计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2024,33(11):15−26 [doi: 10.15888/j.cnki.csa.009681] [CSTR: 32024.14.csa.009681]

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

联合判别性外观和运动线索的行人多目标跟踪①

王　军, 李迎春, 程　勇

(南京信息工程大学 软件学院, 南京 210044)
通信作者: 李迎春, E-mail: 202212210018@nuist.edu.cn

摘　要: 在多目标跟踪任务中, 外界噪声的干扰会导致传统方法的系统建模不可靠, 从而降低目标位置预测的准确性; 而密集人群引起的拥挤和遮挡问题则会严重影响目标外观的可靠性, 导致错误的身份关联. 为了解决这些问题, 本文提出一种多目标跟踪算法 Ecsort. 该算法在传统运动预测的基础上, 引入噪声补偿模块, 降低噪声干扰引起的误差, 提高位置预测的准确性. 其次, 引入特征相似度匹配模块, 通过学习目标的判别性外观特征, 并结合运动线索和判别性外观特征的优势, 从而实现精确的身份关联. 通过在多目标跟踪基准数据集上进行的大量实验结果表明, 与基线模型相比, 该方法在 MOT17 测试集上的 *IDF*1 (ID *F*1 score)、HOTA (higher order tracking accuracy)、AssA (association accuracy)、DetA (detection accuracy) 分别提高了 1.1%、0.5%、0.6%、0.3%, 在 MOT20 测试集上的 *IDF*1、HOTA、AssA、DetA 分别提高了 2.3%、1.9%、3.4%、0.2%.

关键词: 多目标跟踪; 运动线索; 判别性外观特征; 噪声补偿; 数据关联

引用格式: 王军,李迎春,程勇.联合判别性外观和运动线索的行人多目标跟踪.计算机系统应用,2024,33(11):15–26. http://www.c-s-a.org.cn/1003-3254/9681.html

Pedestrian Multi-object Tracking Combining Discriminative Appearance and Motion Cues

WANG Jun, LI Ying-Chun, CHENG Yong

(School of Software, Nanjing University of Information Science & Technology, Nanjing 210044, China)

Abstract: In multi-object tracking tasks, the interference of external noise can lead to unreliable system modeling of traditional methods, thus reducing the accuracy of object position prediction; and the congestion and obstruction caused by dense crowds seriously affect the reliability of the object appearance, resulting in incorrect identity association. To address these issues, this study proposes a multi-object tracking algorithm Ecsort. This algorithm improves position prediction accuracy by introducing a noise compensation module based on traditional motion prediction to reduce errors caused by noise interference. Secondly, this algorithm introduces a feature similarity matching module. It can achieve accurate identity association by learning discriminative appearance features of objects and combining the advantages of motion cues and discriminative appearance features. Extensive experimental results on multi-object tracking benchmark datasets demonstrate that, compared to the baseline model, this method improves ID *F*1 score (*IDF*1), higher order tracking accuracy (HOTA), association accuracy (AssA), and detection accuracy (DetA) by 1.1%, 0.5%, 0.6%, and 0.3% respectively on the MOT17 test set, and by 2.3%, 1.9%, 3.4%, and 0.2% respectively on the MOT20 test set.

Key words: multi-object tracking (MOT); motion cue; discriminative appearance feature; noise compensation; data association

① 基金项目: 国家自然科学基金 (41975183)
收稿时间: 2024-04-25; 修改时间: 2024-05-29; 采用时间: 2024-06-04; csa 在线出版时间: 2024-09-27
CNKI 网络首发时间: 2024-09-30

多目标跟踪 (multi-object tracking, MOT) 是利用目标的空间和时间特征在整个视频序列中预测目标位置的一项任务. 多目标跟踪的目标是在图像或视频的连续帧中识别和连接多个移动目标, 同时在时间序列中保持它们的身份. 更具体地说, 多目标跟踪是获取初始检测集合、分配唯一 ID 并在视频帧之间跟踪它们, 同时保持分配的 ID 的过程. 目前, 多目标跟踪已经成为深度学习中的重要课题, 在自动驾驶等领域有着广泛应用.

在最近的研究中, 多目标跟踪任务主要遵循检测后跟踪 (tracking-by-detection, TBD) 范式[1–5]进行探索. TBD 范式包括两个阶段: 目标检测阶段和跟踪阶段. 在目标检测阶段, 对视频帧中的目标进行定位和识别. 在跟踪阶段[6,7], 连接被检测到的目标到现有轨迹或通过建模被跟踪对象的状态变化[8,9], 并将它们与检测结果匹配, 建立新的轨迹.

与 TBD 范式不同, 联合检测和跟踪 (joint detection and tracking, JDT) 范式将目标检测阶段和跟踪阶段耦合.

尽管一些研究[10,11]表明 JDT 范式取得了一些进展, 但检测任务与重新识别 (re-identification, ReID) 任务有很大的区别, 并且 JDT 范式的共享功能模型可能会降低每个任务的性能. TBD 范式通过将目标检测和跟踪分为两个独立的阶段, 可以更有效地利用计算资源, 适用于资源受限的环境或实时应用.

许多 TBD 方法在实现跟踪阶段的方式上存在差异. 跟踪阶段通常包括两个部分: (1) 运动建模和状态估计. 卡尔曼滤波器 (Kalman filter, KF)[12]通常是这类任务的典型选择. (2) 数据关联. TBD 方法通常通过逐帧匹配, 在当前帧的检测和先前存在的轨迹集之间建立一对一的关联. 解决数据关联问题, 主要有两种方法[13]: (1) 定位视频帧中的目标, 使用二部图匹配算法将预测轨迹边界框和当前检测边界框的交并比 (intersection of union, IoU) 关联起来. (2) 对目标的外观进行建模以解决 ReID 任务. 在跟踪阶段, Sort[1,14] 和 DeepSort[3] 使用卡尔曼滤波器来估计目标状态, 并假设恒定速度模型作为帧之间的过渡函数, 是最经典且应用最广泛的算法.

然而, Sort 和 DeepSort 算法仍存在一些局限性. 首先, 高帧率下可以将行人移动近似为线性运动, 但也使模型对噪声更为敏感. 具体而言, 目标的运动和状态存在不确定性, 运动和状态不确定性带来的噪声在高帧率视频中被放大[14]; 另外, 传感器误差和测量过程中的干扰也会向原始模型引入噪声. 其次, Sort 算法的数据关联策略不够鲁棒. Sort 算法使用 IoU 作为相似性度量, 其性能依赖于目标检测算法的质量. Sort 算法在利用基于运动的线索进行短期关联方面取得了一些成功, 但 Sort 算法无法在一段时间内为相同目标保持相同的身份, 导致频繁的 ID 切换, 如图 1 所示. DeepSort 通过引入外观特征来恢复长时间遮挡对象的身份, 在原始 IoU 匹配的基础上, 它还引入级联匹配和 ReID 网络减少 ID 切换. 然而, 目标在遭受严重遮挡或强烈的外界光照变化时, DeepSort 的外观嵌入模型[9]提取的外观信息变得不可靠. 此外, 许多外观追踪器采用了指数移动平均 (exponential moving average, EMA) 方法[15]作为特征更新策略. EMA 减小了外界噪声的干扰, 然而, 在外观特征更新时, 会带来滞后性. 换句话说, EMA 对当前帧的外观特征响应较慢, 导致一些高度可信的外观被忽视.

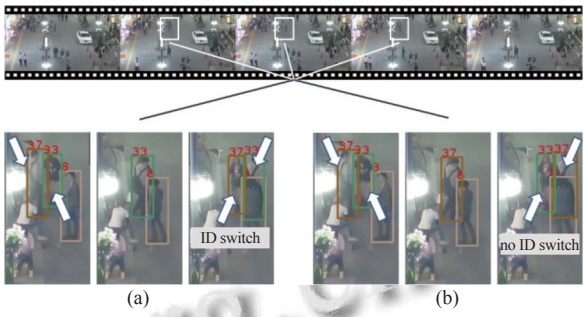

图 1 拥挤和遮挡场景的示例

为了缓解这些局限性的影响, 本文在这项工作中提出了一些改进. 方法的流程如图 2 所示. 首先, 在 MOT 场景中的运动模型主要受到过程噪声和测量噪声的影响, 因此本文设计了一个噪声补偿模块来减小由过程和测量噪声引起的误差. 具体而言, 由于遮挡程度的变化和每个目标实例运动和状态不确定性的差异, 每个实例对噪声的敏感性在视频序列中也会有所变化, 通过动态调整过程噪声和测量噪声, 以减小噪声带来的误差, 产生更准确的预测. 其次, 为了减少由严重遮挡引起的错误匹配, 本文设计了一个新的数据关联模块. 它可以通过结合运动和外观线索来恢复长时间丢失的对象身份. 此外, 研究发现使用 EMA 更新特征的外观追踪器可能会忽略最近的重要外观信息, 为此, 本文提出了一个新颖的特征更新策略, 以获取更可信的外观线索并减少对当前特征的滞后.

本文提出的方法被命名为 Ecsort, 主要贡献如下.

(1) 提出了一种新颖的跟踪器 Ecsort, 它结合了强大的运动和外观线索, 以解决由遮挡和密集场景引起的预测误差和错误匹配问题.

(2) 在原始的运动模型上设计并添加了一个噪声补偿模块, 可以减小由场景噪声引起的模型误差, 并防止轨迹偏移, 以产生更精准的预测.

(3) 提出了一种新颖的特征更新策略, 用于学习具有判别性的外观特征. 此外, 设计了一个新的数据关联模块, 用于匹配相邻帧之间的轨迹和检测. 本文将以上两个模块集成到一个特征相似度匹配模块中, 可以在遮挡和拥挤场景中重新识别对象并恢复丢失的轨迹.

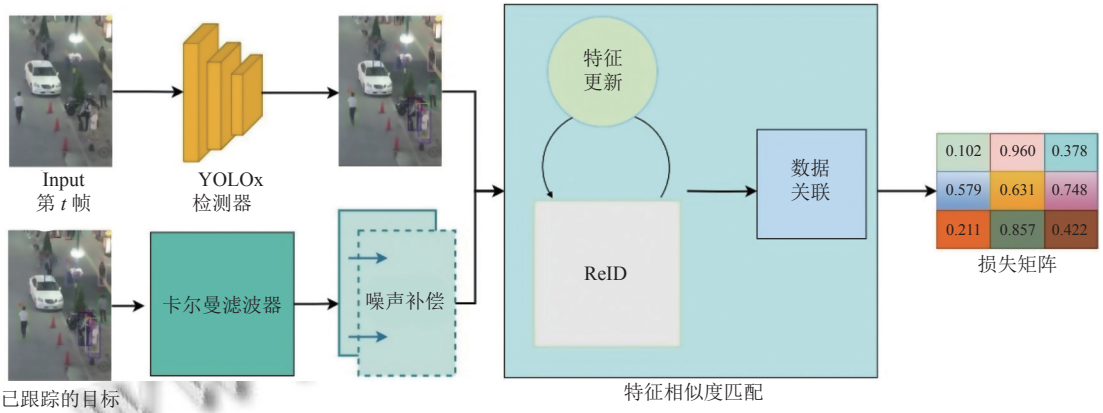

图 2 Ecsort 的 pipeline

1 相关工作

1.1 检测后跟踪 (TBD) 范式

随着目标检测器[16–20]的发展, 越来越多的跟踪器利用强大的检测器实现卓越的跟踪性能[5,13]. 许多现代跟踪器遵循 TBD 范式, 将 MOT 分为两个独立的任务, 即目标检测和跟踪. 目标检测器, 如 Faster R-CNN[16] 、CenterNet[17] 、SDP[21] 、DPM[22]等, 在预测目标边界框时表现出卓越的性能. 然后, 通过关联网络, 将检测结果与相邻帧之间的轨迹进行匹配. 由于检测可以直接由检测器生成, 因此, TBD 范式更侧重于在跟踪阶段提高性能.

在早期的研究中, Sort 采用卡尔曼滤波器生成候选边界框, 并利用匈牙利算法构建轨迹, 通过计算边界框的重叠来关联不同帧的检测结果. IoU-Tracker[2]则是将前一帧的轨迹与当前帧的检测结果相关联. 尽管这些方法简单高效, 但它们依赖于最近邻假设[23], 导致 Sort 和 IoU-Tracker 在遮挡和拥挤的场景中性能发挥较差. 一些研究致力于优化 Sort 一类的算法, 而不使用 ReID 特征以实现更好的跟踪性能. 例如, ByteTrack[5]通过利用低分检测与轨迹的相似性来提高跟踪性能, OC-Sort[14]提出了一种以观测为中心的 Sort 算法来解决遮挡和非线性物体运动的跟踪误差. 然而, 大多数工作[6–8,10]侧重于通过引入 ReID 网络来区分目标外观, 以产生更强大的预测并提高匹配的准确性[24].

1.2 运动模型

大多数 MOT 算法采用的运动模型基于贝叶斯滤波器[25], 将运动预测视为状态估计, 通过最大化后验估计[14]来预测下一帧的状态. 经典的卡尔曼滤波器以线性恒定速度模型为假设, 是一个遵循预测-更新循环的递归贝叶斯滤波器. 鉴于其简单的特性, 许多研究利用卡尔曼滤波器来预测目标运动. 考虑到线性运动假设的局限性, 一些研究开始探索卡尔曼滤波器的更高级变体, 以增强不同方面的运动预测. 例如, 扩展卡尔曼滤波器 (extended Kalman filter, EKF)[26]和无迹卡尔曼滤波器 (unscented Kalman filter, UKF)[27], 它们被用来处理非线性运动. 然而, 它们仍依赖于对近似 KF 假设的高斯先验, 并且需要运动模式假设. 一些研究采用 KF 的变体, 如 NSA-Kalman 滤波器[13,15,28]将检测分数合并到 KF 中. 另一方面, 粒子滤波器[14,29]用于处理非线性运动, 但消耗大量计算资源. 因此, 粒子滤波器很少在 MOT 任务中被使用, 大多数采用的运动模型仍然基于卡尔曼滤波器.

1.3 外观模型

为了在数据关联中实现卓越的性能, 一些工作[30–32]引入了 ReID 特征作为外观模型, 通过外观线索来区分对象. 在早期的工作中, DeepSort 使用外观特征

进行数据关联. 此后, 大多数方法倾向于通过训练复杂的外观模型来提取可靠的外观特征. MOTDT[31]通过在共享特征图上应用 RoI-pooling[33]实现实时跟踪. JDE[34]在一阶段目标检测器 YOLOv3[18,32]中添加了额外的 ReID 分支, 以减少成本计算并获得高效的 ReID 特征. FairMOT[9]通过共享的骨干网络学习目标的定位和外观特征, 以获得更好的外观嵌入. 最近, TransTrack[35]、TrackFormer[10]和 MOTR[11]尝试利用注意力机制来跟踪对象. 尽管这些算法训练技术很先进, 但这些方法需要较高的推理计算成本. 本文提出了一种低滞后性的外观特征更新策略, 使模型在不增加推理成本的情况下, 学习物体的判别性特征.

2 方法

在本节中, 基于现有 TBD 方法的缺点进行了几项改进. 通过将这些改进引入到 ByteTrack 跟踪器中, 本文提出了 Ecsort 跟踪器, 其总体架构如图 3 所示, 包括目标检测和跟踪两个阶段. Ecsort 算法的流程图如图 4 所示.

2.1 卡尔曼滤波器概述

在 Sort 中, 状态向量被表示为一个七元组$x=[x_c, y_c, s, a, \dot{x}_c, \dot{y}_c, \dot{s}]$, 其中$(x_c, y_c)$是对象中心的水平和垂直坐标. s表示对象的边界框缩放, a表示对象的边界框纵横比. $\dot{x}_c, \dot{y}_c, \dot{s}$分别是相应时间的导数 (变化率). 最近一些跟踪器[9,5,15,34]已经改变了状态向量的表示, 将七元组替换为八元组$x=[x_c, y_c, a, h, \dot{x}_c, \dot{y}_c, \dot{a}, \dot{h}]$, 相较于七元组, 八元组新增了向量$h$来表示对象的边界框高度, 但是去掉了表示边界框缩放的向量s. $\dot{x}_c, \dot{y}_c, \dot{a}, \dot{h}$分别表示相应时间的导数. Ecsort 选择八元组来表示状态向量, 观测值表示为一个四元组$z=[x_c, y_c, a, h]$, 其中(x_c, y_c)是对象中心位置的坐标, a和h分别表示候选边界框的宽高比和高度.

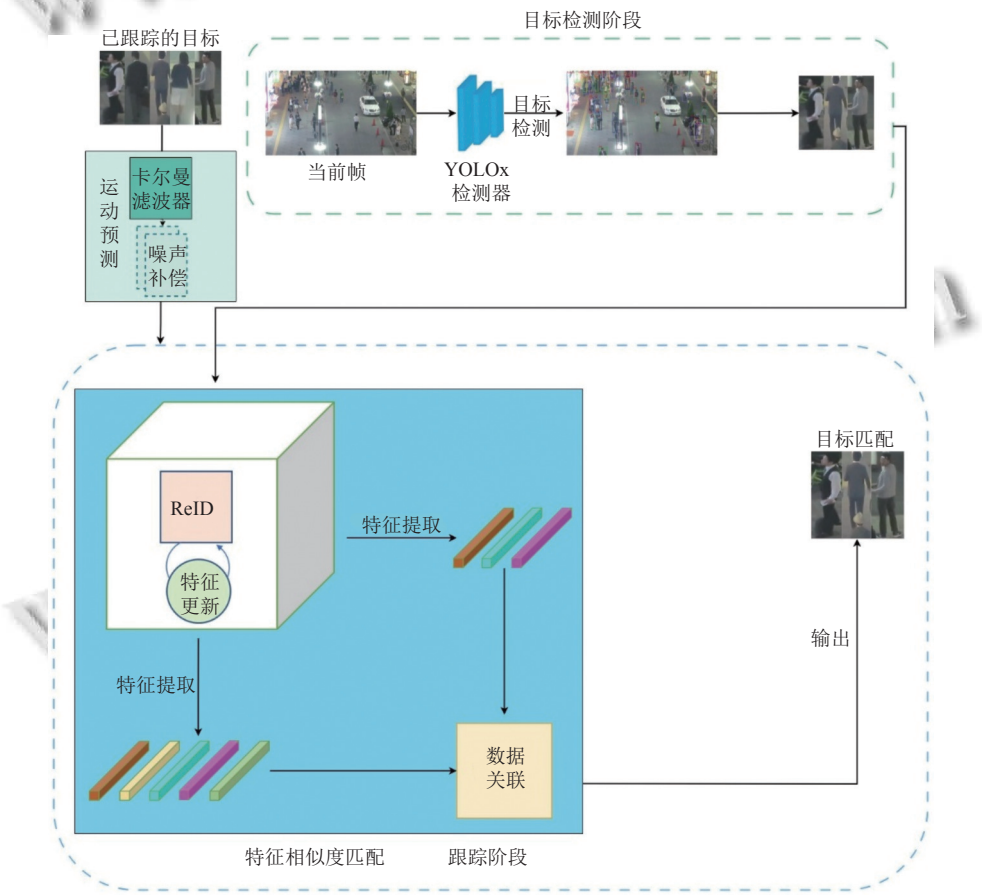

图 3 Ecsort 跟踪器总体架构图

2.2 噪声补偿模块

为了对图像中不同帧之间的目标运动进行建模, 通常使用带有恒定速度模型的卡尔曼滤波器 (KF). KF 通过利用依赖于前一帧和当前测量的状态估计来估计

下一个时间步长的目标状态.

KF 的目标是在给定测量$z \in R^m$和已知x_0的情况下, 对运动目标进行先验状态估计. 在不存在主动控制的目标跟踪任务中, 卡尔曼滤波器由以下线性随机差分方程进行控制:

$$x_k = F_k x_{k-1} + n_{k-1} \tag{1}$$

$$z_k = H_k x_k + v_k \tag{2}$$

其中, F_k是转移矩阵, H_k是观测矩阵, n_{k-1}和v_k分别表示过程和测量噪声, 它们遵循正态分布.

$$n_{k-1} \sim N(0, Q_k),\ v_k \sim N(0, R_k) \tag{3}$$

卡尔曼滤波器包括预测阶段和更新阶段, 遵循递归方程:

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1}$$

其中, Q_k和R_k分别表示过程噪声协方差和测量噪声协方差, 预测阶段和更新阶段分别如式 (4) 和式 (5) 所示. 在每个时间步长k, KF 在预测阶段推导出状态的先验估计$\hat{x}_{k|k-1}$和协方差矩阵$P_{k|k-1}$. 在更新阶段, 给定目标状态的测量z_k, KF 更新后验状态估计$\hat{x}_{k|k}$和估计协方差$P_{k|k}$.

$$\begin{cases} P_{k|k-1} = F_k P_{k-1|k-1} F_k^{\top} + Q_k \\ K_k = P_{k|k-1} H_k^{\top} (H_k P_{k|k-1} H_k^{\top} + R_k)^{-1} \end{cases} \tag{4}$$

$$\begin{cases} \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (z_k - H_k \hat{x}_{k|k-1}) \\ P_{k|k} = (I - K_k H_k) P_{k|k-1} \end{cases} \tag{5}$$

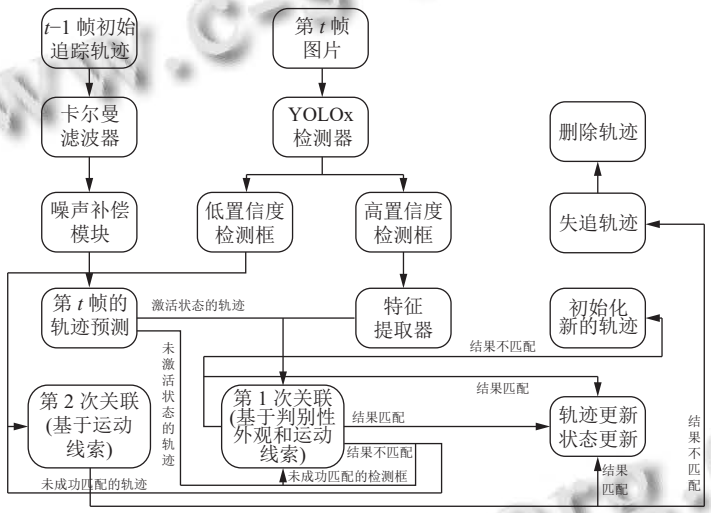

图 4 Ecsort 算法的流程图

由于运动和状态的不确定性引入的过程噪声, 在高帧率视频中的连续帧之间被放大, 同时, 目标遮挡、摄像机运动和光照变化导致测量结果不太可靠, 会为运动模型引入测量噪声. 在更新阶段, 原始运动模型忽略由目标遮挡、状态和运动不确定性以及摄像机运动带来的噪声, 后验状态估计经过预测-更新循环后, 容易造成误差积累, 导致系统建模不可靠. 为了解决这些问题, 通过调整不同场景的噪声, 以获得更可靠的运动模型:

$$\tilde{Q}_k = \left(1 + \frac{\delta}{\epsilon_k}\right) \times Q_k \tag{6}$$

$$\tilde{R}_k = \gamma \times \epsilon_k^{1-\gamma} \times R_k \tag{7}$$

其中, ϵ_k是第k帧的检测分数, $\tilde{Q}_k$和$\tilde{R}_k$分别表示噪声补偿后的过程噪声协方差和测量噪声协方差, δ和γ是超参数. 考虑到 MOT 场景中目标运动和状态的不确定性, 以及相机运动等外部干扰, Ecsort 设置的自适应噪声略高于原始运动模型的噪声. 直观上, 当检测得分ϵ_k越高, 过程噪声协方差$\tilde{Q}_k$和测量噪声协方差$\tilde{R}_k$就越低. 根据式 (4) 和式 (5), 较低的过程噪声协方差$\tilde{Q}_k$可以得到较低的先验误差协方差$P_{k|k-1}$, 先验误差协方差矩阵的减少意味着测量不准确, 而估计更可靠 (误差小), $P_{k|k-1}$减小导致卡尔曼增益减小, 系统对先验状态的预测更有信心, 可以减少因测量不准确而带来的模型误差; 相反, 较低的$\tilde{R}_k$会导致卡尔曼增益增加, 这意味着测量是可靠的, 而估计是不准确的. 在更新步骤中, 测量值具有更大的权重, 后验状态估计更接近测量值, 可以减少由于估计不准确而导致的模型误差.

2.3 特征相似度匹配模块

2.3.1 特征更新

大多数配备外观嵌入模型的跟踪器[13,15]尝试通过引入复杂的 ReID 网络来提取可靠的外观特征, 这在性能上带来了轻微的提高, 但使网络结构更加复杂, 牺牲了大量的计算资源和时间成本.

在最近的工作[13,15]中, 轨迹的外观嵌入是由逐帧深度检测嵌入的指数移动平均 (EMA) 描述的. 它需要一个平滑因子α来调整先前和当前帧外观嵌入的比例. 标准 EMA 公式如下:

$$e_i^t = \alpha \times e_i^{t-1} + (1-\alpha) \times f_i^t \tag{8}$$

其中, e_i^{t-1}是第i个轨迹在$t-1$帧的平均外观嵌入, f_i^t是当前匹配检测的外观嵌入, α是平滑因子. EMA 利用了帧间特征变化的信息, 它可以稍微抑制检测噪声.

然而, 标准 EMA 有一些局限性. EMA 给予早期特征更大的权重, 过于依赖先前的特征. 同时, 它对当前特征的变化响应不及时, 并且在特征更新期间可能会忽略外观特征的短期变化, 潜在地导致重要特征信息的丢失, 这对于恢复被遮挡对象是不利的. 平滑因子的选择对 EMA 的性能有重要影响. 平滑因子的选择不当可能导致过度平滑或平滑不足, 从而影响外观嵌入的可靠性.

因此, 本文提出了一种新颖且高效的低滞后 EMA 来替代标准 EMA. 首先, 通过减少先前特征的比例, 并同等增加当前特征的比例来减少 EMA 中的滞后. 其次, 设计一个自适应平滑因子α_a替换原来的α, 该平滑因子取决于先前帧和当前帧的检测置信度. 低滞后 EMA 如下:

$$e_i^t = \alpha_a \times [e_i^{t-1} + \beta \times (f_i^t - e_i^{t-1})] + (1-\alpha_a) \times sf_i^t \tag{9}$$

$$\alpha_a = \alpha_c + (1-\alpha_c) \times \min\left(1 - \frac{\delta_t - \delta_{t-1}}{1-\delta_{t-1}}, 1\right) \tag{10}$$

$$\beta = \begin{cases} \beta - \psi_{\text{high}}, & \beta > \psi_{\text{dis}} \\ 0.1, & \psi_{\text{high}} < \beta < \psi_{\text{dis}} \\ 0.05, & \psi_{\text{low}} < \beta < \psi_{\text{high}} \\ 0.01, & \text{otherwise} \end{cases} \tag{11}$$

其中, β是一个滞后因子, ψ_{dis}, ψ_{low}, ψ_{high}分别表示不同的检测得分阈值, 滞后因子β是基于外观可靠性动态引入的. f_i^t是当前匹配检测的外观嵌入, e_i^{t-1}是第 i 个轨迹在$t-1$帧的平均外观嵌入. δ_{t-1}是第$t-1$帧的检测置信度, δ_t是第t帧的检测置信度, α_c是外观嵌入常量. 与标准 EMA 相比, 低滞后 EMA 不引入额外的超参数.

2.3.2 数据关联模块

在最近的研究中, 一些不依赖外观的跟踪器[5,14], 例如, Sort 简单地使用 IoU 作为相似性度量, 随着检测器的发展, 基于 IoU 的数据关联算法在跟踪性能上进步显著, 但它们的整体性能不如配备外观模型的跟踪器. 其次, 许多配备外观模型的跟踪器采用了外观和运动成本的一般加权和作为数据关联策略, 它们过度依赖外观并相对性地忽视了运动线索, 导致目标在外观不可靠时性能表现较差.

为了充分利用最新的多目标跟踪研究, 将外观模型集成到跟踪器 Ecsort 中. 外观模型基于 BOT, 使用 ResNet50 作为主干网络.

回顾一些最近的研究, 并发现仅依赖外观特征可能不够有效甚至是无效的. 因此, 本文提出了一种新的方法, 使用余弦距离和 IoU 掩码将外观特征和运动线索结合作为数据关联算法. 首先, 过滤掉余弦距离和 IoU 距离超过给定阈值的候选对象, 仅保留余弦相似度和 IoU 相似度较高的候选目标. 然后, 根据余弦距离低于给定阈值且 IoU 距离高于给定阈值的条件将剩余候选目标分为两部分. 接下来, 对剩余划分的候选目标分别应用不同的加权和作为它们的代价矩阵. 最后, 选择每个矩阵元素中余弦距离和 IoU 距离的最小值作为最终成本矩阵.

$$d_{i,j} = \begin{cases} \zeta \times d_{i,j}^{\cos} + (1-\zeta) \times d_{i,j}^{\text{iou}}, & d_{i,j}^{\cos} < \eta_{\text{emb}}, \eta_{\text{iou}} > d_{i,j}^{\text{iou}} \\ 1, & d_{i,j}^{\cos} > \eta_{\text{emb}}, \eta_{\text{iou}} < d_{i,j}^{\text{iou}} \\ \lambda \times d_{i,j}^{\cos} + (1-\lambda) \times d_{i,j}^{\text{iou}}, & \text{otherwise} \end{cases} \tag{12}$$

$$Cost_{i,j} = \min(d_{i,j}, d_{i,j}^{\text{iou}}) \tag{13}$$

其中, $d_{i,j}^{\cos}$是第i个轨迹外观和第j个检测之间的余弦距离, $d_{i,j}^{\text{iou}}$是第i个轨迹和第j个检测之间的 IoU 距离, $d_{i,j}$是初始成本矩阵. ζ和λ表示余弦距离的权重. η_{emb}是外观阈值, 用于拒绝不可靠的外观嵌入. η_{iou}是 IoU 阈值, 用于拒绝不可能的轨迹和检测之间的边界框对.

3 实验

3.1 实验设置

本文在典型的 MOT 基准测试上评估了跟踪器

Ecsort: MOT17[36]和 MOT20[37]数据集, 它们处于“私有检测”协议下. MOT17 包括用静止和移动摄像机捕捉的视频序列, 而 MOT20 的场景看起来更拥挤. 两个数据集都由训练集和测试集组成, 不包含验证集. 对于消融研究, Ecsort 遵循 ByteTrack, 使用 MOT17 训练集中每个视频的前半部分进行训练, 后半部分进行验证[5,38].

评估指标: 采用广为接受的 CLEAR 指标[39]作为本文的评估指标. 它包含多目标跟踪精度 (*MOTA*)、ID *F*1 score (*IDF*1)、关联精度 (AssA)、检测精度 (DetA)、ID switch (*IDs*)、高阶跟踪精度 (HOTA) 等.

$$MOTA = 1 - \frac{\sum(FN + FP + IDs)}{GT} \tag{14}$$

其中, *GT* 是所有真实值的数目, *MOTA* 基于 *FN*、*FP*、*IDs* 综合判定跟踪错误次数, 更强调检测性能, *IDs* 与检测器稳定性相关[40].

$$IDF1 = \frac{2 \times IDTP}{IDTP + IDFP + IDFN} \tag{15}$$

*IDF*1 一般被用来评价跟踪性能[41], *IDF*1 指标是正确关联的目标占所有真实目标和检测目标之和的比例, 其中 *IDTP*、*IDFP*、*IDFN* 指标均考虑了 ID 信息[42], 因此 *IDF*1 能体现目标身份维持能力[43]. 对于外观特征提取器, Ecsort 采用 FastReID 中的 SBS50 模型, 对 MOT17 和 MOT20 数据集训练 60 epoch 的默认训练策略.

实现细节: 所有实验都在 Intel(R) Xeon(R) Gold5218R CPU@2.10 GHz 和 NVIDIA A10 GPU 上运行. 本文的基线 (baseline) 基于 ByteTrack, 检测器采用了和 ByteTrack 相同的 YOLOx, 来对跟踪性能进行公平的比较[1,2,28]. 本文遵循 ByteTrack, 在 MOT17 验证集上进行消融研究. 在所有实验中, 将ψ_{dis}, ψ_{low}, ψ_{high}分别设置为 0.9, 0.8 和 0.7, 将α_c设置为 0.5, ζ和λ分别设置为 0.8 和 1. 若未特殊指定, 检测分数阈值τ被设置为默认值 0.6, 噪声补偿因子σ和γ取不同值时的跟踪效果如表 1, 实验结果表明, 当σ=1 和γ=1 时, *IDF*1 相对 baseline 提升了 0.8, 产生的 *IDs* 减少了 5, 达到最佳跟踪效果. 相似度阈值设置为 0.2, 如果检测与轨迹之间的相似度小于阈值, 则拒绝匹配. 本文对η_{emb}和η_{iou}也做了灵敏度实验, 当η_{emb}和η_{iou}取不同的值对时, 跟踪效果也有所差异, 从表 2 中可以看出, 当η_{emb}=0.3, η_{iou}=0.3, 跟踪效果最好, 产生的 *IDs* 最少, 跟踪更为稳定. 对于低滞后 EMA 策略, 在 MOT17 验证集上做了灵敏度实验, 实验结果如表 3 所示, 当α_c取 0.5 时, *MOTA* 和 *IDF*1 分别取得最优, 跟踪效果达到最佳, 产生的 *IDs* 也更少, 跟踪效果达到最佳. 滞后因子β设置如式 (11) 所示. 对于丢失的轨迹, 如果在 30 帧内再次出现, 将保留它们, 以防轨迹重新出现. 线性轨迹插值的最大间隔为 20, 以补偿检测的缺失.

表 1 MOT17 验证集的δ和γ超参数实验

σ	γ	*MOTA* (%)	*IDF*1 (%)	*IDs*
1	2	76.1	77.6	339
1	1	76.6	80.1	242
0	1	76.6	79.3	247
0	2	75.7	77.5	408
2	2	76.4	77.8	314
2	1	76.6	79.9	249
1.5	1	76.6	79.9	249
−1	∀	—	—	—
∀	0	—	—	—
1	0.5	76.5	79.0	254
0	0.5	76.5	79.0	268
2	0.5	76.5	78.9	256
1	0.3	76.6	78.7	254
2	0.3	76.5	78.6	255
0	0.3	76.5	78.9	265
1	0.8	76.6	79.5	255
0	0.8	76.6	79.3	254
2	0.8	76.6	79.4	252

注: —代表不存在这种情况

表 2 MOT17 验证集的η_{emb}和η_{iou}灵敏度实验

η_{emb}	η_{iou}	*MOTA* (%)	*IDF*1 (%)	*IDs*
0.6	0.6	75.4	74.8	274
0.5	0.6	75.4	74.8	267
0.6	0.5	76.0	77.4	248
0.5	0.5	76.0	77.6	251
0.4	0.5	75.9	77.8	250
0.5	0.4	77.1	80.1	178
0.4	0.4	77.1	79.5	175
0.3	0.4	77.1	80.3	192
0.4	0.3	76.9	80.4	174
0.3	0.3	77.0	80.5	177
0.2	0.3	76.8	80.3	184
0.3	0.2	76.7	80.7	187
0.2	0.2	76.5	80.7	194

3.2 消融实验

首先, 进行消融实验来验证和量化每个组成部分的贡献. 本文主要关注追踪器的性能, 所以直接使用 ByteTrack 的 YOLOx 检测器. 为了公平比较, 所有实验的跟踪参数和 baseline 的其他设置都是相同的. 在本节中, 验证噪声补偿模块和特征相似度匹配模块对 baseline 的影响. 与 baseline 相比, 在 MOT17 验证集上, 单独引

入噪声补偿模块对 *MOTA*、*IDF*1 的性能分别提升 0.1、0.8, 产生的 ID switch 相较于 baseline 减少了 19, 单独引入特征相似度匹配模块对 *MOTA*、*IDF*1 的性能分别提升 0.2、0.9, 产生的 ID switch 减少了 20. 两个模块组合后引入对 *MOTA*、*IDF*1 的性能分别提升 0.4、1.2, 产生的 ID switch 减少了 29, 如表 4 所示. 实验结果表明噪声补偿模块和特征相似度匹配模块的引入可以实现更强大的预测和更准确的关联. 本文将部分跟踪结果的轨迹可视化, 如图 5、图 6 所示.

表 3 MOT17 验证集的α_c灵敏度实验

α_c	*MOTA* (%)	*IDF*1 (%)	*IDs*
0.9	76.8	80.4	182
0.8	76.8	80.5	189
0.7	76.8	80.5	189
0.6	76.8	80.4	187
0.5	77.0	80.5	177
0.4	76.9	80.5	177
0.3	76.9	80.5	178
0.2	76.8	80.5	179
0.1	76.8	80.4	182

表 4 MOT17 验证集的消融实验

噪声补偿	特征相似度匹配	*MOTA* (%)	*IDF*1 (%)	*IDs*
×	×	76.6	79.3	206
×	√	76.8	80.2	186
√	×	76.7	80.1	187
√	√	77.0	80.5	177

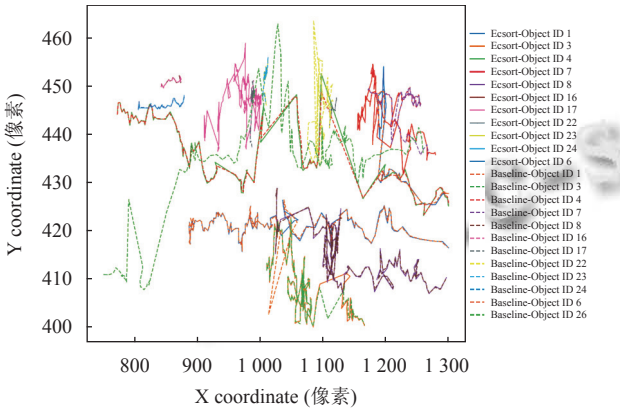

图 5 MOT17-02-FRCNN 序列的轨迹比较

3.3 实验分析

噪声补偿模块分析. 卡尔曼滤波器使用线性运动模型来预测物体轨迹在下一帧中的位置. 然而, MOT 场景中包含由行人状态和运动的不确定性、摄像机运动和光照变化引起的噪声, 受这些噪声的影响, 原始卡尔曼滤波器的运动模型容易造成误差累积, 产生不可靠的预测结果. 如图 7 所示, 原始卡尔曼滤波器预测视频序列下一帧目标轨迹的位置时, 它通常会导致轨迹边界框发生偏移, 例如图 7 中, (a) 中 ID 为 13 的行人、(b) 中 ID 为 20 的行人和 (c) 中 ID 为 23 的行人, 上一帧预测产生的边界框没有完整的覆盖当前帧的目标, 这将为 IoU 关联引入误差, 从而导致 IoU 关联不准确, 也会影响目标的特征提取. 为此, Ecsort 入了噪声补偿模块, 以减少噪声引起的模型误差, 产生更准确的预测. 表 4 显示了噪声补偿模块的影响. 本文将表 4 的结果部分可视化, 如图 8 所示, 添加噪声补偿模块后的卡尔曼滤波器能产生更精准的预测. 实验结果和可视化结果表明, 噪声补偿模块可以减少轨道边界框的偏移, 并能产生更准确的预测.

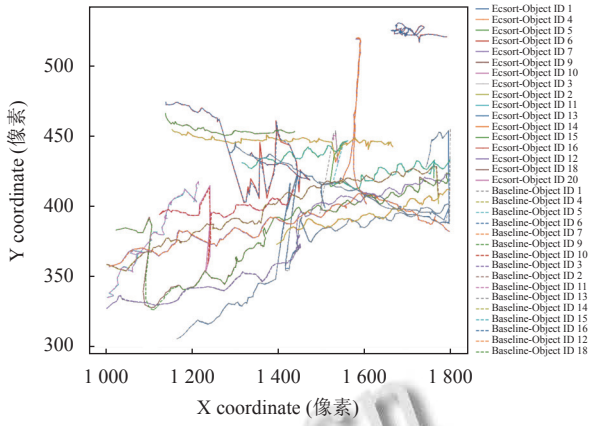

图 6 MOT17-09-FRCNN 序列的轨迹比较

图 7 原始卡尔曼滤波器预测目标轨迹位置的结果

特征相似度匹配模块分析. 基于 IoU 的方法通常被用于解决拥挤和遮挡问题, 但这些方法通常会导致遮挡场景中频繁的 ID switch, 如图 9 中 (a1)–(d1) 所示, 在发生拥挤和遮挡时, (a1) 中 ID 为 30 的行人在经过数帧后 ID 切换成了 34, (b1) 中 ID 为 13 和 25 的行人由原来的 ID 切换成了 36 和 13, (c1) 中 ID 为 71 的行

人由原来的 ID 变换成 67, (d1) 中 ID 为 37 和 33 的行人由原来的 ID 切换成了 33、37. 而 Ecsort 通过学习物体的判别性外观和运动线索来解决遮挡和拥挤场景, 将有效减少 ID 切换, 例如图 9 中 (a2)–(d2) 所示.

(a) (b) (c)

图 8 添加噪声补偿后预测目标轨迹位置的结果

(a1) (a2) (b1) (b2) (c1) (c2) (d1) (d2)

图 9 对比是否使用外观线索对跟踪性能的影响

本文对传统的 IoU 方法和 Ecsort 在处理遮挡问题方面的能力进行了比较, 可视化结果如图 9 所示, 实验证明了特征相似度匹配模块的有效性. 此外, 本节对特征相似度匹配模块做了消融研究, 实验表明特征更新和数据关联模块在 *MOTA* 和 *IDF*1 指标方面都提高了性能, 都有效减少了 ID switch, 结果如表 5 所示. 本文将表 4 的结果部分可视化, 如图 9 所示. 图 9(a1)–(d1), 采用运动线索 (IoU) 用于将当前检测与历史轨迹相匹配, 产生了 ID switch; 图 9(a2)–(d2), 使用特征相似度匹配模块学习目标的判别性外观特征和运动线索, 有效避免了 ID switch. 可视化结果证明了特征相似度匹配模块提高了目标身份关联性能, 可以有效解决拥挤和遮挡场景下身份关联错误问题.

表 5 MOT17 验证集特征相似度匹配模块的消融实验

特征相似度匹配		*MOTA* (%)	*IDF*1 (%)	*IDs*
低滞后EMA	数据关联			
×	×	76.9	76.4	309
×	√	76.7	80.1	187
√	×	77.3	76.3	296
√	√	76.8	80.2	186

3.4 基准评价

本文将 Ecsort 与最先进的跟踪器在 MOT17 和 MOT20 测试集上进行比较, 所有结果都来自官方的 MOTChallenge 服务器, 结果如表 6–表 9 所示.

MOT17 测试集的结果: 如表 6、表 7 所示, Ecsort 在许多主要指标 (如 *IDF*1、HOTA、AssA 和 DetA) 上优于所有最先进的跟踪器. 与 TBD 范式的跟踪器相比, 跟踪器 Ecsort 在 *IDF*1、HOTA、AssA、DetA、*IDs* 性能指标方面表现最好; 与 JDT 范式的跟踪器相比, Ecsort 在 *MOTA*、*IDF*1、HOTA、AssA、DetA 性能指标方面表现更加优异. 与基线跟踪器 ByteTrack 相比, 在 MOT17 测试集上, Ecsort 在 *IDF*1、HOTA、AssA、DetA 方面的性能分别提高了 1.1%、0.5%、0.6%、0.3%, 产生的 ID switch 减少了 645, 测试集的稳定提升证明了所提方法的鲁棒性.

表 6 在 MOT17 测试集上与 TBD 范式的跟踪方法对比

Tracker	*MOTA* (%)	*IDF*1 (%)	HOTA	AssA	DetA	*IDs*
Tube_TK	63.0	58.6	48.0	45.1	51.4	4 137
QuasiDense	68.7	66.3	53.9	52.7	55.6	3 378
MAT	69.5	63.1	53.8	57.2	55.1	2 844
SiamMOT	76.3	72.3	—	—	—	—
CountingSort	78.0	74.8	—	—	—	3 453
OcSort	78.0	77.5	63.2	—	—	1 950
ByteTrack	**80.3**	77.3	63.1	62.0	64.5	2 196
Ours	80.2	**78.4**	**63.6**	**62.6**	**64.8**	**1 551**

MOT20 测试集的结果: MOT20 被认为是难度更高、更复杂的基准测试, 与 MOT17 相比, MOT20 的拥挤和闭塞场景更为严重. 如表 8、表 9 所示, 在这个更具挑战性的场景中, 与 TBD 范式相比, Ecsort 在 *IDF*1、HOTA、AssA、DetA 性能指标方面排名第一; 与

JDT 范式相比, Ecsort 在所有性能指标方面都有明显领先. 与基线 ByteTrack 相比, Ecsort 在 *IDF*1, HOTA, AssA, DetA 性能方面分别提高了 2.3%, 1.9%, 3.4%, 0.2%, 测试集的稳定提升证明了所提方法的鲁棒性.

表 7 在 MOT17 测试集上与 JDT 范式的跟踪方法对比

Tracker	*MOTA* (%)	*IDF*1 (%)	HOTA	AssA	DetA	*IDs*
MOTR	65.1	66.4	—	55.7	60.3	2 049
CTracker	66.6	57.4	49.0	—	—	5 529
CenterTrack	67.8	64.7	52.2	51.0	53.8	3 039
SOTMOT	71.0	71.9	—	—	—	5 184
TransCenter	73.2	62.2	54.5	—	—	4 614
FairMOT	73.7	72.3	59.3	58.0	60.9	3 303
RelationTrack	73.8	74.7	61.0	61.5	60.6	**1 374**
PermaTrackPr	73.8	68.9	55.5	53.1	58.5	3 699
CsTrack	74.9	72.6	59.3	57.9	61.1	3 567
TransTrack	75.2	63.5	54.1	47.9	61.6	3 603
CorrTracker	76.5	73.6	60.7	58.9	62.9	3 369
Ours	**80.2**	**78.4**	**63.6**	**62.6**	**64.8**	1 551

表 8 在 MOT20 测试集上与 TBD 范式的跟踪方法对比

Tracker	*MOTA* (%)	*IDF*1 (%)	HOTA	AssA	DetA	*IDs*
MLT	48.9	54.6	43.2	—	—	2 187
Tracktor++	52.6	52.7	42.1	42.0	42.3	1 648
MPNTrack	57.6	59.1	46.8	47.3	46.6	1 210
SiamMOT	67.1	69.1	—	—	—	—
OcSort	75.7	76.3	62.4	60.8	60.5	**942**
ByteTrack	**77.8**	75.2	61.3	59.6	63.4	1 223
Ours	77.4	**77.5**	**63.2**	**63.0**	**63.6**	1 266

表 9 在 MOT20 测试集上与 JDT 范式的跟踪方法对比

Tracker	*MOTA* (%)	*IDF*1 (%)	HOTA	AssA	DetA	*IDs*
SOTMOT	68.6	71.4	43.2	57.3	57.7	4 209
TransCenter	61.9	50.4	—	—	42.3	4 653
FairMOT	61.8	67.3	54.6	54.7	54.7	5 243
RelationTrack	67.2	70.5	56.5	56.4	56.8	4 243
CsTrack	66.6	68.6	54.0	54.0	54.2	3 196
TransTrack	65.0	59.4	48.5	—	—	3 608
CorrTracker	65.2	69.1	—	—	—	5 183
Semi-TCL	65.2	70.1	55.3	—	—	4 139
Ours	**77.4**	**77.5**	**63.2**	**63.0**	**63.6**	**1 266**

4 结论

本文提出了 Ecsort 算法来处理多目标跟踪问题. 通过学习目标的判别性外观特征和运动线索来处理噪声干扰、拥挤和遮挡问题, 减少多目标跟踪中预测不准确、外观不可靠、关联错误现象的发生. 噪声补偿模块的引入可以减小由场景噪声引起的模型误差, 提高预测精度, 防止了轨道偏移. 特征相似度匹配模块的引入带来了更鲁棒的外观和更准确的数据关联. MOT17 和 MOT20 数据集上的结果表明了 Ecsort 的有效性和鲁棒性. 未来的工作针对如何在保证提升精度和稳定性的同时, 提升跟踪器的速度, 达到实时性跟踪这一方面进行研究.

参考文献

1 Bewley A, Ge ZY, Ott L, *et al*. Simple online and realtime tracking. Proceedings of the 2016 IEEE International Conference on Image Processing. Phoenix: IEEE, 2016. 3464–3468.

2 Bochinski E, Eiselein V, Sikora T. High-speed tracking-by-detection without using image information. Proceedings of the 14th IEEE International Conference on Advanced Video and Signal Based Surveillance. Lecce: IEEE, 2017. 1–6.

3 Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. Proceedings of the 2017 IEEE International Conference on Image Processing. Beijing: IEEE, 2017. 3645–3649.

4 Yu FW, Li WB, Li QQ, *et al*. POI: Multiple object tracking with high performance detection and appearance feature. Proceedings of the 2016 Computer Vision —ECCV 2016 Workshops. Amsterdam: Springer, 2016. 36–42.

5 Zhang YF, Sun PZ, Jiang Y, *et al*. ByteTrack: Multi-object tracking by associating every detection box. Proceedings of the 17th European Conference on Computer Vision. Tel Aviv: Springer, 2022. 1–21.

6 Wang YH, Hsieh JW, Chen PY, *et al*. SMILEtrack: Similarity learning for occlusion-aware multiple object tracking. arXiv:2211.08824, 2022.

7 Cai JR, Xu MZ, Li W, *et al*. MeMOT: Multi-object tracking with memory. Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 8090–8100.

8 Brasó G, Leal-Taixé L. Learning a neural solver for multiple object tracking. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 6247–6257.

9 Zhang YF, Wang CY, Wang XG, *et al*. FairMOT: On the fairness of detection and re-identification in multiple object tracking. International Journal of Computer Vision, 2021, 129(11): 3069–3087. [doi: 10.1007/s11263-021-01513-4]

10 Meinhardt T, Kirillov A, Leal-Taixé L, *et al*. TrackFormer: Multi-object tracking with Transformers. Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 8844–8854.

11 Zeng FG, Dong B, Zhang Y, *et al*. MOTR: End-to-end multiple-object tracking with Transformer. Proceedings of the 17th European Conference on Computer Vision. Tel Aviv: Springer, 2022. 659–675.
12 Welch GF. Kalman filter. In: Ikeuchi K, ed. Computer Vision: A Reference Guide. Cham: Springer, 2020. 1–3.
13 Aharon N, Orfaig R, Bobrovsky BZ. BoT-SORT: Robust associations multi-pedestrian tracking. arXiv:2206.14651, 2022.
14 Cao JK, Pang JM, Weng XS, *et al*. Observation-centric SORT: Rethinking SORT for robust multi-object tracking. Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver: IEEE, 2023. 9686–9696.
15 Du YH, Zhao ZC, Song Y, *et al*. StrongSORT: Make DeepSORT great again. IEEE Transactions on Multimedia, 2023, 25: 8725–8737. [doi: 10.1109/TMM.2023.3240881]
16 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal: MIT Press, 2015. 91–99.
17 Duan KW, Bai S, Xie LX, *et al*. CenterNet: Keypoint triplets for object detection. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 6569–6578.
18 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv:1804.02767, 2018.
19 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934, 2020.
20 Ge Z, Liu ST, Wang F, *et al*. YOLOx: Exceeding YOLO series in 2021. arXiv:2107.08430, 2021.
21 Yang F, Choi W, Lin YQ. Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 2129–2137.
22 Felzenszwalb P, McAllester D, Ramanan D. A discriminatively trained, multiscale, deformable part model. Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition. Anchorage: IEEE, 2008. 1–8.
23 Liang C, Zhang ZP, Zhou X, *et al*. Rethinking the competition between detection and ReID in multiobject tracking. IEEE Transactions on Image Processing, 2022, 31: 3182–3196. [doi: 10.1109/TIP.2022.3165376]
24 Qin Z, Zhou SP, Wang L, *et al*. MotionTrack: Learning robust short-term and long-term motions for multi-object tracking. Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver: IEEE, 2023. 17939–17948.
25 Gurajala R, Choppala PB, Meka JS, *et al*. Derivation of the Kalman filter in a Bayesian filtering perspective. Proceedings of the 2nd International Conference on Range Technology. Chandipur: IEEE, 2021. 1–5.
26 Yang SS, Baum M. Extended Kalman filter for extended object tracking. Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing. New Orleans: IEEE, 2017. 4386–4390.
27 Menegaz HMT, Ishihara JY, Borges GA, *et al*. A systematization of the unscented Kalman filter theory. IEEE Transactions on Automatic Control, 2015, 60(10): 2583–2598. [doi: 10.1109/TAC.2015.2404511]
28 Du YH, Wan JF, Zhao YY, *et al*. GIAOTracker: A comprehensive framework for MCMOT with global information and optimizing strategies in VisDrone 2021. Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops. Montreal: IEEE, 2021. 2809–2819.
29 Jinan R, Raveendran T. Particle filters for multiple target tracking. Procedia Technology, 2016, 24: 980–987. [doi: 10.1016/j.protcy.2016.05.215]
30 Seidenschwarz J, Brasó G, Serrano VC, *et al*. Simple cues lead to a strong multi-object tracker. Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver: IEEE, 2023. 13813–13823.
31 Chen L, Ai HZ, Zhuang ZJ, *et al*. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. Proceedings of the 2018 IEEE International Conference on Multimedia and Expo. San Diego: IEEE, 2018. 1–6.
32 Wang Q, Zheng Y, Pan P, *et al*. Multiple object tracking with correlation learning. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 3876–3886.
33 Girshick R. Fast R-CNN. Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 1440–1448.
34 Wang ZD, Zheng L, Liu YX, *et al*. Towards real-time multi-

object tracking. Proceedings of the 16th European Conference on Computer Vision. Glasgow: Springer, 2020. 107–122.

35 Sun PZ, Cao JK, Jiang Y, *et al*. TransTrack: Multiple object tracking with Transformer. arXiv:2012.15460, 2020.

36 Milan A, Leal-Taixe L, Reid I, *et al*. MOT16: A benchmark for multi-object tracking. arXiv:1603.00831, 2016.

37 Dendorfer P, Rezatofighi H, Milan A, *et al*. MOT20: A benchmark for multi object tracking in crowded scenes. arXiv:2003.09003, 2020.

38 Zhou XY, Koltun V, Krähenbühl P. Tracking objects as points. Proceedings of the 16th European Conference on Computer Vision. Glasgow: Springer, 2020. 474–490.

39 Weng XS, Wang JR, Held D, *et al*. 3D multi-object tracking: A baseline and new evaluation metrics. Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems. Las Vegas: IEEE, 2020. 10359–10366.

40 秦泽宇, 黄进, 杨旭, 等. 基于注意力机制和卡尔曼滤波的多目标跟踪. 计算机系统应用, 2021, 30(12): 128–138. [doi: 10.15888/j.cnki.csa.008214]

41 Ristani E, Solera F, Zou R, *et al*. Performance measures and a data set for multi-target, multi-camera tracking. Proceedings of the 2016 European Conference on Computer Vision. Amsterdam: Springer, 2016. 17–35.

42 王林, 郑有玲. 结合孪生网络重检的长期目标跟踪算法. 计算机系统应用, 2022, 31(4): 188–195. [doi: 10.15888/j.cnki.csa.008425]

43 Bergmann P, Meinhardt T, Leal-Taixé L. Tracking without bells and whistles. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 941–951.

(校对责编: 张重毅)