**分类号**  **学号**

**学校代码10487 密级**

****

**硕士学位论文**

**（学术型□ 专业型🗹）**

**基于视觉的多目标跟踪算法问题分析与改进研究**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **学位申请人** | **：** |  | **XXX** |  |  |
| **学科专业** | **：** |  | **XXXXX** |  |  |
| **指导教师** | **：** |  | **XXX 教授** |  |  |
| **答辩日期** | **：** |  | **XXXX年XX月XX日** |  |  |

**A Dissertation Submitted in Partial Fulfillment of the Requirements**

**for the Master Degree in Engineering/Science （工学/理学硕士）/the Professional Master Degree (专业学位)**

**英文题目，Times New Roman，小二号，  
实词的首字母大写**

**Candidate : xxx（中文习惯，姓在前且姓全部大写）**

**Major : \*\*\***

**Supervisor : Prof. xxx**

**Huazhong University of Science and Technology**

**Wuhan 430074, P. R. China**

**May, 2021**

**独创性声明**

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期： 年 月 日

**学位论文版权使用授权书**

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保 密□，在 年解密后适用本授权书。

本论文属于

不保密□。

（请在以上方框内打“√”）

学位论文作者签名： 指导教师签名：

日期： 年 月 日 日期： 年 月 日

# 摘 要

目标跟踪是传统计算机视觉的一个重要研究分支，在航天、制导、监控等领域有广泛应用，具有较高研究价值。近年来随着深度学习技术的发展，越来越多基于深度学习的计算机视觉技术被提出并应用。目前已有大量基于深度学习的视觉跟踪技术研究与应用，其中多行人跟踪课题在自动驾驶、区域监控等领域有重要研究价值。当前大量行人跟踪技术在人员密集、复杂等场景跟踪效果较差，具有较大改进空间。

本文以目前有具有较高应用价值的多目标跟踪算法FairMOT[1]为对照组，分析算法在行人密集场景下存在：镜头晃动、检测丢失、身份互换等问题，并提出：运动补偿、轨迹插帧、丢失再匹配等算法优化跟踪效果。实验证明本文提出的改进算法能有效提升算法跟踪精度，极大缓解身份互换问题，在多个数据集上指标达到优秀水平。同时改进措施计算成本较低，相比于其他跟踪算法帧率较高，具有较高实用价值。

本文主要贡献如下：

1. 分析单镜头场景下，镜头运动对算法跟踪效果存在的影响，并提出一种基于镜头运动补偿的非线性插帧算法优化运动影响。
2. 分析在行人密集场景下，算法存在的行人身份互换问题，并提出一种新的特征匹配策略优化身份互换问题。
3. 分析算法对丢失目标再匹配问题，提出一种轨迹再匹配机制，优化算法对跟踪丢失目标的跟踪精度。
4. 实验证明上述改进措施均能有效缓解对应问题，相较于其他算法具有显著优势，具有较高应用价值。

**关键词：** 多目标跟踪；镜头运动补偿；非线性插帧；轨迹再匹配

# Abstract

Target tracking is an important research branch of traditional computer vision. It has high research value. In recent years, with the development of deep learning technology, a large number of deep learning-based visual tracking technologies have been researched and applied, among which the multi-pedestrian tracking topic has important research value. At present, a large number of pedestrian tracking technologies have poor tracking effect in crowded and complex scenes, and have great room for improvement.

This paper takes the current multi-target tracking algorithm FairMOT[1] with high application value as the control group, and analyzes the problems of the algorithm in dense pedestrian scenes: lens shaking, detection loss, identity exchange, etc., and proposes: motion compensation, trajectory frame interpolation, Algorithms such as loss and rematch optimize the tracking effect. Experiments show that the improved algorithm proposed in this paper can effectively improve the tracking accuracy of the algorithm, significantly alleviate the problem of identity exchange, and achieve excellent indicators on multiple data sets. At the same time, the calculation cost of the improvement measures is lower, and the frame rate is higher than other tracking algorithms, which has higher practical value.

The main contributions of this paper are as follows:

1. A nonlinear frame interpolation algorithm based on lens motion compensation is proposed to optimize the motion influence.
2. Analyze the pedestrian identity exchange problem, and propose a new feature matching strategy to optimize the identity exchange problem.
3. A trajectory re-matching mechanism is proposed to optimize the tracking accuracy of the algorithm for tracking the lost target.
4. Experiments show that the above improvement measures can effectively alleviate the corresponding problems, and have high application value.

**Keywords:** Multiple object track; motion compensation; unlinear frame insertion; trajectory rematching;

目录编写要求：

1. 目录自动生成时，只需列出二级标题；
2. 中文宋体，英文和数字Times New Roman，四号，行距：固定值25磅；
3. 注意避免出现不具辨识度的标题：如材料与方法、实验结果、分析与讨论、结论，若属于这种类型的论文时，可以把结果部分拆开，根据结果拟定题目。

# 绪论

本章首先介绍多目标跟踪算法的研究背景和价值，并对基于视觉的多目标跟踪任务进行定义。然后阐述目前国内外课题研究现状并分析其优缺点，最后引出本文主要工作内容和文章组织结构。

## 研究背景与意义

计算机视觉技术在航空、制造、监控、通信等诸多领域发挥重要作用，计算机视觉技术通常可分为三个阶段：数据采集、数据处理、数据输出，其中数据处理阶段是当前研究热点。计算机视觉领域包含大量任务，其中目标跟踪任务是其中一个重问题。当前目标跟踪技术广泛用于：体育转播、安防监控、自动驾驶等诸多领域，其中安全防控和自动驾驶领域，急需突破目标跟踪任务上的瓶颈。

近年来各大芯片厂商不断推出算力更强的芯片，于此同时互联网的发展使用户产生了PB级的海量数据，这两股推力推动当下深度学习技术的蓬勃发展。其中深度学习技术在计算机视觉中的应用尤为引人注目，目前基于深度学习的计算机视觉技术在大量领域打败传统视觉技术，包括但不限于：图像分类、人脸识别、目标跟踪等诸多领域。深度学习的成功主要源于：神经网络的强大拟合能力和大量标注样本。

综上所述本文研究内容为基于深度学习的计算机视觉多目标跟踪技术（Multiple Object Track），该项技术对于自动驾驶、区域安防领域具有重要价值。目前业内普遍对任务定义为：通过输入一组由单摄像头拍摄的连续视频帧，对每一帧图片标注目标位置并输出身份信息，同一目标在不同视频帧具有相同身份信息。目前该任务存在以下难点：

1. 形态变化问题：这个问题在体育竞赛转播中较为常见，运动物体发生姿态变化很容易导致外观模型发生改变，导致跟踪失败。除此之外在低帧率环境下，外观模型也较难提取高速变化目标的有效特征。
2. 镜头运动问题：由于目标曝光时间变短，镜头运动会导致拍摄画面模糊，目标形态改变等问题。当前大多数目标跟踪算法会对短期丢失目标进行轨迹预测，通常选用线性运动模型，因此镜头运动会导致目标运动线性度变差，对丢失目标再匹配精度下降。
3. 场景遮挡问题：在商场、学校等人员密集场景下广泛存在该问题，该场景存在大量人与人、人与物之间的遮挡问题。人与人之间的遮挡会导致不同的行人具有相似的外观特征信息，人与物的遮挡会导致行人位置检测失败。因此在这样的场景下会存在大量检测丢失，身份信息互换问题，这也是多目标跟踪算法最具挑战性的场景。
4. 图像噪声：由于视频拍摄存在噪声，噪声会影响到目标检测器的性能，尤其是小目标的跟踪对噪声的影响十分敏感，往往小目标的检测失败同时会导致身份信息的错误提取等一系列问题。
5. 丢失目标再匹配问题：目标大多数算法对于丢失目标设置固定时间窗口，一旦丢失时间超过该阈值就直接丢弃。当目标再次出现时，给目标分配新的身份信息。并且长时间的跟踪丢失，目标可能发生较大形态变化，因此单纯增大时间窗口并不能有效缓解这一问题。

因此对以上问题分析并提出相应算法缓解问题对于该项技术的应用是十分必要的，且具有很高的实用价值。本文对于上述1、2、4问题提出运动补偿和非线性插帧算法缓解，针对3问题提出一种新的特征匹配机制显著较低身份互换问题，对于5问题提出一种基于对比神经网络的轨迹再匹配机制。本文研究背景基于安全防控和自动驾驶领域，提出有效改进措施的缓解上述问题，对于这项技术的实际落地应用具有较大意义。

## 国内外研究现状

目前MOT（Multiple Object Track）技术在国内外有广泛研究，以使用技术划分可分为：传统MOT技术和基于深度学习的MOT技术。传统MOT技术定义为：不使用任何基于深度学习技术的跟踪算法，这些算法大都使用DPM[13]、HOG[18]+SVM[16]、Haar[15] +Adaboost[14]等算法进行目标检测，这些检测算法相较于目前主流的基于神经网络的目标检测器精度较差，且很难使用GPU进行加速运算，因此跟踪效果较差，本文不讨论传统跟踪算法。

当前基于深度学习的MOT技术可分为两大类：在线跟踪算法和离线跟踪算法。

定义在线跟踪算法：当前视频帧的目标身份与位置信息，只能由过去或未来固定时长若干帧生成。定义离线跟踪算法：当前帧目标信息可通过过去及未来所有视频帧生成。因此在线跟踪算法实时性较好，可使用的场景更多。离线跟踪算法精度更高，鲁棒性更强。

其中在线跟踪算法可以根据范式划分为：Track by detection 和 Track by SOT(Single Object Track)。TBD(Track by detection) 跟踪范式为：以时间顺序处理视频帧，对于新的一帧首先进行目标检测。对于每个检测到的目标提取有效特征，再将之前帧跟踪的目标和这些新检测的目标进行匹配，完成跟踪任务。TBS（Track by SOT）跟踪范式为：对每个跟踪目标以单目标跟踪形式进行处理，对于新的一帧先进行目标检测，并对每个跟踪目标预测在这一帧位置，最后对两者进行匹配。TBD范式使用较为广泛因为TBD范式能更好的对丢失目标进行再匹配，但TBS范式能更好的规避身份互换问题。

离线跟踪算法大多数采用以下过程完成任务：先对每一视频帧进行目标检测生成位置信息，根据外观或运动特征匹配相邻帧之间的目标。这样就能生成以时间t轴、图像x轴、图像y轴为空间的三维管道，然后对这些三维管道进行时域信息提取，并于之前特征进行融合。最终以通过混合特征匹配三维管道完成跟踪任务。

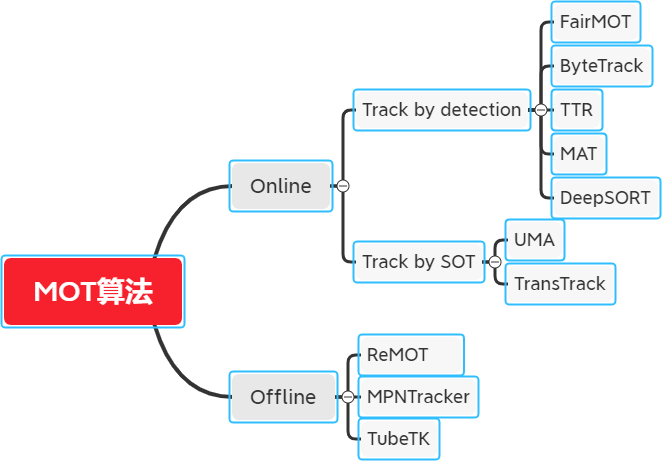


图1.1 MOT算法分类图

### 在线跟踪算法

#### TBD范式

SORT[17]可谓是TBD范式的开篇之作，论文首先使用了TBD跟踪范式之大量工作基于这种方式进行。SORT使用IOU(Intersection-Over-Union)特征对detection和track进行匹配，同时算法使用Kalman Filter[5]对跟踪目标位置进行预测，并使用匈牙利算法（Hungarian Algorithm）对IOU特征矩阵进行匹配。由于IOU特征计算简单，因此在不考虑目标检测器的计算成本下，跟踪算法能保持很高的帧率，往往能达到100fps。因此算法对于大多数人员稀疏的简单环境有较好的跟踪效果，对于存在大量跟踪目标丢失并再现场景效果较差。

Wojke N[2]等人于2017年在SORT算法基础上进行改进并提出了DeepSORT，主要改进有：使用卷积神经网络提取目标外观特征、使用马氏距离代替IOU特征描述目标间的运动相似度、提出级联匹配对不同跟踪长度的轨迹分别处理。DeepSORT相较于SORT有显著提升，主要原因可以归功于卷积神经网络提取外观特征对于长时间丢失目标再匹配精度更高。并且级联匹配算法有助于避免不同目标之间外观特征的互相干扰。DeepSORT由于引入了外观特征提取，因此不可避免地导致了跟踪帧率的降低。由于对于每一帧算法首先需要使用检测器标注目标位置，然对每个目标区域使用卷积网络提取外观特征，因此运算速度随着目标数量增加会显著下降，并且这种外观特征受目标位置影响较大，在人员密集场景下跟踪效果较差。

Wang Z[3]等人针对上述问题提出了JDE算法，该算法首次将目标检测网络和外观特征提取网络进行融合，提出一种端到端的跟踪神经网络模型和对应的训练方法，相比于DeepSORT算法在人员密集场景下有更高跟踪帧率。除此之外JDE算法提出了一种多次特征匹配算法，相较于DeepSORT的级联匹配有更优的跟踪效果。JDE算法以YOLOv3[60]为基本框架，结合FPN[61]结构生成多尺度特征图，并对特征图上的每个特征向量进行目标检测和外观特征提取。虽然这种方法加快了目标跟踪速度，但是同时降低了跟踪精度。由于目标检测任务与身份信息提取任务有不兼容属性，目标检测任务在于区分目标和背景的差异，身份信息提取任务在于区分目标和目标之间的差异。JDE算法在IDF1[58]指标上表现明显较差相较于DeepSORT算法，且存在较多身份互换问题。

Zhang Y[1]等人提出的FairMOT算法使多目标跟踪算法的一次重要突破，该算法不仅跟踪帧率高而且精度优秀，当时在多个数据集都达到了SOTA（State Of The Art）水平。该算法的主要贡献为提出了一种继承目标检测和外观特征提取的模型，首次使用Track By Point范式，相较于其他目标检测器避免了先验Anchor在人员密集环境下对目标身份的错误提取。FairMOT算法以CenterNetv1[55]检测器为基本框架，以JDE算法作为基本匹配策略。模型在DLA34[54]骨干网络输出的特征图后添加目标检测和身份信息提取分支，实验证明相比于JDE和其他基于Anchor的目标检测算法，具有更好的鲁棒性。并且该算法在速度和精度上都大幅超过两部式跟踪算法如：DeepSORT等。

接下来在TBD范式的工作大部分探索都是基于以上工作进行，或者在以上优化以上工作中存在的问题。其中这些工作大都围绕：目标检测和特征提取的冲突问题、复杂环境下身份信息地有效提取、检测与目标匹配等问题展开。

Zhang Y[7]等人提出的ByteTrack算法是目前精度最高的跟踪算法，ByteTrack算法在之前的JDE跟踪策略上进一步优化，提出将目标检测器的低分目标的再匹配策略，这种策略可以有效避免人与人遮挡时发生的目标被固定阈值过滤掉的问题，这种策略在多个跟踪方法都被证明有效。ByteTrack使用YoloX[62]目标检测器，在多个数据集上均达到SOTA水平且跟踪速度平均能达到25fps。

Han S[25]等人首先镜头运动问题的解决方案，他们提出的MAT跟踪算法，对于相机运动场景文章使用ECC[20]算法对前后两帧计算映射矩阵，并将映射矩阵作用在卡尔曼滤波算法上，使算法对丢失目标的轨迹预测更精准。除此之外，算法提出了动态插帧窗口运算公式，该公式根据窗口运动程度和目标运动速度计算插帧窗口大小，实验证明了公式的有效性。

Wang Q[9]等人提出CorrTrack算法，算法主要优化特征提取分支，相较于FairMOT直接的从特征图上生成身份信息特征。CorrTrack算法首先在特征图上计算不同特征点与当前位置的关联系数，然后通过多层前馈网络生成不同特征点的权值矩阵，将特征点特征进行加权平均。此外算法还对时域上的特征图进行同样处理，意在得到对相同目标更有效的外观特征信息。实验证明相较于FairMOT该算法具有MOTA[58]、IDF1指标均有显著提升。

除此以外还有大量在TBD范式下的MOT算法尝试。目前有大量尝试使用图神经网络在MOT算法上的工作[19][33][39][23][34][32][12][24]，这些工作大都集中在优化复杂环境下相邻两帧目标匹配问题，这些工作通常使用运动特征信息建立detection和track之间的全连通图，根据全连通图对外观信息进行进一步优化。除此之外CSTracker[30]、RelationTrack[51]等工作使用注意力机制，优化网络目标检测能力和身份特征提取能力的冲突问题。

#### TBS范式

目前也有大量工作将多目标跟踪问题转换至同时跟踪多个单目标任务上，这种跟踪方式对于目标与目标之间互相部份遮挡时有较好的跟踪效果，但对于长时间丢失目标再匹配效果较差。

Yin J[28]等人提出的UMA是一个端到端的训练模型，将目标运动和关联模型融合在一起。除此之外模型引入注意力机制，提高目标身份特征提取效果。算法的主要创新点是引入了孪生网络的概念，孪生网络主要由主干网络和ROI分支构成。孪生网络仅仅在网络训练过程时使用，UMA推理过程中仅使用主干网络和ROI分支。算法流程可概括为三步：通过检测器检测当前帧所有目标detection、对已有的track进行SOT 模板匹配，将模板匹配结果和detection进行匹配，如果匹配失败则判定为目标丢失、对于未匹配的detection判定为新的跟踪目标，对于未匹配的track判定为丢失或遮挡。同时UMA创新性的引入了目标丢失和被其他目标遮挡的判定模块，当目标丢失时仅使用卡尔曼滤波器预测其位置，对于目标被其他目标遮挡时则取其他目标的平均位置作为输出。

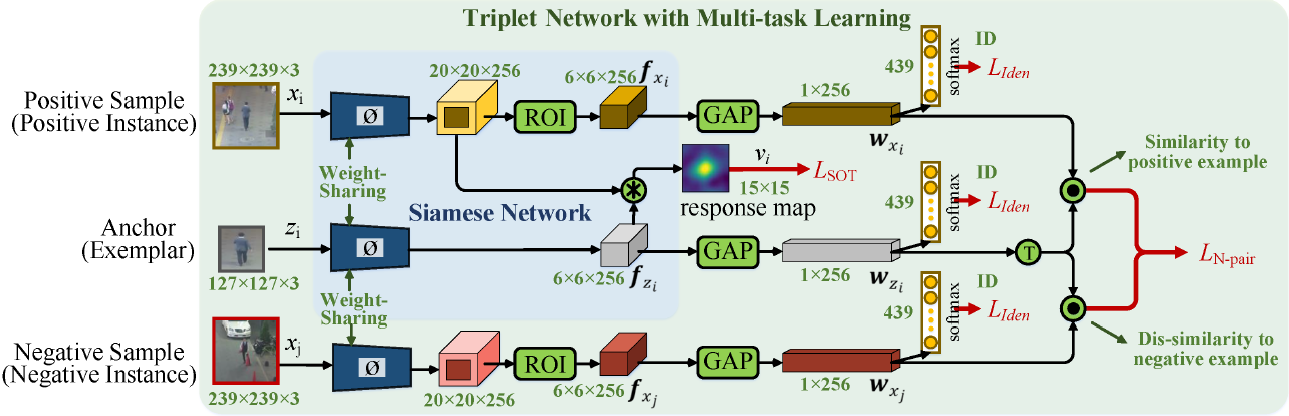


图1.2 UMA训练孪生网络图

Sun P[38]等人提出的TransTrack是首次将Transformer[21]模型引入MOT领域，TransTrack思想较为朴素，将SOT的模板匹配过程转换成Transformer的编码解码和查询过程。相较于其他算法TransTrack的目标检测过程也是基于Transformer模型的，而不是主流的主干网络加分支结构。这样的作法创新型较强但是实际效果并不理想，可能原因是基于Transformer的目标检测器检测精度较差导致。

其中此处应就与本文相关的国内外研究概况进行全面综述，这样相关内容在后面章节中就可以点到为止，无需再大段大段地分别介绍了。

### 离线跟踪算法

离线跟踪算法使用场景相对于在线跟踪较少但精度更高，主要原因当前帧目标位置身份信息可由过去和未来的视频帧共同推断。其中具有代表性的工作有：ReMOT[10]、TubeTK[6]等。

Pang B[6]等人提出TubeTK离线MOT算法，这个方法首先将所有视频帧以时间轴顺序排列形成一个有时间t轴、视频帧x轴、视频帧y轴构成的三维立方体，然后在这个三维立方体上进行三维目标检测。这样在三维立方体上就能形成多个三维管道，每个管道代表一个目标在一组连续的视频帧上位置。然后算法会对三维管道进行三维IOU特征匹配，最后输出跟踪轨迹图。

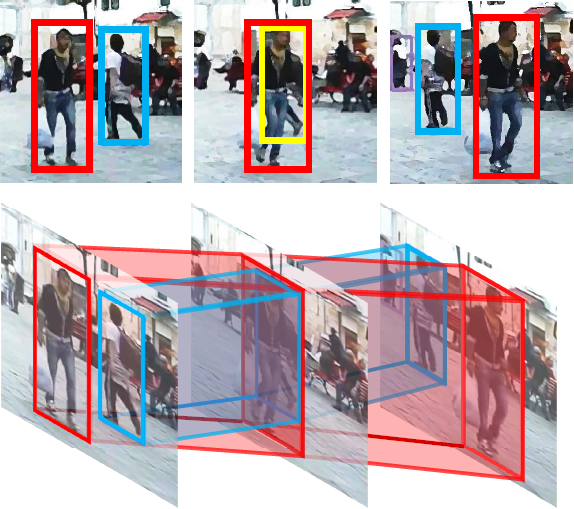


图1.3 TubeTK跟踪原理图

Yang F[10]等人提出ReMOT算法，算法将错误的跟踪轨迹分为两类：Mix-up错误（不同的目标分配至同一个轨迹），Cut-off错误（同一目标轨迹不连续），针对这两个问题文章提出一种新型的轨迹特征提取、轨迹特征比较和轨迹匹配策略。轨迹特征提取策略将目标外观特征和运动特征巧妙地结合在一起，特征对比策略用于比较两个轨迹是同一个目标的相似程度，轨迹融合策略则使用聚类算法将相同目标轨迹组成新的轨迹。由于ReMOT算法对于轨迹特征提取以及轨迹匹配的巧妙设计，使得ReMOT有很高的跟踪精度，并且具有极少的身份信息互换问题出现，但由于算法存在大量重复计算导致实时性很低。

## 存在的问题

目前国内外MOT相关技术普遍存在以下问题：1、过分依赖目标检测器，检测器的性能直接决定跟踪精度的高低2、在人员密集环境下存在大量身份互换问题3、目前基于外观的身份特征提取对长时间丢失的目标再匹配难度较大。问题1是目前所有MOT算法的通病，因为需要通过视频帧生成目标对应位置信息，所以必须使用目标检测器从图片中标定目标位置。问题2也是所有MOT需要面临的挑战，原因如下：首先检测器对于人员密集场景会存在丢失、漏检等问题，再者行人重合会导致难以提取目标外观特征，外观特征匹配失败是造成身份互换的重要原因。问题3是当前在线跟踪算法面临的重要困难，因为在线跟踪算法需要对detection和track进行匹配，然而detection只包含当前视频帧信息，且受检测器和目标形变影响较大，因此对于长时间丢失目标匹配成功率极低。

## 本文主要内容

本文主要研究内容如下：

1. 分析单镜头场景下，镜头运动对算法跟踪效果存在的影响，并提出一种基于镜头运动补偿的非线性插帧算法优化运动影响。
2. 分析在行人密集场景下，算法存在的行人身份互换问题，并提出一种新的特征匹配算法优化身份互换问题。
3. 分析算法对丢失目标再匹配问题，提出一种轨迹再匹配机制，优化算法对跟踪丢失目标的跟踪精度。
4. 实验证明上述改进措施均能有效缓解对应问题，相较于其他算法具有显著优势，具有较高应用价值。

本文段落结构如下：第二章 跟踪场景问题分析，着重分析目前MOT算法在各种复杂场景下出现的跟踪问题。第三章 跟踪问题缓解策略，针对以上问题提出相应缓解策略与算法。第四章 实验分析论文，通过实验证明改进策略的有效性。第五章 总结与展望，给出全文的主要内容及结论，总结本文的创新点，并对未来的工作进行展望。

# 场景问题分析

## 引言

本章首先介绍当下主流基于TBD范式的MOT算法跟踪流程，然后分析当前主流MOT算法在复杂场景下出现的跟踪问题。本章主要分为三节：第一节分析在镜头运动场景下MOT算法存在的问题、第二节分析在人员密集场景下MOT算法存在的问题、第三节分析MOT算法对于长时间丢失目标的再匹配问题。

### 基于TBD范式的MOT算法跟踪流程

当前国内外MOT算法研究主要以TBD范式为主，因为TBD范式相较于TBS范式有较好的跟踪精度，相较于离线跟踪算法有较好的应用价值。TBD跟踪范式可抽象为以下步骤：

1. 以时间顺序处理视频帧。
2. 对于新的一帧首先进行目标检测。
3. 对于每个检测到的目标提取有效特征。
4. 将之前帧跟踪的目标和这些新检测的目标进行特征匹配。
5. 对于匹配成功的跟踪目标更新其身份特征，并输出目标当前帧位置信息。
6. 对于匹配失败的跟踪目标将其标记为跟踪丢失目标。
7. 对于匹配失败的当前帧检测目标初始化为新的跟踪目标。

当前大多数基于TBD范式的MOT算法研究都以DeepSORT[2]或JDE[3]的跟踪策略为基本框架开展工作，因此本文使用JDE跟踪策略作为基本跟踪框架。接下来的小节将介绍JDE策略的基本实现流程。

### JDE跟踪策略

JDE将每个跟踪目标（track）划分为四种状态，分别为：Unconfirmed、Tracked、Losted、Removed。Unconfirmed对应于跟踪目标第一次初始化时状态，设置这种状态主要由于两个原因：1、如果目标只在一帧出现很有可能是检测器错误检测导致的2、如果目标只出现一帧则无法对其建立观测运动模型。当处于Unconfirmed状态的track在下一帧被成功匹配之后则进入Tracked状态，否则进入Removed状态。Tracked状态表示track正在被跟踪，当track匹配失败时进入Losted状态。Losted状态表示当前目标由于某些原因跟踪丢失，如果track与当前detection匹配则重回Tracked状态，如果处于Losted的track长期没有和detection匹配则进入Removed状态。处于Removed状态的track表示已被跟踪器移除。跟踪目标状态转移图如下所示：



图2-1 JDE算法track状态转移图

在此基础上JDE提出外观特征和目标运动特征结合的三次匹配跟踪策略，JDE算法对每一帧视频的跟踪策略步骤如下：

1. 对当前视频帧进行目标检测和目标外观特征提取生成detections。
2. 使用Kalman Filter[5]预测所有处于Tracked以及Losted状态的tracks在当前帧的位置。
3. 根据外观特征对所有处于Tracked以及Losted状态的tracks和detections生成代价矩阵，对矩阵进行二分图匹配。删除detections中匹配成功项，并更新匹配成功的track状态。
4. 对于处于Tracked状态但在step3中未匹配成功的track使用运动特征生成代价矩阵，并进行进二次匹配。
5. 将剩余detections通过运动特征与Unconfirmed状态track进行三次匹配。删除detections中匹配成功项，并更新匹配成功的track状态。
6. 将剩余detections初始化为Unconfirmed状态track。

## 镜头运动场景问题分析：

当前MOT算法在镜头运动场景下表现较差根本原因为：当拍摄镜头存在运动时，同一物体曝光时间短并且在相机感光层成像面积大，会导致跟踪目标的拍摄模糊。因此跟踪模糊画面的目标存在三个难点：1、画面模糊目标检测器难以检测目标在视频帧中具体位置2、跟踪目标运动模型线性度较低，使用Kalman Filter等线性运动模型预测track在当前帧位置偏差更大3、跟踪目标姿态变化更大更难提取有效外观特征信息。接下来的三小节分别仔细阐述以上三个问题。

### 镜头运动对目标检测器的影响

当前主流视觉目标检测技术以深度学习技术为主，其中比较具有代表性的有Yolo[60][62]、CenterNet[55]、EfficientNet[63]等，这些方法通常使用骨干网络（Backbone）+特征金字塔网络（FPN）提取原始图片特征，在输出特征图上使用回归、分类等分支结构输出目标检测信息。相较于传统目标检测算法，基于深度学习的算法精度更高，但同时受图像噪声影响较大。

以MOT17[56]数据集为例，MOT17-05、MOT17-10、MOT17-11、MOT17-13等数据子集存在剧烈镜头运动问题，下图展示MOT17数据集子集中的镜头运动导致的画面模糊情况。



图2-2 镜头运动效果图

从上图不难发现镜头运动会导致画面模糊，并且这种影响对小目标和运动速度较快运动目标影响较大。本文在MOT17测试集上测试了多组不同检测器检测效果，并得出了以下表格描述不同检测器对MOT17数据集目标检测能力，表使用DetA[59]指标衡量检测器精度。

表2-1 MOT17数据集目标检测性能

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 目标检测器 | MOT17-03 | MOT17-06 | MOT17-12 | MOT17-14 |
| FairMOT[1] | 73.4 | 53.4 | 47.8 | 41.5 |
| CSTrack[30] | 72.2 | 51.2 | 49.8 | 40.3 |
| CorrTrack[9] | 74.2 | 54.7 | 49.1 | 45.5 |
| ByteTrack[7] | 73.4 | 52.8 | 58.0 | 45.0 |

其中MOT17-03子数据集不存在镜头运动效果，其他三组子数据集存在不同程度的镜头运动问题。其中MOT17-14子集镜头运动最剧烈，且拍摄图片效果相较于其余两组最差。MOT17-06和MOT17-12子数据集也存在镜头运动问题，但MOT17-14中存在大量待检测小目标，因此效果最差。通过表数据可得到以下结论：镜头运动问题会显著造成目标检测能力下降，这种影响对于存在大量待检测小目标时更明显。

下图统计的是FairMOT在MOT17不同子集上相邻视频帧目标检测丢失比例：

图2-3 相邻视频帧目标检测丢失比例

柱状图统计数字含义为track跟踪丢失后5帧内再匹配数量除以所有匹配数量得到的商，这样做的目的在于track丢失5帧内在匹配一般由于检测丢失造成而不是其他原因。其中MOT17-10和MOT17-13子数据集相机运动程度最剧烈，柱状图可得到与上述表格相同结论。

### 镜头运动对运动模型影响

镜头运动对检测效果另一个重要影响为对目标运动模型的影响，目前大多数MOT算法基于线性运动模型预测track当前帧位置。这种预测方式计算成本低，在高帧率目标低运动速度时效果较好，对于低帧率高速运动目标跟踪性能较差，镜头运动则会导致目标运动线性度下降。这一影响对短距离预测影响较小，对长距离位置预测影响较大。下图展示不同环境下相邻两帧目标预测与实际位置偏差：



图2-4 目标位置真实与预测差异图

上图中橙色方框表示目标真实位置，蓝色框表示线性运动模型预测位置，橙色箭头表示运动模型观测目标运动方向。从上图可看出在镜头运动场景下运动模型观测误差较大。下图反应卡尔曼滤波器在不同场景下对同一目标的相邻视频帧的预测偏差，下图使用IOU（Intersection-Over-Union）距离预测与实际误差。

图2-5 相邻视频帧目标实际与预测位置距离

镜头运动对于长期处于丢失状态下的track的位置预测差异远大于短期预测，原因归结于目标长时间运动线性度远低于短时间线性度。下图反应丢失状态下的track位置预测误差与丢失时间关系。

图2-6 预测误差和时间关系图

从上图数据可以看出相对于固定镜头场景（MOT17-02、MOT17-04、MOT17-09），运动镜头场景下（MOT17-05、MOT17-10、MOT17-11、MOT17-13）目标预测位置差异随时间抖动较为严重，这恰恰反应了的在运动场景运动模型预测的不稳定性，这种不稳定性导致在对长时间丢失目标再匹配时难度较大。

### 镜头运动对外观特征的影响

当前主流MOT算法使用深度神经网络提取目标外观特征信息，对同一目标在不同镜头下提取相似特征任务业内成为Reid（Re-Identification），当前大部分方法通常使用卷积神经网络生成目标特征图，使用先验信息对特征图的不同区域进行不同处理策略，最后生成目标特征向量。目标表现较好的算法有：[35][36][37]，这些工作大部分基于ResNet[11]，因此拍摄图片质量严重影响目标特征提取。下图展示镜头运动对目标外观特征提取的影响，图中可看出运动导致相同目标特征差异大于不同目标特征差异。

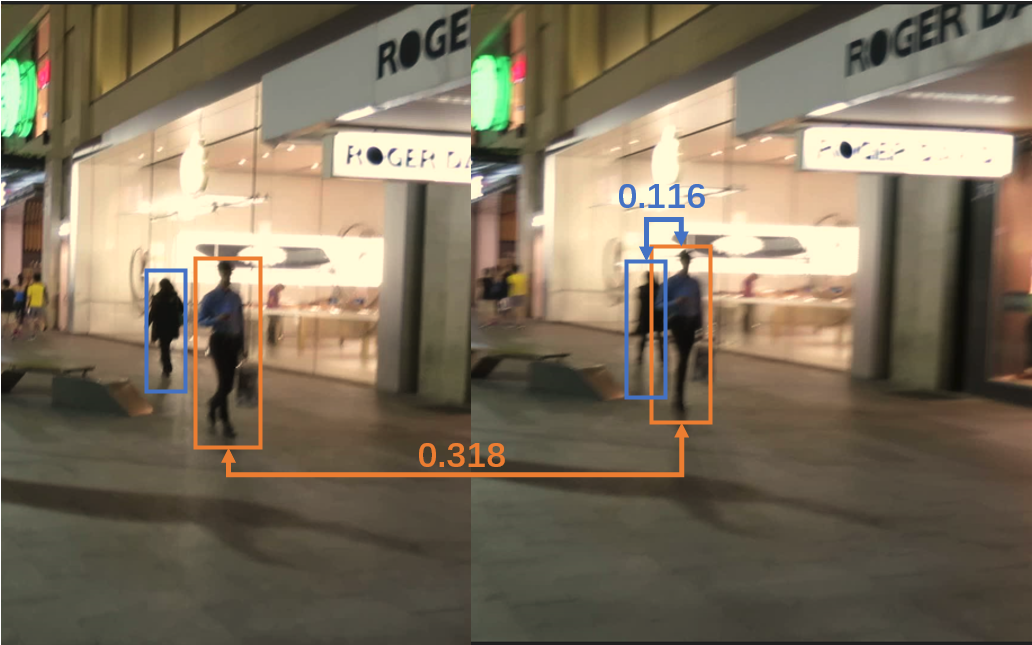


图2-7 运动场景下目标外观特征差异

下图对比FairMOT在不同数据集上特征提取对比统计直方图，本文使用FairMOT算法提取目标外观特征，并使用特征余弦距离衡量特征差异：

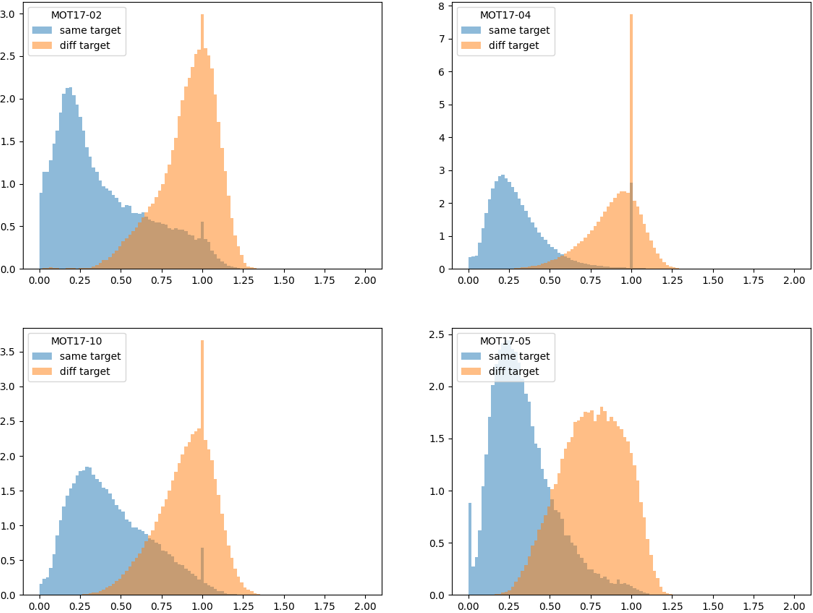


图2-8 目标外观特征统计直方图

上图x轴为外观特征差异，y轴为对应概率密度。从上图可以明显看出，在镜头运动场景（子图3、子图4）目标外观特征有着更大的类内距离和更小的类间距离。因此相比于静态场景，镜头运动场景提取外观特征更不可靠。由此得出以下结论：在镜头运动场景下，目标外观特征与目标运动特征均受到较大影响，因此优化镜头运动场景下的跟踪效果是MOT算法一大重点与难点。

## 密集场景场景问题分析

本节从两个方面分析人员密集场景对MOT算法影响，分别为：密集场景对目标检测器性能影响和密集场景对外观特征提取效果影响。密集场景存在大量人员遮挡问题，较多被遮挡目标被检测器以低分检测框给出，固定阈值的MOT算法通常将这些低分检测框丢弃，因此造成目标跟踪轨迹不连续。除此以外人员密集场景下可能导致目标外观提取问题，因为目标存在大量遮挡、重合、交叉等问题。被遮挡目标往往无法提取完整外观特征信息，由此可能导致发生目标遮挡时出现身份信息互换问题，这种影响对于DeepSORT[2]这种两步式跟踪算法影响较大。

本文定义RO（Ratio Occlusion）指标描述场景下目标遮挡程度，指标计算过程如下：

 (2.1)

其中I（x）函数表示当x为真返回1，否则返回0，其中F表示视频总帧数，n表示每一帧跟踪目标数。RO指标含义为遮挡与被遮挡目标数在所有跟踪目标下占比，其中thres为重合阈值。下图反应MOT17和MOT20子数据集下不同重合阈值RO指标。

图2-9 MOT数据集RO指标

上图可展示了不同数据集下RO指标，可发现MOT20数据集目标重合更严重，实验同样表明在MOT20数据集MOT算法表现更差。

### 密集场景下检测器性能

密集场景对目标检测器性能影响最大，对于被部份遮挡的物体，检测器一般给出较低置信度的检测框，这些检测框容易被固定阈值的MOT算法过滤。由于检测丢失往往导致对被遮挡目标跟踪精度显著下降。下图展示检测器对MOT20数据集中被遮挡目标检测效果图。

图2-10 目标可视化系数与被检测概率关系

从上图可以明显发现随着遮挡程度的上升，目标被检测概率显著提高。因此可得出以下结论：人员密集场景下检测器性能受目标可视化系数影响较大。当前主流的MOT算法受检测器性能影响较为严重，因此可以得出结论：相对于人员稀疏场景，密集场景下MOT算法跟踪性能较差。

### 密集场景下特征提取性能

由于密集场景存在大量人员重叠、遮挡、较差现象，并且目标检测器生成的矩形检测框无法精准标定目标可视化区域，因此这种情况通常导致被遮挡的目标难以提取有效外观特征：1、容易和遮挡其目标具有相似特征2、无法提取自身被遮挡部份有效特征，下图展示了MOT数据集中常见遮挡效果图。



图2-11 密集场景人员遮挡效果

上图中蓝色框标识被遮挡目标位置，橘色框表示遮挡其目标。观察上图易得出结论：在密集场景下存在大量目标遮挡现象，遮挡现象将严重影响目标外观特征提取。这种现象对两步式MOT算法影响较为严重，原因在于两步式MOT算法外观提取器和目标检测器基于不同的骨干网络，特征提取器需要较为精确的检测框才能提取有效外观特征。两步式MOT算法特征提取器通常假设检测框从上到下依次对应行人头部、腹部、腿部，但是由于检测器性能问题，检测框可能不满足以上假设。但是单步式MOT算法通常没有如此假设，因此单步式算法对人员密集场景效果更好。

本文使用FairMOT算法模型提取的外观特征，对比不同遮挡程度下相同目标外观特征差异和不同目标之间外观特征差异，特征差异使用归一化后的余弦距离进行度量。特征差异统计图如下：

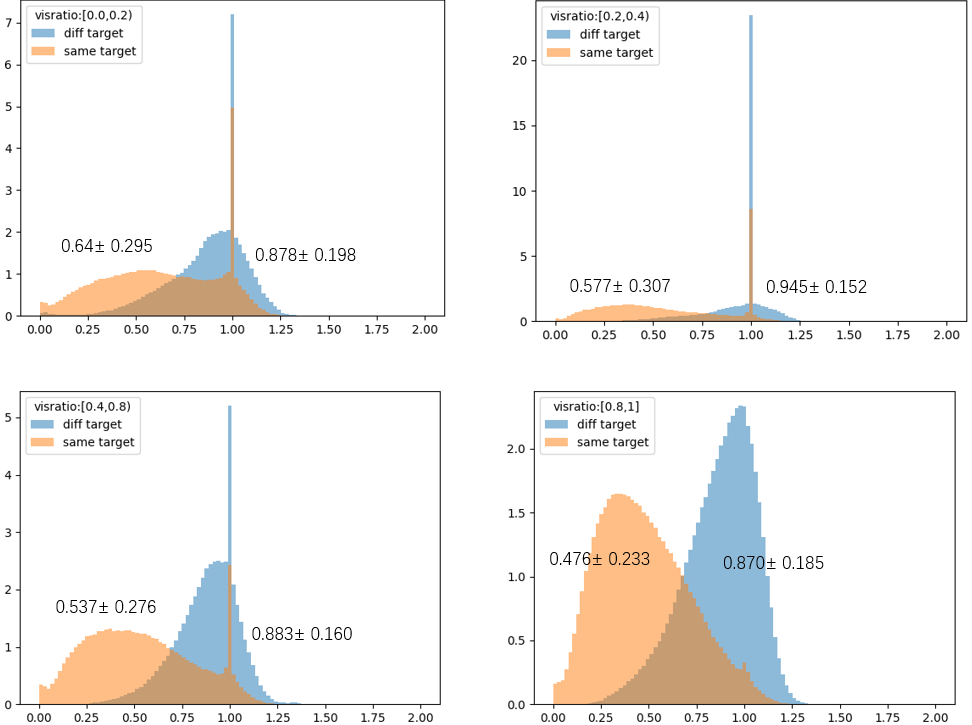


图2-12 密集场景可视化程度与特征差异统计图

上图可以清晰反应的反应对于被遮挡目标难以提取有效外观特征进行，当可视化程度低于0.5时可明显发现相同目标特征差异较大（相同目标间存在存在大量外观特征余弦距离大于等于1），当可视化程度超过0.5时这一现象有所改善。因此可得出以下结论：密集场景下目标间存在大量遮挡交互、遮挡、重合现象，外观特征提取器受目标可视化程度影响较大，目标可视化程度较低时难以提取目标有效外观特征。

## 丢失目标再匹配问题分析

MOT跟踪算法可能由于多种原因导致被跟踪目标丢失，当被跟踪目标丢失后可能会一段时间内重新被跟踪器跟踪。如果目标当前特征与跟踪器记录的最后一次特征匹配则丢失目标则会被正确关联，如果匹配失败则会给丢失目标分配新的ID或和其他已丢失目标ID，导致身份互换问题。并且如果MOT跟踪算法存在对丢失目标插帧轨迹补偿机制，丢失目标的错误匹配会导致大量错误的插帧补偿。下图展示了丢失目标再匹配失败出现的问题，图上半部分反应成功后插帧轨迹，下半部分反应和其他丢失目标ID互换并插帧导致问题。



图2-13 丢失目标再匹配失败展示图

丢失目标再匹配问题是MOT算法中另一大难点，尤其是对于长时间丢失目标。主要原因在于两点：1、长时间丢失目标外观存在较大改变，目标与丢失前外观特征差异较大2、长时间丢失目标运动线性度低难以有效预测目标位置，因此难以使用目标运动特征辅助外观特征优化跟踪精度。由于在线MOT算法需要在有限时间内输出目标位置信息，因此在线跟踪算法难以解决该问题。目前大多数跟踪算法对于丢失超过一定时间阈值目标直接从跟踪队列中丢失，因此当丢失目标重新出现时，可能会和其他跟踪目标错误关联或被分配新ID跟踪。

本文测试MOT17和MOT20数据集上目标特征差异与丢失时间关系，并绘制以下统计图，下图仍使用特征余弦距离描述特征差异：

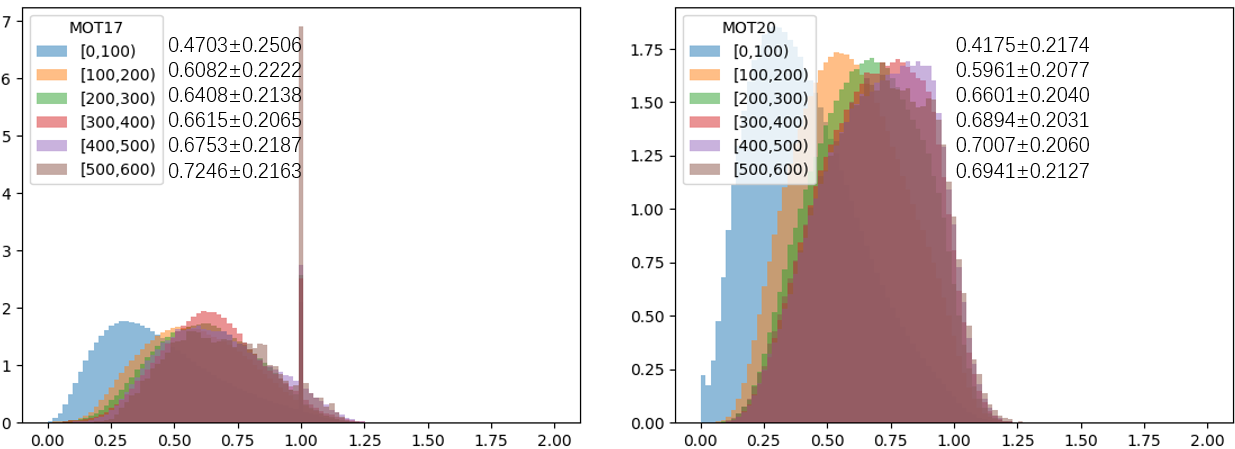


图2-14 丢失目标外观特征差异统计图

上图清晰展示随着目标丢失时间增加，目标外观特征差异逐渐变大，当丢失时间大于100帧后可发现，相同目标外观特征几乎不具备关联性（余弦距离在0左右，特征向量几乎互相正交）。由此可以得出结论：丢失目标外观特征差异随着丢失时间逐步增大，丢失时间越长差异越大。本章图2-6同样展示了丢失目标运动特征随丢失时间长度增加而逐渐变大，尤其在镜头运动场景下目标运动特征极不稳定。对于镜头运动虽然可以进行补偿缓解目标运动线性度降低问题，但丢失目标外观特征巨大变化仍难以处理，因此对于长时间丢失目标再匹配仍是目前大多数MOT算法的痛点。

## 本章小节

本章分析了以FairMOT算法为代表分析了当前主流MOT算法在三个场景下出现的问题。三个场景分别为：镜头运动场景、人员密集场景、目标长时间丢失场景，通过分析MOT算法中：目标检测器、外观特征提取器、运动观测模型、等模块在三个场景下的表现，可以得出三点结论：

1. 镜头运动导致画面模糊会导致目标检测性能下降、目标运动线性度变低、外观特征提取难度、等问题，这些问题进一步影响跟踪精度。
2. 人员密集场景下存在大量目标遮挡、重叠等现象，目标检测器对可视化程度低的目标难以检测，可视化程度低的目标难以提取有效外观特征。
3. 丢失目标特征差异随着丢失时间长度增加而变化，丢失超过100帧的目标外观之间几乎不具备关联性。

由此可知当前MOT算法在上述三个场景下性能都会受到不同程度的影响，本文接下来的章节将会对以上三个场景出现的问题，结合当下MOT算法研究提出相应策略优化算法性能。

# 算法场景优化

## 引言

本章将针对第二章描述的三个场景问题，提出相应优化策略。针对镜头运动场景提出镜头运动补偿和非线性插帧措施优化算法跟踪效果。针对人员密集场景提出一种新式目标特征匹配策略，克服目标之间身份互换问题。针对长时间丢失目标提出一种再匹配机制优化跟踪效果。本章将分为三节介绍针对不同场景下的MOT算法优化，分别为：1、镜头运动场景优化2、人员密集场景性能优化3、丢失目标再匹配性能优化，第四节将以上三种优化策略进行整合提出一种新型的MOT跟踪算法。

## 镜头运动场景优化方法

镜头运动场景会导致：目标检测器能力下降、目标运动观测模型线性度降低、难以提取有效外观特征。针对目标运动线性度降低的问题，本文结合传统计算机视觉领域图像对齐技术进行优化。针对目标检测器能力下降问题，提出一种新型的非线性插帧补偿机制优化检测效果。

### 图像对齐技术

图像对齐技术被广泛用于摄影、医疗、文本检测等领域，图像对齐技术目标为：对于相同目标拍摄图片，但因拍摄角度导致的目标图像畸变，使用算法将目标畸变图像转换为目标参考图像。图像对齐技术通常输出从畸变图像到参考图像的仿射变换矩阵，畸变图像上每一像素点坐标均能使用该矩阵映射至参考图像坐标。下图展示了图像对齐技术常见应用场景和效果，图左边部份是参考图像，图中间是畸变图像，图右部份是矫正后图像。

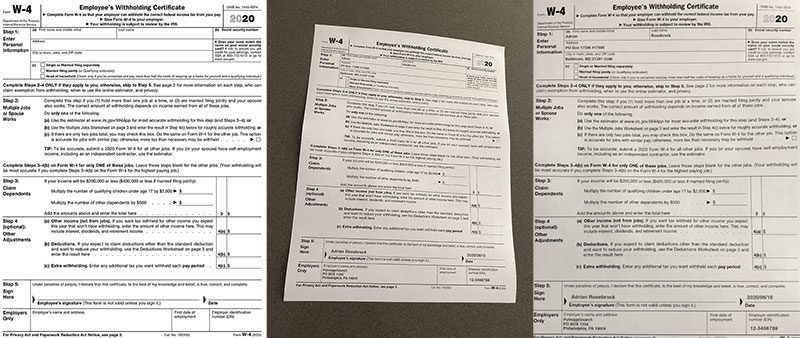


图3-1 图像对齐技术效果图

传统图像对齐技术一般首先提取两幅图像特征点，再对特征点进行KNN或贪心匹配得到匹配对。最后通过匹配对计算图像变换矩阵，一般特征点提取和变换矩阵计算成本较高，下图展示特征点提取与匹配效果

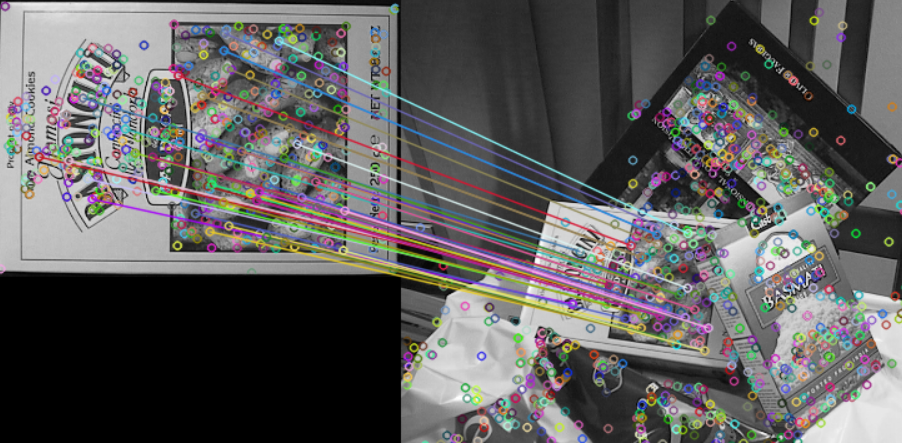


图3-2 特征点匹配效果图

常见的特征点提取算法有：ORB（Oriented FAST and Rotated BRIEF）[31]、SIFT[22]（Scale-invariant feature transform）、SURF[29]（Speeded Up Robust Features），这些算法生成特征点位置与特征向量通常分为三个步骤：

1. 使用高斯差分函数计算并搜索高斯金字塔上的的图像位置，提取出图像上对尺度和梯度方向不变的潜在点。
2. 通过一个非线性函数拟合潜在点上的位置、尺度和梯度信息，使用非线性函数拟合主要是为了计算特征点的稳定程度，过滤稳定度低的潜在点得到特征点位置信息。
3. 在特征点周围相邻像素提取梯度特征信息，对这些梯度信息进行加权归一化处理，使特征具有尺度、光照不变性。

算法生成这些特征点和对应特征向量后，通常使用KNN或贪心算法根据特征汉明距离或余弦距离进行匹配。目前图像匹配技术使用成本较高的算法计算图像变换矩阵，考虑到视频在较高较高帧率图像变化幅度不大，因此本文使用最小二乘法对置信度高的特征点计算图像仿射矩阵，计算公式如下：

 (3.1)

 (3.2)

 (3.3)

 (3.4)

 (3.5)

上述公式中是源特征点坐标的伴随矩阵，是目标特征点坐标的伴随矩阵。图像仿射矩阵，其中表示图像逆时针旋转弧度，是横坐标运动幅度，是纵坐标运动幅度，左上角四个值构成旋转矩阵。公式（3.4）表示通过仿射矩阵将源伴随矩阵转化成目标伴随矩阵计算过程，公式（3.5）表示最小二乘法求解过程。

另一类算法将图像对齐任务转换成优化问题，将仿射矩阵作为优化目标中的一个参数，因此能直接输出仿射变换矩阵，这类算法以ECC[20]（Enhanced Correlation Coefficient Maximization）为代表。ECC算法将图像对齐任务定义为对以下指标进行参数优化问题：

 (3.6)

 (3.7)

上述公式中表示目标图像，表示畸变图像，表示图像坐标，表示仿射矩阵参数，表示简单的图像亮度转换。ECC算法使用梯度下降法优化目标得到最优参数和，根据参数生成仿射矩阵。

由此可以联想到在镜头运动场景下，可以使用图像对齐技术缓解镜头运动带来目标运动观测模型线性度降低问题。具体做法如下：对于相邻两帧图像使用对齐技术计算仿射矩阵，将该仿射矩阵作用于运动观测模型上，使其对目标的位置预测上更加精准。本文使用卡尔曼滤波器作为目标运动观测模型，因此可得出以下计算公式：

 (3.8)

 (3.9)

 (3.10)

 (3.11)

 (3.12)

上述公式中表示卡尔曼滤波器观测状态值，其中表示目标跟踪矩形框左上角坐标值，表示矩形框宽度和高度，表示观测值速度，为仿射矩阵中的旋转矩阵。公式（3.10）、（3.11）、（3.12）表示仿射矩阵作用在不同观测值上的计算过程。本文使用对相邻两帧视频使用ORB特征点提取，并使用最小二乘法计算仿射矩阵，相较于MAT[25]中使用的ECC算法计算成本更低，实验证明本文使用图像对齐技术能对镜头运动场景下的目标运动非线性问题有较好的缓解。

### 非线性插帧

目前存在大量MOT[7][25][4]算法使用线性插帧机制对丢失目标进行轨迹补偿，插帧机制能有效缓解目标在短时间内丢失再匹配后轨迹不连续问题。通常被跟踪目标丢失并在短时间内重新匹配，这一现象大都是由于检测器短时间内检测失误导致的。因此在目标检测器检测能力受限的场景，插帧机制能有效提升算法跟踪精度。线性插帧公式计算过程如下：

 (3.13)

其中表示开始插帧时间，表示结束插帧时间，表示第帧被跟踪目标位置。这种线性插帧机制在镜头静止场景下表现较好，对于镜头运动场景或长距离插帧时工作较差。虽然MAT[25]将镜头运动补偿机制做用在线性插帧上，但是由于仿射矩阵运算不准确性，长距离运动补偿存在的累计误差，导致在运动场景下插帧的不准确性。下图展示线性插帧效果和运动补偿后的效果：

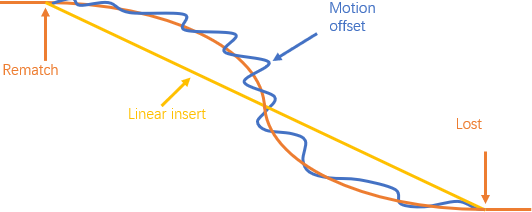


图3-3 线性插帧与运动补偿插帧效果图

上图可以清晰反应使用运动补偿插帧算法相对于线性插帧有较高的精度，但平滑性较差，存在一定的运算累计误差。本文提出一种新的运动补偿插帧算法并使用高斯过程对插入的轨迹进行平滑，实验表明相对于其他方法本文提出的非线性插帧方法具有更高精度。本文提出的运动补偿插帧算法计算过程如下：

 (3.14)

 (3.15)

 (3.16)

 (3.17)

上述公式中表示从第帧到第帧的仿射变换矩阵，表示从帧到第帧的仿射变换矩阵。表示在帧时刻的像素坐标系下时刻目标的相对位置，表示在帧时刻的像素坐标系下时刻目标的相对位置。公式（3.17）则将两个坐标系下的线性插帧映射至当前帧坐标系后位置，进行平均然后输出每一时刻的目标插帧位置。插帧结束后会使用高斯过程进行平滑处理。

由于在镜头运动场景下图像对齐技术计算的仿射变换矩阵具有一定误差，因此本文使用MAT[25]中提出的动态插帧窗口，根据镜头运动剧烈程度计算插帧数量阈值，超过阈值的场景只进行普通线性插帧。

## 人员密集场景优化机制

密集场景存在两个问题：1、密集场景下被遮挡目标难以被检测2、被遮挡目标难以提取有效外观特征，因此在密集场景下存在大量目标丢失、身份互换问题。目前有大量工作着重优化密集场景下跟踪器的跟踪性能，本文将通过以下三小节阐述本文对该场景下的优化措施。

### 目标检测能力优化

对于目标漏检问题最有效的措施为使用精度更高的目标检测器，但这样同时会造成计算成本增加，精度高的目标检测器往往花费更多计算时间。下图展示不同系列目标检测器检测精度与检测帧率关系，图片来源于EfficientDet[52]。

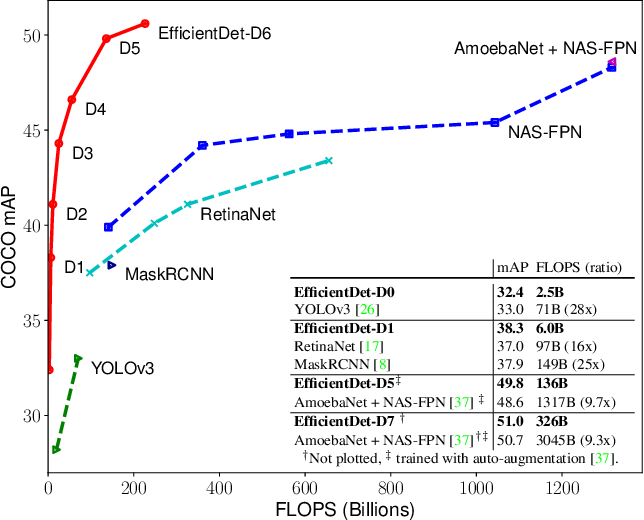


图3-4 检测精度与模型参数关系图

另一些工作尝试检测人体其他头部或对被遮挡位置进行多目标检测以提升被遮挡目标检测能力，这些目标检测算法往往计算代价较大。并且由于当下主流MOT算法都将目标检测器与特征提取器进行融合，因此对于单步式MOT算法直接更换目标检测器未必是一个明治之举，因此需要有一个较为通用策略提升MOT算法目标检测能力。ByteTrack[7]中提出一种检测补偿机制，即使用双阈值对检测器输出bbox（bounding box）进行过滤，具体步骤为：定义两个阈值T1、T2，其中T1>T2首先对于置信度高于T1的bbox使用与JDE跟踪策略相同的特征匹配机制，对于置信度大于T2小于T1的bbox最后与之前未匹配的目标通过运动特征进行匹配。实验证明这一改进在多个MOT算法中都被证明有效，因此本文也引入该算策略进行优化。下图展示了低分检测框二次匹配效果图，图片来源于ByteTrack[7]：

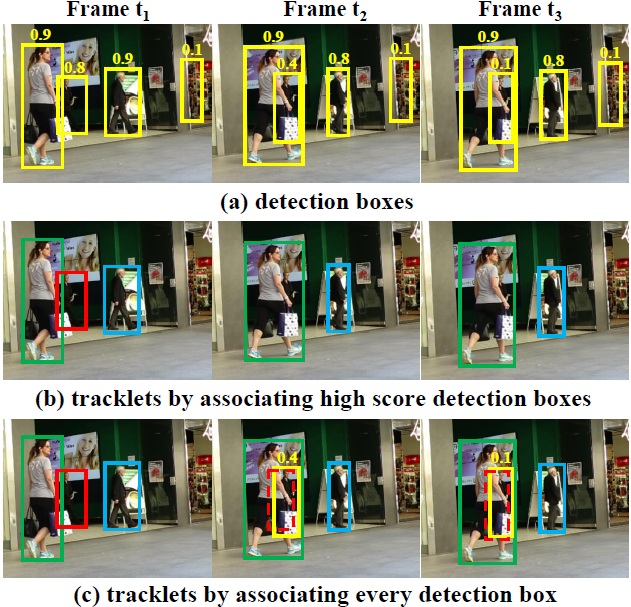


图3-5 低分检测框二次匹配效果图

### 身份信息互换问题优化

密集场景下发生人员互相遮挡交互现象，这样会导致同一个bbox内存在不同目标身体的两部分，因此不同间可能会存在相似的外观特征。因此在密集场景下完全依赖外观特征提取并不可靠，因此本文结合实践观察提出了一种基于遮挡关系的纠错机制，当目标间发生身份信息互换后能丢弃错误跟踪结果。

首先结合实践观察不难发现，如果被跟踪目标都在地面上运动，则离镜头越近的目标的检测框越靠近图片底边，下图展示了这种拍摄现象：

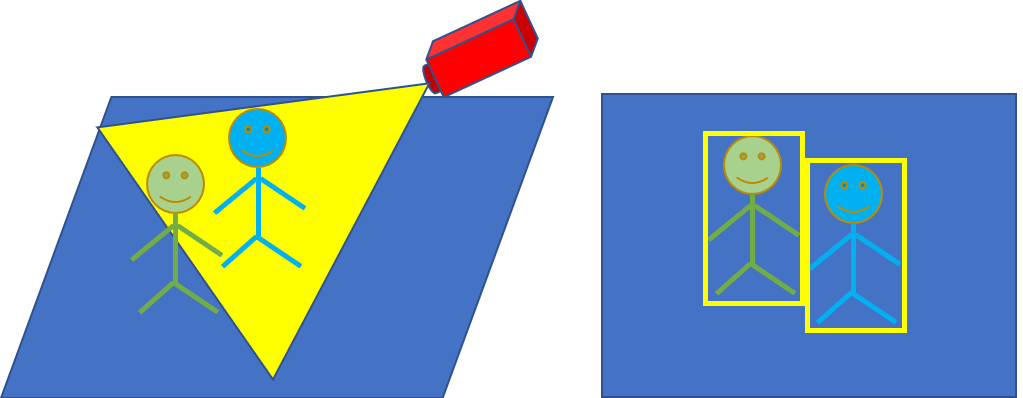


图3-6 地面运动目标拍摄现象展示

上图展示了这种拍摄现象，图左半部分展示拍摄场景，蓝色行人相较于绿色行人距离镜头更近。图右半部分为镜头拍摄画面图，可以看出相较于绿色行人，蓝色色行人检测框更靠近图片底边。这种现象的根本原因在于摄像头相对于地面永远是从高向低拍摄，因此地面上的目标离得越远则其像素坐标越靠近底边。由于这种现象的存在则对于存在遮挡现象的目标便可以很好的判断其遮挡关系，根据目标间的遮挡关系当目标出现身份互换时便可及时纠正错误匹配。

本文根据上述原理进一步处理，对遮挡与被遮挡目标生成其遮挡矩阵，通过遮挡矩阵的变化判定是否发生目标身份交换问题。具体做法是：

1. 对每一个bbox都生成一个W×H大小的全1矩阵，矩阵上的元素代表目标该位置处是否被遮挡（0表示被遮挡，1为目标自身）。
2. 将bbox底边坐标作为key从小到大排序，以从上到下的顺序遍历bbox。
3. 遍历过程中对检测框A来说，如果它和之前遍历过的检测框B存在相交关系，则说明A遮挡了B的一部分，将B遮挡矩阵对应位置元素清零即可。

这样就可以得到所有目标检测框在当前帧下的遮挡矩阵，对于任何detection和track之间的匹配，匹配前和匹配后跟踪目标间的遮挡关系不应发生显著变化，因为在帧率较高的场景下不可能出现这一情况：目标A在前一帧遮挡目标B大部分区域，但是在后一帧目标B又遮挡了目标A的大部分区域，因此一旦这种情况发生则说明detection和track间存在错误匹配。对于任何错误匹配，本文简单地将对应detection和track进行丢弃。本文使用以下计算判定目标是否发生错误匹配：

 (3.18)

代表匹配后目标间遮挡矩阵，表示匹配前目标遮挡矩阵，在0和1之间，表示遮挡矩阵元素个数。上述公式含义在于：匹配前后遮挡矩阵差异元素个数应当小于一定阈值数。下图展示该纠错机制效果，其中最左边部份表示匹配前目标位置和对应遮挡矩阵，图中间部份表示目标检测器检测框，图右边部份表示错误匹配后目标位置和对应遮挡矩阵。



图3-7 错误匹配遮挡矩阵效果展示

从上图可以看出当发生错误匹配时，由于遮挡关系发生变化所以遮挡矩阵在匹配前后有较大差异，可判定为匹配错误，实验表明本文纠错机制能有效缓解目标间身份互换问题。

### 匹配策略

基于以上匹配纠错机制，本文对JDE[3]跟踪策略进行调整提出一种新式的匹配策略。本文第二章引言中层介绍过JDE匹配策略，2.1.2小节内容展示该策略，该策略使用两次匹配法（step3中对处于Tracked和Losted状态的track进行一次匹配，step4对step3中未匹配成功的Tracked状态的track进行二次匹配）。本文在此基础上进行改进，结合目标检测框补偿机制提出一种四次匹配跟踪策略。匹配策略具体步骤如下：

1. 对当前视频帧进行目标检测和目标外观特征提取生成detections，将置信度高于T1的detections放入detection1s队列中，将置信度低于T1高于T2的detections放入detection2s队列中。
2. 使用运动观测模型预测所有处于Tracked以及Losted状态的tracks在当前帧的位置，如果使用图像对齐技术则利用仿射矩阵补偿tracks预测位置。
3. 根据外观特征对所有处于Tracked状态的tracks和detection1s生成代价矩阵，对矩阵进行基于遮挡图机制的二分图匹配。删除detection1s中匹配正确的detections，并更新正确匹配的tracks状态。
4. 对于处于Losted状态的track和detection1s通过外观特征生成代价矩阵，对矩阵进行基于遮挡图机制的二分图匹配。删除detection1s中匹配正确的detections，并更新正确匹配的tracks状态。
5. 对于处于Tracked状态但在step3中未匹配的track和detection1s通过运动特征生成代价矩阵，对矩阵进行基于遮挡图机制的二分图匹配。删除detection1s中匹配正确的detections，并更新正确匹配的tracks状态。
6. 对于处于Tracked状态但在step5中未匹配的track和detection2s通过运动特征生成代价矩阵，对矩阵进行基于遮挡图机制的二分图匹配，更新正确匹配的tracks状态。
7. 将剩余detection1s通过运动特征与Unconfirmed状态track进行匹配。删除detection1s中匹配成功项，并更新匹配成功的track状态。
8. 将剩余detection1s初始化为Unconfirmed状态track。

从上述匹配策略中可以看出本文匹配策略相对于JDE匹配策略主要做了三点改进：1、对前三次匹配均使用遮挡矩阵纠错机制进行优化2、将JDE策略中的step3进行拆分，对Traked状态的track和Losted状态的track进行分别匹配3、将目标检测补偿机制加入匹配策略中作为第四次匹配。对于第一点改进3.3.2小节已较为清晰的阐述，第三点改进则参考了ByteTrack[7]工作。对于第二点改进主要基于两个原因：1、在帧率较高场景下，处于Tracked状态的目标在遮挡较少场景下状特征变化程度不大。2、长时间处于Losted状态的目标外观特征可能发生较大改变，存在对Tracked状态目标匹配影响。因此本文将两种状态的track分别进行匹配，并使用IOU差异对所有基于外观匹配生成的距离矩阵进行过滤。

## 丢失目标再匹配问题优化模型

目标丢失后再匹配问题是目前大部分MOT算法的痛点，原因在于长时间丢失目标外观特征会发生较大变化，目标运动线性度较低难以预测目标位置。目前在线跟踪算法对于该问题对于该问题应对较差，原因在于基于TBD范式的算法，往往对新的视频帧的detection难以提取足够多的有效信息。当某一跟踪跟踪目标重新出现时会被目标检测器重新检测位置，但此时由于目标形变或遮挡原因会导致当前外观特征和目标原特征相差较大，因此会导致算法给这一目标分配新的ID或和其他丢失目标错误匹配。相较于TBD范式的MOT算法，离线跟踪算法在这个问题上有更好的表现，但仍未从根本解决此问题。本文使用离线方式处理丢失目标再匹配问题，提出一种结构对称的对比神经网络模型，该神经网络接受两个目标跟踪轨迹序列特征，并输出两个目标轨迹间相似度。

### 设计初衷

首先介绍对比神经网络设计初衷，目前大多数外观特征提取器都生成一个固定长度的目标特征向量，这种方式较为简单但存在一定缺陷。假设特征向量仅由三个元素组成，则归一化后的特征向量在三维空间内构成一个三维球面，任何目标特征均可由球面上的一个点表示。通常使用特征的余弦值计算特征差异，因此可知相同目标特征在三维球面上需要离得越近越好，不同目标越远越好。但是由于相同目标在图片上不同的拍摄效果，相同目标特征之间存在一定距离，因此每一个目标特征应该分布在球面上的一个区域内（本文称：为特征域），特征域占据一定面积。由于球面大小有限，因此一旦存在大量跟踪目标就难以保证目标特征域不存在相互重叠问题。

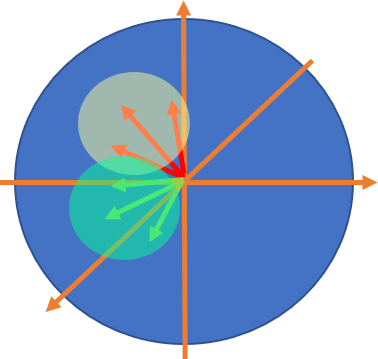


图3-8 不同目标特征域与球面展示图

从上图可以看出红色目标特征域与绿色目标特征域存在重合区域，因此无法根据这部分特征关联相关目标，这种现象在高维数据中也普遍存在。因此为了避免直接将目标映射固定特征空间，本文受到差分模拟电路启发，结合Transformer[21]网络设计了一种结构对称的对比神经网络，网络接受两个目标外观特征序列信息，输出两个目标相似度。

### 模型结构

Transformer网络最初被设计用于自然语言处理领域，但近年来有大量研究将该网络引入计算机视觉领域并取得了不错的效果。Transformer最出彩的设计在于其完全基于注意力机制对序列信息进行编码解码，首先通过多层编码器将原始序列进行编码，再使用解码器将编码数据进行转化。下图展示Transformer网络结构图，图片来源于工作[21]。

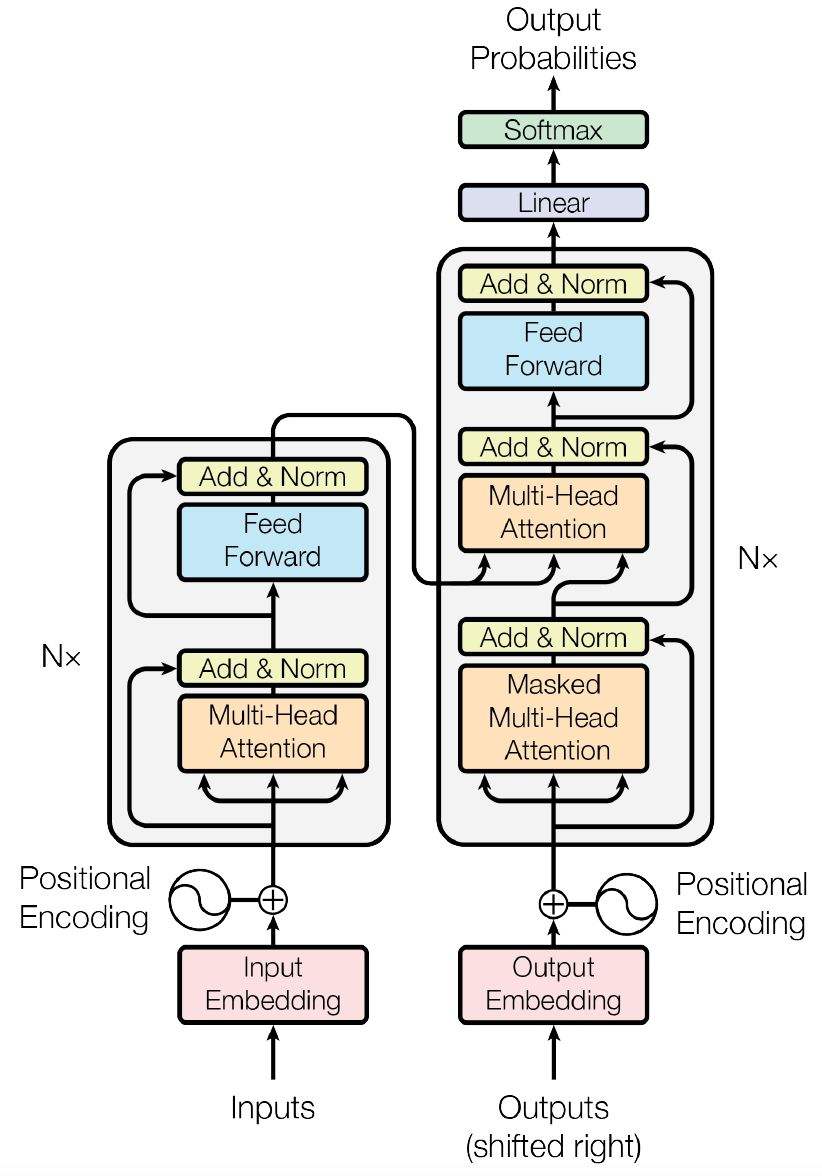


图3-8 Transformer结构图

因此Transformer网络最初被设计用于机器翻译，从宏观角度看Transformer将处在A空间下的特征序列，转化至B空间。由此不难联想到将这种非对称网络重复两次，如果两个目标是同一目标因满足：将目标A特征序列映射至目标B特征域和将目标B特征序列映射至目标A特征域输出几乎相同，通过对比两次输出向量差异便可以估计两个不同序列特征的差异。基于这种想法本文设计如下模型，模型结构图如下：

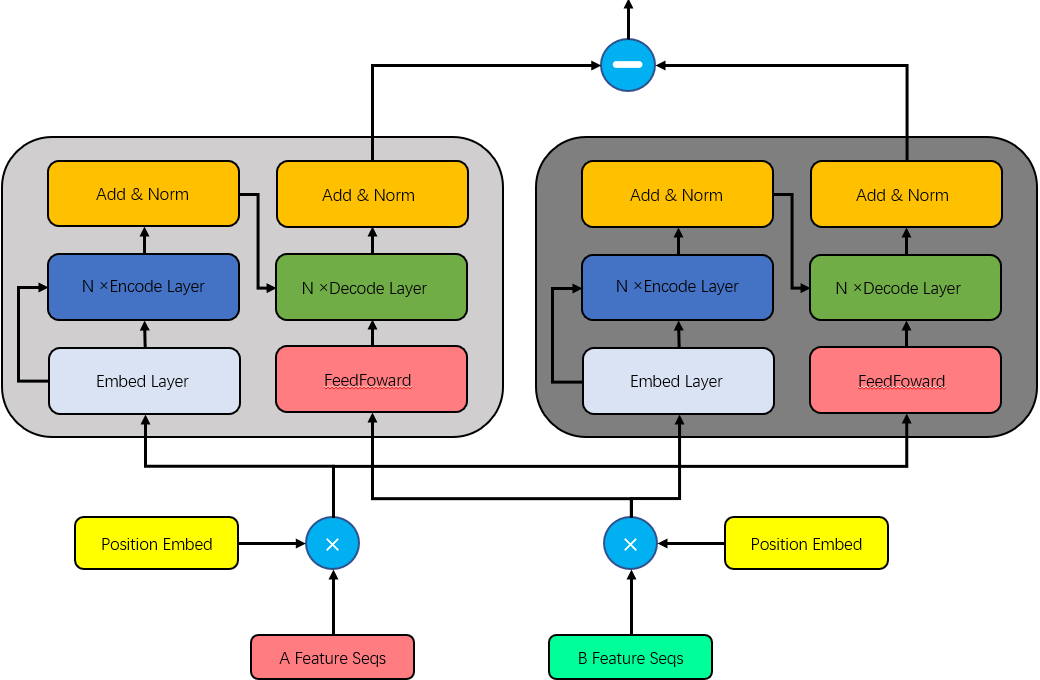


图3-9 对比网络模型结构图

上图中左右两部分模型均参考Transformer结构进行设计，有相同结构并共享相同参数，因此可看作两个特征序列互相映射然后差分的对比网络，差分操作计算两个特征余弦距离，除此之外模型还需要输入sequence -mask。模型输入是两个轨迹不同视频帧的特征向量组和两个对应的sequence-mask，输出为了两个轨迹特征组的相似程度。

本文使用上述对比网络对目标轨迹进行离线匹配，具体流程是：1、在线跟踪过程中记录在线跟踪目标在不同帧对于外观特征与位置2、跟踪结束后，将可能是相同目标的轨迹序列特征送入网络匹配3、根据匹配修正在线跟踪输出结果，其中可能发生匹配的目标轨迹是时空差异均在一定阈值范围内的两个目标。

## 算法流程图

本节将上述三节中提到的优化措施进行整合，提出一种新的MOT算法，该算法基于FairMOT[1]工作，且可被移植到大部分基于JDE跟踪策略的MOT算法上。下图展示算法流程图：



图3-10 算法流程图

上图展示了算法基本流程，可看出本文算法既可以在线跟踪目标，也可以对在线跟踪结果进行离线处理进一步优化。其中运动补偿、遮挡矩阵纠错机制、非线性插帧三项改进优化在线跟踪性能，对比网络再匹配、高斯过程平滑提升离线跟踪性能。实验表明本文提出的算法相较于对照组有较大精度提升，且跟踪速度下降在可接收范围内。

## 本章小结

本章结合第二章对当前MOT算法在不同场景下存在问题，给出了具体的优化措施。并介绍了这些措施基于哪些技术，设计这些措施的初衷是什么，这些机制运行的基本假设是什么。

本章第二小节介绍了图像对齐技术，并基于这一技术进行进一步拓展将其应用在镜头运动场景下，对运动观测模型进行运动补偿。在此基础上本文提出一种新式的非线性插帧方法，缓解因镜头运动造成的检测丢失问题。

本章第三小节介绍了针对密集场景下的算法优化措施，首先针对目标遮挡丢失的情况，引入了检测补偿策略。然后对于目标重合时发生的身份信息互换问题提出了一种基于遮挡矩阵的纠错机制，并介绍了该纠错机制基于的场景假设，在场景假设下机制的有效性。最后结合检测补偿和遮挡矩阵纠错提出了一种四次匹配策略。

本章第四小节介绍了使用离线方式对长时间丢失目标再匹配问题的优化，首先阐述了模型设计初衷和灵感来源，然后结合Transformer网络提出了一种对称结构的对比神经网络。

本章最后一节将上述优化手段进行整合，提出一种新型的MOT算法，并给出算法的流程图，可以将算法划分为在线跟踪部份和离线优化部份，实验表明相较于对照组本文算法具有较高的跟踪精度。

# 实验验证

## 引言

本章将通过实验展示第三章提出的优化措施的有效性，本章使用MOT16[56]、MOT17[56]、MOT20[57]测试集作为测试数据。由于上述三个数据集测试集标签不公开，在官网提交预测结果有次数限制，因此本文的部份实验会将MOT17训练集前半部分作为模型训练集，将MOT17训练集后半部份和MOT20训练集作为验证集进行消融性实验。本章将分为五个小节，第一节介绍测试数据集和评价指标，第二节通过实验展示运动补偿和非线性插帧对镜头运动场景下的优化效果，第三节展示匹配纠错机制和四次匹配策略对人员密集场景下的优化效果，第四节展示对比神经网络对丢失目标再匹配的优化效果，最后一节将三种方法进行聚合通过实验展示本文算法优势。

## 数据集和评价指标

当前大多数多目标跟踪领域研究均以MOT16、MOT17、MOT20三个数据集上的指标作为性能评价标准，三个数据集下的子数据集分布和对应帧率如下表所示：

表4-1 MOT数据集子数据集分布

|  |  |  |  |
| --- | --- | --- | --- |
|  | MOT16 | MOT17 | MOT20 |
| Train | MOT16-02（30 fps）、MOT16-04（30 fps）、  MOT16-05（15 fps）、  MOT16-09（30 fps）、  MOT16-10（30 fps）、  MOT16-11（30 fps）、  MOT16-13（25 fps）、 | MOT17-02（30 fps）、MOT17-04（30 fps）、  MOT17-05（15 fps）、  MOT17-09（30 fps）、  MOT17-10（30 fps）、  MOT17-11（30 fps）、  MOT17-13（25 fps）、 | MOT20-01（25 fps）、MOT20-02（25 fps）、  MOT20-03（25 fps）、  MOT20-05（25 fps）、 |
| Test | MOT16-01（30 fps）、MOT16-03（30 fps）、  MOT16-06（15 fps）、  MOT16-07（30 fps）、  MOT16-08（30 fps）、  MOT16-12（30 fps）、  MOT16-14（25 fps）、 | MOT17-01（30 fps）、MOT17-03（30 fps）、  MOT17-06（15 fps）、  MOT17-07（30 fps）、  MOT17-08（30 fps）、  MOT17-12（30 fps）、  MOT17-14（25 fps）、 | MOT20-04（25 fps）、MOT20-06（25 fps）、  MOT20-07（25 fps）、  MOT20-08（25 fps）、 |

每个子数据集都由一组完整视频截图和官方提供的目标检测器构成，训练集数据还有对应的标签。由于MOT17数据集和MOT16数据集由相同的视频流组成，但MOT17数据集标签相对要求更加苛刻，因此一般使用MOT17数据集作为训练集并在MOT16、MOT17两个数据集同时测试结果。上图还反应了不同子数据的视频帧率，高帧率子数据集通常具有更好的跟踪效果，除此之外每个子数据集视频帧尺度也不完全相同。

本文使用[58]、[58]、[58]、[59]、[59]指标衡量算法跟踪效果，指标计算公式如下：

 (4.1)

指标中的表示未检测到的目标，表示错误检测的目标，表示目标间的互换，表示所有正确标签数量。

 (4.2)

指标受目标检测器性能影响较大，且仅使用指标刻画跟踪身份信息精度，相比之下指标能更有效描述跟踪身份精度。是和的调和平均数，表示轨迹正确匹配数，表示轨迹错误匹配数，表示目标丢失匹配数，指标能更有效的描述跟踪轨迹和目标之间关联精度。

 (4.3)

指标将以为代表更好衡量检测器性能的指标和以为代表更好衡量关联精度的指标进行整合。并测量在不同匹配阈值下的指标表现，其中表示目标与轨迹的正确关联数，表示目标与轨迹未发生关联数，表示轨迹与错误目标发生关联数，表示对于中每一种正确轨迹与目标匹配对。

 (4.4)

指标和指标类似主要受目标检测器影响较大，其中、、分别表示在不同匹配阈值下，匹配成功的轨迹、匹配失败的轨迹、匹配失败的目标。

本文将通过和指标展示对算法目标检测能力的提升，使用和指标展示算法对轨迹与目标关联能力的提升，使用指标展示算法综合能力的提升。

## 镜头运动场景优化效果

本节通过实验展示第三章提到的镜头运动场景优化的实际运行效果。本节将分为两个小节，首先展示基于图像对齐技术的镜头运动补偿机制性能，然后展示基于运动补偿后的非线性插帧算法效果，本节使用MOT17训练集子数据集后半部份作为验证集计算性能指标，实验算法为FairMOT[1]+运动补偿+非线性插帧，算法对照组为FairMOT、FairMOT+线性插帧。

### 基于图像对齐技术的运动补偿

本小节从三个实验展示运动补偿机制对跟踪性能的提升，三个实验分别是：相邻视频帧相同目标IOU（Intersection over Union）距离、运动观测模型预测误差与时间关系图、相邻视频帧丢失目标占比。

相邻视频帧相同目标IOU距离实验与第二章图2-5实验流程一致，图2-5中展示了不同数据集的差异，本实验展示不同方法的差异，为了展示更加直观本文仅在MOT17-05、MOT17-10、MOT17-11、MOT17-13存在镜头运动的子数据集上测试。下图展示了补偿机制实际效果：

图4-1 相邻帧相同目标IOU差异

从上图可以清晰看出当算法中加入镜头运动补偿后，相邻两帧视频相同目标IOU距离显著下降，ECC[20]算法效果略好于ORB[31]特征点提取加仿射矩阵计算。下图展示不同运动补偿方法对运动观测模型预测误差影响。

图4-2 运动观测模型预测差异

上图实验与图2-6实验流程相同，可以清晰看出随着时间在镜头运动场景下，目标预测误差随时间抖动较大。在对镜头进行运动补偿后，抖动幅度显著减小，ECC算法生成曲线与ORB方法生成曲线均始终在Kalman Filter生成曲线之下。下图将展示运动补偿后相邻视频帧目标检测丢失比例：

图4-3 相邻视频帧目标检测丢失比例

上图实验与图2-3实验流程相同，从上述三个实验中可以得出结论基于图像对齐技术的镜头运动补偿，能有效提升观测模型对运动目标观测的线性度，对因镜头抖动造成的目标检测丢失有一定优化效果，因此运动补偿机制能在一定程度上提高算法跟踪精度。

### 非线性插帧优化效果

本小节将通过两个实验展示非线性插帧相较于线性插帧的优势，两个实验分别是：正确插帧占比实验、插帧后算法跟踪精度实验。

正确插帧占比实验对比和目标正确匹配的插帧轨迹与总的插帧轨迹比值，比值越高说明插帧精度越高。下表展示实验结果：

表4-1 插帧算法效果展示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | TP↑ | IDSW↓ | FP↓ | Accuracy↑ |
| Linear Insert | 1026 | 230 | **2135** | 32.71% |
| ECC Offset | **1272** | **206** | 2478 | **33.92%** |
| ORB Offset | 1243 | 210 | 2496 | 33.24% |

上表中ECC Offset表示基于ECC镜头运动补偿的非线性插帧算法，ORB Offset表示基于ORB方法的运动补偿的插帧算法，Accuracy计算公式为TP/（TP+FP）。上上表中可观察结果相较于线性插帧法本文提出的非线性插帧，在插帧精度上有一定程度的提高，尤其是在镜头运动场景下提升更为明显，下表展示不同插帧算法在数据集上的跟踪精度。

表4-2 插针后算法跟踪精度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ |
| No Insert | 69.1 | 72.8 | 57.2 | 56.2 |
| Linear Insert | 71.0 | 73.9 | 58.6 | 58.3 |
| ECC Offset | **71.6** | **74.3** | **59.1** | **58.8** |
| ORB Offset | 71.4 | 73.2 | 58.6 | 58.7 |

从上表清晰展示插帧机制对算法跟踪目标精度有显著提升，使用插帧机制后MOTA和DetA指标均至少上升2%，所以不难得出结论插帧机制能有效提升MOT算法目标检测能力。上表同样展示了本文提出的基于运动补偿的非线性插帧技术相较于传统线性插帧技术有一定提升，相比于线性插帧基于ECC算法和ORB方法的非线性插帧在MOTA和DetA指标上均有领先。

综上所述本文将图像对齐技术和线性插帧机制相结合，提出了基于运动补偿非线性插帧方法，实验表明本文提出的方法能有效提升运动观测模型观测目标线性度以及提升MOT算法目标检测能力。

## 人员密集场景优化效果

本节通过实验展示本文3.3.3小节提出的四次匹配策略对算法在人员密集场景下跟踪性能的提升。实验算法为FairMOT+运动补偿+四次匹配策略，实验对照组为FairMOT、FairMOT+运动补偿+检测补偿机制（ByteTrack[7]中提出）。实验在MOT17训练集子数据集后半部和MOT20训练集子数据集上进行测试。

图4-4 验证数据集MOTA指标

图4-5 验证数据集IDF1指标

图4-6 验证数据集HOTA指标

图4-7 验证数据集DetA指标

图4-8 验证数据集IDSW指标

上述五幅图片清晰展示了本文提出的基于遮挡矩阵的纠错机制和四次匹配机制能有效的提升密集场景下的跟踪算法精度。图2-9展示了不同子数据集上的人员密集程度，其中MOT17-02、MOT17-04、MOT20-01、MOT20-02、MOT20-03、MOT20-05五个子数据集人员密集程度大场景较为复杂。从上述五幅图片中可以看出本文提出的优化机制在这七个子数据集上IDF1和HOTA指标均有显著提高，其中值得一提的是本文提出优化机制对IDSW指标优化更为明显，从图4-8中可以看出相较于对照组身份互换错误数有较大幅度下降，这在很大程度上得益于本文提出的遮挡矩阵纠错机制。但是由于MOTA和DetA指标主要受目标检测器性能影响较大，因此基于遮挡矩阵的纠错机制和四次匹配策略对该指标优化并不显著。

## 丢失目标再匹配问题优化效果

本节将分为两小节介绍展示丢失目标再匹配问题优化效果，第一小节展示如何训练3.4小节提出的对比神经网络模型，第二小节将通过两个实验展示模型对该问题的优化效果，实验分别是：优化后丢失目标特征差异与时间关系实验、算法跟踪精度实验。

### 实验环境与模型训练

本文使用Python语言作为基础开发工具，使用Pytorch[40]深度学习框架搭建神经网络模型，实验软硬件环境如下表所示：

表4-3 实验软硬件环境

|  |  |
| --- | --- |
| Operating System | Ubuntu 18.04 LTS 64bit |
| CPU | Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz |
| GPU | GeForce RTX 2080Ti |
| DRAM | 16G DDR4 2400 MHz |
| Disk | Colorful SL500 256GB |
| Python version | 3.8.12 |
| CUDA version | 11.0 |
| Pytorch version | 1.7.0 |

模型训练可分为：模型搭建、数据输入、损失函数优化三个部份。本文使用Pytorch搭建神经网络模型，从3.4节可知模型以Transformer[21]网络作为基础架构，并在此基础上加入了一些残差连接以提升模型鲁棒性。模型超参数如下表所示：

表4-4模型结构超参数设置

|  |  |
| --- | --- |
| Input Sequence Size | 30 |
| Input Feature Size | 128 |
| Mid-Layer Feature Size | 128 |
| Number of Encoder | 8 |
| Number of Decoder | 8 |
| Number of Mutiple Attention Head | 1 |
| Dropout Rate | 0.2 |

模型训练时需要持续的数据输入，模型输入为三个特征张量和三个掩模张量，特征张量形状为[Batch\_Size,Sequence\_Size,Featrue\_Size]，掩膜张量形状为[Batch\_Size,Sequence]。其中Batch\_Size表示模型训练时批处理的大小，因此特征张量可看作是由Batch\_Size个轨迹的序列特征构成，而掩膜张量则表示特征张量对应位置特征是否能参与模型计算。

模型在训练时需要三个特征张量和三个掩膜张量的原因在于，本文使用Re-Identify任务常用损失函数Triplet-Loss，Triplet-Loss计算公式如下：

 (4.5)

上述公式中函数返回和中较大的元素，则表示模型接受两个轨迹特征张量返回相似度，分别表示三个不同的特征张量。因此两个特征张量和它们对应的掩膜张量，是由相同目标在不同视频帧上的连续轨迹的序列特征组成，它们在Batch\_Size维度上相同的位置是同一个目标的外观特征。而则表示两个特征张量由不同目标特征序列组成。则是超参数，一旦模型训练满足一定要求后则停止继续优化损失函数。

综上所述在模型训练过程中，会随机选择两个不同的目标，对其中一个目标选择其轨迹上的两个连续不重叠区域对应的视频帧，提取外观序列特征。对另一个目标则随机选择其轨迹上的一段连续区域提取外观序列特征。提取模型使用FairMOT[1]的一体式跟踪模型中的特征提取分支。在进行训练时会对视频帧进行随机反转、仿射变换、HSV变换等数据增强操作，以提升模型泛化能力。数据输入后经过模型和损失函数计算后便可使用优化器对损失进行优化，本文使用Adam[41]优化器，优化器参数采用Pytorch框架默认设置，训练阶段超参数如下表所示：

表4-5模型训练超参数设置

|  |  |
| --- | --- |
| Optimizer | Adam |
| Batch\_Size | 16 |
| Number of Epoch | 100 |
| Base\_Learing\_Rate | 0.001 |
| Learning\_Rate\_Decay\_Weight | 0.1 |
| Learning\_Rate\_Decay\_Step | 25，50，75 |
| Weight\_Decay | 1e-5 |
| Margin | 0.5 |

### 优化效果展示

本节通过优化后丢失目标特征差异与时间关系实验和算法跟踪精度实验展示优化效果。下图实验使用MOT17训练集子数据集前半部分作为对比神经网络的训练集，使用MOT17训练集子数据集后半部份和MOT20训练集作为验证集展示效果。下图展示相同目标经过对比神经网络后外观差异与丢失时间关系图：

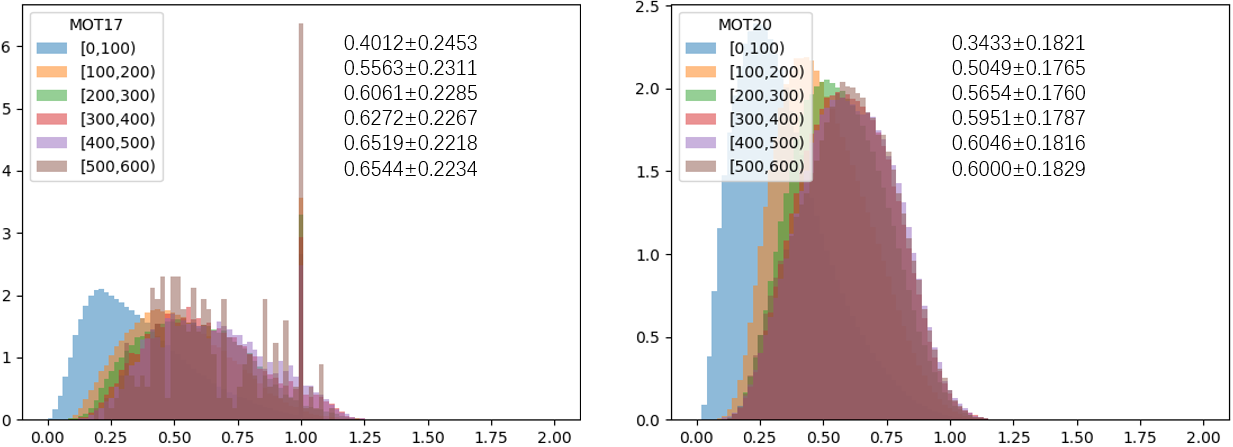


图4-10 对比网络外观差异与丢失时间关系

上图展示了相同目标的外观特征经过对比网络输出后差异，和图2-14对比后可以看出同对照组相比对比网络对相同目标外观差异较小，对于丢失超过100帧的目标外观特征仍均有一定的相关性，实验同时也证明了对长时间丢失目标再匹配问题离线方法由于在线方法，因为离线方法能提取更多有效的序列特征信息。下表展示使用对比网络优化后MOT算法再MOT17训练集子数据集后半部和MOT20训练集子数据集上的跟踪精度：

表4-5 MOT17验证集优化精度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ |
| FairMOT | 69.1 | 72.8 | 57.2 | 56.2 |
| Linear Insert | 71.0 | 73.9 | 58.6 | 58.3 |
| Linear Insert + Rematch | **71.2** | **74.2** | **58.9** | **58.6** |

表4-6 MOT20验证集优化精度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ |
| FairMOT | 69.1 | 72.8 | 57.2 | 56.2 |
| Linear Insert | 54.6 | 50.3 | 39.0 | **45.7** |
| Linear Insert + Rematch | **54.8** | **50.6** | **39.4** | **45.7** |

从上述两张表可以观察得出本文提出的对比神经网络模型对MOT算法跟踪精度均有一定程度提高，由于跟踪场景较为复杂外观特征提取受目标检测器影响较大，因此相较于直接测试外观特征差异，对MOT算法跟踪精度提升幅度较小。

## 算法整体性能

本节将上述第三章提出的优化机制进行整合，在MOT17验证集和MOT20验证集上进行测试，并最终提交至MOTChallenge官网评估算法在MOT16、MOT17、MOT20三个测试集上的跟踪精度，下表展示测试结果：

表4-7 MOT17 验证集算法跟踪精度

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ | Fps↑ |
| FairMOT[1] | 69.1 | 72.8 | 57.2 | 56.2 | **25.9** |
| Ours | **73.8** | **79.1** | **62.2** | **59.7** | 14.2 |

表4-8 MOT20 验证集算法跟踪精度

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ | Fps↑ |
| FairMOT | 49.6 | 49.4 | 37.7 | 40.6 | **12.3** |
| Ours | **58.5** | **57.8** | **43.9** | **47.5** | 9.3 |

表4-9 MOT16 测试集算法跟踪精度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ | IDSW↓ | Fps↑ |
| FairMOT | 74.9 | 72.8 | 59.5 | 56.2 | 1074 | **25.9** |
| CSTracker[30] | 75.6 | 73.3 | 59.8 | 61.8 | 1121 | 15.8 |
| FUFET[32] | 76.5 | 68.6 | 58.3 | 63.3 | 1026 | 6.6 |
| GRTU[50] | 76.5 | 75.9 | **62.6** | **63.3** | 584 | 3.6 |
| Ours | **76.7** | **76.6** | 62.1 | 63.1 | **373** | 14.2 |

表4-10 MOT17 测试集算法跟踪精度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ | IDSW↓ | Fps↑ |
| FairMOT | 73.7 | 72.3 | 59.3 | 60.9 | 3303 | **25.9** |
| RelationTrack[51] | 73.8 | 74.7 | 61.0 | 60.6 | 1374 | 8.5 |
| CSTracker | 74.9 | 72.6 | 59.3 | 61.1 | 3567 | 15.8 |
| GRTU | 74.8 | 75.0 | **62.0** | **62.1** | 1812 | 3.6 |
| Ours | **75.3** | **75.7** | 61.3 | 62.0 | **1155** | 14.2 |

表4-11 MOT20 测试集算法跟踪精度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Method | MOTA↑ | IDF1↑ | HOTA↑ | DetA↑ | IDSW↓ | Fps↑ |
| FairMOT | 61.8 | 67.3 | 54.6 | **54.7** | 5243 | 13.2 |
| TrTrack[38] | 65.0 | 59.4 | 48.9 | 53.3 | 3608 | **19.4** |
| CSTracker | **66.6** | 68.6 | 54.0 | 54.2 | 3196 | 4.5 |
| Ours | 63.3 | **69.6** | **55.2** | 54.0 | **1745** | 9.3 |

从上述五张表中均可以看出，相对于对照组FairMOT算法，本文算法跟踪精度均存在显著提升。在MOT17验证集和MOT20验证集上，MOTA、IDF1、HOTA、DetA指标均存在4%~9%左右的提升，在MOT16和MOT17测试集上述指标均有2%~7%左右的提高，在MOT20测试集上述指标均有1%~2%左右的提高。但由于MOT20测试集跟踪场景过于复杂，可能导致引用ByteTrack[7]目标检测补偿机制可能存在负面影响，DetA指标有一定下降。值得注意的是本文算法能大幅度降低IDSW指标，在三个测试集上IDSW指标均下降66%左右，这进一步证明了本文提出的遮挡矩阵纠错机制和四次匹配策略能有效克服目标身份互换错误。本文提出的算法具有一定计算复杂度，相较于对照组跟踪速度均有25%~45%左右的下降，但对于大多数场景仍可以维持在15fps左右的跟踪速度，具有一定的实用价值。

## 本章小节

本章通过实验展示了第三章针对三种场景优化策略的有效性，本章正文内容从第二节开始，通过五个小节介绍了实验环境、实验流程、实验结果。

第二节介绍了实验所使用的数据集，以及每个数据集对应的子数据集。由于所有MOT数据集测试集需要在MOTChallenge官网提交运行结果，提交有次数限制，因此本文部分实验使用验证集测试，第二节也介绍了验证集划分方法。除此之外第二节还介绍了评估算法使用的指标以及对应的计算过程。

第三节介绍了运动补偿和非线性插帧两种机制的优化效果，通过实验可以得出运动补偿能有效提升MOT算法中的运动观测模型线性度，以及非线性插帧相较于线性插帧能在一定程度上提升算法目标检测能力。

第四节介绍了遮挡矩阵纠错机制和四次匹配策略对算法的优化效果，通过实验可以看出这种优化措施对MOT算法的匹配能力（IDF1、HOTA指标）有较为显著的提升，尤其是IDSW指标下降幅度较大，但对MOT算法目标检测能力（MOTA、DetA指标）提升幅度较小。

第五节介绍了对比神经网络的实验软硬件环境、实验流程和模型和训练超参数，实验表明相同外观特征经过对比神经网络处理后差异有一定降低，随着丢失时间增加外观差异变化程度相较于未处理之前较小。但在实际跟踪环境测试时发现对跟踪精度只有略微提升，这可能是由于外观特征提取受目标检测器影响较大。

第六节将以上三种优化策略进行整合，在MOT数据集测试集上测试算法实际效果。实验表明本文提出的新式MOT跟踪算法，相较于对照组在各个指标上均存在显著提升，算法尤其是对IDSW指标超过了绝大多数算法。由于任何算法均存在过拟合风险，本文主要在验证集上优化算法设计，因此算法在验证集上提升幅度大于测试集提升幅度。

# 总结与展望

## 本文主要内容与结论

本文对当今研究热点基于视觉的多目标跟踪算法进行了深入分析和研究，主要分析了多目标跟踪算法在不同场景下存在的问题，以及针对不同场景提出相应的措施优化算法效果，最终通过实验验证优化措施的可行性和优化后效果，实验证明本文提出的优化措施能提升算法精度且计算成本在可接受范围内，具有一定的实用价值。

本文主体内容分为四章，第一章为绪论首先分析了基于视觉的多目标跟踪算法研究背景于研究意义。然后分析了该领域国内外的研究现状，并将该领域的算法分为两大类：在线跟踪算法和离线跟踪算法，在线跟踪算法又可以分为TBD范式和TBS范式，最后总结了当前该领域研究国内外存在的问题。

第二章为场景问题分析，主要内容为分析当前MOT算法在三种场景种存在的问题，三种场景分别是：镜头运动场景、人员密集场景、目标长距离丢失场景。第二章分析得出三个场景分别存在的问题是：镜头运动场景对目标检测器影响较大，运动场景下运动观测模型观测目标线性度较差、人员密集场景存在大量目标遮挡现象，被遮挡程度越大的目标越难以检测和提取有效的外观特征、相同目标外观差异随着丢失时间变长而增大，丢失超过100帧的目标外观特征几乎不具备相关性。

第三章针对以上问题提出了优化措施，首先针对运动场景提出一种基于图像对齐技术的运动补偿，在此基础上提出了一种非线性插帧算法以提高运动场景下的目标检测能力。其次在人员密集场景下引入了ByteTrack种的目标检测补偿机制，并且通过实践观测提出了一种基于遮挡矩阵的匹配纠错机制，然后将这两种机制进行融合生成一种新的四次匹配策略。最后分析了当前外观特征提取方法对长时间丢失目标特征提取存在的问题，并受到差分电路启发设计了一种基于Transformer网络的对比神经网络模型，该模型具有对称结构。

第四章通过实验证明以上优化措施的有效性，在验证集上运动补偿能显著提升目标运动线性度，分线性插帧相较于线性插帧具有更高的插帧精度。遮挡矩阵纠错机制和四次匹配策略能有效提升MOT算法的匹配能力并能极大抑制身份互换错误，但对目标检测能力提升有限。第四章同时介绍了对比神经网络训练环境、训练流程和超参数，实验证明这个模型能有效优化相同目标外观特征差异，但对算法精度提升较为有限，可能是受到目标检测器性能影响。最后将上述优化措施进行整合形成一种新的MOT算法，在MOT数据集测试集上评估算法性能，结果表明本文提出的算法在不同数据集不同指标上均存在1%~7%的提升，对于IDSW指标更是达到了66%幅度的下降。但算法具有一定的计算复杂度，相较于对照组跟踪速度下降幅度达25%~45%，但跟踪帧率仍保持在15fps左右，因此具有一定实用价值。

## 本文的主要创新点

本文主要创新点可以归纳如下：

1. 结合图像对齐技术提出一种非线性插帧算法，相较于线性插帧算法能有效提升MOT算法在镜头运动场景下的目标检测能力。
2. 引入目标检测补偿机制，提出了一种基于遮挡矩阵的匹配纠错机制，并将两者结合形成一种新的四次匹配策略。相较于JDE匹配策略能显著提升MOT算法匹配能力，并抑制身份互换问题发生。
3. 提出了一种基于Transformer网络的对比神经网络模型和对应的训练方法，模型能较好地降低相同目标跟踪丢失前后外观特征差异。
4. 将以上三种优化措施结合生成一种新的MOT算法，跟踪精度相较于对照组有显著提升。

## 展望

本文算法相较于ByteTrack等算法跟踪精度仍有劣势，主要原因在于本文使用检测器性能较差。因此可以得出结论目前MOT算法仍然受目标检测器性能影响较大。这可能是无法解决的难题因为必须对目标进行定位，因此较为公平的措施是使用相同的跟踪器对比算法匹配性能。并且检测器目标定位能力极大地影响了目标外观特征，检测框定位的波动容易导致提取到不同的目标特征以及背景特征，这个问题在密集场景下十分明显，并且目前几乎无法解决。

除此之外算法仍存在目标身份互换问题，这可能由于目标重合程度过大或目标丢失时间过长导致。本文算法对于长时间丢失目标匹配能力仍较差，因此如何提取目标有代表性的外观特征，并通过一系列波动的目标检测框和对应的外观特征对长时间丢失目标进行正确匹配仍是一个难题。这个问题的解决可能需要整个深度学习技术的进步，而非单纯在神经网络结构上的优化。

本文算法在FairMOT基础上进行改进，FairMOT具有较高的实用价值，对于非密集场景有较好的跟踪性能，因此相较于该算法本文提出的算法也可以在稍复杂一些的场景下使用。除此之外本文提出的优化措施应具有一定指导意义，这些优化措施对其他跟踪算法也能起到一定的精度提升。

# 致 谢

研究生生涯即将结束，在此对这三年帮助过我的组织、群体、个人统一致谢。

首先要感谢中国共产党，是党的领导让我有机会进入华中科技大学攻读硕士学位，是党的领导带我们打赢了疫情攻坚战，是党的领导让我脱贫重回温饱线，是党的领导让我能在中国有一个稳地的环境攻读学位，是党的领导为我们每个研究生发放每月经济补贴，是党的领导为我指引了奋斗的方向。

其次要感谢的是华中科技大学，华中科技大学作为985工程院校，代表了全国最高水平的大学教育与实验设备环境。在这个优渥的学习环境下，我才有机会能做出一定的研究成果。学校良好的生活环境与严格的疫情管控方式让我能安心学业，不必担心疫情给生活带来的麻烦。

最应该感谢的是路松峰导师和朱建新导师，两位导师对我三年研究生生涯的指导，您们的指导为我指点迷津，在课题迷茫时为我指引正确的前进方向。在我遇到困难时引导我思考问题并给出正确的解题思路，对我的科研成果起到了极大地帮助。

最后感谢我在研究生生涯遇到的各位同学，你们的同行给我带来了陪伴的力量，由衷希望我们能在今后的生活中再相遇。

# 参考文献

1. Zhang Y, Wang C, Wang X, 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.
2. Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric[C]//2017 IEEE international conference on image processing (ICIP). IEEE, 2017: 3645-3649.
3. Wang Z, Zheng L, Liu Y, et al. Towards real-time multi-object tracking[C]//European Conference on Computer Vision. Springer, Cham, 2020: 107-122.
4. Stadler D, Beyerer J. Modelling Ambiguous Assignments for Multi-Person Tracking in Crowds[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2022: 133-142.
5. Kalman R E. A new approach to linear filtering and prediction problems[J]. 1960.
6. Pang B, Li Y, Zhang Y, 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.
7. Zhang Y, Sun P, Jiang Y, et al. ByteTrack: Multi-Object Tracking by Associating Every Detection Box[J]. arXiv preprint arXiv:2110.06864, 2021.
8. Zhou X, Koltun V, Krähenbühl P. Tracking objects as points[C]//European Conference on Computer Vision. Springer, Cham, 2020: 474-490.
9. Wang Q, Zheng Y, Pan P, et al. Multiple object tracking with correlation learning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 3876-3886.
10. Yang F, Chang X, Sakti S, et al. ReMOT: A model-agnostic refinement for multiple object tracking[J]. Image and Vision Computing, 2021, 106: 104091.
11. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
12. Wang Y, Weng X, Kitani K. Joint detection and multi-object tracking with graph neural networks[J]. arXiv preprint arXiv:2006.13164, 2020, 1(2).
13. Felzenszwalb P F, Girshick R B, McAllester D, et al. Object detection with discriminatively trained part-based models[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 32(9): 1627-1645.
14. Freund Y, Schapire R E. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of computer and system sciences, 1997, 55(1): 119-139.
15. Papageorgiou C P, Oren M, Poggio T. A general framework for object detection[C]//Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271). IEEE, 1998: 555-562.
16. Cortes C, Vapnik V. Support-vector networks[J]. Machine learning, 1995, 20(3): 273-297.
17. Bewley A, Ge Z, Ott L, et al. Simple online and realtime tracking[C]//2016 IEEE international conference on image processing (ICIP). IEEE, 2016: 3464-3468.
18. Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Ieee, 2005, 1: 886-893.
19. Jiang X, Li P, Li Y, et al. Graph neural based end-to-end data association framework for online multiple-object tracking[J]. arXiv preprint arXiv:1907.05315, 2019.
20. Evangelidis G D, Psarakis E Z. Parametric image alignment using enhanced correlation coefficient maximization[J]. IEEE transactions on pattern analysis and machine intelligence, 2008, 30(10): 1858-1865.
21. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
22. Lowe D G. Object recognition from local scale-invariant features[C]//Proceedings of the seventh IEEE international conference on computer vision. Ieee, 1999, 2: 1150-1157.
23. Li J, Gao X, Jiang T. Graph networks for multiple object tracking[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2020: 719-728.
24. 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. 2019: 253-261.
25. Han S, Huang P, Wang H, et al. Mat: Motion-aware multi-object tracking[J]. Neurocomputing, 2022.
26. Brasó G, Leal-Taixé L. Learning a neural solver for multiple object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6247-6257.
27. Huang Y, Zhu F, Zeng Z, et al. Sqe: a self quality evaluation metric for parameters optimization in multi-object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 8306-8314.
28. Yin J, Wang W, Meng Q, et al. A unified object motion and affinity model for online multi-object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6768-6777.
29. Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF)[J]. Computer vision and image understanding, 2008, 110(3): 346-359.
30. Liang C, Zhang Z, Lu Y, et al. Rethinking the competition between detection and reid in multi-object tracking[J]. arXiv preprint arXiv:2010.12138, 2020.
31. Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF[C]//2011 International conference on computer vision. Ieee, 2011: 2564-2571.
32. Shan C, Wei C, Deng B, et al. Tracklets predicting based adaptive graph tracking[J]. arXiv preprint arXiv:2010.09015, 2020.
33. Liu Q, Chu Q, Liu B, et al. GSM: Graph Similarity Model for Multi-Object Tracking[C]//IJCAI. 2020: 530-536.
34. Zhang Y, Sheng H, Wu Y, et al. Multiplex labeling graph for near-online tracking in crowded scenes[J]. IEEE Internet of Things Journal, 2020, 7(9): 7892-7902.
35. Wieczorek M, Rychalska B, Dąbrowski J. On the unreasonable effectiveness of centroids in image retrieval[C]//International Conference on Neural Information Processing. Springer, Cham, 2021: 212-223.
36. Wang G, Lai J, Huang P, et al. Spatial-temporal person re-identification[C]//Proceedings of the AAAI conference on artificial intelligence. 2019, 33(01): 8933-8940.
37. Luo H, Gu Y, Liao X, et al. Bag of tricks and a strong baseline for deep person re-identification[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2019: 0-0.
38. Sun P, Cao J, Jiang Y, et al. Transtrack: Multiple object tracking with transformer[J]. arXiv preprint arXiv:2012.15460, 2020.
39. Wang Y, Kitani K, Weng X. Joint object detection and multi-object tracking with graph neural networks[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021: 13708-13715.
40. Paszke A, Gross S, Massa F, et al. Pytorch: An imperative style, high-performance deep learning library[J]. Advances in neural information processing systems, 2019, 32.
41. Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
42. Kim C, Fuxin L, Alotaibi M, et al. Discriminative appearance modeling with multi-track pooling for real-time multi-object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 9553-9562.
43. Pang J, Qiu L, Li X, et al. Quasi-dense similarity learning for multiple object tracking[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 164-173.
44. Shuai B, Berneshawi A, Li X, et al. Siammot: Siamese multi-object tracking[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 12372-12382.
45. Guo S, Wang J, Wang X, et al. Online multiple object tracking with cross-task synergy[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 8136-8145.
46. He J, Huang Z, Wang N, et al. Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 5299-5309.
47. Dai P, Weng R, Choi W, et al. Learning a proposal classifier for multiple object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 2443-2452.
48. Wu J, Cao J, Song L, 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. 2021: 12352-12361.
49. Saleh F, Aliakbarian S, Rezatofighi H, et al. Probabilistic tracklet scoring and inpainting for multiple object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 14329-14339.
50. Wang S, Sheng H, Zhang Y, et al. A General Recurrent Tracking Framework Without Real Data[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 13219-13228.
51. Yu E, Li Z, Han S, et al. Relationtrack: Relation-aware multiple object tracking with decoupled representation[J]. IEEE Transactions on Multimedia, 2022.
52. Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.
53. He S, Luo H, Wang P, et al. Transreid: Transformer-based object re-identification[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 15013-15022.
54. Yu F, Wang D, Shelhamer E, et al. Deep layer aggregation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 2403-2412.
55. Zhou X, Wang D, Krähenbühl P. Objects as points[J]. arXiv preprint arXiv:1904.07850, 2019.
56. Milan A, Leal-Taixé L, Reid I, et al. MOT16: A benchmark for multi-object tracking[J]. arXiv preprint arXiv:1603.00831, 2016.
57. Dendorfer P, Rezatofighi H, Milan A, et al. Mot20: A benchmark for multi object tracking in crowded scenes[J]. arXiv preprint arXiv:2003.09003, 2020.
58. Ristani E, Solera F, Zou R, et al. Performance measures and a data set for multi-target, multi-camera tracking[C]//European conference on computer vision. Springer, Cham, 2016: 17-35.
59. Luiten J, Osep A, Dendorfer P, et al. Hota: A higher order metric for evaluating multi-object tracking[J]. International journal of computer vision, 2021, 129(2): 548-578.
60. Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.
61. Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.
62. Ge Z, Liu S, Wang F, et al. Yolox: Exceeding yolo series in 2021[J]. arXiv preprint arXiv:2107.08430, 2021.
63. Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10781-10790.
64. Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, Cham, 2014: 740-755.

# 附录1 攻读硕士学位期间取得的研究成果

**项目成果**

[1] 国家重点研发计划

项目名称: 开放式数控系统安全技术

项目编号: 2021YFB2012202

起止时间: 2021年11月至2024年10月

担任角色：研发人员

[2] 国防基础研究重点项目（涉密）

项目名称: 基于\*\*\*安全防护技术

项目编号: \*\*\*

起止时间: 2021年1月至2023年12月

担任角色：研发人员

[3] 湖北省重点研发计划

项目名称: 区块链交易追踪溯源监管关键技术研究

项目编号: 2020BAB100

起止时间: 2020年9月至2022年7月

担任角色：研发人员

[4] 湖北省科技重大专项

项目名称: 多装备协同制造过程状态在线检测与智能调控

项目编号: 2020AEA011

起止时间: 2020年7月至2023年6月

担任角色：研发人员

[5] 湖北省重点研发计划

项目名称: 5G承载网故障智能诊断关键技术研究及应用

项目编号: 2021BAA171

起止时间: 2022年1月至2023年12月

担任角色：研发人员

[6] 湖北省重点研发计划

项目名称: 大规模工控网络的安全威胁智能诊断技术研究

项目编号: 2021BAA038

起止时间: 2021年12月至2023年12月

担任角色：研发人员

[7] 深圳市基础研究重点项目

项目名称: 数控系统信息安全防护技术研究

项目编号: JCYJ20210324120002006

起止时间: 2021年8月至2024年8月

担任角色：研发人员

[8] 企业合作项目

项目名称:中国工商银行物联网赋能网点数字化转型能力提升项目（基于数字孪生技术的实体建模研究）

项目编号: 001010000520200078

起止时间: 2020年6月至2022年6月

担任角色：课题技术负责人

# 附录2 其它附录

可包括详细的公式推导、实验数据、计算程序、援引他人的原始资料、数据及其设备条件等。