Deep_Sort：具有深度关联指标的简单在线实时跟踪

1. Deep_Sort 概述

Tracking-by-Detection 是多目标跟踪领域的主要范式，流网络和概率图模型是该领域的流行框架,但是这种批处理方法无法直接用于在线场景。传统方法，包括多假设跟踪和联合概率数据关联滤波器，在逐帧的基础上进行了数据关联，在 Tracking-by-Detection 场景取得了较好的结果,但也带来了较高的计算和实现复杂度。

简单在线实时跟踪（SORT）是一种更简单的框架，在图像空间进行卡尔曼滤波，使用匈牙利算法进行逐帧数据关联，在高帧率下取得了良好的结果。这不仅突显了目标检测器性能对整体跟踪结果的影响,而且从从业者的角度来看也是一个重要的见解。

尽管 SORT 具有较好的性能，但由于其较低的状态估计准确度，其结果中具有较多的身份切换，在遮挡场景中具有明显的不足。因此，Deep_Sort 使用了结合运动信息和外观信息的关联度量,通过一个预训练的卷积网络来提取目标的外观信息，以克服 SORT 算法存在的问题。

2 技术细节

Deep_Sort 采用传统的单假设跟踪方法，使用递归卡尔曼滤波器和逐帧数据关联。其主要流程为，检测器获取视频当前帧中的目标框，卡尔曼滤波器根据当前帧的轨迹集合预测下一帧轨迹集合，预测轨迹与下一帧检测目标框进行匹配，卡尔曼滤波器更新匹配成功的轨迹。

2.1 轨迹处理与状态估计

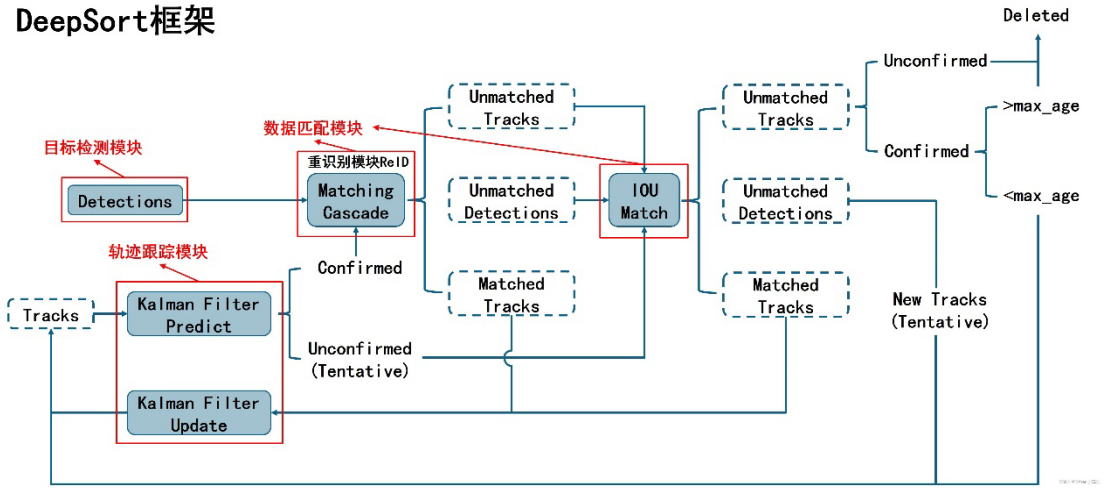
Deep_Sort 的轨迹处理和卡尔曼滤波框架与 SORT 算法基本一致,跟踪场景被定义为 8 维状态空间$(u, v, \gamma, h, \dot{u}, \dot{v}, \dot{\gamma}, \dot{h})$，分别表示锚框中心点坐标、纵横比、高度，及其各自的速度。使用具有匀速运动和线性观测模型的标准卡尔曼滤波器，取(u, v, γ, h)为目标状态的直接观测。在 Deep_Sort 中卡尔曼滤波的预测表达式为

$$\tilde{x}_k = A_k x_{k-1}$$

$$\tilde{P} = A_k P_{k-1} A_k^T + Q$$

其中，x_t表示系统在t时刻系统状态的均值向量，A_k表示k时刻的状态转移矩阵，P_k表示k时刻的协方差矩阵，Q表示系统的噪声矩阵，带~表示预测的估计矢量。

对每个轨迹记录自上次关联以来的帧数a_k，其在卡尔曼滤波预测期间递增，并在再次关联时重置为 0。超过最大年龄的轨迹视为离开场景并删除。新轨迹在前三帧中被视为暂定帧。



图表 1 Deep_Sort 模型架构图

2.2 分配问题

Deep_Sort 通过结合两个适当的度量来整合运动和外观信息。使用预测的卡尔曼状态和新到达的测量值之间的（平方）马氏距离来表示运动信息，其计算方式为

$$d^{(1)}(i,j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i)$$

其中y_i为轨迹预测的均值，S_i为协方差矩阵，d_i为检测框，通过χ^2分布设定$t^{(1)} = 9.4877$（95%置信区间）排除不合理的关联。使用检测和轨迹之间在外观空间的最小余弦距离作为第二个度量，其计算方式为

$$d^{(2)}(i,j) = \min\left\{1 - r_j^T r_k^{(i)} \,\middle|\, r_k^{(i)} \in \mathcal{R}_i\right\}$$

引入二进制变量来表示这个模型是否允许关联。两个指标相互补充，前者提供了运动信息，用于短期预测；后者考虑了外观信息，用于长期遮挡后的身份恢复。通过二者的加权平均结合两指标。

2.3 匹配级联

引入级联来解决一系列子问题，而不是通过求解指标来跟踪关联，用于减少

因卡尔曼滤波不确定性增加导致的关联模糊问题。使用匹配级联，优先考虑出现更频繁的目标，具体来说，在轨迹年龄 n 上迭代，以求解年龄增长的线性分配问题。这种匹配级联优先考虑年龄更小的轨迹，每次仅处理当前组与未匹配检测的关联，逐步减少不确定性对全局匹配的影响。在匹配的最后，在未确认和未匹配的轨迹上，运行 SORT 的联合关联交集，即进行 IOU 匹配，以应对突发外观变化，提到对初始错误的鲁棒性。

2.4 深度外观描述子

在大规模行人重识别数据集上训练一个 CNN，即 ReID 网络，使其适合在行人跟踪下进行深度度量学习，用于提取目标的外观信息，输出归一化的 128 维向量表示特征。ReID 网络的架构如下，在目标跟踪的流程是，对检测到的每个锚框的位置，去原始的图片中，截取对应的目标方框图，然后放入到 ReID 网络中，由于已经加载好了预先训练好的权重，所有，放入这个 ReID 网络后，进行前向推理后，经过 embedding 后，获得对应的每个锚框的特征向量，也就是对应着每个锚框的外观信息。

Name	Patch Size/Stride	Output Size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max Pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10		128
ℓ_2 normalization		128

图表 2 ReID 网络架构

3 模型表现

Deep_Sort 是基本 SORT 算法的改进版本，相比原始的 SORT 算法，添加了一

个 CNN 用于提取目标的外观信息，记录每个轨迹的年级进行匹配级联等。Deep_Sort 具有较高的可扩展性,并在多个指标上具有与 SORT 相近甚至优于 SORT 的性能。Deep_Sort 的优势在于，能够进行实时的目标跟踪，并且能够应对长时间的遮挡。

在 MOT16 基准下，使用预训练的 Faster-RCNN 作为检测器，在 7 个测试序列中评估了模型的性能。相比于 SORT，Deep_Sort 的身份切换此时从 1423 降低到 781，下降了将近 45%；其 MOTA 指标为 61.4，在在线方法中具有较强的竞争力；运行时间（频率）在 GPU 加速下达到 20Hz，展现出较好的实时性。由于保持目标在遮挡和未命中时的身份，轨迹碎片略有增加；此外，大部分被追踪的物体数量显著增加，大部分丢失的物体数量减少。可见，整合外观信息可以有效维持目标在长时间遮挡下的身份。