CVPR2020 《ArTIST: Autoregressive Trajectory Inpainting and Scoring for Tracking》

Abstract

在线多目标跟踪(MOT)框架的核心组件之一是将新的检测与现有的 tracklets 关联起来，通常通过一个评分函数来实现。尽管 MOT 取得了很大的进步，但是设计一个可靠的评分函数仍然是一个挑战。在本文中，我们**引入了一个概率自回归生成模型，通过直接测量小轨代表自然运动的可能性来对小轨建议进行评分**。我们的模型的一个关键特性是，它**能够在部分观察的情况下生成一个轨迹的多个可能的未来**。这使得我们**不仅可以对轨迹进行评分，还可以在检测器长时间无法检测到某些对象(例如，由于遮挡)的情况下，有效地维护现有的轨迹，通过对轨迹进行采样，来填补由于误检测而造成的缝隙**。我们的实验证明了我们的方法在几个 MOT 基准数据集上评分和嵌入轨迹的有效性。此外，我们还展示了我们的生成模型的通用性，通过使用它来产生具有挑战性的人类运动预测任务的未来表征。

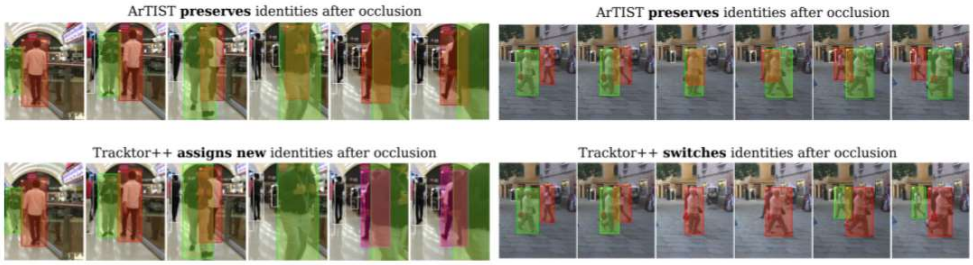

Fig. 1. Qualitative comparison of ArTIST (first row) and the SOTA Tracktor++ [7] (second row). These results evidence the effectiveness of our inpainting and scoring strategies at handling occlusions in complex and crowded scenes. Note that although Tracktor++ makes use of a person re-identification model trained on the MOT challenge, it failed to preserve the identities during the tracking process and assigned new identity (left) or switched identities (right) after an occlusion.

1 Introduction

在视频中跟踪多个目标是许多计算机视觉应用程序成功的关键，如运动分析、自动驾驶、机器人导航和视觉监视。近年来，随着目标检测技术的发展，tracking-by-detection 已成为多目标跟踪的一种实际方法;它包括首先检测单个帧中的目标，然后将这些检测与轨迹(称为轨迹条)联系起来。在这种背景下，现有的跟踪系统大致可以分为在线[36,43,58,52,21,70,99,50,93,19,7,20,85]，（小轨在每个时间步长上生长）,和 batch-based(又名离线)的[79,38,18,80,76,16,39,92,51]（小轨是在处理整个序列之后计算的）,通常在多假设跟踪(MHT)(11 日,38)框架。在本文中，我们开发了一个在线跟踪系统。

大多数检测跟踪框架的核心是一个评分函数，该函数的目的是在分配一个新的检测之后评估跟踪器的质量。定义这样一个评分函数的最常见的信息来源可能是外观。例如，受到行人再识别方法[35]的启发，依赖于单目标跟踪器的多目标跟踪算法[14,88,95,21,19,43][94,97,96,77,9]通常根据检测到的外观随时间的距离来设计评分函数。然而，**在多个目标跟踪场景中，外观可能不太可靠，这不仅是因为姿态变化和遮挡可能会显著影响它，还因为多个目标可能看起来非常相似**，例如在团队运动中。此外，这样的人员重新识别模块需要额外的训练，并被证明高度依赖于目标领域[29,22,26]。因此，许多方法[58、10、91、44、33、59、90、72、66、23]宁愿利用几何信息，也不受这些限制。为了提高鲁棒性，最近的研究[70,51,39,27]侧重于将外观与几何和社会信息相结合，使用递归神经网络(RNNs)学习评分函数。**正如他们在各自的论文中所承认的那样，虽然训练结果模型是有效的，但是需要大量的手工数据准备，例如创建一个数据集来训练一个好与坏的二进制 tracklet 分**

类器[39]或仔细平衡数据[51]，以及详细的训练过程。

与以前的方法不同，在本文中，我们建议**直接从跟踪数据中学习评分轨迹，而不需要额外的数据准备成本**。 为此，我们**设计了一个概率自回归模型显式学习自然轨迹分布的回归模型**。 这允许我们在**只给定一个边界框位置的序列时估计一个轨迹的可能性**。 因此，**我们不仅可以在指定新的检测后计算小轨的质量，而且还可以通过对已学习分布的采样来弥补小轨丢失的几个检测**。 据我们所知，我们的方法构成了填补由于探测器故障而造成的空白的第一次尝试。这样做是通过对自然人类轨迹分布的采样，给定一个观察到的部分轨迹，就可以自然地画出缺失的探测。

总而言之，我们的贡献如下:

(1)我们引入了一种概率自回归生成模型，能够通过直接测量轨迹代表自然运动的可能性来可靠地对轨迹进行评分。

(2)由于我们的模型学习了自然人体运动的分布，因此它能够生成多个可信的未来轨迹表示，并在包含漏检的轨迹内绘制轨迹。

(3)我们证明了我们基于几何的评分函数可以超越它所训练的数据集，允许我们在不同的情况下部署它，即使新领域与训练领域有很大的不同。这是因为**我们的评分函数有效地学习了自然运动的分布，而不依赖于外观、摄像机视角或特定的跟踪指标**。

(4)根据 MOT [7,86]的最新趋势，我们另外证明了当与[7]的边界框优化头结合使用时，我们的概率评分功能和 tracklet 修复方案的有效性，这使我们的性能优于最先进的。

(5)最后，我们评估模型在人类运动预测的艰巨任务中生成合理的未来表示的能力，即在给定观察序列的情况下预测未来 3D 人体姿势。

我们的模型名为 ArTIST，用于绘画中的自回归轨迹和跟踪评分，设计简单，并使用简单的负对数似然损失函数进行训练。

2 Related Work

在本节中，我们将重点讨论以前处理多对象跟踪任务的工作。简要回顾一下现有的人体运动预测方法，我们建议读者参考附录。

多目标跟踪在计算机视觉领域有着悠久的历史。随着该领域的大趋势，最近的跟踪系统都遵循一种深度学习的形式[17、39、78、99、20、7、85、51、76、70、58、67、47、82、27]。其中，**与我们的方法最接近的是使用递归神经网络的方法**，因此我们在这里重点关注它。最早的基于 rnn-based 的跟踪框架[58]旨在模仿贝叶斯滤波器的行为。为此，一个 RNN 被用来建模运动，另一个被用来计算小轨和新检测之间的关联向量。随着[58]在 RNNs 运动建模上的成功，提出了几种常见的运动仿真方法。在[70]中，三个 LSTMs 被用来模拟小轨的外观、运动和交互之间的时间依赖关系。在没有遮挡的情况下，使用单一的对象跟踪器跟踪场景中的不同对象。为了处理遮挡，这个单一的目标跟踪器被一个匈牙利算法[60]所取代，该算法基于 LSTMs 计算的分数/成本矩阵，将检测分配给 tracklets。类似地，在[67]中，提出了一种基于三流 lstm 的网络，将姿态、外观和运动信息结合起来。在[47]中，Siamese LSTM 被用来为场景中物体的位置和速度建模，以达到评分和分配的目的。在[82]中，Siamese LSTM 在运动和外观上被用于为匈牙利算法提供分数，该算法合并了短轨，最初通过卡尔曼滤波获得。在[27]中，使用两个递归网络来维护外部和内部的记忆，以建模运动和外观特征，从而计算分配过程中使用的分数。

虽然以前的算法是在线工作的，但在离线跟踪管道中也使用了递归模型。例如，在[51]中，LSTMs 用于在 MHT 框架中对小轨进行评分。为此，我们训练了一个利用外观、动作和社会信息的循环评分函数来优化 IDF1 评分代理[68]。正如作者所承认的那样，虽然这取得了

很好的性能，但是需要手动调整参数、增加数据并仔细设计训练过程。在[39]中，LSTMs 用于决定何时修剪 MHT 框架中的分支。这种方法称为双线性 LSTMs，它使用一个修改后的 LSTM 单元，将外观和动作作为输入。然而，基于外观和基于动作的模型首先分别进行了预训练。当学习较长范围依赖关系时，LSTM 单元处理外观信息的方式被证明对检测的质量很敏感。

通常，大多数性能最好的方法都使用外观信息[7、69、20、85、51、70、39、99、50]。然而，为了获得外观的最佳效果，需要在每个目标数据集上重新培训/微调外观模型。这限制了这些方法对新数据集的适用性。此外，在诸如 MOT17[57]这样的数据集中，所有的测试序列在训练序列中都有一个相似的对应序列，这大大简化了基于外观的模型任务，但并没有反映现实。

与这些方法不同，ArTIST 仅利用几何信息进行训练，而不依赖于目标数据集的外观。事实上,将在我们的实验中,ArTIST 甚至不需要看到目标数据集的几何信息,因为它只使用这些信息来学习一个分布在自然人类的运动,可以实现与任何年检数据集覆盖足够多样的场景,如动/静态照相机、摄像机视点不同,拥挤的场面。此外,相比之前的方法,使用多个流来处理不同形式(39 岁,70 年,27 日,67 年),操纵训练数据[39],或设计数据敏感和复杂的损失函数[51],我们的模型依赖于一个非常简单的周期性网络体系结构与一个简单的负对数似损失函数,可以直接在任何跟踪训练数据集没有任何数据操作或增加。

注意，许多方法，如 Social LSTM[1]和 Social GAN[32]，利用生成模型来编码人群运动的社会行为。由于他们专注于社会信息的建模，这些方法通常不能与 MOT 方法相比，因此超出了这项工作的范围。

3 Proposed Method

我们解决了在线跟踪场景中多个对象的问题。 我们的方法依赖于每个时间范围内的两个主要步骤: 对检测在现有 Tracklet 中的拟合程度进行评分, 并将检测结果分配给 Tracklet。以下，我们首先描述整个跟踪流程。 然后，我们深入研究评分功能和分配策略的细节。

3.1 Multiple Object Tracking Pipeline

像在许多其他在线跟踪系统中一样，我们遵循“按检测跟踪”范例[5]。 让我们考虑一个T帧的视频，其中，对于每个帧，我们提供了一组由例如 Faster-RCNN [30]，DPM [28]或 SDP [89]计算的检测值。这产生了由$D^{1:T} = \{D^1, D^2, \ldots, D^T\}$表示的整个视频的整体检测集，其中$D^t = \{d_1^t, \ldots, d_n^t\}$是时间$t$的所有检测的集合，$d_i^t \in \mathbb{R}^4$，即左上边界框角的 2D 坐标$(x, y)$，其宽度$w$和高度$h$。我们用第一帧$D^1 = \{d_1^1, \ldots, d_n^1\}$中的检测值来初始化第一组小轨迹$\mathrm{T}$。 从第二时间步长到视频结束，目标是通过将新的检测结果分配给相应的轨迹来扩展轨迹。 在整个视频中，可以创建新的运动轨迹，并将其附加到运动轨迹T的集合中，现有的运动轨迹也可以终止，并从 T 中删除。

为了在时间t上增长小轨迹T_j，我们通过将检测附加到T_j来为每个新检测计算小轨迹提议$\widehat{\mathrm{T}}_j^i$，并在评分模型下计算每个提议的可能性。 我们在时间t上为所有在T中的小轨迹计算这样的可能性，然后通过使用匈牙利算法求解线性程序将检测结果分配给小轨迹[60]。作为这种线性分配的结果, 某些检测将分配给某些小轨迹。其他检测可能未分配给任何小轨迹，因此可以用作新小轨迹的起点。 相反，某些小轨迹可能未分配任何检测，如果在一定时期内未分配它们，则可能导致其终止。

在此 MOT pipeline 的情况下，在本节的其余部分中，我们将描述 ArTIST 体系结构，该体系结构允许我们对每个 Tracklet 建议进行评分并在检测器由于例如遮挡或运动模糊而失败时修补 Tracklet。

3.2 ArTIST Architecture

ArTIST 是一种概率自回归生成模型，旨在明确学习自然小径的分布。 作为估计量，ArTIST 能够确定每个轨迹的可能性。 作为一种生成模型，ArTIST 能够通过从每个时间步长的估计分布中进行多项式采样来生成小轨迹的多个可能的连续性。

自回归框架中的小轨迹T_j的概率定义为:

$$p(\mathcal{T}_j) = p(b_1) \prod_{t=2}^{T} p(b_t \mid b_{t-1}, b_{t-2}, ..., b_1) \ , \tag{1}$$

其中b_t是在时间t处分配给T_j的边界框表示。为了对此建模，因为每个边界框都由其位置表示，该位置是一个连续变量，因此可以想到学习在给定先前位置的情况下在下一帧中回归该位置。 但是，回归并未明确提供自然小径的分布。 此外，回归只能产生轨迹的单个确定性连续性，而不能反映例如人类运动的随机性，对于该连续性，同等可能存在多个连续性。

为了解决这个问题，受 PixelRNN [61]的启发，我们建议离散化边界框位置空间。这使我们可以将$p(\mathrm{T})$建模为离散分布，其中每个条件分布都在等式 1 中。建模为具有 softmax 层的多项式分布。 但是，与像 PixelRNN 一样的生成模型通过独立于数据的量化（例如，通过分箱）使空间离散化，我们建议通过将运动速度（即 δx，δy，δw 和连续帧之间的 δh,由相应帧的宽度和高度标准化。）聚类来定义离散值的数据相关集合。 这使我们的输出空间移位和缩放不变。然后，我们将离散运动类定义为聚类质心。在实践中，我们使用非参数 k-means 聚类算法来获得 K 聚类。

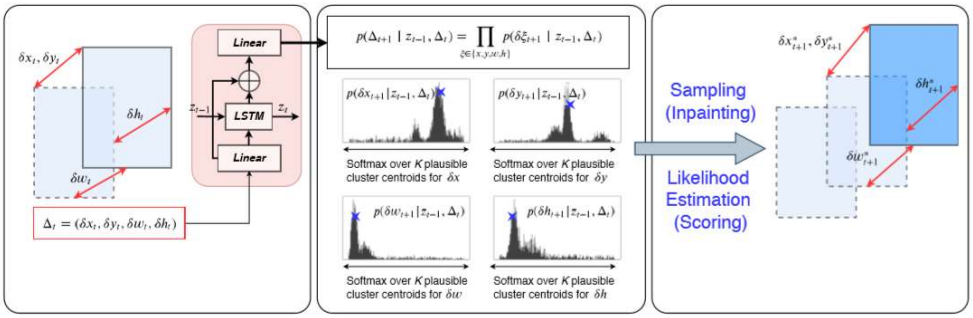

Fig. 2. ArTIST framework. (Left) ArTIST relies on a recurrent residual architecture to represent motion velocities. **(Middle)** Given the motion velocity representation at time t, ArTIST estimates a distribution over the next (i.e., at time $t+1$) bounding box location. **(Right)** Given the estimated distributions, one can either generate a new bounding box velocity $(\delta x^*_{t+1}, \delta y^*_{t+1}, \delta w^*_{t+1}, \delta h^*_{t+1})$ by sampling (indicated by blue stars over distributions) from the distributions, or evaluate the likelihood of an observed bounding box under the model.

我们的 ArTIST 体系结构如图 2 所示。ArTIST 依赖于递归残差体系结构来表示运动速度。在每个时间步长 t 处，以Δt =（δxt，δyt，δwt，δht）表示的运动速度作为输入。 给定该输入和在最后一个时间步z_{t-1}中计算出的隐藏状态，它随后预测了时间 t + 1 的运动速度分布，即$p(\Delta t + 1 \mid z_{t-1}, \Delta t)$。 这与等式 1 中的定义匹配。因为$z_t$包含有关所有先前时间步长的信息。

训练 ArTIST 仅需要跟踪数据集的可用性，并且我们可以利用单个 Tracklet，而无需同时考虑场景中的多个 Tracklet。 数据集中的每个小轨迹由一系列边界框定义，并通过相应帧的宽度和高度进行归一化，然后从中提取速度{Δ·}。 由于我们仅打算在下一时间步长估计边界框位置上的概率分布，因此我们使用简单的负对数似然损失函数训练模型。 从估计的分布

中，可以在给定检测边界框的情况下测量小轨迹的可能性，也可以为小轨迹补漏以填补由于缺少检测而导致的间隙。 我们在下面讨论这两种情况。

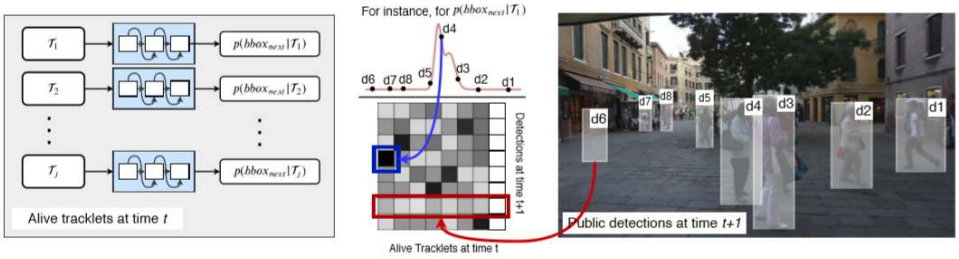

Fig. 3. Scoring tracklets with ArTIST. To assign each of the detections provided at time $t+1$ (right image) to a tracklet, we compute the probability distribution of the next bounding box for each tracklet $\mathcal{T}_1, ..., \mathcal{T}_j$, as shown on the left. Then, for each tracklet, we compute the negative log-likelihood of every detection, and take the negative log-likelihood of a detection d_i under the model for tracklet $\mathcal{T}_j$ as the cost of assigning d_i to $\mathcal{T}_j$. The Hungarian algorithm then takes the resulting cost matrix (middle) as input and returns an assignment. Here, the blue box indicates that d_4 is the best match for $\mathcal{T}_1$, resulting in the lowest assignment cost. The red box shows that d_6, which is a false detection, results in high assignment costs for all tracklets.

Tracklet Scoring. 为了计分在时间 t 处检测将成为小轨迹的延续的可能性，我们将与现有小轨迹对应的运动速度序列输入到 ArTIST，如图 3 所示。给定该序列，模型然后估计概率分布在时间 t 超出边界框的位置。然后，在给定的估计分布的情况下，我们将观察到的检测的可能性作为小径检测对的分数。具体而言，我们针对与先前观测值（或如果绘有前一个时间步长的则绘有边界框）的任何检测计算出Δ。然后，我们将最接近此Δ的质心的估计概率作为可能性。实际上，我们假设边界框参数即 δx，δy，δw 和 δh 独立。因此，我们有四组聚类，因此在每个时间步长估计了四个概率分布。然后，我们将边界框的可能性计算为组件概率的乘积，即

$$p(\Delta_{t+1} \mid z_{t-1}, \Delta_t) = \prod_{\xi \in \{x,y,w,h\}} p(\delta\xi_{t+1} \mid z_{t-1}, \Delta_t). \tag{2}$$

实际上，我们在日志空间中进行此操作，对概率的日志求和。

Tracklet Inpainting. 在现实世界中，由于例如遮挡或运动模糊，几帧的检测失败非常普遍。这样的故障使将来的检测与小轨迹的关联复杂化，因此可能导致错误的小轨迹终止。我们的方法通过在无法检测到的小轨迹上进行修补来解决此问题。让我们考虑在过去的几帧中未给小轨迹分配任何检测的情况。现在，我们尝试检查当前时间步的新检测是否属于该检测。为了让我们的模型计算出新观测值的可能性，我们需要访问上一个时间步长之前的完整边界框序列。为此，我们使用模型来修补缺失的观测值。具体来说，由于 ArTIST 每次都会估算未来边界框位置的分布，因此，如果在给定时间没有将边界框分配给 Tracklet，我们可以从前一时间步长估算的分布中采样一个。实际上，可以以递归方式进行采样，以创建完整的观察序列并修补缺失的边界框，从而使我们能够对新的检测结果进行评分。

为了说明运动本质上是随机的这一事实，尤其是对于人类而言，我们从估计的分布中抽样用于整个子序列的 S 个候选对象进行修复，并获得多个看起来可行的修复轨迹。 由于 ArTIST 本身仅依赖于几何信息，因此无法估计 S 修复选项中的哪些有效。 为此，我们使用了 Tracklet 拒绝方案，该方案允许我们做出更可靠的决定，即已修补的 Tracklet 不会偏离其实际方向。 我们在下面详细说明我们的小径拒绝方案。

Tracklet Rejection Scheme. 如上所述，当检测器在几帧内未能检测到物体时，我们的

模型能够修复缺失的观测值。我们的模型还考虑了人体运动的随机性，因此生成了多个可能的修复对象。要选择这些候选者之一，如果有一个要选择，我们将计算最后生成的边界框与场景中的所有检测结果的交集相交（IOU）。然后，如果 IOU 超过阈值，则选择具有最高 IOU 的候选者。然而，在某些情况下，一个候选者的最后生成的边界框可能与错误检测或对另一对象（即属于不同轨迹的对象）的检测重叠。**为了解决这些歧义，我们继续为 tTRS 帧的所有候选对象预测框。然后，我们不仅检测当前帧，还检测前面的 tTRS 帧，计算 IOU。然后，ArTIST 选择具有最大 IOU 总数的候选者**。这使我们可以忽略与错误检测或与在不同方向上移动的另一个对象的检测相匹配的候选。然而，这可能不足以消除所有情况的歧义，例如，属于附近且沿相同方向移动的其他小轨道的检测。我们在下面讨论的分配策略中处理这些情况。注意，实际上，我们使用较小的 tTRS，例如 2 或 3 帧，因此我们的方法仍可以视为在线跟踪。附录中的图 4 提供了我们的轨迹跟踪拒绝方案的更详细说明。

3.3 Assignment

要将检测结果分配给每个时间步长，我们使用匈牙利算法找到的线性分配。 匈牙利方法依赖于成本矩阵C，存储将每个检测分配给每个小轨迹的成本。 在我们的案例中，成本是 ArTIST 计算得出的负对数似然率。让我们用$C_{ij} = \log\ \mathrm{p}(< d_i, \mathrm{T}_j >)$) 表示将检测$i$分配给轨迹$j$的负对数可能性。然后，匈牙利算法通过求解$A^* = argmin_A \sum_{i,j} C_{ij} A_{ij}$返回关联的轨迹检测对的索引，其中$A \in [0,1]^{N \times M}$是分配概率矩阵，$N$是检测的数量，$M$是轨迹的数量。该矩阵满足约束条件$\sum_j A_{ij} = 1$、$\sum_i A_{ij} = 1$，$\forall j$。

在实践中，要考虑到我们对所修补的小轨迹的不确定性，我们运行了两次匈牙利算法。首先，仅使用通过实际检测获得其前一时间得分的小轨迹； 其次，使用通过修复获得的其余轨迹和未分配的检测。