基于轨迹预测增强的复杂场景多目标跟踪方法

刘培刚[1)], 王奔[1)], 李亚传[1)], 崔振东[1)], 王君伍[1)], 杨少波[1)], 李宗民[1,2)]

[1)] (中国石油大学(华东)计算机科学与技术学院 青岛 266580)
[2)] (中国石油大学胜利学院 东营 257061)
(dongfangwy@upc.edu.cn)

摘 要: 以冬奥会的短道速滑比赛场景为例, 针对短道速滑中运动员的目标外观差异性小、运动变化快、目标间遮挡频繁等运动特点, 设计一个应用于短道速滑场景的多目标跟踪数据集, 并提出一种基于轨迹预测增强的多目标跟踪方法. 首先计算包围框交并比距离与外观特征余弦距离, 联合判断检测响应与跟踪轨迹的相似性解决目标外观相似问题; 然后通过跟踪轨迹的全局特征和运动线索恢复被遮挡目标丢失的信息, 提高中断轨迹的重关联能力; 最后根据检测先验控制新轨迹的初始化, 减少噪声检测对轨迹跟踪中身份交换的影响. 实验结果表明, 与 DeepSORT 方法相比, 所提方法在短道速滑场景中能够稳定地跟踪轨迹, 有效地减少了轨迹中断, 其中, IDF_1 提升 21 个百分点, MOT 准确度提高 14.3 个百分点; 该方法在目标差异性小、运动变化快的短道速滑场景中保证长期稳定跟踪, 对多目标跟踪在复杂场景中的应用具有启发意义.

关键词: 深度学习; 多目标跟踪; 短道速滑; 卡尔曼滤波; 轨迹预测
中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2024.19851

Trajectory Prediction Enhancement Method for Multiple Object Tracking in Complex Scenes

Liu Peigang[1)], Wang Ben[1)], Li Yachuan[1)], Cui Zhendong[1)], Wang Junwu[1)], Yang Shaobo[1)], and Li Zongmin[1,2)]

[1)] (*School of Computer Science and Technology, China University of Petroleum(East China), Qingdao* 266580)
[2)] (*Shengli College, China University of Petroleum, Dongying* 257061)

Abstract: Taking the short-track speed skating match of the Winter Olympics as an example, a multiple object tracking dataset applicable to the short-track speed skating scene is designed based on the athletes' motion characteristics, such as small distinctions in target appearance, fast motion changes, and frequent mutual occlusion between targets. A multiple object tracking method based on trajectory prediction enhancement was proposed. First, the intersection-over-union of bounding boxes and the cosine distance of appearance features are calculated to jointly judge the similarity between detection and trajectories, solving the problem of similar target appearances. Secondly, the tracking trajectory's global features and motion clues are used to recover the lost information of occluded targets, improving the re-association ability of interrupted trajectories. Finally, the detection prior is used to control the initialization of new trajectories, reducing the impact of noisy detection on identity exchange during trajectory tracking. Experimental results show that compared with the DeepSORT method, this method can guarantee stable tracking in the short-track speed skating scene and effectively reduce trajectory in-

收稿日期: 2022-07-25; 修回日期: 2023-05-05. **基金项目**: 国家重点研发计划(2019YFF0301800); 国家自然科学基金(61379106); 山东省自然科学基金(ZR2013FM036, ZR2015FM011). **刘培刚**(1979—), 男, 博士, 讲师, 主要研究方向为图像处理、数据科学及应用; **王奔**(1997—), 男, 硕士研究生, 主要研究方向为计算机视觉; **李亚传**(1995—), 男, 博士, 主要研究方向为计算机视觉、边缘计算; **崔振东**(1997—), 男, 硕士研究生, 主要研究方向为计算机视觉; **王君伍**(1998—), 男, 硕士研究生, 主要研究方向为计算机视觉; **杨少波**(1997—), 男, 硕士研究生, 主要研究方向为计算机视觉; **李宗民**(1965—), 男, 博士, 教授, 博士生导师, 主要研究方向为计算机图形学、图像处理.

terruptions, with an increase of 21 percentage points in IDF_1 and 14.3 percentage points in multiple object tracking accuracy (MOTA). This method ensures long-term stable tracking in short-track speed skating scenes with small differences in targets and fast motion changes. It has inspirational significance for the application of multiple object tracking in complex scenes.

Key words: deep learning; multiple object tracking; short-track speed skating; Kalman filter; track predict

短道速滑是我国冬奥会的传统强项，传统的训练方法是由观察员拍摄训练视频，并在训练后由教练员观察分析运动员的运动状态，制定针对性的训练计划. 如何在现有的训练基础上进一步提高训练效率，从而提高比赛成绩是目前仍需解决的问题. 一个有效的方法是将人工智能技术与短道速滑运动结合，借助人工智能辅助运动员训练，通过比赛视频获取运动员的运动数据，针对关键问题设计训练方案，切实地解决该问题.

基于视频的多目标跟踪(multiple object tracking，MOT)是计算机视觉领域的一个重要研究分支，在智能安防、自动驾驶等任务中应用广泛. MOT 方法通常在视频帧中定位目标，并在后续图像帧中维持目标正确的身份信息. 与教练员肉眼观察分析对比，MOT 方法能够更准确地获取运动员的运动数据，充分挖掘多源因素对竞技成绩的影响，并将这些信息进行量化，为运动员制定训练计划提供数据支撑. 然而，MOT 方法在短道速滑场景中的应用面临诸多挑战，如运动员的外观差异小、运动状态相似，且变化快、频繁地相互遮挡等，容易导致跟踪模型出现身份交换以及轨迹中断等问题，影响跟踪效果.

近年来，随着深度学习在目标检测中的广泛应用，基于检测的跟踪(track-by-detection，TBD)范式成为当前主流的 MOT 方法[1-3]，TBD 范式通常包括目标检测、数据关联和轨迹预测 3 部分. 首先利用检测器定位每帧中出现的运动员，然后通过运动模型预测轨迹在当前帧的位置，再将检测定位的结果与轨迹预测值进行数据关联，通过关联结果更新运动模型，最终得到目标的完整轨迹. 得益于目标检测算法在深度学习领域取得的突破，TBD 范式的算法在 MOT 方法中取得了不错的跟踪精度. 但在复杂场景中，目标检测算法受制于目标尺度较小、运动变化快等问题，整体跟踪精度无法得到有效的提升. 因此，本文关注数据关联和轨迹预测部分，提高 MOT 方法在复杂场景中的跟踪精度和跟踪鲁棒性.

数据关联采用身份分配的方式解决检测结果与跟踪轨迹之间的匹配问题. 传统的 MOT 方法常采用联合概率数据关联(joint probabilistic data association，JDPA)[4]或多重假设跟踪(multiple hypothesis tracking，MHT)[5]分配轨迹身份. 其中，JDPA 通过观测结果与轨迹估计值的关联概率进行匹配；MHT 通过跟踪轨迹与检测结果的所有假设进行匹配，根据后续的目标观测值来减少误差. 当检测结果与跟踪估计值之间的不确定性增加时，MHT 和 JDPA 匹配的不确定性也随之增加. 匈牙利匹配算法[6]根据检测结果与轨迹估计值的权值二分图(也称代价矩阵)，计算完备匹配下最优权匹配的结果，能够得到数据关联的全局最优解. 代价矩阵的设计是匈牙利匹配的关键. Bewley 等[1]使用检测与轨迹的交并比(intersection-over-union，IoU)距离构建代价矩阵；Wojke 等[2]和 Yu 等[3]均是在 IoU 距离的基础上引入目标外观的余弦距离，通过特征提取网络提取检测响应的嵌入特征向量，构建检测与轨迹外观的代价矩阵；Cao 等[7]和 Lu 等[8]使用轨迹的方向一致性距离构建代价矩阵，通过轨迹方向估计值与检测的偏差降低错误匹配的概率. 上述相似性距离可以通过传统算法计算得到，也可以与检测模型一起学习得到. Wang 等[9]和 Zhang 等[10]利用联合检测和嵌入(joint detection and embedding，JDE)模型同时学习目标检测和外观特征提取，在 ResNet34[11]的基础上添加深层特征聚合结构，融合不同层级的深度特征进行预测，通过多任务学习同时训练中心点损失、偏移损失和身份特征的交叉熵损失. JDE 通过一个网络便能得到目标检测结果和对应的外观特征向量. 在目标差异性明显的稀疏场景中，JDE 具有不错的性能；但当目标差异性小时，JDE 难以通过单独的网络学习到相似样本之间细小的差异，直接影响数据关联模块的匹配准确性.

轨迹预测需要通过时间序列的运动变化估计当前图像帧中的轨迹位置. 卡尔曼滤波算法[12]是目前较为通用的轨迹预测算法，通过时间序列上分布偏差的协方差矩阵以及上一时刻的观测值估计目标在当前时刻的运动位置，在速度和精度上取得不错的效果. 此外，Fang 等[13]通过循环神经网

络构建轨迹预测模型，根据跟踪轨迹的历史位置计算当前时刻位置. 当目标运动简单且有规律时，通过循环神经网络估计的轨迹位置较为准确；但当目标运动方式变得复杂时，如非线性运动或者变速运动，循环神经网络的估计结果与目标真实值之间就会出现较大的误差，造成错误的关联结果，出现轨迹身份交换现象[14].

为了切实解决 MOT 方法在复杂场景中的面临的挑战，本文针对短道速滑场景中存在的目标相对尺度小、遮挡严重、运动变化复杂等问题，设计了一个应用于短道速滑场景的 MOT 数据集，提出一种应用于复杂场景的 MOT 方法，通过外观线索和IoU距离构建联合代价矩阵，解决检测与轨迹之间因特征差异小引起的错误匹配问题；提出一种基于预测增强的轨迹预测方法，通过轨迹的全局特征和运动线索增强轨迹的特征表现力，提高遮挡轨迹的重新匹配能力，保证目标长期稳定的轨迹序列；提出一种基于检测先验的轨迹状态控制(trajectory state controller, TSC)方法，根据检测先验结果控制新轨迹的初始化，减少误检目标对跟踪对象的影响.

1 短道速滑数据集

本文提出一个应用于短道速滑场景的 MOT 数据集(简称 SkatingTrack 数据集)，为复杂场景中的目标跟踪任务提供基准. SkatingTrack 数据集采集于北京体育大学二七国家冰雪运动训练科研基地和首都体育馆，在 2 个场馆分别标注了 44 402 张和 8 776 张数据图片，采集的图片分辨率为 1 920×1 080 像素，视频帧速为 30 帧/s. 通过 Labelme 工具对采集的视频进行注释，带有注释的标签包括轨迹的编号和包围框. 对于完全遮挡的目标，在目标遮挡结束重新出现时标注与之前相同的身份标记. SkatingTrack 数据集采集运动员训练视频共 56 个，其中, 36 个视频作为训练集, 10 个作为验证集, 10 个作为测试集，在划分过程中，保证每个子集在视频总长度上的合理划分. SkatingTrack 的基本数据信息如表 1 所示，采集图像中的待检测目标特点如下:

(1) 目标的相对尺度小、外观相似. 与 MOT 数据集相比，SkatingTrack 数据集中运动员目标在固定相机下的相对尺度较小，且比赛时衣着相似，导致跟踪轨迹间外观差异小，容易造成数据匹配时的错误关联;

(2) 跟踪轨迹快速变化的运动状态. 运动状态包括运动方向和速度，与其他场景相比，运动员比赛过程中的运动方向和速度变化大，容易导致运动模型预测的估计值与真实值之间有较大偏移;

(3) 跟踪遮挡严重且频繁，交叉区域相互遮挡. 运动员在入弯以及出弯时抢占有利位置，短时间内目标会在小范围内聚集，使得更容易出现遮挡现象，容易造成检测器漏检.

表 1 SkatingTrack 数据集与常用的 MOT 数据集对比

数据集	视频	平均轨迹数	总轨迹数	帧速/(帧·s^{-1})	总图片帧
MOT17[15]	14	96	1 342	30	11 235
MOT20[16]	8	432	3 456	25	13 410
DanceTrack[17]	100	9	900	20	105 855
SkatingTrack	56	6	336	30	53 178

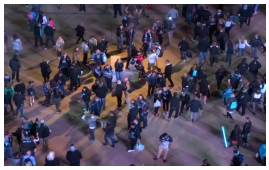
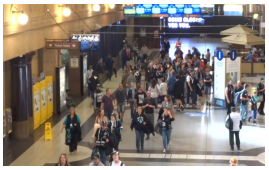
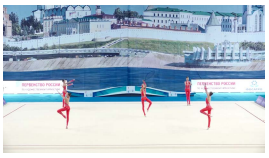
目前，针对一些复杂场景的MOT数据集(如表 2 所示)，现有算法仍能取得较好的跟踪效果，如 MOT20 数据集[15]中通过增加目标密度提升跟踪难度；但是，个体间外观差异明显，无法在数据匹配时设置匹配障碍，跟踪器很容易根据外观或者位置区分目标的不同，轨迹预测器也可以通过规律的运动状态预测目标的轨迹位置. DanceTrack 数据集[17]通过减少目标之间的差异性提高跟踪难度，使目标对象之间的姿态、衣着、运动状态相似，增加数据关联的不确定性. 虽然在一定程度上能验证跟踪算法在不同场景中的鲁棒性，但是跟踪个体在视频中相对尺度较大，个体间仍保持着一定的相对距离，现有的算法仍可根据目标的外观差异或位置差异得到较好的跟踪精度，却无法验证 MOT 方法在复杂场景中的有效性. 与 MOT17 和 MOT20 数据集相比, SkatingTrack 数据集在总视频数量上要多，虽然其中每个视频的轨迹数较少，但目标对象的运动变化比 MOT 数据集更难处理. 与 DanceTrack 数据集相比，SkatingTrack 数据集更关注于目标对象在时空上复杂的运动状态变化，而非关注轨迹的初始化或者中断的处理操作. 因此，在 SkatingTrack 场景中的 MOT 方法需要着重关注跟踪过程中的稳定性和鲁棒性.

2 本文方法

2.1 方法框架

本文提出一种基于轨迹预测增强(trajectory prediction enhancement, TPE)的 MOT 方法(prediction enhancement tracker, PETracker)，其框架如图 1 所示.

表 2　不同 MOT 数据集场景对比

数据集	场景 1		场景 2		场景 3	
	编号	图片	编号	图片	编号	图片
MOT17[15]	01		08		12	
MOT20[16]	04		06		07	
DanceTrack[17]	09		48		67	
SkatingTrack	01		05		41	

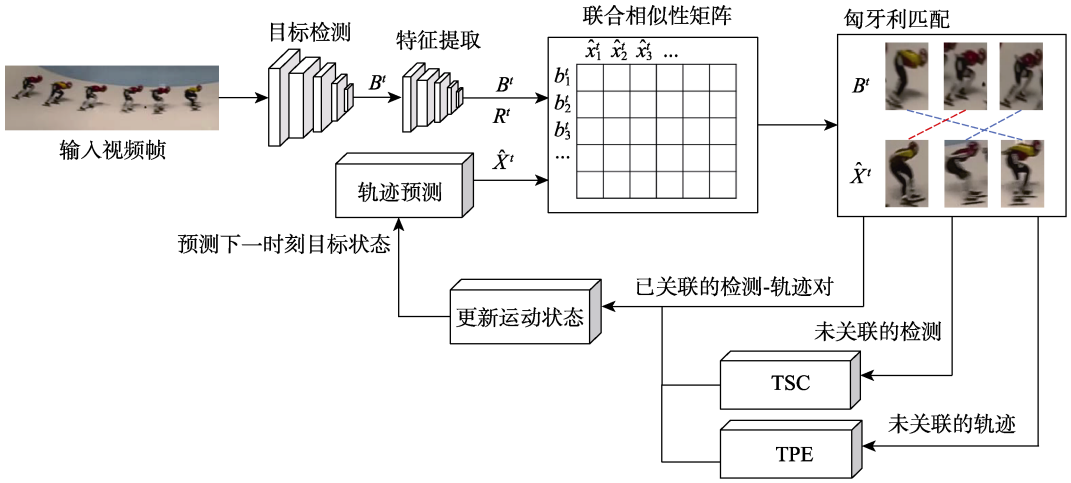


图 1　PETracker 算法框架

(1) 检测. 给定视频序列 $V^n=\{I^1,I^2,\cdots,I^n\}$，其中，$n$ 表示视频长度. 通过 Yolo5 检测器定位图像帧中的运动员目标，t 时刻每帧图像中的检测集合表示为 $B^t=\{b_1^t,b_2^t,\cdots,b_N^t\}$.

(2) 特征提取. 对每个目标包围框使用特征提取网络提取高维特征向量 $\boldsymbol{r}_i^t=\mathbb{R}^D$.

(3) 预测更新. 使用卡尔曼滤波在时域上估计目标状态的运动变化，根据上一个时刻的估计状态和当前时刻的观测状态更新当前时刻的估计状态 $\hat{\boldsymbol{X}}^t$.

(4) 数据关联. 将估计的位置 $\hat{\boldsymbol{X}}^t$ 和检测器定位的对象集合进行匹配. 根据检测与轨迹之间的 IoU 距离和外观特征的余弦距离构建联合相似性矩阵，使用匈牙利匹配算法计算检测目标响应与跟踪轨迹估计值的最优匹配.

2.2　联合相似性矩阵

通过检测结果与轨迹预测结果之间的 IoU 距离和外观特征的余弦距离构建联合相似性矩阵[2]，能够有效地表现目标之间差异，从而更好地使得轨迹与正确的目标进行匹配. 由马氏距离表示的检测响应与轨迹预测的 IoU 距离可表示为

$$\boldsymbol{D}_{i,j}^t=\left(b_i^t-\hat{x}_j^t\right)^{\mathrm{T}}\boldsymbol{P}_j^{-1}\left(b_i^t-\hat{x}_j^t\right) \tag{1}$$

其中，$\boldsymbol{D}_{i,j}^t$ 表示 t 时刻检测响应和跟踪轨迹间的空间距离代价；$\boldsymbol{P}_j$ 表示卡尔曼滤波所预测的当前观测空间的协方差矩阵；$\hat{x}_j^t$ 表示轨迹 j 在当前时刻通过运动模型预测的空间位置；b_i^t 表示 t 时刻的检测响应.

采用余弦距离表现运动员间的外观差异，通过行人重识别网络计算检测响应的外观特征向量 $\boldsymbol{r}_i^t$，跟踪轨迹的全局外观特征表示为 $R_j=\left\{\boldsymbol{r}_j^k\right\}_{k=1}^t$，检测目标与轨迹之间的外观相似度表示为

$$\boldsymbol{E}_{i,j}^t=\min\left\{1-\boldsymbol{r}_i^{\mathrm{T}}\boldsymbol{r}_j^k \mid \boldsymbol{r}_j^k\in R_j\right\} \tag{2}$$

将 2 种相似性度量方式联合，最终检测与轨迹估计之间的联合相似性矩阵表示为

$$\boldsymbol{C}_{i,j}^t=\lambda\boldsymbol{D}_{i,j}^t+(1-\lambda)\boldsymbol{E}_{i,j}^t \tag{3}$$

2.3 TPE 方法

本文提出轨迹预测增强(trajectory prediction enhancement, TPE)方法，如图 2 所示. 当跟踪目标因视角遮挡而中断时, TPE 通过轨迹的全局外观特征和运动模型的运动线索维护中断轨迹的特征状态，提高跟踪轨迹在遮挡结束后与正确的检测响应重新关联的能力.

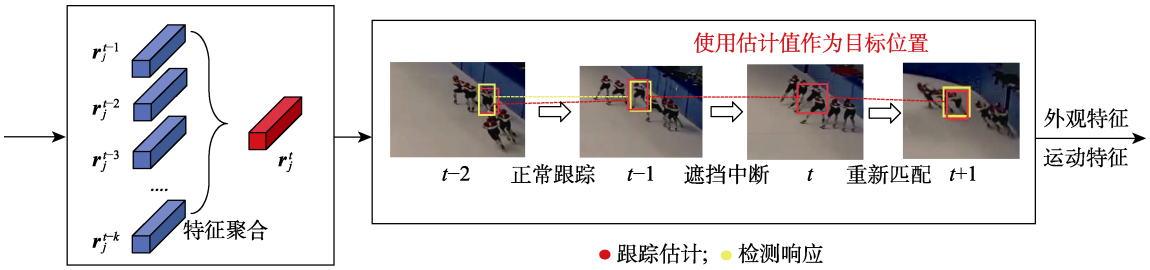


图 2 TPE 方法结构

假设 t 时刻轨迹 j 没有检测响应关联，即跟踪中断. 此时，跟踪轨迹的外观特征集合为 R_j，则遮挡时跟踪轨迹的特征可表示为

$$\boldsymbol{r}_j^t=\frac{1}{t-1}\sum_{u=1}^{t-1}\boldsymbol{r}_j^u \tag{4}$$

考虑时间与精度的权衡，计算时通常使用最近的 K 个外观特征向量作为轨迹的特征与检测的特征进行关联. 通过式(4)计算遮挡时轨迹特征能够融合更长时间内的外观特征变化，有效地提高轨迹在遮挡结束时与正确的检测结果关联的概率. 此外，运动模型根据上一时刻目标的预测值和观测值计算当前时刻目标位置. 假设 $\hat{\boldsymbol{X}}^t$ 是运动模型在时刻 t 的预测值，$\boldsymbol{X}^t$ 是轨迹集合的真实位置，$\boldsymbol{F}_t$ 是运动变换矩阵，$\boldsymbol{P}^t$ 是实际值与计算值的协方差矩阵，$\dot{\boldsymbol{P}}^t$ 是实际值与估计值的协方差矩阵，$\boldsymbol{Q}^t$ 为过程噪声协方差矩阵；则 t 时刻跟踪轨迹的预测值表示为

$$\begin{cases}\hat{\boldsymbol{X}}^t=\boldsymbol{F}_t\boldsymbol{X}^{t-1}\\ \boldsymbol{P}^t=\boldsymbol{F}_t\dot{\boldsymbol{P}}^{t-1}\boldsymbol{F}_t^{\mathrm{T}}+\boldsymbol{Q}^t\end{cases} \tag{5}$$

若此时刻有观测值 b_m^t 与预测值 $\hat{\boldsymbol{X}}^t$ 匹配，则卡尔曼滤波更新后验估计可表示为

$$\begin{cases}\boldsymbol{K}^t=\dot{\boldsymbol{P}}^t\boldsymbol{H}_t^{\mathrm{T}}\left(\boldsymbol{H}_t\boldsymbol{P}^t\boldsymbol{H}_t^{\mathrm{T}}+\boldsymbol{N}\right)^{-1}\\ \boldsymbol{X}^t=\hat{\boldsymbol{X}}^t+\boldsymbol{K}^t\left(b_m^t-\boldsymbol{H}_t\hat{\boldsymbol{X}}^t\right)\\ \dot{\boldsymbol{P}}^t=\left(1-\boldsymbol{K}^t\boldsymbol{H}_t\right)\boldsymbol{P}^t\end{cases} \tag{6}$$

其中，$\boldsymbol{H}_t$ 表示观测矩阵；$\boldsymbol{N}$ 表示观测噪声的协方差矩阵；$\boldsymbol{K}^t$ 表示卡尔曼增益. 当 t 时刻跟踪轨迹中断时, TPE 方法通过运动模型将中断轨迹的估计位置作为当前时刻的实际位置更新轨迹状态.

TPE 通过聚合目标的全局外观特征预测遮挡轨迹的外观特征变化，通过运动模型预测遮挡轨迹的空间位置变化，可以有效地减少遮挡问题对跟踪过程的影响，提高跟踪轨迹的重关联能力.

2.4 TSC 方法

本文提出的 TSC 方法根据检测结果的先验估计引导最大可生成轨迹数，减少检测器误检的异常影响，提高跟踪模型中数据关联阶段的鲁棒性.

假设视频序列的长度为 L，通过 TSC 方法计算整个视频序列中可生成的最大目标身份数和该视频序列中允许出现的最多跟踪轨迹身份数的公式分别为

$$G_{\max}=S\left(T_i\right),\ 1\leqslant i\leqslant n \tag{7}$$

$$T_{\max}=\begin{cases}G_{\max}, & \text{if } G_{\max}\geqslant L/2\\ M, & \text{otherwise}\end{cases} \tag{8}$$

其中，T_i 表示 t 时刻视频帧中检测响应的数量；M 表示预设目标轨迹数. 设 t 时刻有检测响应未关联轨迹，若当前跟踪轨迹数少于 $T_{\max}$，此时允许检测响应作为新轨迹生成；否则将检测响应作为非必要的误检响应抑制.

SkatingTrack 数据集中运动员目标运动状态变化较快，当轨迹长时间中断时，运动模型预测位置与目标实际位置差距较大. 本文设置中断控制阈值 α 控制运动模型连续预测的最大时长，当轨迹连续中断帧超过 α 时，TPE 方法不再显式地预测轨迹位置，能够有效地解决卡尔曼滤波连续预测产生的漂移问题.

TSC 方法步骤如下:

输入. 检测结果 B^t.

输出. 视频可生成的最大目标身份数 $T_{\max}$.

Step1. 统计视频每帧中检测数量.

Step2. 统计视频序列中所有检测数量出现的次数，最多出现次数为 $G_{\max}$.

Step3.

if ($G_{\max} \geqslant L/2$){

允许生成的最多轨迹数 $T_{\max}$ 为 $G_{\max}$.

}

else {

允许生成的最多轨迹数 $T_{\max}$ 为预设值 M.

}

3　实　验

PETracker 算法基于 Pytorch 实现，训练环境为 CUDA11.0+torch1.8，训练及测试时显卡型号为 NVIDIA GeForce GTX 2080Ti 11 GB. 算法参数设置如下：检测相应的最小置信度设置为 0.3，用于数据关联的外观特征集合最大记录为 100，预设目标轨迹数 M =6，轨迹初始化在视频的前 30 帧内完成，初始化时，需要在第 1 次检测后连续 5 帧内均有配对的检测响应. 中断轨迹阈值设置为 5 帧，代价矩阵中外观特征与运动特征的比例为 9∶1.

3.1　评价指标

为了从数据角度体现 PETracker 算法的性能，本文采用 MOTChallenge[15]公开测试的量化标准进行评估. 以伪正例(false positives, FP)、伪负例(false negatives, FN)、身份交换(identify switch, IDs)和真实轨迹(ground truth, GT)值 GT_t 为指标进行计算.

MOT 准确度 A_{MOT} 的公式为

$$A_{\mathrm{MOT}} = 1 - \frac{\sum_{t=1}^{n} \mathrm{FN}_t + \mathrm{FP}_t + \mathrm{IDs}_t}{\sum_{t=1}^{n} \mathrm{GT}_t} \tag{9}$$

MOT 精确度 P_{MOT} 以真实轨迹包围框与预测包围框的交并比 $D_{t,i}$ 和时刻 t 为预测轨迹，与真值轨迹成功匹配的数量 C_t 作为指标计算，其更加注重检测质量. P_{MOT} 的公式为

$$P_{\mathrm{MOT}} = \sum_{t=1}^{n}\sum_{i} D_{t,i} \Big/ \sum_{t=1}^{n} C_t \tag{10}$$

身份 F_1 得分(identify F_1 score, IDF_1)表示为正确识别的检测数和计算检测数的比值，通过识别精确度 P_{ID} 和识别召回率 R_{ID} 计算 IDF_1 得分. IDF_1 的公式为

$$\mathrm{IDF}_1 = \frac{2 \times P_{\mathrm{ID}} \times R_{\mathrm{ID}}}{P_{\mathrm{ID}} + R_{\mathrm{ID}}} \tag{11}$$

多数跟踪率(mostly tracked, MT)表示为成功跟踪的轨迹与真值轨迹的比值超过 80%的轨迹数.

多数丢失率(mostly lost, ML)表示为成功跟踪的轨迹与真值轨迹的比值小于 20%的轨迹数.

高阶跟踪准确度(higher order tracking accuracy, HOTA) A_{HOT}，通过检测准确性(detection accuracy, DA)得分 S_{DA} 和关联准确性(association accuracy, AA)得分 S_{AA} 验证 MOT 算法的综合准确性，A_{HOT} 的公式为

$$A_{\mathrm{HOT}} = \sqrt{S_{\mathrm{DA}} \times S_{\mathrm{AA}}} \tag{12}$$

A_{MOT} 和 IDF_1 是衡量 MOT 算法有效性的重要指标，前者考虑误检、漏检和身份交换对跟踪结果的影响，后者关注轨迹的身份是否正确.

3.2　实验与分析

3.2.1　权重对比实验

权重对比实验分为 2 部分:

(1) 验证不同权重比的联合相似性矩阵在 SkatingTrack 数据集中的表现，结果如图 3 所示. 可以看出，当外观特征占比为100%时，在SkatingTrack数据集的测试集中，PETracker 算法的 A_{MOT} 可以达到 84.2%，IDF_1 达到 84.0%；当外观特征距离与 IoU 距离比例为 9∶1 时，A_{MOT} 在原基础上提高 1.9 个百分点，IDF_1 提高 1.5 个百分点；随着 IoU 距离比例的增加，整体性能表现下降，这是因为通过深度网络提取的目标外观线索能够反映目标在深层向量空间上的语义差异，而 IoU 距离能够提供空

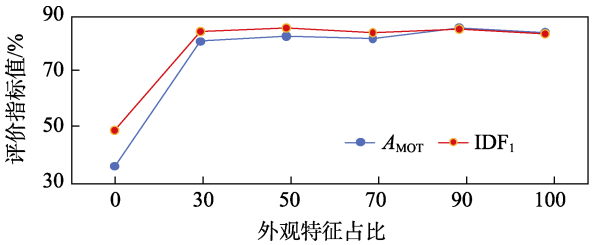


图 3　联合相似性矩阵特征比例精度对比图

间约束，可以有效地防止目标与位置上差异大的其他目标关联，因此，联合相似性矩阵能够较好地反映目标与轨迹之间的空间差异以及外观差异.

(2) 验证中断控制阈值 α 减少时运动模型预测偏移的有效性，结果如表 3 所示. 可以看出，在 α 为 7 帧时跟踪模型整体表现最佳，与 α 为 3 帧时相比，A_{MOT} 提高 3.1 个百分点，$\mathrm{IDF_1}$ 提升 1.7 个百分点；当 α 为 9 帧时，由于运动员目标运动速度较快，运动模型预测的位置与实际位置相差较大，因此与 α 为 7 帧时相比，A_{MOT} 下降 0.9 个百分点，$\mathrm{IDF_1}$ 下降 0.5 个百分点.

表 3 中断控制阈值不同设置对性能的影响

α	A_{MOT}/%	$\mathrm{IDF_1}$/%	IDs
3	85.1	87.8	95
5	86.7	88.5	94
7	**88.2**	**89.5**	76
9	87.3	89.0	**76**

注. 加粗表示最优值.

3.2.2 模块消融实验

在 SkatingTrack 数据集的验证集上验证 PETracker 算法的有效性，结果如表 4 所示. 可以看出，在基线跟踪器(BaseLine)上使用 TSC 方法时，$\mathrm{IDF_1}$ 提高 0.9 个百分点，IDs 的数量由原来的 82 降低到 68；TSC 方法可以减少检测器误检对数据匹配的影响，抑制错误轨迹的生成；在 BaseLine 上使用 TPE 方法时，A_{MOT} 提高 7 个百分点，$\mathrm{IDF_1}$ 提升 4.6 个百分点，在跟踪轨迹因遮挡或漏检现象而中断时，TPE 方法利用运动线索和特征线索维持轨迹的状态变化，当轨迹与检测响应重新匹配时，能有效地提升与正确检测之间的配对概率；TPE 方法通过运动模型预测被遮挡对象的空间位置，被遮挡目标的不定性会随被遮挡时间的增加而增加，IDs 相比增加了 28，FP 从 160 增加到 697. 在 BaseLine+TPE 上使用 TSC 方法可以减少运动模型对较长时间丢失轨迹的不可靠的预测，使 IDs 由 110 下降到 95.

表 4 PETracker 算法消融实验对比

方法	A_{MOT}/%	$\mathrm{IDF_1}$/%	IDs	FP	FN
BaseLine	75.7	78.9	82	**160**	5 779
BaseLine+TSC	75.9	79.8	**68**	276	5 279
BaseLine+TPE	82.7	83.5	110	697	3 315
BaseLine+TSC+TPE	**85.1**	**87.8**	95	704	**2 586**

注. 加粗表示最优值.

3.2.3 算法对比实验

在 SkatingTrack 数据集的测试集中与目前主流的 MOT 算法进行实验，结果如表 5 所示，其中，测试时所有算法使用相同的检测输出结果. 可以看出，在相互遮挡严重、目标外观差异小、运动变化迅速的 SkatingTrack 数据集上，PETracker 算法的精度优于目前主流的算法，保持着较高的跟踪精度；与 DeepSORT 算法相比，PETracker 算法在保证长期稳定的跟踪前提下提高了中断轨迹的重匹配能力，其中，A_{MOT} 提高 14.3 个百分点，$\mathrm{IDF_1}$ 提高 21 个百分点. 实验结果证明，当跟踪场景中目标遮挡严重时，使用 JDE 模型难以得到理想的跟踪效果；PETracker 算法能够有效地提高中断轨迹的重匹配能力，降低检测器误检和漏检对跟踪过程的影响.

表 5 5 种算法在 SkatingTrack 数据集上精度对比

算法	A_{MOT}/%	$\mathrm{IDF_1}$/%	IDs	P_{MOT}/%	MT/%	ML/%
DeepSORT[2]	74.2	68.8	**138**	81.9	49.0	10.2
FairMOT[10]	72.2	24.9	375	83.6	22.2	6.6
ByteTrack[18]	82.6	61.7	144	**86.9**	77.8	**0.0**
OC-SORT[7]	77.2	56.3	211	84.7	38.9	**0.0**
PETracker	**88.5**	**89.8**	146	82.0	**89.8**	**0.0**

注. 加粗表示最优值.

在其他复杂场景的 MOT 数据集 DanceTrack 中，验证 PETracker 算法的性能，结果如表 6 所示. 可以看出，PETracker 算法仍然能够保持较高的准确度和较好的鲁棒性，与 DeepSORT 算法相比，该算法在跟踪精度上有较明显的提升，其中，A_{HOT} 提升 4 个百分点，$\mathrm{IDF_1}$ 提升 1.2 个百分点.

3.3 可视化对比

为了更加直观地体现本文跟踪算法的有效性，对 PETracker 算法与 DeepSORT 算法在 SkatingTrack 数据集上部分比赛视频的可视化效果进行对比，结果如表 7 所示. 可以看出，场景 1 中，跟踪个体在远点弯道时目标相对尺度小，容易出现低质量检测以及漏检问题，此时 DeepSORT 算法出现较为严重的轨迹中断现象，而 PETracker 算法则能较为准确地维持跟踪轨迹的状态，在检测器漏检严重时对目标对象的轨迹位置进行预测，避免出现轨迹碎片，保证长期稳定跟踪；场景 2 中，跟踪个体位于近点弯道时目标之间位置相近，导致 DeepSORT 算法在检测筛选时出现漏检现象，因此该算法在数据关联时缺少正确的检测结果与跟踪轨迹对应，导致错误的匹配结果，PETracker 算法

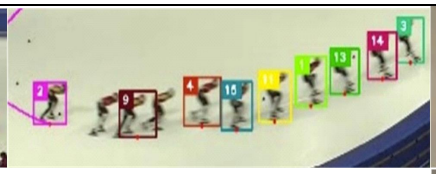
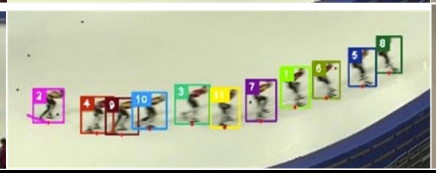
表 6　8 种算法在 DanceTrack 数据集上精度对比

算法	A_{HOT} /%	IDF_1/%	A_{MOT} /%	S_{DA} /%	S_{AA} /%
CenterTrack[19]	41.8	35.7	86.8	78.1	22.6
FairMOT[10]	39.7	40.8	82.2	66.7	23.8
MOTR[20]	48.4	46.1	79.2	74.5	32.7
TraDes[21]	43.3	41.2	86.2	71.8	25.4
ByteTrack[18]	47.3	52.5	89.5	71.6	31.4
OC-SORT[7]	**55.1**	**54.2**	**89.4**	**80.3**	**38.0**
DeepSORT[2]	38.4	43.9	81.3	66.7	22.2
PETracker	42.4	45.1	80.5	66.1	27.3

注. 加粗表示最优值.

通过 TSC 方法能够维持正确数量的跟踪轨迹数，并在关联阶段根据轨迹数进行筛选，有效地提高了数据关联阶段的正确率；场景 3 中，在检测器出现漏检时，DeepSORT 算法只将检测的目标结果与轨迹预测结果进行关联，在目标之间差异性小时容易造成错误的匹配，导致轨迹在短时间内出现连续的身份交换问题，而本文方法提出 TPE 方法，根据运动模型记录的运动状态预测被遮挡轨迹的实际位置，保证了轨迹跟踪的连续性，减少跟踪轨迹的身份交换，有效地提高跟踪准确度.

表 7　2 种算法在 SkatingTrack 数据集上可视化效果对比

算法	场景 1	场景 2	场景 3	场景 4
DeepSORT[2]				
PETracker				

4　结　语

本文设计了一种应用于短道速滑场景的 MOT 数据集 SkatingTrack，填补了当前对短道速滑场景 MOT 数据集的空白；提出轨迹预测增强方法 PETracker，当轨迹因遮挡而发生中断时，通过全局特征和运动线索补全中断期间的轨迹运动状态和外观变化，并提出 TSC 方法，根据检测的先验减少检测器误检对数据关联的影响. 实验结果表明，与现有 MOT 方法相比, PETracker 在 SkatingTrack 数据集上取得了目前最好的跟踪精度，可以在不重新训练网络的前提下，有效地提高跟踪模型在复杂场景下的鲁棒性和准确性. PETracker 已成功地应用在短道速滑国家队的日常训练中，为 MOT 方法在实际复杂场景中的应用提供了新的思路和改进方向.

参考文献(References):

[1] Bewley A, Ge Z Y, Ott L, *et al*. Simple online and realtime tracking[C] //Proceedings of the IEEE International Conference on Image Processing. Los Alamitos: IEEE Computer Society Press, 2016: 3464-3468

[2] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric[C] //Proceedings of the IEEE International Conference on Image Processing. Los Alamitos: IEEE Computer Society Press, 2017: 3645-3649

[3] Yu F W, Li W B, Li Q Q, *et al*. POI: multiple object tracking with high performance detection and appearance feature[C] //Proceedings of the Conference on Computer Vision. Heidelberg: Springer, 2016: 36-42

[4] Rezatofighi S H, Milan A, Zhang Z, *et al*. Joint probabilistic data association revisited[C] //Proceedings of the IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2015: 3047-3055

[5] Reid D. An algorithm for tracking multiple targets[J]. IEEE transactions on Automatic Control, 1979, 24(6): 843-854

[6] Kuhn H W. The Hungarian method for the assignment problem[J]. Naval Research Logistics Quarterly, 1955, 2(1-2): 83-97

[7] Cao J K, Pang J M, Weng X S, *et al*. Observation-centric SORT: rethinking SORT for robust multi-object tracking[OL]. [2022-07-25]. https://arxiv.org/abs/2203.14360

[8] Lu Z C, Rathod V, Votel R, *et al*. RetinaTrack: online single stage joint detection and tracking[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2020: 14668-14678

[9] Wang Z D, Zheng L, Liu Y X, *et al*. Towards real-time multi-object tracking[C] //Proceedings of the 16th European Conference on Computer Vision. Heidelberg: Springer, 2020: 107-122

[10] Zhang Y F, Wang C Y, Wang X G, *et al*. FairMOT: on the fairness of detection and re-identification in multiple object tracking[J]. International Journal of Computer Vision, 2021, 129(11): 3069-3087

[11] He K M, Zhang X Y, Ren S Q, *et al*. Deep residual learning for image recognition[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2016: 770-778

[12] Kalman R E. A new approach to linear filtering and prediction problems[J]. Journal of Fluids Engineering, 1960, 82(1): 35-45

[13] Fang K, Xiang Y, Li X C, *et al*. Recurrent autoregressive networks for online multi-object tracking[C] //Proceedings of the IEEE Winter Conference on Applications of Computer Vision. Los Alamitos: IEEE Computer Society Press, 2018: 466-475

[14] Zou Z B, Huang J J, Luo P. Compensation tracker: reprocessing lost object for multi-object tracking[OL]. [2022-07-25]. https://arxiv.org/abs/2008.12052

[15] Milan A, Leal-Taixé L, Reid I, *et al*. MOT16: a benchmark for multi-object tracking[OL]. [2022-07-25]. https://arxiv.org/abs/1603.00831

[16] Dendorfer P, Rezatofighi H, Milan A, *et al*. MOT20: a benchmark for multi object tracking in crowded scenes[OL]. [2022-07-25]. https://arxiv.org/abs/2003.09003

[17] Sun P Z, Cao J K, Jiang Y, *et al*. DanceTrack: multi-object tracking in uniform appearance and diverse motion[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 20993-21002

[18] Zhang Y F, Sun P Z, Jiang Y, *et al*. ByteTrack: multi-object tracking by associating every detection box[C] //Proceedings of the 17th European Conference on Computer Vision. Heidelberg: Springer, 2022

[19] Zhou X Y, Koltun V, Krähenbühl P. Tracking objects as points[C] //Proceedings of the 16th European Conference on Computer Vision. Heidelberg: Springer, 2020: 474-490

[20] Zeng F G, Dong B, Zhang Y, *et al*. MOTR: end-to-end multiple-object tracking with transformer[C] //Proceedings of the 17th European Conference on Computer Vision. Heidelberg: Springer, 2022

[21] Wu J J, Cao J L, Song L C, *et al*. Track to detect and segment: an online multi-object tracker[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2021: 12352-12361