更好，复杂度也更低。

表 2　加入注意力结构的实验结果对比

模型	AP/%	参数增量	计算量增量
V4Tiny	77.8	0	0
V4Tiny+SE	78.0	43 008	649 600
V4Tiny+GC	78.2	174 051	782 397
V4Tiny+GCA	78.3	914	613 725

2)目标检测算法对比试验

选取主流的检测算法进行对比试验，如表 3 所示，YOLOv4 和采用 ResNet50 作为主干的 Faster R-CNN 在精度上分别比本文改进的检测算法高 8.6%和 4.39%，但他们的参数量和计算量都非常大，不够高效，也无法在一般设备上实时运行。使用 MobileNetV2 替换 YOLOv4 主干部分，

计算量减少,但整体性能低于改进模型。与 YOLOv3-Tiny 和 YOLOv4-Tiny 相比,改进模型的精度分别提高 6%和 3.6%,模型的参数量分别减少 54%和 32%。在 GPU 和 CPU 上推理时间的增加,主要原因是引入深度可分离卷积和多尺度融合,导致 1×1 卷积数量和卷积层数的增加,提升了系统的内存访问成本,影响 GPU 并行处理的高效性。

表 3 目标检测算法效果对比

算法	AP/%	输入尺寸	Paras/M	FLOPs/G	FPS(GPU)	FPS(CPU)
YOLOv4	86.9	416	64.0	29.9	35.6	1.5
Faster R-CNN	82.7	600	28.3	250.0	10.3	0.3
YOLOv4-MobileNetV2	76.6	416	10.4	3.8	58.9	6.8
YOLOv3-Tiny	72.3	416	8.7	2.8	134.5	15.4
YOLOv4-Tiny	74.7	416	5.9	3.4	140.9	16.5
改进 YOLOv4-Tiny	78.3	416	4.0	2.4	121.3	14.8

3)改进多目标跟踪算法消融试验

选取 MOT16 训练序列对改进算法进行消融实验,如表 4 所示,将 YOLOv4-Tiny+DeepSORT 算法作为基线,引入深度可分离卷积后,YOLOv4-Tiny 的参数量降低 25%,在 GPU 和 CPU 上的整体推理速度提升,但 MOTA 下降了 0.9%;加入多尺度融合后,模型对小目标的检测能力增强,MOTA 上升了 5.8%,仅增加少量推理时间,跟踪质量得到较大提升;增加的 3 个 GCA 模块在几乎不增加模型复杂度的条件下,提升了跟踪质量;跟踪部分,将 DeepSORT 中 CV 替换为 CA 模型后,MOTA 上升 1.6%,MT 上升了 1.6%,ML 降低了 1.3%,实时性有所降低;用 VOVNetV1 替换 WRN 后,降低了 MMF 和 CA 模型增加的推理负担,维持跟踪精度的同时,压缩参数量,提升了 FPS。相较于基线,改进后算法的 MOTA 由 37.4%上升至 44.2%,IDF1 上升了 3.8%,MT 提高了 7.2%,ML 降低了 19%,参数量压缩了 52%,整体推理速度得到提升,但在增加对小目标检测能力的同时,IDs 发生次数有所增加。

表 4 改进模块对跟踪结果的影响

Method	MOTA/%	IDF1/%	MT/%	ML/%	IDs	Paras/M	FPS(GPU)	FPS(CPU)
Baseline	37.4	45.1	11.6	50.1	358	8.7	31.2	11.2
+Dconv	36.5	45.4	11.4	52.0	347	6.5	33.1	12.1
+MFF	42.3	50.2	16.0	35.1	550	6.8	28.7	10.2
+GCA	42.6	50.6	16.4	34.5	531	6.8	28.6	10.1
+CA	44.2	51.2	18.0	33.2	516	6.8	28.0	10.0
+VOVNetV1	44.2	48.9	18.8	31.1	470	4.2	32.9	10.8

4)多目标跟踪算法对比试验

在 MOT16 测试序列与其他 4 种经典且性能较好的跟踪算法对比,如表 5 所示,MOTDT 算法本质上与 DeepSORT 类似,但其使用全卷积网络组成的编码和解码器对行人的特征进行提取和分类,然后用非极大值抑制对 ROI 进行处理,将检测和跟踪的结果都用于数据关联,整体跟踪效果较本文算法好,但速度略低。DMAN 算法将单目标跟踪的思想应用到多目标跟踪,引入时间和空间注意力机制来对目标进行关联与跟踪,虽然效果好,但采用的 ResNet50 和 LSTM 网络需要较高的算力,在一般设备上达不到实时的效果,在相同 GPU 上处理速度在 0.3 帧左右。OTCD_1 算法设计了以 R-FCN 架构为主干的检测器,将含有目标的关键帧和非关键帧分别处理,保持跟踪性能的同时提升速度。GMPHD_ReId 算法将高斯混合概率假设密度滤波器和 CNN 建立的外观模型结合用于解决在线跟踪的遮挡问题,速度快,但整体跟踪效果较本文算法略差。与经典算法相比,本文算法在精度相近的情况下,推理速度更快。

表 5 多目标跟踪算法效果对比

算法	MOTA/%	IDF1/%	MT/%	ML/%	IDs	FPS(GPU)
Baseline	40.1	41.2	13.3	48.7	782	29.6
MOTDT[22]	47.6	50.9	15.2	38.3	792	20.6
DMAN[23]	46.1	54.8	17.4	42.7	532	0.3
OTCD_1[24]	44.4	45.6	11.6	47.6	759	17.6
GMPHD_ReId[25]	40.4	50.1	11.5	43.1	789	31.6
本文算法	45.3	43.1	15.9	40.2	973	33.9

3　结　　论

本文在 YOLOv4-Tiny 和 DeepSORT 算法的基础上进行改进，提出一种能实时运行在低算力设备上的多目标跟踪算法。首先，针对 YOLOv4-Tiny 对小目标检测能力差的问题，增加一条检测分支并使用多尺度融合策略；使用 GCA 注意力模块，加强网络对上下文特征的提取能力；在主干和分支中引入深度可分离卷积，进一步降低模型的复杂度；并选取多个数据集训练，增加模型的泛化能力。在跟踪部分，使用 CA 模型替代原运动模型以减少由行人运动不确定性带来的影响；使用 VOVNetV1 行人重识别网络，在保证精度的前提下，加快跟踪速度。最后，将二者结合，在 MOT16 上测试，结果表明，相较于改进前，整体算法的参数量和计算量更少，多目标跟踪精度也得到了较大的提升。与其他几种算法相比，本文算法也能实现速度与精度的平衡，完成复杂场景下的跟踪任务。

参考文献

[1] 李建良，张婷婷，陶知非，等. 基于改进 Camshift 与 Kalman 滤波融合的领航车辆跟踪算法[J]. 电子测量与仪器学报，2021，35(6)：131-139.

[2] 张相胜，沈庆. 基于改进 YOLOv3 的多目标跟踪算法研究[J]. 激光与光电子学进展，2021，58(16)：190-200.

[3] 董美琳，任安虎. 基于深度学习的高速公路交通事件检测研究[J]. 国外电子测量技术，2021，40(10)：108-116.

[4] 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.

[5] He K，GKIOXARI G，P DOLLÁR，et al. Mask R-CNN[C]. Proceedings of the IEEE International Conference on Computer Vision，2017：2961-2969.

[6] LIU W，ANGUELOV D，ERHAN D，et al. SSD：Single shot multi-box detector [C]. European Conference on Computer Vision，Springer，Cham，2016：21-37.

[7] REDMON J，FARHADI A. YOLO9000：Better，faster，stronger[C]. IEEE Conference on Computer Vision and Pattern Recognition，IEEE Press，2017：6517-6525.

[8] REDMON J，FARHADI A. YOLOv3：An incremental improvement[J]. ArXiv Preprint，2018，ArXiv：1804.02767.

[9] BOCHKOVSKIY A，WANG C Y，LIAO H. YOLOv4：Optimal speed and accuracy of object detection[J]. ArXiv Preprint，2020，ArXiv：2004.10934.

[10] ZHANG X，ZHOU X，LIN M，et al. ShuffleNet：An extremely efficient convolutional neural network for mobile devices[C]. Proceedings of IEEE conference on computer vision and pattern recognition. Washington D. C，USA：IEEE Press，2018：6848-6856.

[11] SANDLER M，HOWARD A，ZHU M，et al. MobileNetV2：Inverted residuals and linear bottlenecks[C]. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C，USA：IEEE Press，2018：4510-4520.

[12] HAN K，WANG Y，TIAN Q，et al. GhostNet：More features from cheap operations[C]. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D. C，USA：IEEE Press，2020：1580-1589.

[13] BEWLEY A，GE Z，OTT L，et al. Simple online and realtime tracking [C]. International Conference on Image Processing. Phoenix：IEEE，2016：3464-3468.

[14] YU F W，LI W B，LI Q Q，et al. Poi：Multiple object tracking with high performance detection and appearance feature [C]. European Conference on Computer Vision. Springer，2016：36-42.

[15] WANG Z D，ZHENG L，LIU Y X，et al. Towards real-time multi-object tracking [C]. European Conference on Computer Vision. Springer，2020：107-122.

[16] PANG B，LI Y Z，ZHANG Y F，et al. Tubetk：Adopting tubes to track multi-object in a one-step training model [C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition，2020：6308-6318.

[17] WOJKE N，BEWLEY A，PAULUS D. Simple online and real-time tracking with a deep association metric[C]. 2017 IEEE International Conference on Image Processing(ICIP)，2017：3645-3649.

[18] CAO Y，XU J，LIN S，et al. Global context networks[C]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2020，DOI：10.1109/TPAMI.2020.3047209.

[19] WANG Q，WU B，ZHU P，et al. ECA-Net：Efficient channel attention for deep convolutional neural networks [C]. Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)，Seattle，WA，USA，14 June 2020.

[20] LEE Y，PARK J. Centermask：Real-time anchor-free instance segmentation[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition，2020：13906-13915.

[21] JIE H，LI S，GANG S，et al. Squeeze-and-excitation

networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020,42(8):2011-2023.

[22] CHEN L, AI H, ZHUANG Z, et al. Real-time multiple people tracking with deeply learned candidate selection and person re-identification [C]. IEEE International Conference on Multimedia & Expo. IEEE Computer Society, 2018:1-6.

[23] JI Z, HUA Y, NIAN L, et al. Online multi-object tracking with dual matching attention networks[C]. In Proceedings of the European Conference on Computer Vision(ECCV), 2018: 366-382.

[24] LIU Q, LIU B, WU Y, et al. Real-time online multi-object tracking in compressed domain [J]. IEEE Access, 2019: 76489-76499.

[25] BAISA N L. Occlusion-robust online multi-object visual tracking using a GM-PHD filter with a CNN-based re-identification [J]. ArXiv Preprint, 2019, ArXiv:1912.05949.

作者简介

黄凯文,硕士研究生,主要从事计算机视觉方面的研究。
E-mail:2604775893@qq.com

凌六一,教授,博士生导师,主要从事检测技术与智能信息处理方面的研究。
E-mail:lyling@aust.edu.cn