一、行人追踪的背景及研究意义

多目标跟踪，即Multiple Object Tracking(MOT)，其主要任务中是：给定一个图像序列，找到图像序列中运动的物体，并将不同帧的运动物体进行识别——也就是给定一个确定准确的id。在多目标跟踪任务中，行人追踪任务同我们的生活息息相关。如今，安防监控设备遍布校区、街道、校园以及其他公共场合，而行人追踪功能在其中发挥着重要的作用。例如在“天网”系统、视频监控、无人驾驶等。在应用中，可以将行人追踪与安防工作相结合，这对于重点人员的追踪、违法犯罪事件的预警、保障人民的生命财产安全等，都有着极为重要的意义。

行人追踪是计算机视觉研究领域中的一个重要的方向，在研究和实际应用方面都有着巨大的价值。1999 年欧盟 IST(Information Societ-yTechnologies) 的框架程序委员会设立重大项目 ADVISOR-(Annotated Digital Video forSurv-eillance andOptimised Retrieval),旨在开发一个系统来有效地管理公共交通系统(如地铁),从而缓解城市的压力,它覆盖了人群和个人的行为模式分析、人机交互等研究。英国的雷丁大学开展了对车辆和行人的跟踪及其交互作用识别的相关研究.美国马里兰大学的计算机视觉实验室通过分析摄像机采集的灰度视频图像,将外形分析技术与跟踪技术集合来跟踪人体各个主要部分的位置,可在户外环境下实时地检测和跟踪到多个人体.德国戴姆勒克莱斯勒公司也基于城市交通设计了UT4系统和智能Stop&Go系统。在对行人交通的检测与跟踪方面,中国科学院自动化研究所]在跟踪人体运动中采用了运动模型和关节人体模型对行人进行建模分析。

二、主流多目标追踪算法分析

当前主流的多目标追踪(MOT)算法主要由两部分组成：Detection+Embedding。Detection部分即针对视频，检测出每一帧中的潜在目标。Embedding部分则输出检测框中物体的外观特征（通常通过一个ReID网络抽取一个低维的向量，叫做embedding向量）。根据这两部分实现的不同，又可以划分为SDE系列和JDE系列算法。

SDE(Separate Detection and Embedding)系列算法完全分离Detection和Embedding两个环节，最具代表性的就是DeepSORT算法。这样的设计可以使系统无差别的适配各类检测器，可以针对两个部分单独调优。但是，由于其算法流程上的串联性质，导致速度，慢耗时较长，在构建实时MOT系统时将面临较大的挑战。

JDE(Joint Detection and Embedding)系列算法则完全是在一个共享神经网络中同时学习Detection和Embedding，使用一个多任务学习的思路设置损失函数并进行端到端的训练。代表性的算法有JDE和FairMOT。这样的设计兼顾精度和速度，可以实现高精度的实时多目标跟踪。

DeepSORT(SIMPLE ONLINE AND REALTIME TRACKING WITH A DEEP ASSOCIATION METRIC) 扩展了原有的SORT(Simple Online and Realtime Tracking)算法，增加了一个CNN模型用于在检测器限定的人体部分图像中提取特征，在深度外观描述的基础上整合外观信息，将检出的目标分配和更新到已有的对应轨迹上即进行一个ReID重识别任务。DeepSORT所需的检测框可以由任意一个检测器来生成，然后读入保存的检测结果和视频图片即可进行跟踪预测。

JDE(Joint Detection and Embedding)是在一个单一的共享神经网络中同时学习Detection任务和Embedding任务，并同时输出图像画面中的检测框位置和检测框内物体的外观特征。JDE原论文是基于Anchor-Baseed的YOLOv3检测器增加一个ReID分支学习Embedding，训练过程被构建为一个多任务联合学习问题，从而兼顾精度和速度。

FairMOT以Anchor Free的CenterNet检测器为基础，克服了Anchor-Based的检测框架中anchor和特征不对齐问题，深浅层特征融合使得检测和ReID任务各自获得所需要的特征，并且使用低维度ReID特征，提出了一种由两个同质分支组成的简单baseline来预测像素级目标得分和ReID特征，实现了两个任务之间的公平性，并获得了更高水平的实时多目标跟踪精度。

三、 项目实践

1、追踪算法的选择（DeepSORT）

基于上述三种目标追踪算法（DeepSORT、JDE、FairMOT）的性能特点以及我们自己的实践经验，最终选择DeepSORT作为基本方向。

选取依据如下。DeepSORT算法对目标检测模型以及特征提取模型要求较高，两个模型相辅相成。但是当两个模型并没有很好的效果时，追踪的效果也不会太差，也不会出现很大的行人id问题。JDE和FairMot目标追踪算法，是在一个共享神经网络中同时学习Detection和Embedding，但是检测与特征提取这两个功能同时达到一个比较好的效果是比较难的。在实验中我们发现，这种共享网络对于行人检测的效果还是很不错的，然而对于特征提取效果不理想，出现了频繁的切换id现象。当特征提取功能比较理想时，容易出现行人检测过拟合现象，路边的树等物体被误识别为“行人”。总而言之，对于JDE和FairMot追踪算法，目标检测和特征提取同时都取得一个较好的效果，是较为困难的。

最终，我们确定了基本路线，以DeepSORT算法为基础，训练出更好的行人检测模型以及特征提取模型来提升追踪算法的效果。

1.1DeepSORT算法简介

首先简单介绍一下SORT算法。SORT算法使用卡尔曼滤波处理逐帧数据的关联性，使用匈牙利算法进行关联度量。这种简单的算法在高帧速率下获得了良好的性能。但是由于SORT忽略了被检测物体的表面特征，因而在物体的区别较大、相似度较低的情况下SORT算法才能取得较为理想的效果。然而对于密集行人追踪来说，行人与行人之间的相似度较大，再加上人群比较密集，SORT算法无法取得较理的效果。基于以上理由，我们而选取了SORT的升级版本DeepSORT，该算法使用准确度更高的度量来代替关联度量，并使用一个简单的CNN网络提取行人特征。

在SORT中，我们直接使用匈牙利算法去解决预测的Kalman状态和新状态之间的关联度。当目标运动的不确定性较低时，马氏距离是一个很好的关联度量，但是在实际中，相机运动也会造成马氏距离不能进行精确的匹配，使得该度量失效。因此，我们整合第二个度量标准:对每一个BBox检测框 我们计算一个表面特征描述子 ，我们会创建一个gallery用来存放最新的 L个轨迹的描述子，然后我们使用第i个轨迹和第j个轨迹的最小余弦距离作为第二个衡量尺度。最终我们将马氏距离度量和最小余弦距离进行加权融合作为最终的度量。

1、使用检测框与跟踪器预测框之间的马氏距离描述运动关联程度：

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

其中，d_j表示第j个检测框的位置，y_i表示第i个跟踪器对目标的预测位置，S_i表示检测位置与平均跟踪位置之间的协方差矩阵。这意味着，马氏距离通过计算检测位置和平均跟踪位置之间的标准差对状态测量的不确定性进行了考虑。

2、第i个轨迹和第j个轨迹的最小余弦距离：$d^{(2)}(i,j) = \min\{1 - \boldsymbol{r}_j{}^{\mathrm{T}} \boldsymbol{r}_k^{(i)} \mid \boldsymbol{r}_k^{(i)} \in \mathcal{R}_i\}$

对每一个检测块d_j求一个特征向量r_j（通过CNN网络计算对应的128维feature向量r_j），约束条件是$||r_j||=1$），并计算第i个跟踪器的最近100个成功关联的特征集与当前帧第j个检测结果的特征向量间的最小余弦距离。

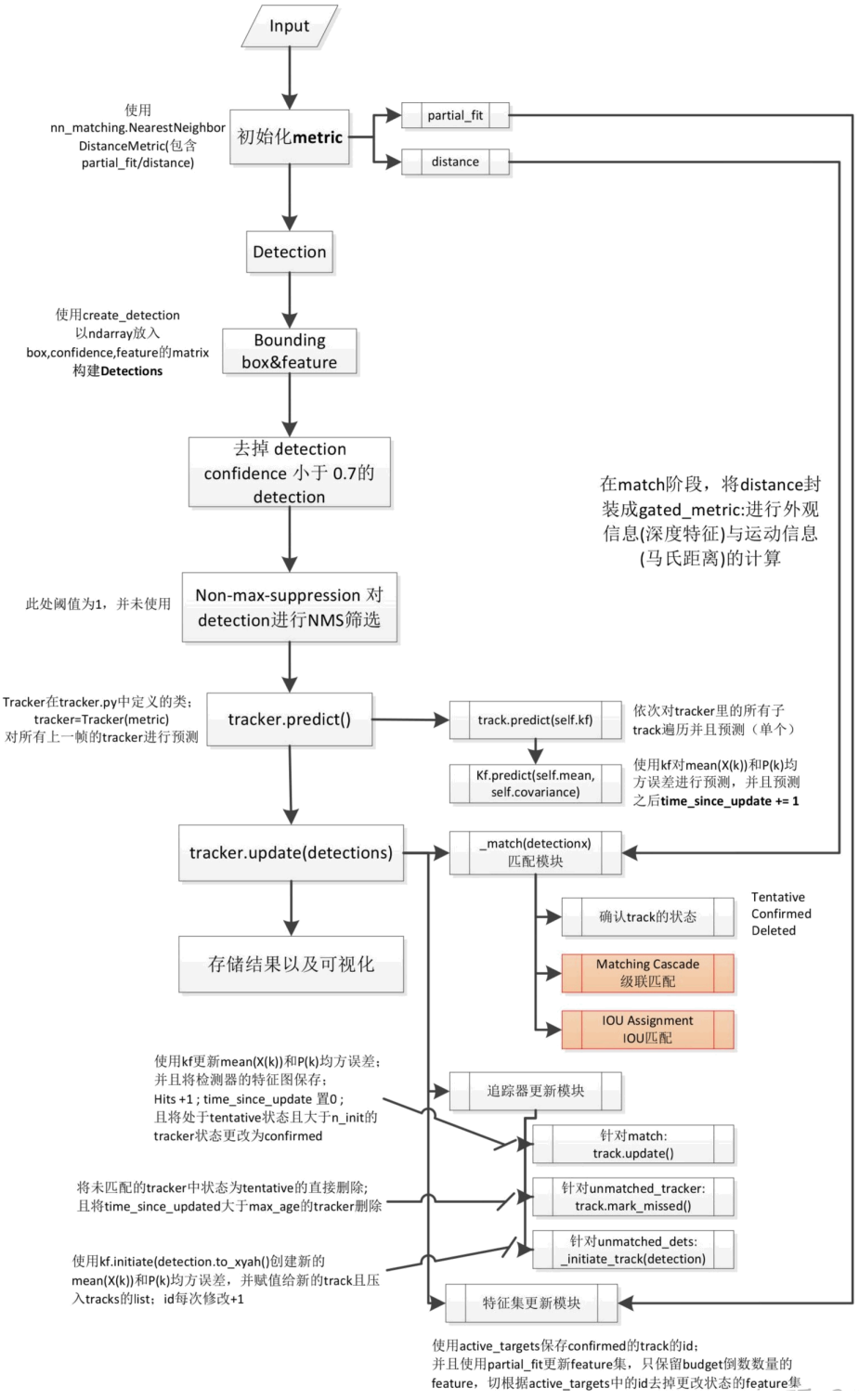
3、使用两种度量方式的线性加权作为最终的度量：$c_{i,j} = \lambda d^{(1)}(i,j) + (1-\lambda) d^{(2)}(i,j)$

总的来说，DeepSORT效果还是很明显的，使用CNN提取的特征进行匹配，大大减少了SORT中的ID switches, 实验结果表面，ID切换率减少了大约45%。

		MOTA ↑	MOTP ↑	MT ↑	ML ↓	ID ↓	FM ↓	FP ↓	FN ↓	Runtime ↑
KDNT [16]*	BATCH	68.2	79.4	41.0%	19.0%	933	1093	11479	45605	0.7 Hz
LMP_p [17]*	BATCH	**71.0**	**80.2**	**46.9%**	21.9%	434	**587**	7880	**44564**	0.5 Hz
MCMOT_HDM [18]	BATCH	62.4	78.3	31.5%	24.2%	1394	1318	9855	57257	35 Hz
NOMTwSDP16 [19]	BATCH	62.2	79.6	32.5%	31.1%	**406**	642	**5119**	63352	3 Hz
EAMTT [20]	ONLINE	52.5	78.8	19.0%	34.9%	910	**1321**	**4407**	81223	12 Hz
POI [16]*	ONLINE	**66.1**	79.5	**34.0%**	20.8%	805	3093	5061	**55914**	10 Hz
SORT [12]*	ONLINE	59.8	**79.6**	25.4%	22.7%	1423	1835	8698	63245	**60 Hz**
Deep SORT (Ours)*	ONLINE	61.4	79.1	32.8%	**18.2%**	**781**	2008	12852	56668	40 Hz

1.2DeepSORT算法流程图

DeepSORT算法可以分为四个核心部分，即轨迹处理和状态估计、相关性度量、级联匹配、深度特征描述器。DeepSORT的输入是行人检测模型的输出bbox。



2、关键点以及难点分析

首先，在分析训练数据集与测试数据集及对行人结果进行对比后，我们发现，对于密集的行人，由于人与人之间的遮挡面积较大，很容易导致漏检测。当人群过于密集，形成“一片”的时候，其漏检可能性大幅提升。其次，视频中较小的目标，对检测模型是一个较大的挑战。

其次，行人检测应用本身也对检测模型的精确度提出了更高的要求。在观察最终的追踪效果时我们发现，当检测模型精确度较低时，目标容易频繁的丢失，从而导致追踪时发生频繁的id切换。

最后，特征提取模型的优化也是一个重要问题。特征提取模型对bbox区域内的目标进行特征提取，其效果直接影响到DeepSort算法的匹配效果。当bbox提取到的特征不够准确及"区别度"不够明显时，很容易导致DeepSORT算法在追踪时发生id切换或者直接“跟丢”。

综上所述，本项目的关键技术如下：

1） 降低密集行人的漏检率。

2） 提高小目标的检测率。

3）提高检测模型的精确度。

4）提升体征提取模型的效果。

3、方法描述

为了提升行人检测模型的准确率。首先我们对MOT20训练集进行筛选，将置信度比较低的以及遮挡面积较大的bbox进行剔除，同时使用k-means方法计算出适用于MOT20的anchors；为了提高小目标的检测率，我们对模型的输入进行适当的调整，避免图片进行较大的缩小，减少信息的丢失。

引用

[1] Simple Online and Realtime Tracking with a Deep Association Metric (arxiv.org)

[2]多目标跟踪：SORT和Deep SORT - 知乎 (zhihu.com)

[3]寂寞你快进去-多目标追踪：DeepSORT_Paddle_

[4]AI算法修炼营的博客:多目标跟踪_| FairMOT：统一检测、重识别的多目标跟踪框架，_

[5]Startapi的博客deepsort算法原理以及代码解析_

[6]JDE:Towards Real-Time Multi-Object Tracking

[7] FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking (arxiv.org)

[8]PaddleDetection/README_cn.md at release/2.1_