一种改进的滑动窗口轨迹数据压缩算法

吴家皋[1,2],刘 敏[1,2],韦 光[1,2],刘林峰[1,2]
(1.南京邮电大学 计算机学院,江苏 南京 210003;
2.江苏省无线传感网高技术研究重点实验室,江苏 南京 210003)

摘 要:移动对象的轨迹信息具有重要的理论和应用价值。为减少轨迹数据存储空间,提高数据分析及传送速度,提出了一种改进的滑动窗口轨迹数据压缩算法。在该算法中,将最大偏移距离参考轨迹点作为当前待压缩的轨迹点能否被压缩的判据,以降低轨迹的压缩时间、提高压缩效率。实验测试结果表明,较现有的滑动窗口轨迹数据压缩算法,改进的滑动窗口轨迹数据压缩算法在压缩时间上显著减少,在压缩率上也有所提高,并且在常用的轨迹压缩阈值范围内,两种算法压缩后得到的轨迹集的相似度很高。

关键词:全球卫星定位系统;轨迹压缩;滑动窗口;压缩率
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2015)12-0047-05
doi:10.3969/j.issn.1673-629X.2015.12.011

An Improved Trajectory Data Compression Algorithm of Sliding Window

WU Jia-gao[1,2],LIU Min[1,2],WEI Guang[1,2],LIU Lin-feng[1,2]
(1. College of Computer,Nanjing University of Posts and Telecommunications,
Nanjing,210003,China
2. Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks,
Nanjing 210003,China)

Abstract:Moving object trajectory information has important theoretical and practical value. To reduce the track data storage space and improve analysis and transmission speed,an improved sliding window track data compression algorithm is proposed,in which whether the current trajectory point can be compressed is based on the maximum offset distance between the reference point,to reduce the compression time and improve the compression efficiency. Experimental results show that compared with the existing track sliding window data compression algorithm,the improved algorithm has advantages on a compressed time significantly,and slightly increases the compression ratio,moreover within a certain trajectory compression threshold,the two algorithms have high similarity of compressed trajectories set.

Key words:GPS;trajectory compression;sliding window;compression ratio

0 引 言

随着全球卫星定位系统(GPS)[1-2]等定位设备在移动终端上的广泛使用以及基于位置服务(Location-Based Service,LBS[3])和移动社交网络(Mobile Social Network[4])的发展和普及,实现了快速的定位并记录空间位置,用户可以方便地获取个人位置信息,研究人员也能方便地通过位置感知设备获取轨迹数据,因此越来越多的移动用户的轨迹数据被收集并存储在移动用户的数据库中,庞大的数据量给数据的存储、查询、分析及传送造成了很大的困难,所以对轨迹数据压缩算法的研究[2-5]成为目前的热点问题之一。

在对移动轨迹进行分析研究时,其实只需要存储准确描述移动轨迹的信息点,在这里称之为特征点,其他数据点可以简化处理,这是对轨迹数据压缩的基本

收稿日期:2015-03-17 **修回日期**:2015-06-23 **网络出版时间**:2015-11-19
基金项目:国家自然科学基金资助项目(61373139);江苏省自然科学基金(BK2012833);江苏省高校自然科学基金(12KJB520011);南京邮电大学科研基金(NY213160)
作者简介:吴家皋(1969-),男,副教授,博士,研究方向为计算机网络、GIS应用等;刘 敏(1989-),女,硕士生,研究方向为移动计算、移动模型等。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20151119.1109.030.html

思想。轨迹数据压缩的目的是在保留数据所包含的信息的前提下,去除冗余定位点,从而减少数据量,缩小数据所占用的存储空间。但在简化数据点的同时,必然会丢失一定量的信息,带来某些风险,所以目前出现的轨迹数据压缩算法[6-8],是在数据信息的准确性和数据存储空间两者之间进行权衡。

根据处理数据格式的不同,将轨迹数据压缩算法分为批处理算法和在线处理算法[9-10]。批处理算法只有在明确轨迹数据的起始轨迹点和终止轨迹点的情况下,才能对轨迹数据进行压缩。该算法适用于静态批量数据的处理,最具代表性的批处理压缩算法是文献[11]提到的道格拉斯-普克轨迹数据压缩算法;在线处理算法从轨迹数据的起始轨迹点开始,不需要事先明确轨迹数据的终止轨迹点,终止轨迹点可以动态增加。该算法适用于数据量相对较小噪声较大的数据,其代表算法有文献[12]提到的滑动窗口轨迹数据压缩算法和文献[13-14]提到的标准开放窗口(Normal Opening Window)算法。但是无论哪一类轨迹数据压缩算法,当它们的起始轨迹点和终止轨迹点有一个变化时,都需要计算位于起始轨迹点和终止轨迹点之间的所有轨迹点到起始轨迹点与终止轨迹点的直线的垂直距离,很多学者认为这样会影响算法的运算效率,处理时间较长。

结合现有压缩算法的优缺点,提出了一种改进的滑动窗口轨迹数据压缩算法。该算法在滑动窗口轨迹数据压缩方法中,将最大偏移距离参考轨迹点作为当前待压缩的轨迹点能否被压缩的判据,只需要计算最大偏移距离参考轨迹点和当前待压缩的轨迹点到滑动窗口的起始轨迹点和终止轨迹点的直线的垂直距离即可。实验测试结果表明,较现有的滑动窗口轨迹数据压缩算法,文中提出的改进算法在压缩时间上显著减少,在压缩率上也有所提高,并且在常用的轨迹压缩阈值范围内,两种算法压缩后得到的轨迹集的相似度极高。

1 相关工作

现有的轨迹数据压缩算法主要分为批处理算法和在线处理算法两大类。

批处理算法的代表是道格拉斯-普克轨迹数据压缩算法[12],它是一种自上而下的算法。该算法首先考虑将一条轨迹的初始轨迹点和终止轨迹点虚连一条直线,求出其余各轨迹点到该直线的垂直距离,选择其最大者与预先规定的阈值相比较,若小于等于阈值,则将直线两端间各轨迹点全部删去,若大于阈值,则将离该直线垂直距离最大的轨迹点保留,并以此为界,把轨迹分成两部分,对这两部分递归使用上述方法,直至最终无法做进一步的压缩为止。道格拉斯-普克轨迹数据压缩算法被广泛应用于制图和计算机图形应用中。很多学者认为该算法是批处理算法中最准确的,但是其处理时间过于昂贵,时间复杂度为 $O(n^2)$, n 为待压缩轨迹点的总数,而且该算法必须事先知道轨迹的终止点,才能对轨迹进行压缩,因此如果轨迹点是动态的增加,则需要使用在线处理算法。

在线处理算法主要包括滑动窗口轨迹数据压缩算法[12]和标准开放窗口轨迹数据压缩算法[13-14]。滑动窗口轨迹数据压缩算法是目前公认的经典算法,它采用逐步逼近的思想,类似于计算机网络流量控制中的滑动窗口协议,给定一个大小已确定的窗口,计算滑动窗口中所有位于起始轨迹点和终止轨迹点之间的轨迹点到起始轨迹点与终止轨迹点的直线的垂直距离,如果所有距离都小于预先规定的阈值,那么滑动窗口沿轨迹序列方向向后滑动一个轨迹点,否则,若出现偏移距离大于该阈值的点,则将滑动窗口的终止轨迹点前的那个轨迹点(即当前待压缩轨迹点)添加到压缩后的轨迹集中,新的滑动窗口从这个轨迹点开始,继续使用上述处理方法,直到处理完成。标准开放窗口轨迹数据压缩算法是在滑动窗口轨迹数据压缩算法的基础上进行的改进,也是一种在线处理算法,与滑动窗口轨迹数据压缩算法不同的是它将偏移距离最大的那个轨迹点作为新的滑动窗口起始轨迹点。

这两种在线处理算法不需要明确轨迹数据的终止轨迹点,可以动态地增加数据,适用于很多实际的应用中,但是这两种算法的时间复杂度仍然比较高,都为 $O(n^2)$ 。

针对上述问题,文中提出了一种改进的滑动窗口轨迹数据压缩算法。

2 算法描述

2.1 算法的改进策略

无论是道格拉斯-普克轨迹数据压缩算法、滑动窗口轨迹数据压缩算法还是标准开放窗口轨迹数据压缩算法,都是以轨迹点偏移轨迹方向的垂直距离作为轨迹特征点的选取标准,但是,当起始轨迹点和终止轨迹点有一个变化时,需要计算所有轨迹点到起始轨迹点和终止轨迹点的直线的垂直距离,从而造成算法复杂度增加,影响算法的运算效率。针对这一问题,提出将滑动窗口中的最大偏移距离参考轨迹点作为当前待压缩的轨迹点能否被压缩的判据。这样,当滑动窗口的起始轨迹点和终止轨迹点有一个变化时,只需要计算最大偏移距离参考轨迹点和当前待压缩的轨迹点到起始轨迹点和终止轨迹点的直线的垂直距离,从而显著降低算法的复杂度,减少压缩处理时间。

设待压缩的轨迹集为 $P=\{P_i\}$,其中,$P_i(x,y)$ 为第 i 个轨迹点,x,y 分别表示轨迹点的经度、纬度转换成平面坐标后的横坐标和纵坐标,$i\in[1,n]$,n 为待压缩轨迹点的总数;Q 是压缩后的轨迹集;令滑动窗口为 $W=(P_{start},P_{cur},P_{end},P_m)$,其中,$P_{start}$ 和 P_{end} 分别为滑动窗口的起始轨迹点和终止轨迹点,P_{cur} 为滑动窗口中当前待压缩轨迹点,P_m 为滑动窗口中最大偏移距离参考轨迹点;设 L 为轨迹压缩的距离阈值。

为了更清楚地阐明文中的改进策略,图 1 给出了一个改进的滑动窗口轨迹数据压缩算法的示意图。

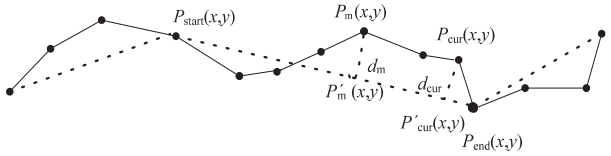

图 1　一个改进的滑动窗口轨迹数据压缩示意图

如图 1 所示,当前轨迹数据压缩滑动窗口 $W=(P_{start},P_{cur},P_{end},P_m)$,分别计算点 P_{cur} 和 P_m 到直线 $P_{start}-P_{end}$ 的垂直距离 d_{cur} 和 d_m,若 d_{cur} 或 d_m 大于阈值 L,说明轨迹点偏移轨迹方向较远,轨迹不能近似拟合,则将 P_{cur} 添加到轨迹集 Q 中,并且令 $P_{start}=P_{cur}$,设置新的滑动窗口;否则,P_{start} 不变,将滑动窗口的 P_{cur} 和 P_{end} 沿轨迹序列同时往后移一个点,并且根据 d_{cur} 和 d_m 的大小判断 P_m 是否需要更新。对于新的滑动窗口,重复上述过程直到压缩过程结束。这样,在每个滑动窗口的处理中,就不需要计算 P_{start} 和 P_{end} 间的其他轨迹点到直线 $P_{start}-P_{end}$ 的垂直距离,虽然会有一些误差,但是大大提高了算法的运算效率,降低了算法的复杂度。

根据上述思想,改进的滑动窗口轨迹数据压缩算法流程图如图 2 所示。

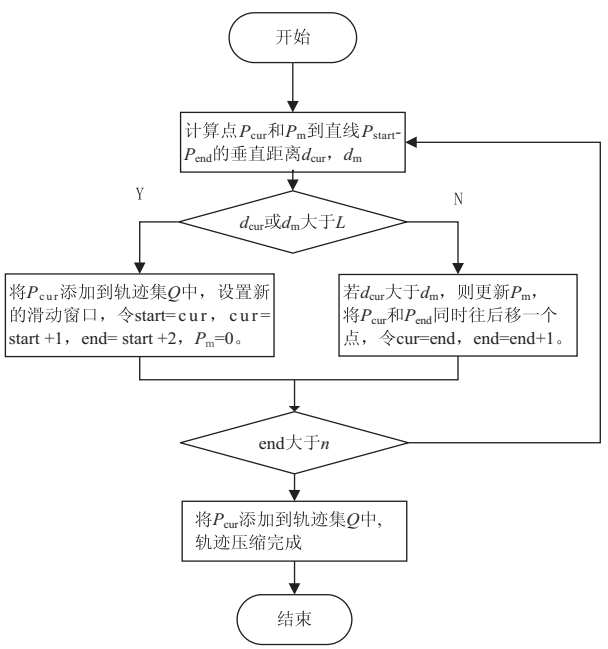

图 2　算法流程图

2.2　改进的滑动窗口轨迹数据压缩算法

(1)垂直距离的计算。

算法中都需要计算点到直线的垂直距离。以 P_{cur} 到直线 $P_{start}-P_{end}$ 的垂直距离 d_{cur} 计算为例。根据平面解析几何原理,垂直距离 d_{cur} 的计算公式为:

$$d_{cur}=\sqrt{(P_{cur}.x-P'_{cur}.x)^2+(P_{cur}.y-P'_{cur}.y)^2} \tag{1}$$

其中,P'_{cur} 为 P_{cur} 到直线 $P_{start}-P_{end}$ 的垂直坐标,如图 1 所示,其位置坐标由公式(2)、(3)给出:

$$P'_{cur}.x=\frac{P_{cur}.x+K\cdot(P_{cur}.y-P_{start}.y)+K^2\cdot P_{start}.x}{1+K^2} \tag{2}$$

$$P'_{cur}.y=P_{start}.x+K\cdot(P'_{cur}.x-P_{start}.x) \tag{3}$$

其中,$K=\dfrac{P_{end}.y-P_{start}.y}{P_{end}.x-P_{start}.x}$,同理,可以计算 d_m。

(2)特征点的判别条件。

通过引入最大偏移距离参考轨迹点 P_m 作为特征点选取的判别条件,根据实际情况设置相应的轨迹压缩距离阈值 L,如果当前待压缩点 P_{cur} 满足以下任一条件,则将该点作为特征点。

条件 1:P_m 到直线 $P_{start}-P_{end}$ 的垂直距离大于给定的阈值 L;

条件 2:P_{cur} 到直线 $P_{start}-P_{end}$ 的垂直距离大于给定的阈值 L。

综上所述,基于滑动窗口的轨迹数据压缩改进算法策略的伪代码如下所示:

```
Compress( )
输入:待压缩的轨迹集为 P = {P_i} ,轨迹压缩距离阈值为 L
输出:特征点集合 Q
{
Q = {P_1}
滑动窗口 W = (P_1 ,P_2 ,P_3 ,0)
do
利用公式(1)~(3)分别计算点 P_cur 和 P_m 到直线 P_start - P_end 的垂直距离 d_cur 和 d_m
if( d_cur > L or d_m > L ) then //对特征点判断
Q = Q ∪ {P_cur}    // P_cur 加入特征点集合
W = {P_cur ,P_start+1 ,P_start+2 ,0} //设置新的滑动窗口
end if
if( d_cur ≤ L and d_m ≤ L )
if( d_cur > d_m ) then
P_m = P_cur //根据距离大小更新 P_m
end if
W = {P_start ,P_end ,P_end+1 ,P_m} //设置新的滑动窗口
end if
while (end> n )
}
```

时间复杂度分析:该算法的时间复杂度为 $O(n)$,

n 为待压缩轨迹点的总数,由此可以看出,改进的滑动窗口轨迹数据压缩算法相对于滑动窗口轨迹数据压缩算法的时间复杂度降低了很多。

3 测试与性能分析

3.1 实验数据集和实验环境

为验证改进的滑动窗口轨迹数据压缩算法的性能,采用微软亚洲研究院 GeoLife 项目组的 182 名用户在一段时间内收集的数据集,从中选取一位用户从 2009-03-26,14:21:33 到 2009-03-27,12:00:07 之间的 GPS 数据,数据每隔 2 s 采集一次,共有 16 084 个轨迹点。

实验 PC 机:Inter(R) Core(TM) i3-3240,主频为 3.4 GHz,Java 环境为 jdk1.6,采用 eclipse 开发工具。

3.2 实验评价准则

为评价压缩效果,采用最通用的性能评价方法:压缩时间 T、压缩率 R、压缩误差 E 以及相似度 S。定义如下:

压缩时间 T:压缩完成后的时间 T_1,压缩前的时间 T_2。

$$T = T_1 - T_2 \tag{4}$$

压缩率 R:压缩前轨迹点的个数 M,压缩后轨迹点的个数 N。

$$R = 1 - \frac{N}{M} \times 100\% \tag{5}$$

压缩误差 E:非特征点到它相邻两个特征点的连线的平均距离,l_k 为非特征点 k 到前后相邻的特征点连线的距离。

$$E = \frac{1}{M-N}\sum_{k=1}^{k=M-N} l_k \tag{6}$$

相似度 S:令 $\boldsymbol{V} = (V_1, V_2, \cdots, V_n)$ 为压缩后的轨迹向量,若 P_i 为特征点,则 $V_i = 1$,否则 $V_i = 0$。设 $\boldsymbol{V}_1$ 为现有滑动窗口轨迹数据压缩后的轨迹向量,$\boldsymbol{V}_2$ 为改进的滑动窗口轨迹数据压缩后的向量,则

$$S = \frac{\boldsymbol{V}_1 \cdot \boldsymbol{V}_2}{|\boldsymbol{V}_1| \cdot |\boldsymbol{V}_2|} \tag{7}$$

其中,S 表示滑动窗口轨迹数据压缩算法和文中改进算法得到的压缩后的轨迹集的相似程度。

3.3 实验结果

为了评价轨迹压缩策略的压缩效果,实验采用了改变轨迹点的个数将滑动窗口轨迹数据压缩算法与改进的滑动窗口轨迹数据压缩算法在压缩时间上进行了比较,结果如图 3 所示。

由图 3 可知,两种算法的压缩时间随着需要压缩的轨迹点的个数的增大而增大,同时可以得出,文中提出的算法在压缩时间上较滑动窗口轨迹数据压缩算法明显减小。且随着轨迹点个数的增大,两种算法的压缩时间相差越大,这说明轨迹点个数越多,文中提出的算法的压缩时间减少的越明显。

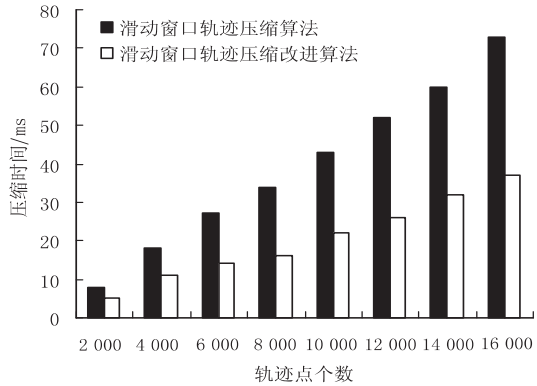

图 3 轨迹点个数改变对压缩时间的影响

另外,实验通过改变轨迹压缩的距离阈值将滑动窗口轨迹数据压缩算法与文中改进算法在压缩率、压缩误差、相似度上进行了比较。由于数据采集间隔时间较短,以下实验每隔 6 s 采集一次,共有 5 361 个轨迹点。实验结果如图 4 ~ 6 所示。

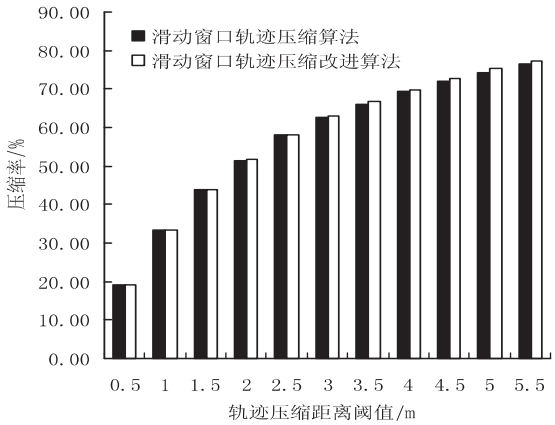

图 4 轨迹压缩距离阈值改变对压缩率的影响

由图 4 可知,文中提出的算法在压缩率上较滑动窗口轨迹数据压缩算法有小幅度的改进,而且可以看出,无论哪种算法,数据压缩率都随着轨迹压缩距离阈值的增大而增加。由此可以说明,选取的轨迹压缩距离阈值越大,能够保留的数据点就越少,压缩率越大,随着轨迹压缩距离阈值的增大,也带来了信息丢失的风险,轨迹数据压缩是在数据信息的准确性和数据存储空间两者之间进行权衡。

由图 5 可知,文中提出的压缩算法在压缩误差上较滑动窗口轨迹数据压缩算法有小幅度的增大,这是因为在对特征点的判断时只考虑了最大偏移距离参考轨迹点和当前待压缩轨迹点到滑动窗口的起始轨迹点和终止轨迹点的直线的垂直距离,而没有将所有的轨迹点到该直线的垂直距离都进行判断;另外,无论哪种算法,随着轨迹压缩距离阈值的增大,其压缩误差都增大,因此可以看出,选取的轨迹压缩距离阈值越大,非

特征点到相邻特征点连线的距离之和越大,压缩后的轨迹与其原来的轨迹相差越远。

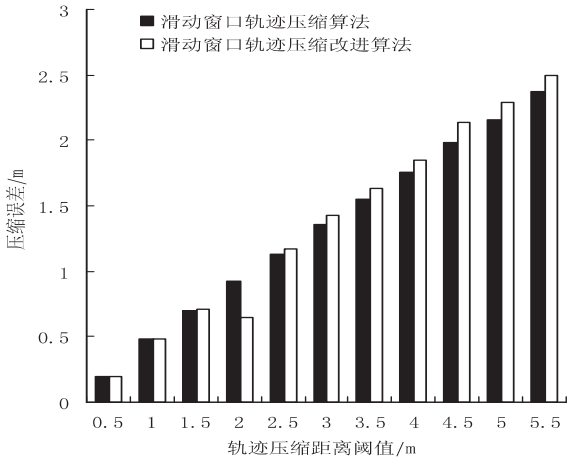

图 5　轨迹压缩距离阈值改变对压缩误差的影响

为了评价滑动窗口轨迹数据压缩算法和改进的滑动窗口轨迹数据压缩算法所得的特征点的异同,文中采用相似度进行比较,如图 6 所示。

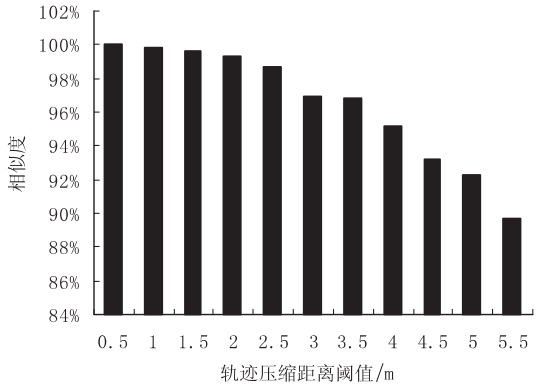

图 6　轨迹压缩距离阈值改变对相似度的影响

由图 6 可知,随着轨迹压缩距离阈值的增大,滑动窗口轨迹数据压缩算法和基于滑动窗口的轨迹数据压缩改进算法的相似度减小,轨迹压缩阈值从 0.5 到 5.5 的范围内,相似度达到 90% 以上。由此可以看出,两种压缩算法得到的特征点非常相似。

4　结束语

对于轨迹数据压缩,道格拉斯-普克轨迹数据压缩算法因其计算简单、压缩效率高而被普遍采用。但是道格拉斯压缩算法只适用于批量数据的处理,且压缩误差相对较大;滑动窗口轨迹数据压缩算法可以适用于在线处理,但其计算量较大,压缩时间较高。文中提出的改进的滑动窗口轨迹数据压缩算法虽然在压缩误差上略有上升,但在压缩率有所提高,压缩时间大大减少,而且其压缩后的轨迹集与滑动窗口轨迹数据压缩算法得到的轨迹集相似度极高,具有一定的研究价值。对于如何减小压缩误差,还有待进一步的探究。

参考文献:

[1] Fan Bo, Leng Supeng, Yang Kun. GPS: a method for data sharing in mobile social networks[C]//Proceedings of networking conference. Trondheim: IEEE, 2014: 1-9.

[2] Pan Gang, Qi Guande, Wu Zhaohui, et al. Land-use classification using taxi GPS traces[J]. IEEE Trans on Intelligent Transportation Systems, 2013, 14(1): 113-123.

[3] Ray S, Blanco R, Goel A. Supporting location-based services in a main-memory database[C]//Proceedings of 2014 IEEE 15th international conference on mobile data management. Brisbane, QLD: IEEE, 2014: 3-12.

[4] Zhao Peikun, Zhao Juanjuan, Wang Wu. Division of mobile social network based on user behavior[C]//Proceedings of 2013 international conference on wavelet analysis and pattern recognition. Tianjin: IEEE, 2013: 148-152.

[5] Chen Minjie, Xu Mantao, Franti P. Compression of GPS trajectories[C]//Proceedings of data compression conference. Snowbird, UT: IEEE, 2012: 62-71.

[6] Hung C, Peng W C. Model driven traffic data acquisition in vehicle sensor network[C]//Proceedings of 2010 39th international conference on parallel processing. San Diego: IEEE, 2010: 424-432.

[7] Chen Long, Wong R C, Jagadish H V. Direction-preseving trajectory simplification[J]. Proceedings of the VLDB Endowment, 2013, 6(10): 949-960.

[8] Beliman R. On the approximation of curves by line segments using dynamic programming[J]. Communications of the ACM, 1961, 4(6): 284-284.

[9] Potamias M, Patroumpas K, Sellis T. Sampling trajectory streams with spatiotemporal criteria[C]//Proceedings of the 18th international conference on scientific and statistical database management. Piscataway: IEEE, 2006: 275-284.

[10] Bogorny V, Valiti J, Alnares L. Semantic-based pruning of redundant and uninteresting frequent geographic patterns[J]. Geoinformatica, 2010, 14(2): 201-220.

[11] Yu Jing, Chen Gang, Zhang Xiao. An improved Douglas-Peucker algorithm aim at simplifying natural shoreline into direction-line[C]//Proceedings of 2013 21st international conference on geoinformatics. Kaifeng: IEEE, 2013: 1-5.

[12] Cao Xin, Cong Gao, Jensen C S. Mining significant semantic locations from GPS data[J]. VLDB, 2010, 3(1): 1009-1020.

[13] Maratnia N, de By R. Spatio-temporal compression techniques for moving point objects[C]//Proc of EDBT. [s. l.]: [s. n.], 2004: 765-782.

[14] 张达夫,张析明. 基于时空特性的 GPS 轨迹数据压缩算法[J]. 交通信息与安全, 2013, 31(3): 6-9.

一种改进的滑动窗口轨迹数据压缩算法

作者： 吴家皋， 刘敏， 韦光， 刘林峰， WU Jia-gao， LIU Min， WEI Guang， LIU Lin-feng
作者单位： 南京邮电大学 计算机学院,江苏 南京 210003;江苏省无线传感网高技术研究重点实验室,江苏 南京 210003

刊名： 计算机技术与发展 ISTIC
英文刊名： Computer Technology and Development
年，卷(期)： 2015,25(12)