控制与决策
Control and Decision

基于帧内关系建模和自注意力融合的多目标跟踪方法

朱姝姝, 王欢, 严慧

引用本文:
朱姝姝,王欢,严慧. 基于帧内关系建模和自注意力融合的多目标跟踪方法[J]. 控制与决策, 2023, 38(2): 335−344.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2021.1188

基于帧内关系建模和自注意力融合的多目标跟踪方法

朱姝姝, 王 欢†, 严 慧

(南京理工大学 计算机科学与工程学院，南京 210094)

摘 要: 多目标跟踪在视频监控领域有重要的应用价值. 随着卷积神经网络(convolutional neural networks, CNN), 尤其是图神经网络(graph neural networks, GNN)的发展, 多目标跟踪的研究现阶段取得了很大突破. 其中, 图神经网络由于引入目标-轨迹间的关系建模, 显示出更稳定的跟踪性能. 然而, 已有的基于GNN的多目标跟踪方法都仅在连续两帧之间建立全局关系模型, 忽视了帧内目标与周围其他目标的交互, 没有考虑在帧内建立合适的局部关系模型. 为了解决该问题, 提出基于帧内关系建模和自注意力融合模型(INAF-GNN)的多目标跟踪方法. 在帧内, INAF-GNN建立目标与邻居目标的关系图模型以获取局部跟踪特征; 在帧间, INAF-GNN建立目标与轨迹关系图模型以获得全局跟踪特征, 并利用注意力机制设计一个特征融合模块整合局部和全局跟踪特征. 在MotChallenge行人标准数据集上进行大量的实验, 与多个基于图神经网络的多目标跟踪方法相比较, 结果显示, MOTA指标提高1.9 %, IDF1指标提高3.6 %. 同时, 在UA-DETRAC车辆数据集上的验证测试表明了所提出方法的有效性和泛化能力.

关键词: 多目标跟踪；图神经网络；数据关联；帧内目标关系建模

中图分类号: TP273 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2021.1188

引用格式: 朱姝姝, 王欢, 严慧. 基于帧内关系建模和自注意力融合的多目标跟踪方法[J]. 控制与决策, 2023, 38(2): 335-344.

Multi-object tracking based on intra-frame relationship modeling and self-attention fusion mechanism

ZHU Shu-shu, WANG Huan†, YAN Hui

(School of Computer Science and Engineering，Nanjing University of Science and Technology，Nanjing 210094，China)

Abstract: Multi-object tracking is a crucial technique of video surveillance. Over the past decade, the convolutional neural networks (CNNs) and especially graph neural networks (GNNs) have made multi-object tracking a great progress, where the GNN show an significant advantages due to modeling the relationship between targets and trajectories. These GNN models, however, mostly consider building a global relationship model for targets and trajectories only in two neighboring frames, neglecting the interactions between an object with the others within a frame. In order to handle this issue, we propose an intra-frame relationship modeling and self-attention fusion method for multi-object tracking. Within a frame, the INAF-GNN builds a relational graph model for an object and its neighboring objects to obtain local tracking features. Across two frames, the INAF-GNN constructs another relational graph model for objects and trajectories to acquire global tracking features. In further, both the local and global tracking features are fed into a feature integration module via a self-attention mechanism. We run various experiments on the pedestrian MotChallenge benchmark datasets, and the experimental results show that the proposed method outperforms GNN-based multi-object tracking methods by 1.9 % of MOTA and 3.6 % of IDF1. Besides, it is also validated over the vehicle UA-DETRAC datasets. Both demonstrate the effectiveness and generalization capability of the proposed method.

Keywords: multi-object tracking；graph neural networks；data association；intra-frame relationship model

0 引 言

多目标跟踪(multiple object tracking, MOT)的目的是在视频序列的每一帧中确定所有目标位置, 保持其身份信息不变. 它是一个经典计算机视觉问题, 其应用包括自动驾驶、机器人导航和视频监控[1]等. MOT中的跟踪一般分为在线跟踪和离线跟踪两种模

收稿日期: 2021-07-07；**录用日期:** 2021-11-26.
基金项目: 国家自然科学基金项目(61703209, 61773215).
责任编委: 陈家伟.
†通讯作者. E-mail: wanghuanphd@njust.edu.cn.

式,主要区别在于是否利用未来帧的信息处理当前帧目标. 在线跟踪要求在处理每一帧时,只能利用当前帧和历史帧的信息决定当前帧的跟踪结果,不能根据当前帧的信息修改历史帧的跟踪结果. 离线跟踪则允许利用所有帧的信息获得全局最优解. 本文工作属于在线多目标跟踪.

目前,已有工作将图神经网络引用到多目标跟踪任务中,利用GNN[2]将特征提取与数据关联模块结合起来,在性能上有了很大提升. 这些方法[3-6]使用CNN学习特征,将GNN用于数据关联模块,并将外观信息和运动信息嵌入图的拓扑结构中,利用图结构的特性使目标节点的信息可以传播、学习和更新,进而得到全局跟踪特征. 但是,这些方法[3-6]只基于单个目标的特征在连续帧之间建立相似度模型,没有考虑帧内目标与周围目标之间的关系,忽略了目标局部关系建模,从而为整体跟踪性能的提升留下一些空间. 图1给出一个行人多目标跟踪场景. ①为相邻两帧图片;②为使用单个目标计算的相邻两帧间目标相似度值;③为建立交互信息后相邻帧间目标相似度值. 如图1所示,加入帧内图模型后,同一目标(穿黑色上衣白色裙子的女子)相似度值从0.67上升至0.87,不同目标(同样穿黑色西装的两名男子)相似度值从0.69下降至0.32,目标的区分性提高了. 可见,在有遮挡以及不同目标(行人)有相似的外观情况下,跨帧之间建立的图模型使用单个目标特征计算相似度值是不可靠的,增加帧内目标与邻居目标关系建模有助于提高相似度计算的准确性.

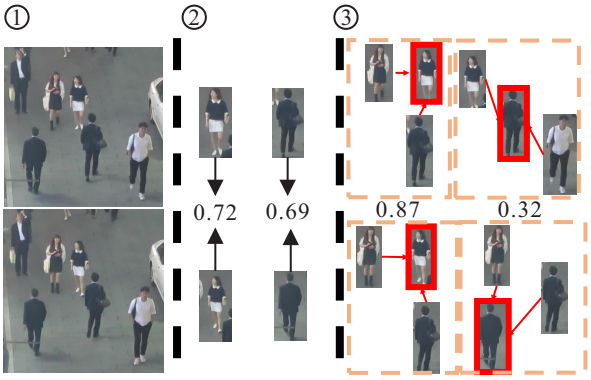


图1 行人多目标跟踪场景

在目标与邻居目标关系建模方面已进行了一些工作. 文献[7]考虑了帧内目标与周围目标的交互,在帧内对目标建立了有向图,通过计算轨迹帧内有向图与检测帧内有向图每个节点之间的相似度值,最终得到两个图之间的相似度值,通过该值计算成本矩阵. 文献[8]利用目标和邻居的特征值,聚合周围目标特征,利用聚合后的特征提高关联的准确性. 文献[7-8]虽然考虑了帧内目标的关系交互,但均忽视了目标的帧间全局跟踪特征.

与以上文献工作不同,本文提出一种新的端到端的多目标跟踪模型,同时考虑帧内目标之间的交互关系和帧间目标与轨迹的关联关系. 在帧内,建立目标与邻居目标关系图模型以获取局部跟踪特征;在帧间,建立目标与轨迹关系图模型以获得全局跟踪特征,并利用注意力机制设计一个特征融合模块,以整合局部和全局跟踪特征. 其次,在帧内建模方面,本文在使用目的和使用方式上也与文献[7-8]不一样:1)使用目的不同. 文献[7-8]都只利用帧内关系,所以帧内建模仅起到计算成本矩阵进行数据关联的作用. 而本文帧内关系建模不仅参与成本矩阵计算,而且能够与帧间跟踪特征融合提高跟踪鲁棒性. 其原因是,基于图网络的多目标跟踪方法[3-6]一般仅考虑单独使用由帧间信息传递所得到的全局跟踪特征,不可避免地会使得节点特征同质化严重,容易导致跟踪不鲁棒,而本文方法通过引入局部跟踪特征缓解这一问题,因此,如何更好地融合局部和全局跟踪特征变得尤其重要,鉴于此,引入被广泛证明有较强特征融合能力的自注意力机制完成这一任务. 2)使用方式不同. 本文方法在帧内建模后,利用欧氏距离选取最近的几个邻居目标(实验得出2个邻居目标最佳),同时,仅使用欧氏距离作为权重值,将卷积后的局部特征与帧间全局特征融合后进行数据关联.

本文贡献如下:

1)提出一个端到端的多目标跟踪框架,将帧内目标与周围邻居目标之间建立关系图模型,结合目标的局部跟踪特征和全局跟踪特征,使得最终学习到的目标特征更具判别性和表示能力.

2)在融合每个目标的局部跟踪特征和全局跟踪特征时,采用注意力机制[7]使局部和全局跟踪特征分别学到各自的权重值,加权求和后得到最终的目标特征表示.

3)在MOT17行人数据集上验证所提出方法,与当前基于图的多目标跟踪方法和一些主流的非图网络的多目标跟踪方法相比较,在绝大部分指标上均取得最好的结果. 同时,在UA-DETRAC车辆数据集上进行验证测试,结果表明所提出方法具有较强的泛化能力.

1 相关工作

1.1 多目标跟踪

MOT大部分均为基于检测的跟踪[5]. 每一帧首先做目标检测,然后在特征提取模块,提取目标的外

观和运动特征,这些特征之后被用于相似度值的计算. 数据关联时,将目标分为不同组,在保持目标实现一对一的关联约束时,能够最大化全局相似性.

在多目标跟踪里,首先需要提取可靠的特征表示. 目前,已有很多方法被用于外观特征的提取,包括传统的手工提取特征、可学习的特征[9]以及深度特征[10-11],其中深度特征方法使用孪生网络[11-12]、自动编码器[13]、相关滤波器[14]和空间注意力[15]等提取边框内目标的外观特征.

当外观处于被频繁遮挡或严重变化时,需要加入其他信息以保证跟踪的准确性,运动信息应用最为广泛. 对于运动特征的提取,可以利用卡尔曼滤波[16]、LSTM[17]等. 提取到外观特征和运动特征后,计算相似度时也有很多方法,有余弦相似性、欧氏距离、多层感知机等. 最后,利用计算的特征相似度做关联时,通常使用多种方法进行处理,如匈牙利算法[4]、动态规划[18]和强化学习[19]. 尽管上述方法能够提高多目标跟踪的性能,但是多目标跟踪在改进上还有很大的空间,上述方法的一个缺陷是没有考虑目标与周围环境的交互.

1.2　图神经网络

传统的神经网络在工作时,输入的数据是结构化数据,如语音、图像、文本等. 而图神经网络则是一种对非结构化数据进行操作的神经网络模型,是一个可以从图到图关系推理计算的结构化体系模型,社交网络、知识图谱、复杂的文件系统等都属于非结构化数据. 图是不规则的,每个图都有一组大小可变的无序节点,图中每个节点有不同数量的相邻节点[20],利用图的层次结构可以通过边将其他邻居节点信息聚合起来,用于捕获实例间的相互依赖关系. 近年来,图神经网络应用于各个领域,包括小样本[21]、半监督[22]、强化学习[19]等.

运动信息是MOT中常用的信息,大多数方法认为目标在图片空间内是平滑的,并且设计不同的运动模型捕捉单个目标的动态行为. 但是,物体的运动并不总是平滑的,当相机处于运动状态时物体的运动是难以预测的. 本文不仅对帧间目标使用图模型建模,在帧内也根据目标间的相对位置建立图模型,使得整个模型对摄像机运动更为鲁棒.

2　方法介绍

假设在$t-1$帧中有M条轨迹目标$o_i \in O, i \in \{1,2,\ldots,M\}$,在$t$帧中有$N$个检测目标$d_j \in D, j \in \{1,2,\ldots,N\}$. MOT的任务是利用轨迹目标和检测目标的特征得到两者的相似度值,根据该值得到轨迹和检测的匹配,最终得到每个目标的轨迹. 模型整体框架如图2所示,分为4个部分:

1) 外观和运动特征提取模块. 使用一个孪生网络(siamese CNN)提取轨迹和检测的外观特征,将前L帧轨迹位置信息输入长短期记忆网络(LSTM),输出轨迹运动特征;将t帧检测的位置信息输入多层感知机(MLP),输出检测的运动信息. 将轨迹(或检测)的外观特征与运动特征拼接后输出.

2) 帧内关系建模模块. 对于每个t帧中检测的d_j和$t-1$帧中的轨迹o_i,选取距离它们最近的k个邻居,为每个d_j和o_i构建一个有向图,图中节点代表轨迹和检测. 每个节点的初始特征为图2(a)输出的拼接后的外观和运动特征,建模后状态如图2(b)所示.

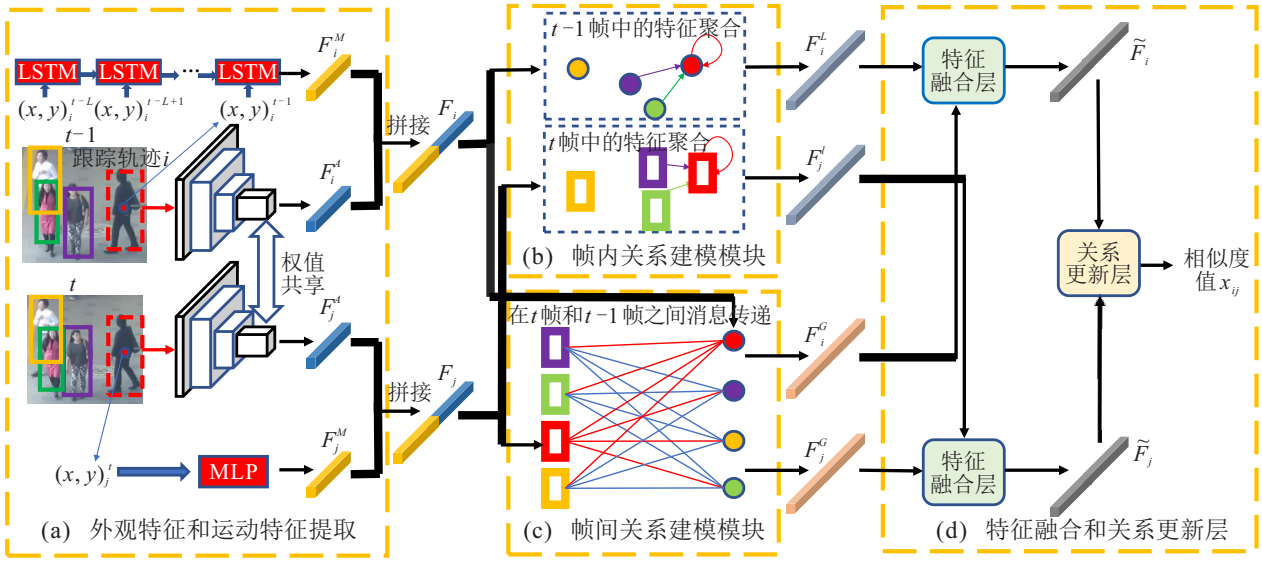


图2　多目标跟踪方法的整体框架

3) 帧间关系建模模块. 在t帧与$t-1$帧之间构造二分图, 即边的连接只存在于轨迹节点o_i与检测节点d_j之间, 检测节点之间与轨迹节点之间不建立连接. 节点的初始特征同帧内关系模块一样, 均是由图2(a)输出的拼接特征, 建模后状态如图2(c)所示.

4) 特征融合和关系更新模块. 利用图注意力机制[23], 将帧内交互局部特征与帧间全局特征融合, 得到最终的轨迹和检测特征, 根据检测后的特征更新轨迹和检测关系, 去掉多余连接.

2.1 外观特征和运动特征提取模块

如图2(a)模块, 对于某一轨迹, 前L帧中轨迹的位置信息表示为$T = \{(x,y)_i^{t-L}, (x,y)_i^{t-L+1}, \ldots, (x,y)_i^{t-1}\}$, 其中$(x,y)_i^{t-L}$为$t-L$帧中轨迹$i$的边界框中心坐标信息. 将集合$T$输入LSTM中, 输出轨迹$o_i$的运动特征$F_i^M \in R^{D_M}$. 当前$t$帧中检测$j$的边界框信息$D = (x,y)_j^t$, 将$D$输入MLP, 输出检测$d_j$的运动特征$F_j^M \in R^{D_M}$. 对于轨迹和检测的表观特征提取, 将裁剪的图片块输入孪生网络(SiameseCNN), 编码轨迹和检测的外观特征为$F_i^A \in R^{D_A}$、$F_j^A \in R^{D_A}$. 将外观特征和运动特征拼接后, 输出轨迹特征$F_i \in R^{D_A+D_M}$和检测特征$F_j \in R^{D_A+D_M}$.

2.2 帧内关系建模模块

已有的基于图的数据关联模块模型[4,6]结构如图2(b)所示, 都是在帧间建立图模型, 只考虑了全局特征, 忽视了帧内行人的局部特征, 用单个的特征表示计算轨迹与检测之间的相似度值. 本文模型对帧内关系建模, 考虑行人与邻居的交互信息, 提取了行人的局部特征. 在帧内关系建模模块, 选取距离每个轨迹(检测)最近的两个邻居轨迹(检测), 如图2(b)所示, 圆圈代表$t-1$帧中轨迹, 方块代表t帧中检测, 不同颜色对应输入图片中不同颜色的边界框, 有向边表示聚合邻居特征信息. 边的权值计算公式为

$$e_{i,s} = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2}, \tag{1}$$

$$e_{j,t} = \sqrt{(x_j - x_t)^2 + (y_j - y_t)^2}. \tag{2}$$

其中: (x_i, y_i)为目标i边界框中心点的坐标位置信息, 目标s为目标i的第s个邻居, 检测t为检测j的第t个邻居, $e_{i,s}$和$e_{j,t}$为边的权值. 根据边权值聚合邻居特征信息, 有

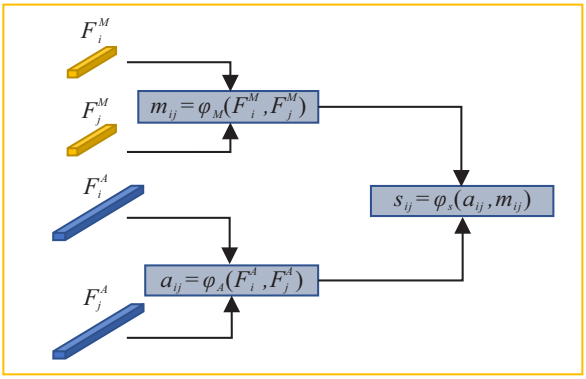
$$F_i^L = F_i + \sum_s F_s e_{i,s}, \tag{3}$$

$$F_j^L = F_j + \sum_t F_t e_{j,t}, \tag{4}$$

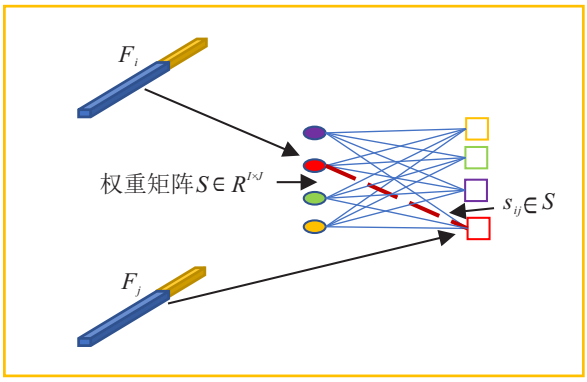
其中轨迹和检测节点赋予自身权重值为1, 得到轨迹和检测在帧内的局部特征$F_i^L \in R^D$和$F_j^L \in R^D$, $D = D_A + D_M$.

2.3 帧间关系建模模块

采用文献[6]方法对帧间关系建模, 如图3所示, 分为两个部分: 1) 特征相似度计算模块, 该模块又分为外观特征相似度矩阵计算、运动特征相似度矩阵计算和两者相似度融合3个部分; 2) 消息传递模块, 具体细节见文献[6].



(a) 特征相似度模块



(b) 消息传递模块

图 3 帧间关系建模

由图3可见, φ_M、φ_A和φ_S是3个全连接神经网络. 将轨迹和检测的外观特征拼接后输入φ_A, 输出轨迹和检测的外观相似值为a_{ij}, 运动特征拼接后输入φ_M, 输出轨迹和检测的运动相似值为m_{ij}, 将外观信息相似值和运动信息相似值输入φ_S, 输出轨迹和检测的亲和力值为$s_{ij}, s_{ij} \in S, S \in R^{I\times J}$为帧间二分图边的权重矩阵, I为轨迹个数, J为检测个数. 在消息传递模块, 将轨迹和检测的特征$F_i \in F_M$、$F_j \in F_N$以及权重矩阵S输入二分图进行消息传递, 公式如下:

$$F_M^G = \mathrm{relu}(\mathrm{softmax}(S)F_N W_\theta), \tag{5}$$

$$F_N^G = \mathrm{relu}(\mathrm{softmax}(S^{\mathrm{T}})F_M W_\theta). \tag{6}$$

其中: $F_M \in R^{I\times D}$, $F_N \in R^{J\times D}$, F_M为$t-1$帧中所有轨迹的全局特征, F_N为t帧中所有检测的全局特征, 参数$W \in R^{D\times D}$, $D = D_A + D_M$.

2.4 特征融合和关系更新模块

2.4.1 特征融合层

特征融合模块将帧内建模模块(2.2节)得到的轨迹(或检测)局部特征与帧间建模模块(2.3节)得到的轨迹(或检测)全局特征进行融合,融合方法使用图注意力机制[23],学习局部特征和全局特征的权重信息,对其进行加权求和,有

$$U_i = (F_i^L, F_i^G),\ U_j = (F_j^L, F_j^G). \tag{7}$$

其中: $U_i, U_j \in R^{2\times C}$, C为局部特征和全局特征的长度,其大小等于$(D_A + D_M)$.

$$a_{i,r} = \mathrm{softmax}(w_r^{\mathrm{T}} \tanh(W_r U_i^{\mathrm{T}})), \tag{8}$$

$$a_{j,r} = \mathrm{softmax}(w_r^{\mathrm{T}} \tanh(W_r U_j^{\mathrm{T}})). \tag{9}$$

其中: $r = 1, 2$; w_r和W_r为可学习的参数,其大小分别为d_a和$d_a \times C$. 最终融合后的轨迹节点i和检测节点j的特征分别为

$$\widetilde{F}_i = a_{i,1} \times F_i^L + a_{i,2} \times F_i^G, \tag{10}$$

$$\widetilde{F}_j = a_{j,1} \times F_j^L + a_{j,2} \times F_j^G. \tag{11}$$

2.4.2 关系更新层

将轨迹节点i和检测节点j的最终特征表示$\widetilde{F}_i$、$\widetilde{F}_j$输入关系更新层,该层中关联矩阵X的元素$x_{ij} \in R$. 利用一个无参数的元素减法将一对节点中的特征聚合到连接两者的边上,作为边的特征进行迭代估算,利用多层感知机MLP将聚合后的特征转换成一个标量值x_{ij}. 该层可形式化为

$$x_{ij} = \mathrm{MLP}_\theta(\sigma(\widetilde{F}_l, \widetilde{F}_J)), \tag{12}$$

其中$\sigma(\cdot)$为一个特征聚合函数.

2.5 损失函数

端到端的训练其损失函数的设计是复杂的,因为帧与帧之间真值(groundtruth)矩阵维度大小是不一样的. 模型生成的关联矩阵由3部分组成: 一对一的关联、新生的轨迹和消失的轨迹,所以参考文献[6]的方法分别计算矩阵损失和向量损失两部分.

首先生成真值(groundtruth)关联矩阵$\hat{Y} \in R^{m\times n}$, $\hat{Y}$中每个元素$\widehat{y_{ij}} \in 0, 1$,矩阵$Y_{O2O} \in R^{k\times k}$为关联矩阵$\hat{Y}$的子矩阵, k表示有k个轨迹与检测成功匹配,其在真值矩阵中对应的行和列是一个one-hot向量. 矩阵$Y_{B\&D}$也是关联矩阵$\hat{Y}$的子矩阵($Y_{B\&D} \bigcup Y_{O2O} = \hat{Y}$),里面存放的是新出生或者消失的轨迹,其在真值矩阵中对应的行和列向量值全部为0.

对于模型最终生成的关联矩阵$Y \in R^{m\times n}$,其中的每一个元素或者为1,或者为0,代表该轨迹与检测是否匹配. 使用如下二元交叉熵损失计算矩阵级别的损失:

$$L_e = \sum_i^I \sum_j^J (-p\widehat{y_{ij}} \log \sigma(y_{ij}) - (1 - y_{ij}) \log(1 - \sigma(y_{ij}))). \tag{13}$$

其中: $y_{ij} \in Y, \widehat{y_{ij}} \in \hat{Y}$; p为权重参数,用来平衡样本. 实验中设置经验值为25,计算向量损失时分别计算一对一关联向量损失和轨迹新生&消失向量损失,对应矩阵Y_{O2O}和$Y_{B\&D}$. 向量$v_{O2O} \in Y_{O2O}$是一个one-hot向量,损失函数计算为

$$L_{O2O} = -\sum_{O2O}^{k} \hat{v}_{O2O} \log(\mathrm{softmax}(v_{O2O})). \tag{14}$$

其中: $\hat{v}_{O2O}$为对应groundtruth中的one-hot向量, v_{O2O}为模型生成矩阵对应的一对一关联向量, k为关联数量. 采用均方误差(MSE)损失计算轨迹新生&消失向量损失,有

$$L_{B\&D} = \sum_{B\&D}^{v} \|\mathrm{sigmoid}(v_{B\&D})\|^2, \tag{15}$$

其中$v = m + n - 2 \times k$. 最终,多级矩阵损失计算为

$$L_{\mathrm{Matrix}} = L_e + L_{O2O} + L_{B\&D}. \tag{16}$$

3 实验分析

3.1 实验环境和实验细节

本文方法在16.04 Linux系统上使用Python3.8和PyTorch1.7实现,同时利用1块NVIDIA GTX 3090 GPU卡.

所有裁剪图像块调整为84×32,以保持目标的纵横比. 用于提取统一的外观特征的Siamese CNNs,有4个卷积层,每层使用非线性激活函数Relu. LSTM网络模块建模轨迹在时域的非线性运动特征,轨迹的长度设置为5,学习率lr = 0.001,共进行40 000次迭代.

3.2 数 据 集

实验使用多目标跟踪基准数据集MotChallenge. 该数据集的跟踪目标是行人,因为行人是目前多目标跟踪领域最主要的测试和验证对象. 由于MOT17子集提供了更为多样的行人检测结果,更能够评测一个多目标跟踪方法在不同目标检测精度上的综合鲁棒性,本文实验结果主要针对MOT17数据集给出. MOT17数据集有14个视频序列,一般划分方法是7个用于训练,7个用作测试,每个视频序列由3个检测器提供检测结果,分别为DPM、FRCNN和SDP,其中SDP检测器效果最好,DPM检测器效果最差.

3.3 评估指标

为了能够公平准确地评价算法性能,采用文献[24]的评价指标评估所提出算法,包括:

1) MOTA (multiple object tracking accuracy): 多目标跟踪的准确度,确定目标个数,用于统计跟踪中的误差积累情况,与FP(没有匹配上的算法结果)、FN(没有匹配上的真实标注)和IDSW(算法匹配的真实标识信息发生变化的次数)有关.

2) MOTP (multiple object tracking precision): 多目标跟踪的精确度,确定目标位置上的精确度,计算预测框与真实标注框之间的重合率,用于衡量目标位置定位的精确程度.

3) MT (mostly tracked targets): 目标大部分被跟踪到的轨迹,若被跟踪到轨迹占比该目标真实轨迹长度大于80 %,则属于目标大部分被跟踪到.

4) ML (mostly lost targets): 目标大部分未被跟踪到的轨迹,若被跟踪到轨迹占比该目标真实轨迹长度小于20 %,则属于目标大部分轨迹缺失.

5) FP (false positives): 错检数,指当前帧中没有得到匹配的轨迹数,即算法结果没有得到匹配的数量.

6) FN (false negatives): 漏检数,指当前帧中未匹配上的真实标注(groundtruth).

7) IDS (identity switches): 跟踪轨迹改变目标身份号的次数.

3.4 实验结果

在MOT17数据集上评估所提出方法,与5种方法进行比较,结果如表1所示. 可见,所提出算法在MOTA、MOTP、FN、ML和MT上取得最好结果,在IDF1和IDSW上取得第2好结果. MASS[25]在MOTA指标上仅次于所提出方法,其考虑了外观、运动信息、结构和边界框重叠率4个方面信息,比较单个轨迹和检测在4个方面的相似度,虽然该方法也考虑了单个检测和轨迹的信息,但忽视了检测和轨迹与周围环境交互的局部特征,同时对于轨迹和检测节点的全局位置关系也无法捕捉. 而EDA_GNN[6]方法虽然在连续帧之间建立二分图,设计了更新机制,使轨迹节点和检测节点特征能够自适应更新,且更新后的节点特征能够捕获全局的位置关系,但忽视了帧内轨迹和节点的局部特征.

本文方法不仅考虑了轨迹和检测在帧内的局部特征,而且考虑了文献[6]提出的更新后的全局特征,与文献[25]不同,本文使用每个检测和轨迹与周围环境交互的局部特征,同时对于两种特征利用一个注意力机制进行最终的融合.

此外,为了验证所提出帧内关系建模模块的有效性,在方法GNMOT[4]中添加帧内关系建模模块,将MOT17-02作为训练集,MOT17-09作为验证集,使用SDP检测器进行检测,结果如表2所示.

因为GNMOT[4]使用单个跟踪器,其外观模型和运动模型作为两个独立模块进行训练,且运动特征为目标的坐标和速度信息,所以只在外观模型上添加帧内消息传递模块. 对于目标局部外观特征与全局外观特征的融合,为了减少参数量以进行更好的训练,只采用简单的融合,结果表明在MOTA、MOTP和FP指标上均有提升,验证了所提出帧内模型的有效性.

3.5 可视化结果

为了更直观地表明算法性能,图4～图6给出了所提出算法在MOT17数据集01、03、06三个视频序列上的部分可视化跟踪结果. 其中MOT17-01的视频拍摄于光线较暗的地方,背景复杂,目标5在第6、121、164帧中始终被正确跟踪.

表1 在MOT17测试集上的跟踪结果

trackers	mode	MOTA(↑)	MOTP(↑)	IDF1(↑)	IDSW(↓)	MT(↑)	ML(↓)	FP(↓)	FN(↓)	Frag(↓)
ours	Online	**47.4**	76.71	44.1	4 276	**19.5**	**33.1**	42 469	**250 314**	6 124
MASS[25]	Online	46.9	76.4	**46**	4 478	16.9	36.3	25 733	269 116	11 994
PHD_LMP[26]	Online	45.9	76.6	42.5	4 977	16.9	37.2	27 964	272 196	6 985
EDA_GNN[6]	Online	45.5	76.3	40.5	**4 091**	15.6	40.6	25 685	277 663	**5 579**
GMPHD_N1Tr[27]	Online	42.1	**77.7**	33.9	10 698	11.9	42.7	**18 214**	297 646	10 864
GMPHD_KCF[28]	Online	39.6	74.5	36.6	5 811	8.8	43.3	50 903	284 228	7 414
GNMOT[4]	Near-Online	50.2	–	47	5 273	19.3	32.7	29 316	246 200	–
jCC[29]	Offline	51.2	75.9	54.5	1 802	20.9	37	25 937	247 822	2 984
NOTA[30]	Offline	51.3	76.7	54.5	2 285	17.1	35.4	20 148	252 531	5 798
CRF_TRA[31]	Offline	53.1	76.1	53.7	2 518	24.2	30.7	27 194	234 991	4 918

表 2　在GNMOT[4]模型中加入帧内关系建模后的跟踪结果

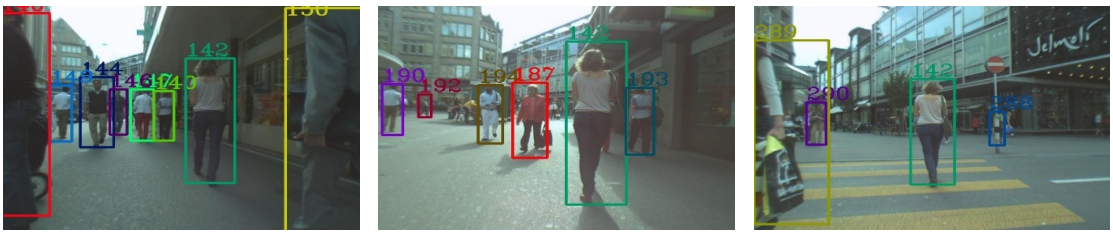
trackers	MOTA(↑)	MOTP(↑)	IDF1(↑)	IDSW(↓)	MT(↑)	ML(↓)	FP(↓)	FN(↓)
GNMOT[4]	62.7	84.9	**52.7**	**54**	9	1	191	**1 740**
GNMOT[4]＋帧内图模型	**63.3**	**85.2**	51.8	63	9	1	**116**	1 777

第 6 帧　　第 121 帧　　第 164 帧

图 4　MOT17-01 跟踪结果

第 155 帧　　第 543 帧　　第 735 帧

图 5　MOT17-03 跟踪结果



第 537 帧　　第 806 帧　　第 1 124 帧

图 6　MOT17-06 跟踪结果

MOT17-03 视频序列行人较多, 行人间易发生频繁遮挡, 且不同目标的衣着很相似; MOT17-06 是在光照条件下不断变化的视频, 且目标尺寸也在发生变化, 目标跟踪难度增加. 即使这样, 目标 142 在第 537、806、1 124 帧仍被正确跟踪.

3.6　消融实验

对所提出的模型进行消融实验, 以验证: 1) 帧内关系建模模块得到的局部特征的有效性; 2) 所提出算法融合帧内和帧间特征的有效性; 3) 邻居数量对于实验结果的影响. 前 2 项以 MOT17-09 作为训练集、MOT17-02 作为验证集设计实验. 第 3 项考虑 MOT17-09 数据集中某些帧中行人数量较少, 以 MOT17-02 作为训练集、MOT17-10 作为验证集设计实验. 如表 3 所示, 其中帧间特征指仅在帧间建立图模型, 帧内不建立模型, 不使用帧内目标的局部特征. 当仅使用帧间特征时, 其 MOTA 为 23.2, 加入帧内特征且将帧内特征与帧间特征进行简单融合后, MOTA 为 25.4. 上述情况表明了加入帧内特征的有效性. 当帧间特征信息传递时, 检测 (轨迹) 的特征在二分图上进行了一次传播, 每个检测 (轨迹) 节点在信息传递后, 特征因卷积了相邻节点的特征变得平滑. 虽然对于计算轨迹和节点的相似度分数有利, 但有时会使得相似度分数值过高, 模糊了轨迹和检测的差异性. 这时加入每个检测 (轨迹) 的局部特征, 可以解决特征平滑问题, 提高跟踪的鲁棒性.

为了验证上述分析, 将帧内特征也进行一次信息传递, 帧内模型与帧间模型从并行模式改为串行模式, 即将帧内建模模块得到轨迹 (检测) 的帧内特征输入帧间建模模块的消息传递部分, 由结果可知其效果是下降的. 与前文分析一致, 帧内特征是检测 (轨迹) 的局部特征, 如果在帧间图模型上进行消息传递, 则会使轨迹 (检测) 特征变得平滑, 特征之间的差异性变小, 那么对于后续跟踪效果下降是显然的. 本文模型对帧内局部特征不作信息传递, 而是在融合模块 (2.4.1 节) 将其融合, 最终特征既有差异性也有相似性.

表 3 消融实验结果

trackers	MOTA(↑)	MOTP(↑)	IDF1(↑)	IDSW(↓)	MT(↑)	ML(↓)	FP(↓)	FN(↓)
用帧内特征做帧间消息传递	21.6	78.7	24.8	937	15	13	5 140	**8 489**
帧间特征	23.2	78.7	26.2	974	**17**	**12**	4 767	8 530
简单融合	25.4	**78.8**	26	812	15	13	4 512	8 533
注意力融合	**26.4**	78.6	**28.7**	**740**	15	13	**4 422**	8 512

为了验证利用注意力机制融合特征的有效性,使用简单融合方法整合帧内和帧间特征. 简单融合是指,对于得到的帧内局部特征和帧间全局特征,只是简单地相加进行融合,与本文所提出的注意力融合相比,显然使用注意力融合效果更好,前者MOTA值为25.4,后者MOTA值为26.4,表明帧内特征与帧间特征的权重值不一样,不能简单融合,应按照其各自权重进行加权融合. 消融实验第3项根据选取的邻居节点数不同,最终的跟踪效果也是不同的,由图7可见,当邻居数为2和3时,其MOTA值最高,为44.6,但邻居数为2的IDF1值比邻居数为3的IDF1值高1.7. 实验表明,邻居数量为2是帧内建模较好的选择,邻居数量过多反而会因为相邻两帧目标邻居交集过大导致特征差异性减小.

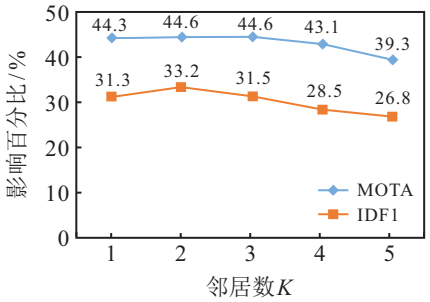


图 7 邻居数量对于跟踪效果的影响

同时考虑损失函数(式(13))中超参数p的最优设置. 在该损失函数中,p作为正样本的权重因子,对模型也起到一定影响. 表4给出了超参数p的消融实验,可以看出,p值设定为25时效果最好.

表 4 参数p优化对比实验结果

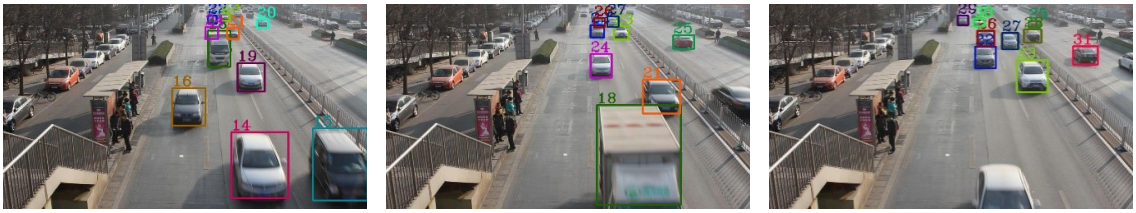
p	MOTA(↑)	MOTP(↑)	IDF1(↑)	IDSW(↓)	MT(↑)	ML(↓)	FP(↓)	FN(↓)
15	51.6	79	43.5	607	**36**	3	2 665	2 945
20	53.8	78.9	45.1	596	34	3	2 424	**2 910**
25	**55.9**	**79.1**	**50.3**	**477**	33	3	**2 265**	2926
30	54.2	79.1	44.1	537	32	3	2 399	2 948
35	53.9	79.1	46.3	530	35	3	2 443	2 942

3.7 车辆跟踪实验

为验证所提出算法的泛化能力,在UA-DETRAC数据集上进行车辆跟踪实验,实验使用UA-DETRAC数据集中的MVI_20011和MVI_39971作为验证集. 跟踪时直接使用在MotChallenge行人标准数据集上训练好的模型,未进行任何参数的调整,运用文献[24]的评估指标评估跟踪算法的准确性. 实验结果如表5所示,可视化结果如图8和图9所示.

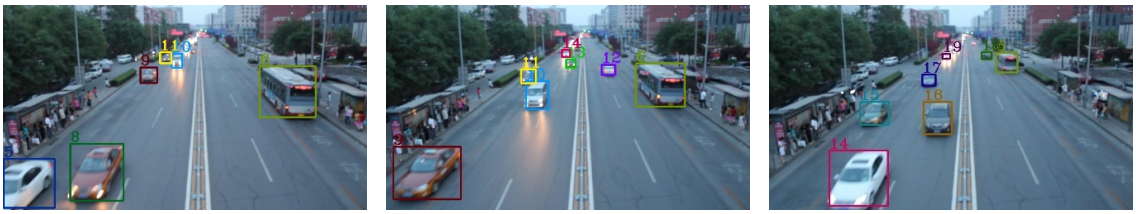
表 5 UA-DETRAC数据集跟踪结果

trackers	MOTA(↑)	MOTP(↑)	IDSW(↓)	MT(↑)	ML(↓)	FP(↓)	FN(↓)
车辆跟踪	55.9	79.1	35	33	16	1165	589



第234帧 第299帧 第350帧

图 8 MVI_20011序列上的跟踪结果



第130帧　　第200帧　　第360帧

图9　MVI_39771序列上的跟踪结果

4　结　论

本文提出了基于图神经网络的端到端多目标跟踪模型. 在EDA_GNN[6]的基础上,引入帧内关系建模模块,使用注意力机制[23]将帧内目标的局部跟踪特征与全局跟踪特征进行融合,得到一个能力更强的特征表示. 利用该特征计算的相似度能更好地区分目标,提高在线多目标跟踪的性能. 在MOT17公开数据集上与当前基于图的多目标跟踪方法和一些主流的非图网络目标跟踪相比,所提出算法取得了综合指标最优的跟踪效果. 在车辆数据集上的测试实验验证了所提出方法的泛化能力.

参考文献(References)

[1] Ciaparrone G, Luque Sánchez F, Tabik S, et al. Deep learning in video multi-object tracking: A survey[J]. Neurocomputing, 2020, 381: 61-88.

[2] Wu Z H, Pan S R, Chen F W, et al. A comprehensive survey on graph neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(1): 4-24.

[3] Ma C, Li Y, Yang F, et al. Deep association: end-to-end graph-based learning for multiple object tracking with conv-graph neural network[C]. Proceedings of the 2019 on International Conference on Multimedia Retrieval. New York: ACM, 2019: 253-261.

[4] Li J H, Gao X, Jiang T T. Graph networks for multiple object tracking[C]. IEEE Winter Conference on Applications of Computer Vision. Snowmass, 2020: 708-717.

[5] Brasó G, Leal-Taixé L. Learning a neural solver for multiple object tracking[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6246-6256.

[6] Jiang X L, Li P Z, Li Y J, et al. Graph neural based end-to-end data association framework for online multiple-object tracking[J/OL]. 2019, arXiv: 1907.05315.

[7] Liu Q, Chu Q, Liu B, et al. GSM: Graph similarity model for multi-object tracking[C]. International Joint Conference on Artificial Intelligence. Piscataway: IEEE, 2020: 530-536.

[8] Moskalev A, Sosnovik I, Smeulders A. Two is a crowd: Tracking relations in videos[J/OL]. 2021, arXiv: 2108.05331.

[9] Yang B, Nevatia R. An online learned CRF model for multi-target tracking[C]. IEEE Conference on Computer Vision and Pattern Recognition. Providence, 2012: 2034-2041.

[10] Weng X S, Wang Y X, Man Y Z, et al. GNN3DMOT: Graph neural network for 3D multi-object tracking with 2D-3D multi-feature learning[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6498-6507.

[11] Shuai B, Berneshawi A, Li X Y, et al. SiamMOT: Siamese multi-object tracking[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville, 2021: 12367-12377.

[12] 宋建辉, 张甲, 刘砚菊, 等. 基于条件对抗生成孪生网络的目标跟踪[J]. 控制与决策, 2021, 36(5): 1110-1118.
(Song J H, Zhang J, Liu Y J, et al. Conditional generative adversarial siamese networks for object tracking[J]. Control and Decision, 2021, 36(5): 1110-1118.)

[13] Ho K, Keuper J, Keuper M. Unsupervised multiple person tracking using autoencoder-based lifted multicuts[J/OL]. 2020, arXiv: 2002.01192.

[14] 赵浩光, 孟琭, 耿欢, 等. 尺度自适应的多特征融合相关滤波目标跟踪算法[J]. 控制与决策, 2021, 36(2): 429-435.
(Zhao H G, Meng L, Geng H, et al. Scale adaptation and multi-feature fusion correlation filtering object tracking algorithm[J]. Control and Decision, 2021, 36(2): 429-435.)

[15] Chu P, Wang J, You Q Z, et al. Spatial-temporal graph transformer for multiple object tracking[J/OL]. 2021, arXiv: 2104.00194.

[16] 李可非, 马晓川, 刘宇, 等. 基于转换量测容积卡尔曼滤波器带多普勒量测的目标跟踪算法[J]. 控制与决策, 2021, 36(6): 1425-1434.
(Li K F, Ma X C, Liu Y, et al. Converted measurement cubature Kalman filter for Doppler-assisted

target tracking[J]. Control and Decision, 2021, 36(6): 1425-1434.)

[17] Saleh F, Aliakbarian S, Salzmann M, et al. ArTIST: Autoregressive trajectory inpainting and scoring for tracking[J/OL]. 2020, arXiv: 2004.07482.

[18] Ullah M, Cheikh F A. A directed sparse graphical model for multi-target tracking[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Salt Lake City, 2018: 1897-18977.

[19] Zintgraf L, Feng L, Igl M, et al. Exploration in approximate hyper-state space for meta reinforcement learning[J/OL]. 2020, arXiv: 2010.01062.

[20] Liu Y X, Pan S R, Jin M, et al. Graph self-supervised learning: A survey[J/OL]. 2021, arXiv: 2103.00111.

[21] Zhang X, Zhang Y J, Zhang Z Y. Multi-granularity recurrent attention graph neural network for few-shot learning[C]. International Conference on Multimedia Modeling. Springer, 2021: 147-158.

[22] Wang X, Liu N, Han H, et al. Self-supervised heterogeneous graph neural network with co-contrastive learning[J/OL]. 2021, arXiv: 2105.09111.

[23] Wang X, Ji H Y, Shi C, et al. Heterogeneous graph attention network[C]. WWW '19: The World Wide Web Conference. Piscataway: IEEE, 2019: 2022-2032.

[24] Milan A, Leal-Taixe L, Reid I, et al. MOT16: A benchmark for multi-object tracking[J/OL]. 2016, arXiv: 1603.00831.

[25] Karunasekera H, Wang H, Zhang H D. Multiple object tracking with attention to appearance, structure, motion and size[J]. IEEE Access, 2019, 7: 104423-104434.

[26] Sanchez-Matilla R, Cavallaro A. Motion prediction for first-person vision multi-object tracking[C]. European Conference on Computer Vision. Berlin: Springer, 2020: 485-499.

[27] Baisa N L, Wallace A. Development of a N-type GM-PHD filter for multiple target, multiple type visual tracking[J]. Journal of Visual Communication and Image Representation, 2019, 59: 257-271.

[28] Kutschbach T, Bochinski E, Eiselein V, et al. Sequential sensor fusion combining probability hypothesis density and kernelized correlation filters for multi-object tracking in video data[C]. The 14th IEEE International Conference on Advanced Video and Signal Based Surveillance. Lecce, 2017: 1-5.

[29] Keuper M, Tang S, Andres B, et al. Motion segmentation & multiple object tracking by correlation co-clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 42(1): 140-153.

[30] Chen L, Ai H Z, Chen R, et al. Aggregate tracklet appearance features for multi-object tracking[J]. IEEE Signal Processing Letters, 2019, 26(11): 1613-1617.

[31] Xiang J, Xu G H, Ma C, et al. End-to-end learning deep CRF models for multi-object tracking deep CRF models[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 31(1): 275-288.

作者简介

朱姝姝 (1994−), 女, 硕士生, 从事计算机视觉、多目标跟踪的研究, E-mail: shushuzhu@njust.edu.cn;

王欢 (1982−), 男, 副教授, 博士, 从事模式识别、图像处理、红外目标检测等研究, E-mail: wanghuanphd@njust.edu.cn;

严慧 (1983−), 女, 副教授, 博士生导师, 从事模式识别、计算机视觉、机器学习等研究, E-mail: yanhui@njust.edu.cn.

(责任编辑：郑晓蕾)