

基于双向 RNN 的私家车轨迹重构算法

肖竹¹, 钱鑫¹, 蒋洪波¹, 蔡成林², 曾凡仔¹

(1. 湖南大学信息科学与工程学院, 湖南 长沙 410082;

2. 湘潭大学信息工程学院, 湖南 湘潭 411105)

摘要: 在复杂的城市环境中, 由于存在难以避免的 GNSS 定位信号中断现象以及车辆行驶过程中的误差累积, 易造成所收集的车辆轨迹数据不准确和不完备, 因此提出一种基于双向 RNN 的私家车轨迹重构算法, 使用了 GNSS-OBd 轨迹采集设备收集车辆轨迹信息, 利用多源数据融合实现双向加权轨迹重构。同时, 在轨迹重构模型中引入神经算术逻辑单元 (NALU), 加强深度学习的外推能力并确保轨迹预测的精度, 提高算法在应对城市复杂路段时轨迹重构的稳健性; 选取了实际城市路段进行了测试实验, 并和现有算法进行了对比分析。通过均方根误差 (RMSE) 以及 Google Earth 对重构轨迹进行可视化展示, 实验结果验证了所提算法的有效性和可靠性。

关键词: 私家车; 车辆定位; 轨迹重构; 循环神经网络

中图分类号: TN914

文献标识码: A

doi: 10.11959/j.issn.1000-436x.2020227

Bidirectional RNN-based private car trajectory reconstruction algorithm

XIAO Zhu¹, QIAN Xin¹, JIANG Hongbo¹, CAI Chenglin², ZENG Fanzi¹

1. College of Computer Science and Electronics Engineering, Hunan University, Changsha 410082, China

2. College of Information Engineering, Xiangtan University, Xiangtan 411105, China

Abstract: To address the problem that in the complex urban environment, due to the inevitable interruption of GNSS positioning signal and the accumulation of errors during vehicle driving, the collected vehicle trajectory data was likely to be inaccurate and incomplete. a bidirectional weighted trajectory reconstruction algorithm was proposed based on RNN neural network. The GNSS-OBd trajectory acquisition device was used to collect vehicle trajectory information, and multi-source data fusion was adopted to achieve bidirectional weighted trajectory reconstruction. Furthermore, the neural arithmetic logic unit (NALU) was leveraged with the purpose of enhancing the extrapolation ability of deep network and ensuring the accuracy of trajectory reconstruction. For the evaluation, real-world experiments were conducted to evaluate the performance of the proposed method in comparison with existing methods. The root mean square error (RMSE) indicator shows the algorithm accuracy and the reconstructed trajectory is visually displayed through Google Earth. Experimental results validate the effectiveness and reliability of the proposed algorithm.

Key words: private car, vehicle positioning, trajectory reconstruction, RNN

1 引言

随着我国工业化和城市化进程的持续加速, 机

动车数量持续迅速增长。特别地, 我国私家车占机动车总数比例高达 83%^[1]。私家车的高保有量一方面给交通运输、安全行驶和环境保护等带来巨大压

收稿日期: 2020-05-26; 修回日期: 2020-10-24

通信作者: 蒋洪波, hongbojiang2004@gmail.com

基金项目: 国家自然科学基金资助项目 (No.U20A20181, No.61732017); 湖南省重点研发基金资助项目 (No.2018GK2014)

Foundation Items: The National Natural Science Foundation of China (No.U20A20181, No.61732017), The Key Research and Development Project of Hunan Province (No.2018GK2014)

力,另一方面,私家车在行驶过程中产生了大量的轨迹数据。通过研究轨迹数据,分析与发掘深层次的车辆轨迹移动特征,这对认知人们的社会活动、优化城市道路资源配置等具有重要的研究价值,能为车联网、移动社交网络、智慧城市等应用领域提供有效服务^[2-3]。

在此背景下,准确获取车辆轨迹数据是上述新兴应用的技术基础。近年来,GNSS(global navigation satellite system)和移动感知等技术日趋成熟,推动了车辆轨迹数据采集、分析、处理及其相关应用领域的快速发展^[4-5]。

应该注意的是,实现车辆轨迹数据的收集需要使用低成本且易于推广的设备;同时,在复杂的城市环境中,由于存在难以避免的GNSS定位信号中断(GNSS outage)现象以及车辆行驶过程中的误差累积,易造成所收集的车辆轨迹数据不准确和不完备。因此,如何开发低成本且用户友好的轨迹收集设备,构建决策层面的异类信息融合模型、进而设计定位信号中断和误差累积时的轨迹预测与重构算法是车辆轨迹数据准确获取所面临的挑战性问题。

针对上述问题,本文通过集成GNSS接收模块和车载诊断(OBD, on-board diagnostics)系统读取,开发了一种GNSS-OBD设备来采集车辆轨迹相关数据,主要包括位置信息(GNSS获取)和运动状态信息(OBD读取器读取),该设备满足大规模部署场景中的轨迹收集的低成本需求,易于被用户接受,尤其适合于私家车用户。针对采集过程中轨迹数据误差和缺失的现象,本文提出了一种基于RNN(recurrent neural network)的双向加权轨迹重构算法——Bi-RNN-NALU(bidirectional recurrent neural network integrate neural arithmetic logic unit),将简单的循环神经网络(RNN, recurrent neural network)作为模型的主体,并将神经算术逻辑单元(NALU, neural arithmetic logic unit)引入提出的轨迹重构模型中,以解决GNSS信号中断发生在复杂路段的挑战性问题。通过对前向与后向2个方向分别进行轨迹重构,再根据误差累积随时间推移而增大的特点,给予轨迹预测不同的权重,来双向加权重构新的车辆轨迹。实验表明本文算法能有效降低轨迹中后期的误差,提高轨迹收集的整体精度。本文工作的技术贡献可归纳如下。

1) 开发了一种GNSS-OBD设备,利用低成

本的GNSS接收模块和OBD读取器,将车辆的多源数据进行融合。该设备具有规范的标准接口,能广泛适用于不同类型的私家车。基于该设备,已经开展了实际场景中大规模轨迹数据的采集工作。

2) 基于轨迹收集设备所获得的异类信息,即位置信息和运动状态信息,提出Bi-RNN-NALU算法,通过利用前向与后向2个方向的信息来加权重构新的车辆轨迹,利用GNSS接收良好时训练OBD读取的运动状态信息,消除其累积误差,基于此对GNSS信号中断时的位置信息进行重构。

3) 为进一步解决轨迹数据不准确和缺失及误差累积的问题,在轨迹重构模型中引入神经算术逻辑单元,加强深度学习的外推能力并确保轨迹预测的精度,同时提高了算法在应对城市复杂路段时轨迹重构的稳健性。

4) 利用GNSS-OBD设备进行了真实路段的测试实验。实验结果表明,即使在连续转弯和加速/减速驾驶等方向和速度频繁变化、GNSS信号中断60 s以上的复杂路段,Bi-RNN-NALU算法能将平均位置误差控制在10 m,且能有效降低轨迹中后期的误差,提高轨迹数据的整体精度。

2 相关工作

车辆轨迹是由机动车在城市道路移动时产生的路径^[6],利用定位技术能记录并收集车辆的轨迹数据。车辆轨迹尤其是私家车的轨迹信息通常是许多移动应用的基础^[7-8]。

一种简单的轨迹收集方案是使用智能手机来采集信息,在智能手机中嵌入GNSS模块和惯性传感器,使其具有定位导航功能。然而,智能手机并不适合大规模的私家车轨迹收集,主要原因如下:

1) 在日常驾驶中,私家车用户大都清楚去往目的地的路线,不需借手机手机的导航软件,因此用户没有长时间开启手机导航的习惯;2) 即使对于某段行程需要使用手机,所收集的车辆轨迹数据可能不完整且不连续,这是因为在某些情况下,如为了节省手机电量或其他目的,用户并不会在整个驾驶过程中使用手机导航。概而言之,使用智能手机来收集轨迹数据在实际应用中并不可行。

另一种解决方案是利用车载导航系统进行车辆跟踪和轨迹采集。然而,车载导航系统提取数据后并不能提供统一的数据接口类型。此外,

低配车型通常并不配置车载导航系统，从市场第三方进行安装，其成本都相对较高。因此，依靠车载导航系统进行大规模轨迹数据采集的可行性较低。

另外，许多研究使用 GNSS 接收机与外部惯性测量单元 (IMU, inertial measurement unit)^[9-11]或附加车载传感器^[12]融合的方法来收集车辆轨迹数据。在复杂城市场景下，多径效应及非视距等会导致 GNSS 信号中断，此时可以通过外部 IMU 传感器来采集相对独立的车辆运动信息。这些解决方案主要是直接利用多源数据融合的优势，但在诸如立交桥等复杂路段，效果不佳。此外，IMU 设备不利于大规模推广，这不可避免地限制了 IMU 数据的准确性，从而降低轨迹采集的性能。

3 轨迹采集与重构算法

3.1 GNSS-OBd 轨迹采集设备

本文开发了一种低成本、易于推广的 GNSS-OBd 轨迹数据采集设备，如图 1 所示。该设备主要包括 3 个模块，具体如下。1) GNSS 模块 ublox M8030，用于获得车辆经纬度的位置信息。2) OBd 读取器在不使用附加 IMU 器件的情况下，能够通过与车辆 OBd 接口相连，使用 CAN 总线来传输车内运动传感器所获取的速度、加速度和转向等运动状态信息。3) 通信单元，可将采集到的轨迹数据发送至数据中心（如图 1 所示）。所采集的车辆相关的信息包含车辆 ID、车辆起始和停止时间、车辆起始和停止位置（GNSS 坐标）、行驶里程、燃油消耗量、行驶时长、车辆位置、GNSS 信号状态及时间（判断 GNSS 是否中断）、车辆加速度和角速度、车辆速度和行驶方向、瞬时累积里程、发送机转数和油门踏板位置等。表 1 给出了部分原始轨迹数据样例。

在车辆启动时，GNSS-OBd 设备开始收集轨迹数据。该设备除了能实时采集车辆的轨迹数据，还能为主提供车辆防盗追踪、电子围栏、行程管理和汽车故障诊断等服务。因此基于该设备开展了实际场景中大规模轨迹数据的采集工作。此外，本文车辆轨迹采集过程中的数据传输是完全匿名的，并不收集关于车辆（车辆 ID 已被匿名为字符串）及驾驶员的信息。所采集的数据存储在合作的第三方服务提供商的服务器中，并受身份验证机制和防火墙的保护^[13]。

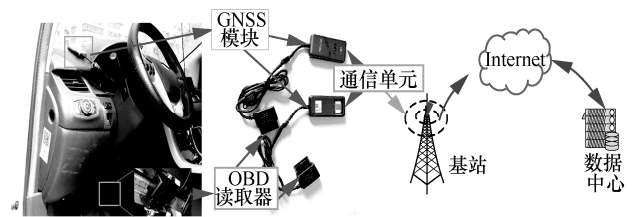


图 1 GNSS-OBd 轨迹采集设备

3.2 问题描述

如前文所述，与轨迹相关的应用高度依赖数据的有效性及其完整性。确保轨迹数据完整性的主要挑战之一是城市中无法避免的 GNSS 信号中断现象，这会导致车辆位置数据的错误或丢失，降低轨迹数据的应用价值，该问题对于细粒度的轨迹数据挖掘尤为严重^[14]。为解决该问题，本文借鉴数据融合的思想，即利用从 OBd 读取器获取的行驶状态，包括车速、行驶方向、加速度、角速度等，在 GNSS 信号中断时恢复轨迹的缺失数据。问题描述如下。

在 GNSS-OBd 设备中，车辆位置和行驶状态分别通过 GNSS 模块和 OBd 读取器获得。在时刻 t ，令 $s_t=(x_t, y_t)$ 表示车辆位置信息（经度和纬度）， $a_t^{obd}, \omega_t^{obd}$ 分别表示在 t 时刻 OBd 读取器读数中的加速度和角速度。

表 1

轨迹原始数据样例

车辆 ID	位置 (经纬度)	GNSS 信号状态	时间	加速度/ (m.s ⁻²)	角速度/ (rad.s ⁻¹)	速度/ (m.s ⁻¹)	方向(以正东方向 逆时针为参考)	里程/km	发动机转数/ (round.min ⁻¹)	油门踏 板位置
382704	东经: 114°2' 50" 北纬: 22°35' 40"	强 (9)	2018-07-05 09:08:51	0.103 782	0.042 725	18	150°	41 990.946	951	25.32
382704	东经: 114°2' 55" 北纬: 22°35' 32"	强 (7)	2018-07-05 09:10:10	-0.389 651	-0.009 16	14	120°	41 992.219	927	23.14
382704	东经: 114°3' 28" 北纬: 22°35' 40"	弱 (1)	2018-07-05 09:14:55	0.443 45	-0.061 03	11	100°	4 1994.083	885	19.23

已知 观测的车辆位置信息 $\mathbf{S}=[\mathbf{s}_{T-N+1}, \dots, \mathbf{s}_T, \tilde{\mathbf{s}}_{T+1}, \dots, \tilde{\mathbf{s}}_{T+L}, \mathbf{s}_{T+L+1}, \dots]$ $=[(x_{T-N+1}, y_{T-N+1}), \dots, (x_T, y_T), (\tilde{x}_{T+1}, \tilde{y}_{T+1}), \dots, (\tilde{x}_{T+L}, \tilde{y}_{T+L}), (x_{T+L+1}, y_{T+L+1}), \dots]$, 已知 OBD 读取器读数 $\mathbf{O}=[(a_{T-N+1}^{\text{obd}}, \omega_{T-N+1}^{\text{obd}}), \dots, (a_{T+L+1}^{\text{obd}}, \omega_{T+L+1}^{\text{obd}}), \dots]$ 。

这里的 $[\tilde{\mathbf{s}}_{T+1}, \dots, \tilde{\mathbf{s}}_{T+L}]$ 表示在轨迹收集过程中 GNSS 信号发生失效或中断时的车辆位置, $[(a_{T-N+1}^{\text{obd}}, \omega_{T-N+1}^{\text{obd}}), \dots, (a_{T+L+1}^{\text{obd}}, \omega_{T+L+1}^{\text{obd}}), \dots]$ 表示 OBD 收集的车辆运动信息, 包括加速度、角速度, N 表示 GNSS 信号未中断的持续时间, L 表示 GNSS 信号中断的持续时间。

求解 在 GNSS 信号中断的情况下重构轨迹数据 $\text{Traj}_{\text{rec}}=[\hat{\mathbf{s}}_{T+1}, \dots, \hat{\mathbf{s}}_{T+L}]=[(\hat{x}_{T+1}, \hat{y}_{T+1}), \dots, (\hat{x}_{T+L}, \hat{y}_{T+L})]$ 。其中, (x_t, y_t) 表示 t 时刻真实的车辆位置坐标; 在 $t = T+1$ 时刻到 $t = T+L$ 时刻之间, GNSS 信号发生中断, 车辆轨迹缺失; (\hat{x}_t, \hat{y}_t) 表示预测的 t 时刻车辆位置坐标。

3.3 算法框架

车辆轨迹缺失问题的根源是 GNSS 信号失效或中断, 因此无法提供轨迹收集过程中所需的车辆的位置信息。值得注意的是, 依据时间维度, 前后的轨迹信息是彼此相关的, 不仅可以由前向的轨迹信息推测出后向的轨迹, 同样可以由后向的轨迹信息反向推测前向的轨迹。受此启发, 本文提出 Bi-RNN-NALU 算法来双向加权重构缺失的轨迹, 算法的总体框架如图 2 所示。原始的轨迹数据中包含 GNSS 信号接收状态, 指示在轨迹采集期间 GNSS 信号的强弱。根据数据融合的思想, 当 GNSS 信号可用时, 将包括 GNSS 数据 (例如车辆位置数据) 和 OBD 读取器读数 (例如车辆运动信息) 在内的收集的轨迹信息加入训练过程中, 以基于 GNSS 轨迹位置来学习 OBD 读取器读数的误差。具体来说, 方案首先基于采集的 GNSS 位置数据和 OBD 读取数据计算 GNSS 信号中断前每一时刻对应 OBD 读取器读数误差。接着, 通过 Bi-RNN-NALU 模型, 分别从前向和后向 2 个方向对进行融合后的 OBD 读取器读数误差进行训练, 并分别预测 GNSS 信号失效或中断时的 OBD 读取器读数误差。然后, 基于航位推测的思想, 结合 2 个方向上预测的 OBD 读取器读数误差, 分别重构缺失的车辆轨迹。最后, 结合 2 个方向的重构轨迹进行加权来重构新的车辆轨迹。

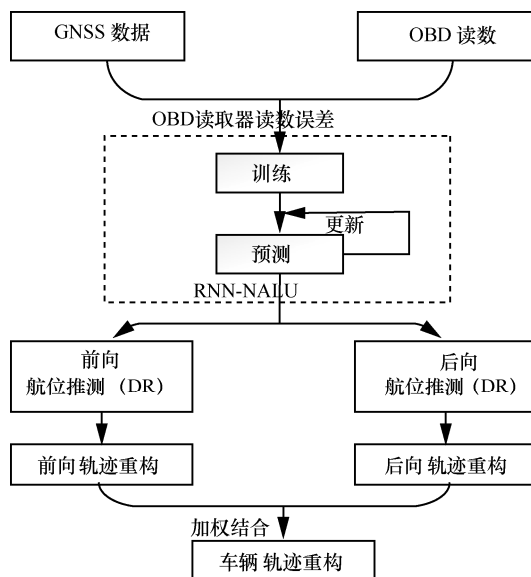


图 2 Bi-RNN-NALU 算法框架

4 Bi-RNN-NALU 算法的构造

4.1 RNN 和 NALU

1) RNN

深度神经网络可以通过使用几乎“原始”数据进行预测, 从而能够将其更好地归纳为不同的输入模式。RNN 是一种特殊的人工神经网络模型^[15]。与其他深度神经网络不同, 实际上 RNN 在空间上并不“深”, 最简单的 RNN 只有一个隐藏层。RNN 的基本特征是在每一步都包含一个反馈连接, 因此可以在时间维度上进行扩展, 从而在时间维度上形成一个“深”的神经网络。

对于单个隐藏层的 RNN, 可表述为

$$\mathbf{h}_t = f_h(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h) \quad (1)$$

$$\mathbf{o}_t = f_o(\mathbf{W}_{ho}\mathbf{h}_t + \mathbf{b}_o) \quad (2)$$

其中, \mathbf{x}_t 和 \mathbf{o}_t 分别表示 t 时刻的输入和输出向量, \mathbf{h}_t 表示 t 时刻的隐藏记忆向量, $\mathbf{W}_{xh}, \mathbf{W}_{hh}, \mathbf{W}_{ho}$ 分别表示 $\mathbf{x}_t, \mathbf{h}_{t-1}, \mathbf{h}_t$ 对应的线性转换权重矩阵, $\mathbf{b}_h, \mathbf{b}_o$ 分别表示 $\mathbf{h}_t, \mathbf{o}_t$ 的偏置向量, f_h 和 f_o 均表示非线性的激活函数。

RNN 由于其特殊的结构特点, 非常适合于序列信号的建模。理论上, 式(1)和式(2)中描述的最简单 RNN 模型可以处理任意长度的序列信号, 因为其在一系列具有挑战性的预测问题上都取得成功, 近年来在数据预测研究中得到了广泛应用。

2) NALU

NALU^[16]是一种通过原始算术运算, 能够系统

地学习表示和操纵数字的模型。

NALU 主要由 2 个神经累加器 (NAC, neural accumulator) 及一个 Sigmoid 型门组成, 其中, NAC 是其输入向量的线性变换, 如式(3)所示。

$$\text{NAC: } \mathbf{a} = \mathbf{W}\mathbf{x} \quad (3)$$

其中, \mathbf{x} 是输入向量; \mathbf{a} 是输出向量; \mathbf{W} 是经过连续且可区分的参数化, 得到的一个无约束的权重参数。 $\mathbf{W} = \tanh(\hat{\mathbf{W}}) \odot \sigma(\hat{\mathbf{M}})$, 它的这种形式便于使用梯度下降进行学习, 并生成矩阵, 其元素保证在 $[-1, 1]$ 之间, 并偏向地接近 -1 、 0 或 1 。NAC 的结构使它具有支持累加的能力, 可以实现线性的外推。NALU 能够学习 2 个 NAC 子单元之间的加权总和, 一个 NAC 子单元能够进行加法和减法运算, 另一个子单元能够进行乘法、除法和幂运算, 如 \sqrt{x} 等, 通过 Sigmoid 型门 \mathbf{g} 来实现控制, 如果加/减子单元的输出值的权重为 1 (开), 则乘/除子单元格为 0 (关), 反之亦然。第一个 NAC 计算累加向量 \mathbf{a} , 该向量存储 NALU 的加/减操作结果, 计算方式与原始 NAC 相同 (即 $\mathbf{a} = \mathbf{W}\mathbf{x}$)。第二个 NAC 在对数空间中运行, 因此能够学习乘法和除法, 将结果存储在 \mathbf{m} 中, 如式(4)所示。

$$\mathbf{m} = \exp \mathbf{W} (\log(|\mathbf{x}| + \varepsilon)) \quad (4)$$

最终 NALU 的输出 \mathbf{y} 表示为

$$\begin{aligned} \text{NALU: } \mathbf{y} &= \mathbf{g} \odot \mathbf{a} + (\mathbf{1} - \mathbf{g}) \odot \mathbf{m} \\ \mathbf{g} &= \sigma(\mathbf{G}\mathbf{x}) \end{aligned} \quad (5)$$

其中, \mathbf{G} 是线性转换权重矩阵, ε 可以防止出现 $\log 0$ 的情况。因此, NALU 单元可以学习由乘法、加法、减法、除法和幂函数组成的算术函数, 可以外推出训练期间观察到的范围之外的数值。

4.2 Bi-RNN-NALU 算法原理

3.3 节概述了本文算法的框架, Bi-RNN-NALU 算法分为 3 个步骤, 具体如下。

1) 第一步的目的是学习 GNSS 设备正常运行时与轨迹数据相对应的 OBD 读取器读数的误差 (即训练过程), 并以此预测 GNSS 信号失效时对应的 OBD 读取器读数误差 (即预测过程)。当 GNSS 完全可用时, 通过 GNSS 获得的轨迹位置对于许多与轨迹相关的应用 (例如驾驶员行为调查/分类和旅行时间/距离估计) 足够准确。因此, 本文利用基于 GNSS 采集的 N 个轨迹点, 推断出车辆的准确加速度 a_t^{gnss} 和角速度 ω_t^{gnss} 。此外, 利用航位推测 (DR, dead reckoning) 的思想, Bi-RNN-NALU

算法借助初始 GNSS 点 (请注意, 此时 GNSS 信号仍然可用) 结合 OBD 读取器读数 a_t^{obd} 和 ω_t^{obd} 来获得轨迹位置。由于车载运动传感器固有的噪声, 更重要的是累积的误差, 基于 OBD 读取器读数的轨迹是不准确的 (参见 5.2 中 DR-OBD 的实验结果)。为了解决这个问题, 在训练阶段, Bi-RNN-NALU 算法根据 GNSS 校准 OBD 读取器读数的误差。Bi-RNN-NALU 算法是学习 GNSS 信号中断前 ($a_t^{\text{gnss}}, \omega_t^{\text{gnss}}$) 与 ($a_t^{\text{obd}}, \omega_t^{\text{obd}}$) 之间的差距, 并预测 GNSS 信号中断时它们的差距。时刻 t 的差距定义为 OBD 读取器读数误差 $\mathbf{e}_t = (e_t^a, e_t^\omega)$, 其中, $e_t^a = a_t^{\text{gnss}} - a_t^{\text{obd}}$, $e_t^\omega = \omega_t^{\text{gnss}} - \omega_t^{\text{obd}}$ 。这里有 2 次预测, 包括前向预测, 利用 GNSS 信号中断前的 OBD 读取器读数误差预测 GNSS 信号中断时的读数误差; 后向预测, 利用 GNSS 信号中断后的 OBD 读取器读数误差预测 GNSS 信号中断时的读数误差。

每一个轨迹点与先前的轨迹点是有关联的, 具体来说 GNSS 信号中断期间车辆的运动信息与 GNSS 信号中断之前的运动信息高度相关。在实际应用中, 运动信息中会有各种误差以及由此产生的误差累积, 即本研究中的 OBD 读取器读数不可避免地会出现误差和误差累积的情况。为了解决该问题, 在 GNSS 信号可用时, Bi-RNN-NALU 算法通过一个基于 RNN 的神经网络模型来学习误差的特征。此外, 本文研究还观察到对于车辆的运动状态信息而言, 例如从不同路段所获得的加速度和角速度具有不同的特征, 能够显著影响轨迹重构的性能。特别是在复杂路段, 例如立交桥路段, 转向和加减速操作较多, 驾驶状态变化较大, 以及车辆状态变化较长, 使立交桥路段的 OBD 读取器读数误差超过了训练集的正常范围。这可以看作一种异常变化, 传统的 RNN 结构无法很好地解决该问题。

本文利用 NALU 作为附加单元加入基于 RNN 的轨迹重构模型, 以便在训练数值范围的内部和外部获得更好的泛化。为此, 本文设计了一个叠加的 RNN 和 NALU 相结合的神经网络结构, 除了输入层和输出层之外, 网络分为四层, 包括 2 个 RNN 层, 其中一个 RNN 层有 128 个单元, 另一个有 64 个单元; 一个 NALU 层, 用于应对复杂路段的轨迹预测问题, 提高对复杂路段的稳健性, 该层单元数

为 64; 一个 Dense 层, 其神经元数目与输出 OBD 读数误差向量长度大小相同。NALU 赋予网络强大的外推能力, 可以应对由复杂路段 (如立交桥) 引起的“异常”速度及方向变化。图 3 给出了 Bi-RNN-NALU 算法第一步的结构示意。

2) 第二步是基于前后向预测的 OBD 读取器读数误差和车辆位置之间的关系, 通过航位推测的数学计算过程重构轨迹。这里有 2 次航位推测的过程: ① 前向航位推测重构轨迹, 利用前向预测的 GNSS 信号中断时的 OBD 读取器读数误差进行前向航位推测; ② 后向航位推测重构轨迹, 利用后向预测的 GNSS 信号中断时的 OBD 读取器读数误差进行后向航位推测。

前向航位推测重构轨迹通过 OBD 读取误差的预测值间接预测位移, 达到重构轨迹的目的。在 GNSS 信号中断的持续时间 $t = T + 1, \dots, T + L$, 轨迹重构过程可以表示为

$$(\hat{a}_{t-1}^{\text{gnss}}, \hat{\omega}_{t-1}^{\text{gnss}}) = (a_{t-1}^{\text{obd}}, \omega_{t-1}^{\text{obd}}) + \hat{e}_{t-1} \quad (6)$$

$$(\hat{v}_t^{\text{gnss}}, \hat{\alpha}_t^{\text{gnss}}) = (\hat{v}_{t-1}^{\text{gnss}}, \hat{\alpha}_{t-1}^{\text{gnss}}) + (\hat{a}_{t-1}^{\text{gnss}}, \hat{\omega}_{t-1}^{\text{gnss}}) \quad (7)$$

其中, \hat{v}_t^{gnss} 和 $\hat{\alpha}_t^{\text{gnss}}$ 分别表示在时刻 t 的车辆的预测速度和预测方向角。

因此, 2 个相邻轨迹点之间的预测位移可通过式(8)来计算, 通过式(9)来重构轨迹。

$$\hat{d}_t = \left[\hat{v}_t^{\text{gnss}} \cos\left(\frac{\hat{\alpha}_t^{\text{gnss}}}{180} \pi\right), \hat{v}_t^{\text{gnss}} \sin\left(\frac{\hat{\alpha}_t^{\text{gnss}}}{180} \pi\right) \right] \quad (8)$$

$$\hat{s}_t = \hat{s}_{t-1} + \hat{d}_t \quad (9)$$

后向航位推测重构轨迹与前向航位推测的过程相似, 只是时间维度从后向前反向推测。Bi-RNN-NALU 算法的第二步的具体结构如图 4 所示。

3) 第三步是基于前向重构轨迹结合后向重构轨迹加权重构新的轨迹, 具体结构如图 5 所示。

由于车载传感器的累计误差会导致随时间推移的车辆位置误差越来越大, 因此在轨迹重构这样的时间序列预测的问题中, 离已知轨迹点时间上越近的重构轨迹点的可信程度更高。因此, 本文定义置信度来表示不同时刻重构轨迹点的可信程度, 假设 GNSS 信号中断的时间是 $t = T + 1, \dots, T + L$, 在 $t = T + i$ 时刻, 前向轨迹重构的车辆位置的置信度为

$$p_i^f = 0.5 \left[\cos\left(\frac{\pi}{L-1}(i-1)\right) + 1 \right], 1 \leq i \leq L \quad (10)$$

其中, 第一个重构点 $p_1 = 1$, 最后一个重构点 $p_L = 0$, 与之相对的后向轨迹重构的车辆位置的置信度 $p_i^b = 1 - p_i^f$ 。

令前向重构轨迹为 $\text{Traj}_f = (\hat{s}_{T+1}^f, \hat{s}_{T+2}^f, \dots, \hat{s}_{T+i}^f, \dots, \hat{s}_{T+L}^f)$, 后向重构轨迹为 $\text{Traj}_b = (\hat{s}_{T+L}^b, \hat{s}_{T+L-1}^b, \dots, \hat{s}_{T+i}^b, \dots, \hat{s}_{T+1}^b)$, 基于置信度给轨迹的每个点分配权重系数, 在 $t = T + i$ 时刻, 前向轨迹权重系数 $w_i^f = p_i^f$, 后向轨迹权重系数 $w_i^b = p_i^b$, 即根据可信程度不同, 置信度大的权重系数大, 反之则小, 双向加权后的重构轨迹为

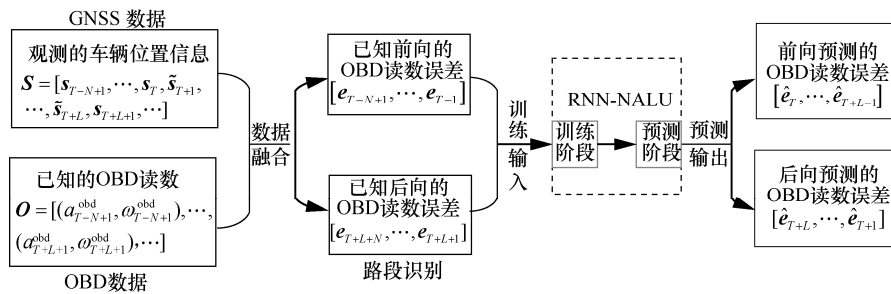


图 3 Bi-RNN-NALU 算法第一步的结构

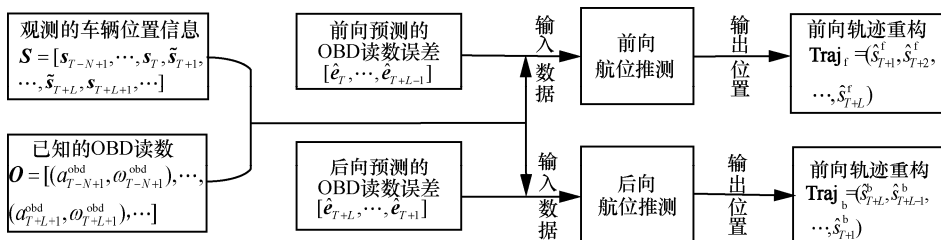


图 4 Bi-RNN-NALU 算法第二步的结构

$$\text{Traj}_{\text{Bi-rec}} = (\hat{s}_{T+1}, \hat{s}_{T+2}, \dots, \hat{s}_{T+L}) = (w_1^f \hat{s}_{T+1}^f + w_1^b \hat{s}_{T+1}^b, w_2^f \hat{s}_{T+2}^f + w_2^b \hat{s}_{T+2}^b, \dots, w_L^f \hat{s}_{T+L}^f + w_L^b \hat{s}_{T+L}^b) \quad (11)$$

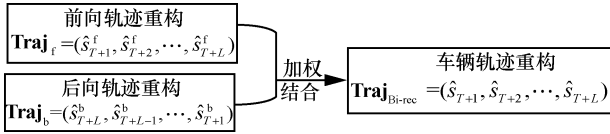


图 5 Bi-RNN-NALU 算法第三步的结构

5 实验结果及分析

5.1 道路测试

为了验证 Bi-RNN-NALU 算法的性能，本文进行了真实道路场景下的轨迹数据采集与测试。图 6 描绘了道路测试实验的车辆轨迹，该轨迹采集于长沙市区，其中椭圆内的线段表示轨迹缺失的路段，包括普通交叉路口转弯、直线、直角转弯、立交桥等路段场景。具体来说，驾驶安装了图 1 所示设备的测试车辆，获得图 6 所示轨迹，轨迹数据采集的采样率为 1 Hz。在道路测试中，当车辆通过具有良好 GNSS 信号的开阔区域时，GNSS 模块的精度通常为 15 m。

5.2 对比方法及结果分析

1) 对比算法

Bi-RNN-NALU。算法中网络使用 5 个时间步长的窗口进行训练，表示过去 5 s 的观测值，每预测下一秒的 OBD 读取器读数误差，此窗口就更新一次（即更新频率为 1 Hz），将预测的新的 OBD 读取器读数误差加入新的输入窗口，继续预测下一秒 OBD 读取器读数误差。考虑到车辆驾驶过程中的短期相关性，即包含在轨迹数据中车辆状态是与最近的轨迹相关的，训练不需要很长的轨迹。在这里，本文使用 GNSS 信号中断前后 15 s 的轨迹数据作为训练集，当车载 OBD 位置终端设备的采样率设置为 1 Hz 时，2 个方向各自输入历史的 15 个

轨迹点作为训练数据。

DR-OBD。如 4.2 节所述，通过运用航位推算的思想^[17]，主要基于 OBD 读取的车辆运动信息，结合 GNSS 信号中断前的初始 GNSS 位置数据恢复的车辆轨迹位置。

DR-RNN/DR-RNN-NALU。与 Bi-RNN-NALU 类似，轨迹重构算法的前两步单独前向预测重构轨迹，但在算法的第一步使用了不同的神经网络结构，将应用 RNN 的方法在以下部分称为“DR-RNN”。特别地，将应用 RNN-NALU 网络的方法在以下部分称为“DR-RNN-NALU”，此算法仅用于立交桥路段对比，验证 NALU 在复杂路段的作用，参数设置与本文算法一致。

支持向量回归 (SVR, support vector regression)^[18]。当 GNSS 信号稳定有效时，训练学习车辆移动信息与传感器的运动状态数据之间的内在联系，得到一个回归模型。当 GNSS 信号无效或中断时，利用已建立的回归模型，结合此时的车辆运动状态数据，预测对应的车辆移动信息。训练集取 GNSS 信号中断前 30 s 的数据。

高斯过程回归 (GPR, Gaussian process regression)^[19]。同 SVR 类似，建立回归模型预测车辆移动信息。训练集同样取 GNSS 信号中断前 30 s 的数据。

实验验证使用的软硬件环境说明如下。CPU 为 Intel(R) Core(TM) i5-8300H CPU @2.30GHz 2.30 GHz，内存为 8 GB，显卡为 NVIDIA GTX1050 Ti，软件为 Pycharm。

2) 结果分析

具体的车辆驾驶场景和 GNSS 信号失效时长参

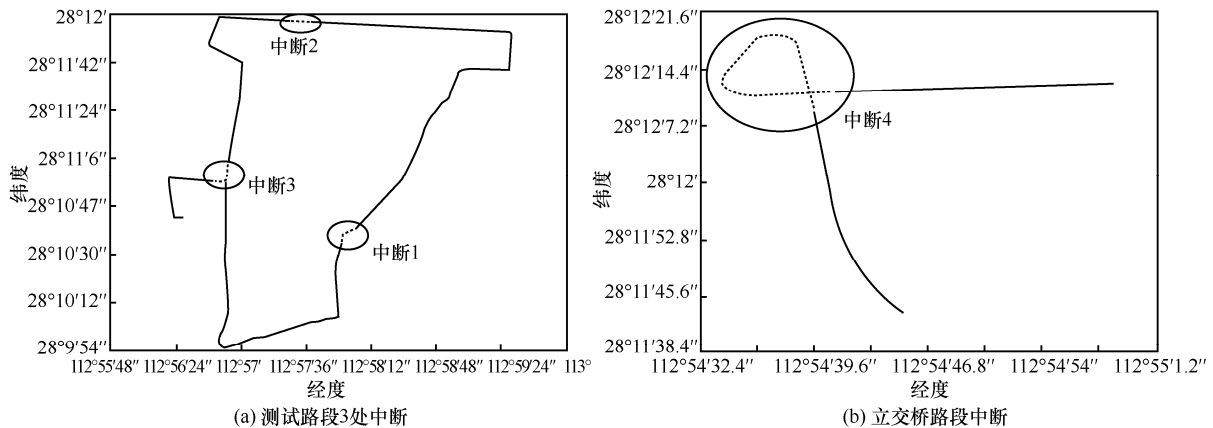


图 6 车辆轨迹

数如表 2 所示。

表 2 道路测试参数设置

测试阶段	GNSS 信号失效时长/s	场景
测试 1	40	交叉路口转弯
测试 2	40	直线路段
测试 3	30	直角转弯
测试 4	66	立交桥路段

图 7 给出了在交叉路口转弯路段各种算法的重构轨迹及其位置误差的结果，GNSS 信号失效时长为 40 s。这个交叉路口近似“直角转弯”，车辆的运动状态也与直角转弯类似，都是低速进入，经过路口后加速离开。在该路段，由于速度变化频繁，来自运动传感器的加速度不准确，角速度较为准确，导致 DR-OBD 的重构轨迹形状与真实轨迹相似但更短。从图 7(a) 可以看出 Bi-RNN-NALU 算法重构的轨迹与真实轨迹最为接近，特别是在轨迹末端，与真实轨迹末端近乎重合。仅前向轨迹重构的 DR-RNN 的效果也不错，SVR 可能由于累计误差过大，导致在轨迹末端无法正确拟合特征，方向偏移较大，而 GPR 的在轨迹中后期近乎失效，轨迹偏移很大。如图 7(b) 所示，Bi-RNN-NALU 算法在各轨迹点的位置误差始终保持最低，相比之下其他算法在轨迹末端的位置误差较大。

图 8 给出了在直线路段各种方法的重构轨迹及其位置误差的结果，GNSS 信号失效时长为 40 s。从图 8(a) 可以看出，SVR 和 GPR 重构轨迹方向在速度和方向上相比真实轨迹误差都很大，而从 DR-OBD 可以看出在该路段车内运动传感器的固有噪声和误差积累较大，结合图 8(b) 可以看到 DR-RNN 表现良好，而 Bi-RNN-NALU 算法的效果在 DR-RNN 的基础上则有明显提升，显著减小了 DR-RNN 在轨迹末端的误差。

图 9 给出了在直角转弯路段各种方法的重构轨迹及其位置误差的结果，GNSS 信号失效时长为 30 s。车辆在经过直角转弯路段时，低速进入，加速离开。在该路段，由于速度变化频繁，来自运动传感器的加速度不准确，但低速短时间经过弯道使角速度较为准确，这些情况导致 DR-OBD 的重构轨迹形状与真实轨迹相似但较短，也使大多数方法重构的轨迹能够与真实轨迹形状相似，其中，

Bi-RNN-NALU 表现出最好的性能，DR-RNN 稍差。SVR 和 GPR 的结果比较差，在形状上偏离真实轨迹，拟合效果不佳。

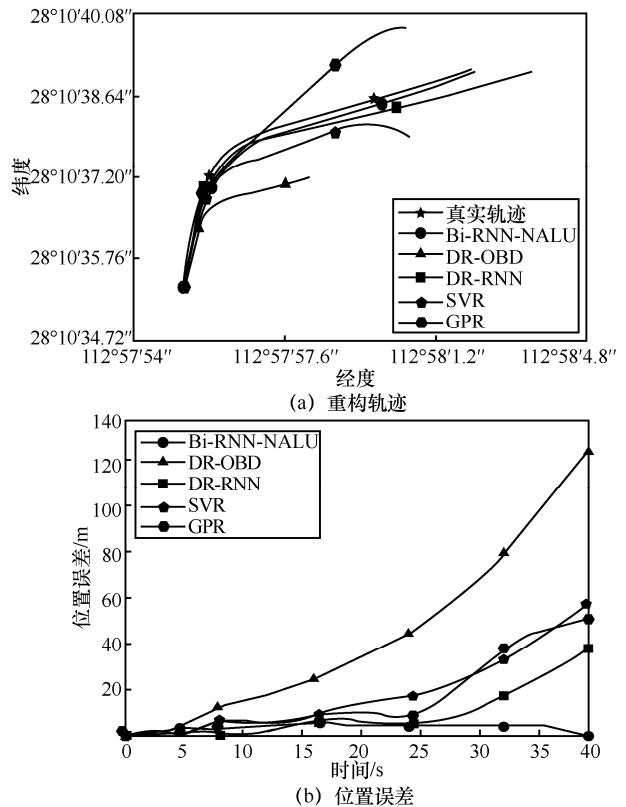


图 7 交叉路口转弯路段重构轨迹及其位置误差

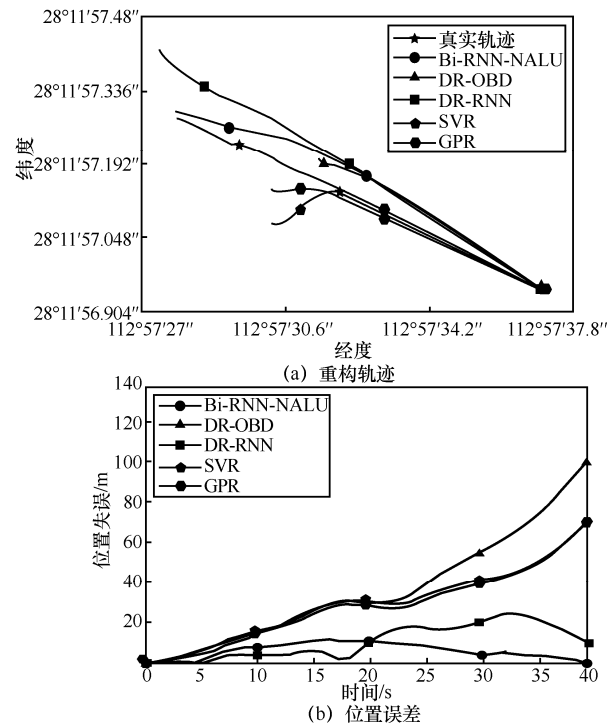


图 8 直线路段重构轨迹及其位置误差

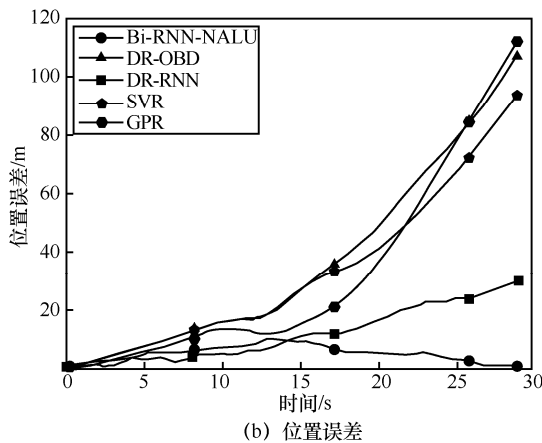
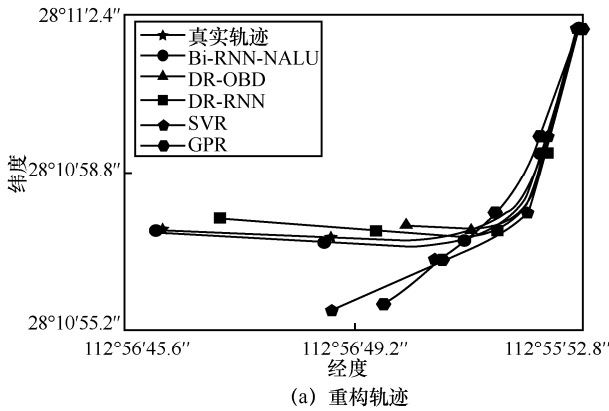
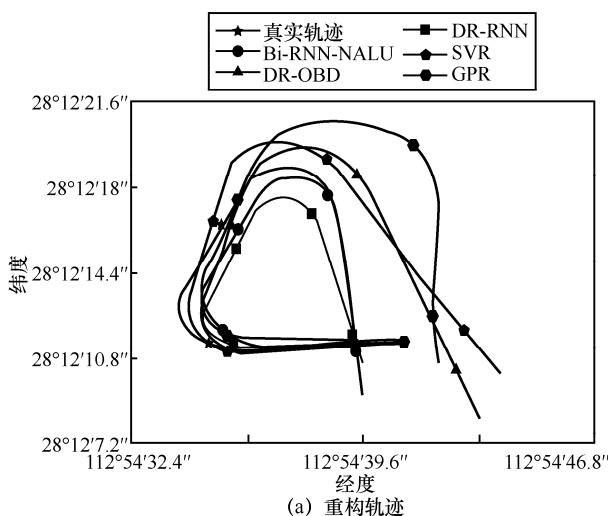


图 9 直角转弯路段重构轨迹及其位置误差

图 10 给出了立交桥路段各种方法的重构轨迹及其位置误差的结果。GNSS 信号失效时长为 66 s。在该路段，车辆以较低的速度持续行驶并保持转弯，这极大影响了车内运动传感器的性能。此外，车辆驾驶状态在立交桥持续变化，这与进入立交桥路段前完全不同。换言之，这种特定的驾驶状态，即保持 40 s 以上的转向，在训练集中很少出现。这



阻碍了模型从训练阶段获得足够的知识。也就是说，当该路段发生 GNSS 信号失效或中断时，从 OBD 读取器获得的运动信息不稳定，可能无法提供足够的信息。因此，如图 10(a)所示，DR-OBD 轨迹与真实轨迹偏离较大，DR-RNN 和 GPR 等重构的轨迹在形状和长度上也呈现不同程度的变化。在这种情况下，DR-RNN 无法正常工作，因为传统的 DNN 结构无法很好地解决预测数值超出训练数据范围的问题。通过重构基本算术运算，NALU 可以很好地解决额外数据的问题。Bi-RNN-NALU 算法由于能有效利用后向重构的轨迹，精度有了明显提升，特别是在轨迹中后期位置误差一直保持在低水平，有效克服了一般单向轨迹重构方法累计误差随时间增大的不足。

为进一步论证所提算法的性能，本文采用均方根误差 (RMSE, root mean square error) 作为指标验证各算法的有效性，其计算式为

$$RMSE = \sqrt{\frac{\sum_{i=1}^L [\hat{s}_i - s_i]^2}{L}} \quad (12)$$

其中， s_i 表示真实的位置数据， \hat{s}_i 表示重构的位置数据， $\hat{s}_i - s_i$ 表示重构位置与真实位置之间的距离， L 为 GNSS 信号中断的时长。

表 3 描述了在 4 个失效阶段各种方法重构轨迹的均方根误差对比。在表 3 中对本文提出的算法的误差结果进行了加粗显示，可以看出在各个路段 Bi-RNN-NALU 算法都能获得最好的精度，这是因为双向重构轨迹可以利用仅使用前向重构所缺乏的信息，确保了轨迹重构的完整性。此外，在立交桥路段，

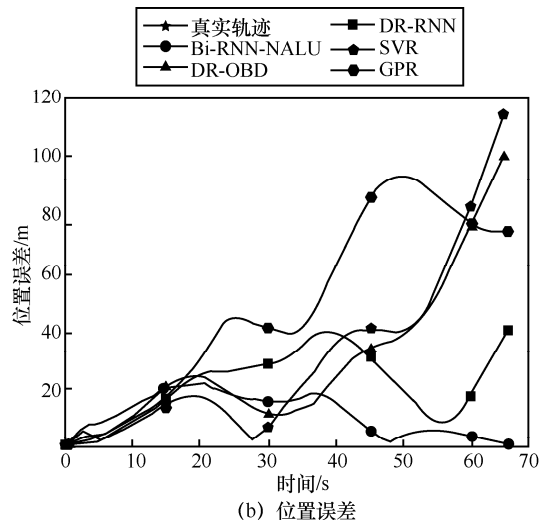


图 10 立交桥路段重构轨迹及其位置误差

表 3

各方法重构轨迹的均方根误差

算法	失效 1 (40 s) /m	失效 2 (40 s) /m	失效 3 (30 s) /m	失效 4 (66 s) /m
Bi-RNN-NALU	3.761	6.625	4.627	12.939
DR-OBD	55.031	44.815	48.609	39.793
DR-RNN	13.539	13.458	14.847	24.567
SVR	23.276	33.815	42.13	43.55
GPR	22.897	32.916	45.642	60.174

加入 NALU 的 Bi-RNN-NALU 及 DR-RNN-NALU 相比 DR-RNN 性能提升很大, 有效地提高了算法在诸如立交桥路段的复杂路段的稳健性和有效性。

6 结束语

本文设计了一种基于 GNSS-OBD 的车辆轨迹采集设备, 在此基础上, 针对城市环境下私家车轨迹采集所面临的不准确和数据缺失的问题, 提出一种新的基于 RNN 神经网络的轨迹重构算法——Bi-RNN-NALU, 该算法将 RNN 作为基本网络结构, 利用轨迹缺失前后的轨迹信息, 同时从前向及后向分别重构轨迹, 加权重构新的车辆轨迹。由于充分利用了单一前向轨迹重构方法所没有使用的后向信息, Bi-RNN-NALU 算法可以降低车辆轨迹后期的累计误差, 提高重构轨迹的准确性。此外, 为了应对复杂路段的挑战, 本文将 NALU 引入重构模型来提高算法在复杂路段的稳健性。通过真实道路测试, 证明了算法的准确性和可靠性。考虑到所采集的轨迹数据量庞大, 下一步工作将研究轨迹压缩算法, 提升轨迹数据处理的效率。

参考文献:

- [1] 中国统计局. 中国统计黄页 [M]. 北京: 中国统计出版社, 2018. National Bureau of Statistics. China statistical yearbook[M]. Beijing: China Statistical Publishing House, 2018.
- [2] YAN L, SHEN H Y, CHEN K. MobiT: distributed and congestion-resilient trajectory-based routing for vehicular delay tolerant networks[J]. IEEE/ACM Transactions on Networking, 2018, 26(3):1078-1091.
- [3] SHAN X N, HAO P, CHEN X H, et al. Vehicle energy/emissions estimation based on vehicle trajectory reconstruction using sparse mobile sensor data[J]. IEEE Transactions on Intelligent Transportation Systems, 2018, 20(2):716-726.
- [4] CHIANG K W, TSAI G J, CHANG H W, et al. Seamless navigation and mapping using an INS/GNSS/grid-based slam semi-tightly coupled integration scheme[J]. Information Fusion, 2019, 50(1): 181-196.
- [5] BARRIOS C, MOTAI Y, HUSTON D. Trajectory estimations using smartphones[J]. IEEE Transactions on Industrial Electronics, 2015, 62(12): 7901-7910.
- [6] BARRIOS C, MOTAI Y. Predicting vehicle trajectory[M]. Boca Raton: CRC Press, 2017.
- [7] HUANG Y R, XIAO Z, WANG D, et al. Exploring individual travel patterns across private car trajectory data[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 1(1): 1-15.
- [8] WANG D, FAN J J, XIAO Z, et al. Stop-and-wait: discover aggregation effect based on private car trajectory data[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(10): 3623-3633.
- [9] SUHR J K, JANG J, MIN D, et al. Sensor fusion-based low-cost vehicle localization system for complex urban environments[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 18(5): 1078-1086.
- [10] LIU L. Enabling large-scale road data acquisition with an easy-to-install and sustainable setup[C]// Workshop on Mobisys Phd Forum. New York: ACM Press, 2017: 3-4.
- [11] LIU R, YUEN C, DO T N, et al. Cooperative positioning for emergency responders using self IMU and peer-to-peer radios measurements[J]. Information Fusion, 2020, 56(1):93-102.
- [12] CARLOS M, RUIZ-GONZALEZ R, GOMEZ-GIL J. A data fusion system of GNSS data and on-vehicle sensors data for improving car positioning precision in urban environments[J]. Expert Systems with Applications, 2017, 80(1): 28-38.
- [13] 李璐璐, 华佳烽, 万盛, 等. 基于高效信息缓存的位置隐私保护方案[J]. 通信学报, 2017, 38(6): 148-157. LI L L, HUA J F, WAN S, et al. Achieving efficient location privacy protection based on cache [J]. Journal on Communications, 2017, 38(6): 148-157.
- [14] 陶桦, 冯富琴, 肖鹏, 等. 基于运行轨迹特征分析的车辆自组织网络路由算法[J]. 通信学报, 2016, 37(6): 144-153. TAO H, FENG F Q, XIAO P, et al. Routing algorithm based on characteristics analysis of vehicle trace in vehicular ad hoc network[J]. Journal on Communications, 2016, 37(6):144-153.
- [15] YANG Z L, GUO X Q, CHEN Z M, et al. RNN-Stega: linguistic steganography based on recurrent neural networks[J]. IEEE Transactions on Information Forensics and Security, 2019, 14(5):1280-1295.
- [16] TRASK A, HILL F, REED S, et al. Neural arithmetic logic units[C]// The Thirty-second Annual Conference on Neural Information Processing Systems. Saarland: DBLP, 2018: 8046-8055.
- [17] WU Y, ZHU H, DU Q, et al. A survey of the research status of pedestrian dead reckoning systems based on inertial sensors[J]. International Journal of Automation and Computing, 2019, 16(1): 65-83.

- [18] XIAO Z, LI P, HAVYARIMANA V, et al. GOI: a novel design for vehicle positioning and trajectory prediction under urban environments[J]. IEEE Sensors Journal, 2018, 18(13): 5586-5594.
- [19] ZHANG W, LI Y, XIPNG W, et al. Adaptive soft sensor for online prediction based on enhanced moving window GPR[C]// 2015 International Conference on Control, Automation and Information Sciences. Piscataway: IEEE Press, 2015: 291-296.

[作者简介]



肖竹（1981- ），男，湖南涟源人，博士，湖南大学副教授、博士生导师，主要研究方向为下一代无线通信网、移动计算、测距与定位导航技术、大数据挖掘等。



钱鑫（1994- ），男，安徽合肥人，湖南大学硕士生，主要研究方向为车联网、测距与定位导航技术。



蒋洪波（1976- ），男，湖南长沙人，博士，湖南大学教授、博士生导师，主要研究方向为下一代无线通信网、移动计算。



蔡成林（1969- ），男，湖南双峰人，博士，湘潭大学教授、博士生导师，主要研究方向为 GNSS 定位与导航技术。



曾凡仔（1971- ），男，湖南郴州人，博士，湖南大学教授、博士生导师，主要研究方向为无线通信和车联网。