ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 30,No. 1,Jan. 2015,pp. 47－58
DOI:10.16337/j.1004-9037.2015.01.004

http://sjcj.nuaa.edu.cn
E-mail:sjcj@nuaa.edu.cn
Tel/Fax:＋86-025-84892742

# 时空轨迹大数据模式挖掘研究进展

吉根林　赵　斌

(南京师范大学计算机科学与技术学院,南京,210023)

**摘　要**:时空轨迹挖掘是数据挖掘领域的前沿研究课题,通过研究和开发时空轨迹挖掘技术,来发现隐藏在轨迹大数据中有价值的规律和知识以供决策支持。本文介绍了时空轨迹大数据模式挖掘与知识发现领域的研究进展;然后对时空轨迹模式挖掘技术产生的背景、应用领域和研究现状作了简介,并探讨了面向时空轨迹大数据模式挖掘的研究内容、系统架构以及关键技术,最后对时空轨迹频繁模式、伴随模式、聚集模式和异常模式的挖掘算法思想进行了阐述。

**关键词**:时空轨迹模式挖掘;时空轨迹大数据;轨迹频繁模式;轨迹伴随模式;轨迹聚集模式;轨迹异常模式

**中图分类号**:TP181　　**文献标志码**:A

## Research Progress in Pattern Mining for Big Spatio-temporal Trajectories

Ji Genlin, Zhao Bin

(School of Computer Science and Technology, Nanjing Normal University, Nanjing, 210023, China)

**Abstract**: Spatio-temporal trajectory pattern mining has emerged as an active research field, focusing on the research and development of mining technology for big spatio-temporal trajectories to discover useful rules and knowledge. This paper attempts to review the recent research progress in spatio-temporal trajectory pattern mining and knowledge discovery. Then the background, application and advances of spatio-temporal trajectory pattern mining are introduced. And the research contents, system infrastructure and key technologies in big spatio-temporal trajectory pattern mining are discussed. Finally, the mining algorithm ideas for frequent pattern, flock pattern, gathering pattern, outlier pattern of spatio-temporal trajectory are expounded.

**Key words**: spatio-temporal trajectory pattern mining; big spatio-temporal trajectory; trajectory frequent pattern; trajectory flock pattern; trajectory gathering pattern; trajectory outlier pattern

## 引　　言

根据2014年《中国移动互联网调查研究报告》[1]显示,截至2014年6月,中国网民规模达6.32亿,中国手机网民规模达5.27亿,占比为83.4%。近年来,手机等移动智能终端已经开始为广大普通用户提供全方位的信息服务。例如,百度地图帮助车辆导航,大众点评推荐兴趣点(Point of interest,POI),

**基金项目**:国家自然科学基金(41471371)资助项目。
**收稿日期**:2014-12-10;**修订日期**:2015-01-21

快的打车提供租车预约服务等。这些常见的移动互联网应用服务的背后需要对其产生的时空数据进行分析与挖掘。

时空轨迹数据作为时空数据的一种，它由 GPS 终端、智能手机等设备产生，记录了移动对象的行为特征，包括位置、时间、速度、方向等属性。随着智能移动终端的广泛应用，时空轨迹数据的存储、分析研究已受到学术界的广泛关注。时空轨迹数据挖掘技术在许多领域得到了应用，如交通协调与管理（如道路流量监控[2]）、旅游路线推荐[3,4]、自然灾害预警（如飓风预测[5]）、环境保护（如空气质量监测[6]）等。

目前学术界对于轨迹数据已经开展了深入研究。文献[7]在 2013 年提出的基于轨迹数据分析与挖掘的智慧城市技术体系框架，如图 1 所示。该框架包含 3 个层次，分别是轨迹数据的感知层、知识发现层和应用层。轨迹数据的感知层通过 GPS，WIFI，蓝牙等技术手段，获取移动对象的位置、时间、访问频率、共现模式等信息；知识发现层通过数据挖掘等技术分析与理解移动对象的活动规律和特性，预测个体行为和群体事件等；有了上述两个层次的支撑，应用层可以为城市的公共安全、医疗服务、交通管理、商业金融等领域提供信息化服务与决策支持。本文介绍时空轨迹大数据模式挖掘与知识发现领域的研究进展。

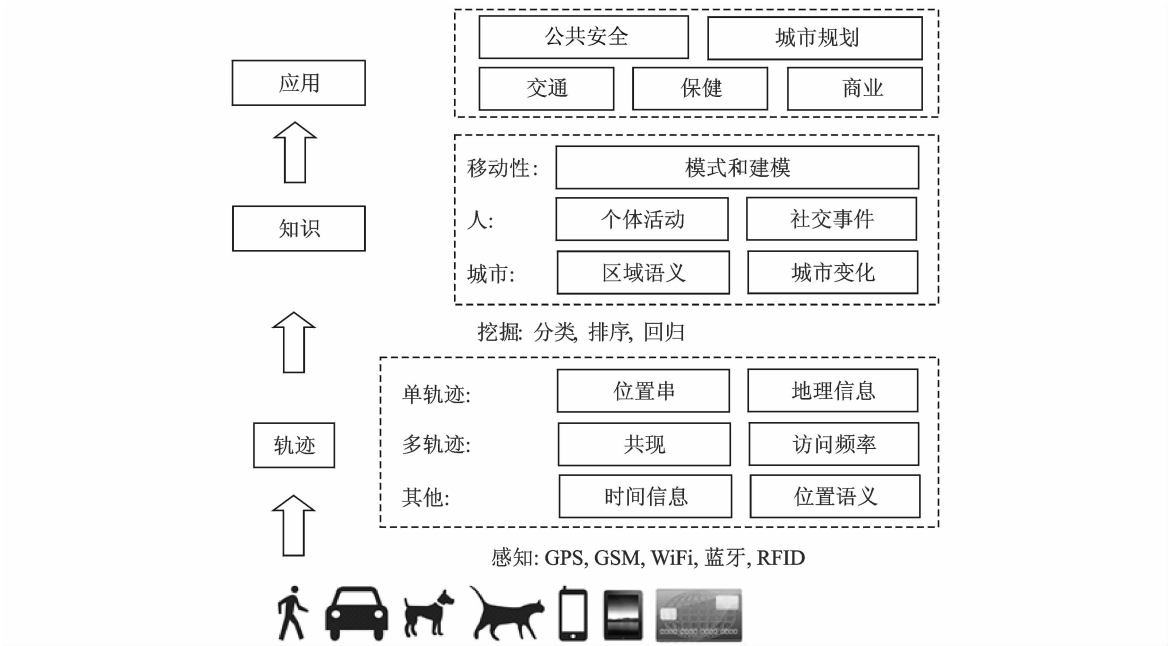


图 1 基于轨迹挖掘的智慧城市技术体系框架[7]

Fig. 1 Framework of smart cities based on trajectory mining

## 1 时空轨迹预处理和模式定义

在无线通讯与移动计算技术快速发展的背景下，大量时空轨迹数据来源于手机、PDA、导航设备等智能移动终端。通常，智能设备的性能和定位技术的特性决定了时空轨迹数据具有两个基本特点：(1) 数据质量低，由于基站、RFID，WIFI 等定位技术精度有限，因而生成的轨迹数据在地理位置上存在偏差，需要修正；(2) 数据规模大，智能终端设备主要采用周期采样的方式生成轨迹数据，日积月累产生的数据规模特别巨大。在轨迹数据挖掘中无法直接使用质量有限的原始轨迹数据。而是通过预处理阶段修正轨迹中的位置偏差和去除冗余的位置点信息，以确保轨迹数据挖掘的最终效果和计算效率。

### 1.1 时空轨迹预处理

时空轨迹数据预处理包括:道路匹配和轨迹压缩两个基本阶段。

(1)道路匹配

移动设备所在场景的环境噪声、设备本身的可靠性以及定位技术自身局限性会影响移动对象的定位精度。以最常使用的 GPS 定位技术为例,通常定位精度在 10 m 左右。如果将电子地图缩放到较大比例时,移动对象位置会出现偏离路网道路的情况。所以,即使采集到了移动对象的轨迹数,也必须首先进行道路匹配工作,修正有偏差的位置数据[8]。

(2)轨迹压缩

轨迹数据通常由智能移动终端的定位模块按照不同的采样频率实时产生。采样频率越高,得到的轨迹数据就越精确,越能够完整反映移动对象的行为。但是,同时会产生大量冗余的位置点数据,这会严重影响挖掘算法的效率[9,10]。所以,在轨迹预处理阶段的一个重要工作就是轨迹的压缩[11]。轨迹压缩按照处理方式的不同分为离线压缩和在线压缩两种。离线方法在访问所有轨迹位置的前提下对原有轨迹进行近似化处理。最经典的方法是 DP(Douglas-Peucker)算法[12,13],它采用线段序列替代原有数据点序列,保持原有轨迹的几何特征。而在线处理方法更适合轨迹实时处理的场景(如移动对象的实时监控)。基于滑动窗口的算法[14,15]在变长的滑动窗口中按照指定偏离错误率的指导对冗余位置点进行替换。

### 1.2 时空轨迹模式定义

挖掘时空轨迹数据中有价值的模式,如频繁模式、伴随模式、聚集模式、异常模式等,一直是时空轨迹数据挖掘研究中的一个重要课题。在定义各种时空轨迹模式之前,首先定义时空轨迹序列。它是轨迹数据中最基本的形式。按照序列中元素类型的不同,可以分为时空轨迹点序列和时空轨迹边序列两种。

**定义 1** 时空轨迹点序列:它是一个元组序列 $S_v=\{v_0,\cdots,v_i,\cdots,v_n\}$,其中 $v_i=\langle x_i,y_i,t_i\rangle$ 是空间中的坐标点,$t_i$ 为该点的时戳。

**定义 2** 时空轨迹边序列:时空轨迹边序列 $S_e$ 是由时空轨迹点序列变换而来,是一个元组序列,$S_e=\{e_1,\cdots,e_i,\cdots,e_n\}$,其中 $e_i=\langle(x_{i-1},y_{i-1},t_{i-1}),(x_i,y_i,t_i)\rangle$ 是空间中的一条边,$(x_i,y_i,t_i)$ 表示点坐标,$t_i$ 表示时戳。

#### 1.2.1 时空轨迹频繁模式

时空轨迹频繁模式是指从时空轨迹集中发现的频繁重复的序列。它们可以协助研究人员完成关于移动对象的分析、预测等任务,进而可将其应用于经营商业、旅游业和管理城市交通等方面决策。

在进行轨迹频繁模式挖掘之前,必须对时空轨迹数据进行预处理。首先采用路网匹配和轨迹压缩技术对时空轨迹序列进行处理,仅保留可能改变轨迹方向的关键点;然后,将时空轨迹序列转变成由兴趣点构成的新序列。不难发现,挖掘兴趣点序列模式要比挖掘单纯的空间坐标点更有意义。

**定义 3** 兴趣点序列:给定阈值 $d$,时空轨迹边序列 $S_e$ 对应的兴趣点序列为 $S_a=\{a_i,\cdots,a_j\}$,其中 $1\leqslant i<j\leqslant m$,且满足以下条件:对于任意 $a_i$,在 $S_e$ 中总存在 $e_j$,使得点 $a_i$ 到边 $e_j$ 的垂直距离小于等于距离阈值 $d$。

**定义 4** 时空轨迹频繁模式挖掘:给定时空轨迹数据库 $D=\{S_{a1},S_{a2},\cdots,S_{an}\}$ 和最小支持度 $S_{\min}$,其中 $S_{ai}\in D$ 为兴趣点序列。时空轨迹模式挖掘问题就是找出所有频繁的兴趣点子序列。频繁兴趣点子序列 $P$ 满足以下条件

$$\mathrm{Support}_D(P)\geqslant s_{\min} \tag{1}$$

式中 $\mathrm{Support}_D(P)=|\{S|P\subseteq S,S\in D\}|/|D|$。

1.2.2 时空轨迹伴随模式

时空轨迹伴随模式是指从时空轨迹数据集中发现具有相同或者相似路线的移动对象群体。通过分析移动对象群体的行为特征和规律,可以帮助实现在时空环境中的群体跟踪、热点事件发现等。2007年,Benkert[16]等首次给出了时空轨迹伴随模式的形式化定义。

**定义 5** 伴随模式:给定 $m, k \in \mathbf{N}, r$ 为大于零的常数。给定时空轨迹集合,且每条轨迹由 $\tau$ 条线段构成。伴随模式是指在时间区间 $I=[t_i, t_j](j-i+1 \geqslant k)$ 中,至少包含 $m$ 个移动对象,在时间区间 $I$ 的每个时刻中所有位置点都集中在半径为 $r$ 的圆形区域内。

1.2.3 时空轨迹聚集模式

文献[9,17]于2013年提出了聚集模式。首先定义3个基本概念:快照簇、群体和参与者。

**定义 6** 快照簇:为某一时刻移动对象形成的簇,并且簇内所有移动对象密度相连。

**定义 7** 群体:由一定数目的快照簇形成的集合,并且任意相邻时刻的快照簇间的距离都小于等于给定的距离阈值。

**定义 8** 参与者:在群体中出现至少 $k_p$ 次的移动对象。

**定义 9** 聚集模式:如果群体中的每个快照簇含有至少 $m_p$ 个参与者,那么这个群体就属于聚集模式。

1.2.4 时空轨迹异常模式

2008年,Lee等人[10]提出了轨迹异常模式挖掘。设有时空轨迹数据集 $D=\{TR_1, \cdots, TR_n\}$, $TR_i = p_1 p_2 p_3 \cdots p_j \cdots p_{len_i} (1 \leqslant i \leqslant n)$是一条轨迹,其中,$p_j$ 为 $d$ 维度的点,$len_i$ 为轨迹 $TR_i$ 的长度。轨迹段是指一条线段 $p_i p_j (i<j)$,$p_i$ 和 $p_j$ 是来自 $TR_i$ 中任意的点。

**定义 10** 离群轨迹段:如果一个轨迹段周围没有足够数量的其他轨迹段与其靠近,则它称为离群轨迹段。

**定义 11** 异常模式:轨迹的异常模式是指 $O=\{O_1, \cdots, O_m\}$,其中 $O_i$ 为离群轨迹段。

## 2 时空轨迹频繁模式挖掘

时空轨迹的频繁模式挖掘可以形式化为频繁序列挖掘[18]。但是两者区别在于:轨迹数据中包括位置维度、时间维度和语义维度[19]等,所以简单地采用传统序列挖掘方法无法有效解决时空轨迹频繁模式挖掘问题。

2007年,Giannotti[20]突破了传统频繁序列挖掘,提出挖掘由兴趣区域(Region of interest, ROI)构成的频繁序列。为此他们提出了3种不同的轨迹频繁模式挖掘算法。在数据预处理阶段,将轨迹的位置点序列转化成由兴趣区域的序列。按照空间离散化形式的不同,兴趣区域分别预设兴趣区域和热门兴趣区域两种形式。前两种轨迹模式挖掘算法分别基于这两种兴趣区域而实现。第3种算法主要考虑如何结合空间和时间维度进行轨迹模式挖掘。根据局部区域密度的变化规律,采用增量式的处理方法发现兴趣区域,在此基础上构建新的挖掘算法。实验结果表明该方法可以识别更准确的轨迹模式。

2013年,Luo等人[21]研究了一种基于时间周期的最频繁路径(Time period-based most frequent path, TPMFP)查询问题。它的主要目标是分析和研究大多数行人最频繁的道路选择情况。为了避免路经边数和非频繁路经对于挖掘结果的影响,他们没有采用基于标量值的函数计算路经被路过的频数(简称,路经频数),而是选择了序列形式描述路经频数。具体步骤为:(1)建立在指定时间区间中含有边权重的足迹图,该步骤为了应对轨迹数据海量规模的挑战,采用足迹索引提高创建足迹图的计算效率;(2)将路经查询问题采用动态规划算法解决,然后采用改进的Bellman-Ford算法进行最终的问题求解。

2014年,Zhang等人[19]提出了时空轨迹的细粒度序列模式挖掘问题,认为在连续空间中的位置点不适合进行序列模式挖掘,而由相同语义的位置点构成的项目更适合该问题。于是,将位置的语义维度

连同空间和时间维度一起加入到细粒度序列模式挖掘中。该模式必须满足3个基本条件:(1)空间紧凑型;(2)语义一致性;(3)时间连续性。将细粒度序列模式挖掘分两步解决:首先,挖掘出一组轨迹片段支持的粗模式;然后,采用自顶向下的方式逐步细化粗模式得到细粒度序列模式。

## 3 时空轨迹伴随模式挖掘

时空伴随模式挖掘的研究经历了3个阶段:(1)群体模式挖掘;(2)伴随模式挖掘;(3)蜂群模式挖掘。

### 3.1 群体模式

2002年,Laube和Imfeld[22]定义了一种基于相似运动方向的时空轨迹模式集合。2004年,他们进一步提出了一系列基于运动方向和位置信息的移动对象运动模式,其中包括群体模式[23]。群体模式只考虑单一时刻移动对象的移动行为,要求在某一时刻至少有 $m$ 个对象在同一区域中并且移动方向相同。

2006年,Gudmundsson等人[16]认为上述群体模式定义并不符合实际应用。因为移动对象群体在被定义成群体模式之前,可能待在一起几天甚至几个星期。Gudmundsson等人给出了群体模式的新定义,即flock($m$,$k$,$r$)。它是指一定数量的移动对象在给定半径的圆形区域内持续移动,其中 $m$ 为群体内移动对象的最小数量;$k$ 为移动对象持续移动的最短时间;$r$ 为移动对象所在圆形区域的最大半径。

### 3.2 伴随模式

2008年,Jeung等人[24,25]为了解决在轨迹模式挖掘过程中对于移动对象群体大小和形状上的限制,提出了伴随模式,要求一定数目的移动对象在持续 $k$ 个时间内密度相连。给定距离阈值 $e$ 和点集 $S$,$S$ 中任意点 $p$ 的 $E$ 邻域表示为 $NH_e(p)=\{q\in S\mid D(p,q)\leqslant e\}$。

**定义12** 直接密度可达:给定距离阈值 $e$ 和最小点数 $m$,如果点 $q\in NH_e(p)$,并且 $|NH_e(p)|\geqslant m$,则点 $q$ 与点 $p$ 直接密度可达。

**定义13** 密度可达:给定距离阈值 $e$ 和最小点数 $m$,如果存在点序列 $p_1,p_2,\cdots,p_n,p=p_1,q=p_n$ 使得点 $p_i$ 与 $p_{i-1}$ 直接密度可达,则点 $q$ 与点 $p$ 密度可达。

**定义14** 密度相连:给定距离阈值 $e$ 和最小点数 $m$,如果存在的一个点 $x$,使得从 $x$ 出发都与点 $p$ 和点 $q$ 密度可达,那么 $p$ 和 $q$ 密度相连。

给定一个时空轨迹集合 $S$,密度约束 $e$ 和 $m$,时间区间 $k$,伴随模式要求在时间区间 $k$ 内任意时刻的簇都至少有 $m$ 个移动对象彼此密度相连。如图2所示,在 $t_1$ 到 $t_4$ 的时刻区间中移动对象形成了一个伴随模式,其中3条折线段分别代表 $o_1$,$o_2$ 和 $o_3$ 三个对象的移动轨迹。

Jeung等人[25]提出了相干移动簇算法(Coherent moving cluster, CMC)来挖掘伴随模式,因为CMC算法中要生成虚拟位置点来为那些缺失的点进行线性插值,并且在每个时刻都要对每个对象进行聚类,时间开销很大,这导致了很高的计算代价。因此在CMC算法基础上引入了DP和考虑时间因素的新DP[35]两种轨迹简化技术来简化轨迹,在得到简化轨迹后对轨迹段进行聚类,并在之上进行区域查询,分别提出了采用轨迹简化技术的伴随模式发现(Convoy discovery using trajectory simplification, CuTS)算法和CuTS+算法。Jeung在CuTS的基础上考虑到轨迹简化和距离计算的时间特性,又提出了CuTS*算法来进一步提高挖掘效率。

2010年,Aung等人改进了Jeung之前提出的伴随模式。主要原因是伴随模式无法解决两个问题:(1)伴随中有一些成员可能暂时离开群体,而这些成员在挖掘中不应该被直接忽视;(2)在现实中伴随可能会演变成更大或者更小的伴随。为了解决这两个问题,文献[26]提出了进化伴随的概念。进化伴随模式含有两种成员:固定成员和动态成员。进化伴随模式在其时间跨度内的任一时刻都必须包含至少

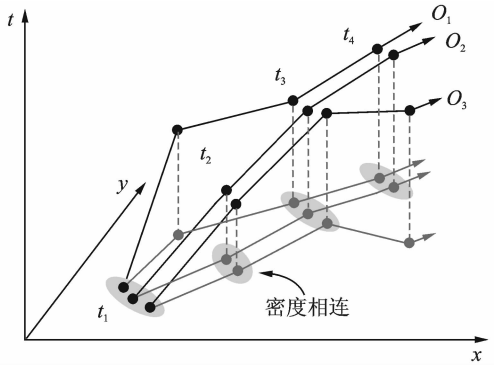


图 2 伴随模式示例[25]

Fig. 2 Example of convoy patterns

$m$(整数阈值)个彼此密度相连的固定成员;同时进化伴随模式可以拥有零个或多个动态成员。但是每个动态成员都必须在移动时间区间中与固定成员至少密度相连 $k$ 次。

Aung 等人提出了一个简单切片式算法来挖掘进化伴随模式,这个算法和之前提出的 CMC 算法类似,都是首先对缺失的数据位置信息进行线性插值,再在每个时刻进行基于密度的聚类,最后再进行查询。与 CMC 一样,该算法也要进行大量的 DBSCAN 聚类,时间开销非常大。Aung 按照 Jeung 提出的 TRAJ-DBSACAN 思想[25],对轨迹进行分段并在分段后的轨迹上进行聚类,从而提出了交错式进化伴随算法 ID-1[26],在对轨迹进行简化后,利用 TRAJ-DBSCAN 代替原来的遍历扫描,这样在每个时刻进行很少的聚类操作;在 ID-1 算法的基础上,Aung 又增加了剪枝操作进一步优化算法,形成了交错式进化伴随算法 ID-2[26]。

### 3.3 蜂群模式

2010 年,Li 等人认为之前的伴随模式挖掘方法在移动对象簇的定义上有很大的限制。这些方法都要求移动物体在连续的时间内在一起同时移动。然而同一簇中的移动对象可能会暂时离开群体,但在后续某个时刻再次相聚。根据这一基本事实,提出了蜂群模式概念[27],即在一定的时间内移动对象在形状任意的簇内一起移动且时间不要求连续。为了避免挖掘冗余的伴随模式,他们进一步给出了闭合蜂群的定义,这样模式挖掘的目标就变为查找完整的闭合蜂群集合。

## 4 时空轨迹聚集模式挖掘

时空轨迹聚集模式的挖掘就是要将具有相似行为的时空对象划分到同一组中,它能帮助人们在日常生活中监控和预测不寻常的群体事件。对于各种聚集模式具有不同的挖掘算法,但总体可以分为两类:关联规则剪枝的聚集模式挖掘算法和基于密度聚类的聚集模式挖掘算法。

### 4.1 基于关联规则剪枝的聚集模式挖掘算法

2003 年,Wang 等人[28]在传统顾客群体购物信息的基础上增加了顾客之间的时空距离信息,从而提出了组的概念,即在连续的指定时间内,如果一个集合内的所有移动对象彼此之间的距离都小于指定阈值,那么这样的集合被称之为组模式。

2004 年,Wang 等人[29]提出了基于 Apriori 的有效组模式挖掘算法(Apriori-like algorithm for mining valid group patterns, AGP)和基于 FP-Growth 的有效组图结构挖掘算法(Algorithm based on valid group graph data structures, VG-Growth)来挖掘组模式。2005 年,Hwang San-Yih 等人[30]认为上述两

个算法存在两个缺陷:为了保证位置信息的精确,对移动对象位置信息的采样频率必须很高,这样就会使得数据库变得非常庞大;由于基站时钟的差异,在现实中对用户的位置信息的采集几乎不可能完全同步。为了解决以上问题,他们采用轨迹模型表示物体的移动,一条轨迹可以分解成一组线性函数的集合,并且每个线性函数的时间区间互不相交,针对 AGP 算法和 VG-Growth 算法的不足,提出了基于轨迹的组模式挖掘(Apriori trajectory-based group pattern mining, ATGP)算法和遍历式 VG-Growth(Traversal VG-Growth,TVG-Growth)算法来挖掘组模式。2008 年,Wang 等人[31]进一步发展了 AGP 和 VG-Growth 算法,提出了基于 Apriori 的最大有效组挖掘(Apriori-based algorithm for mining maximal valid groups, AMG)算法和生成最大有效组的 VG-Growth(an extension of VG-growth for generating maximal valid groups, VGMax)算法。

2010 年,Li 等人[27]提出了一种面向移动对象的深度优先搜索算法 ObjectGrowth,该算法在搜索过程中利用先验剪枝策略,向后剪枝策略和向前闭包检查三种算法来进行剪枝优化。

### 4.2 基于密度聚类的聚集模式挖掘算法

按照轨迹形式的不同,利用基于密度的空间聚类(Density-based spatial clustering of applications with noise, DBSCAN)算法[32]对移动对象进行聚类可分为两种情况:(1)对移动对象原始轨迹进行聚类,例如移动簇模式[33]和旅伴模式[34];(2)对原始轨迹预处理后的轨迹段进行聚类,例如聚集模式。

2005 年,Kalmis 等人[33]提出了挖掘移动簇模式算法 MC1。首先在每个时刻对移动物体进行聚类,然后对相邻时刻的簇求交集来判断它们是否包含足够数目的公共对象。他们在 MC1 的基础上提出 MC2 算法,即在每个时刻进行聚类后,对相邻簇进行求交时增加剪枝操作来降低计算代价。最后继续改进,在 MC2 基础上提出了近似算法 MC3,通过减少 DBSCAN 聚类的对象数量降低算法运行时间。

2012 年,Tang 等人[34]提出了基于旅伴结构挖掘旅伴模式,传统的聚集模式挖掘需要保存每个采样点的坐标和采样时间等信息,但旅伴结构只存储移动对象间的关系,如是否属于同一个簇等,这极大地降低了轨迹聚类时的计算量。

2013 年,郑凯等人[9,17]针对之前提出的聚集模式定义的不足,提出了聚集模式。该定义不仅考虑利用轨迹简化技术来简化轨迹,在轨迹段上进行聚类,而且通过建立网格索引大大降低了空间区域查询的计算量。他们还研究了在轨迹数据库不断增加的情况下,利用增量式算法挖掘聚集模式。

需要说明的是,聚集模式的研究发展中借鉴了伴随模式的研究工作。比较伴随模式和聚集模式挖掘问题可以发现,两者的主要差别在于:(1)伴随模式中的移动对象在时空环境中的组织程度相对紧密、严格,而集中模式定义的要求相对较松;(2)伴随模式通常识别时空环境中的小群体事件,而集中模式更适合识别时空环境中的大群体事件。

## 5 时空轨迹异常模式挖掘

通常按照处理数据类型的不同,将时空轨迹的异常检测分为面向静态数据集和面向数据流的两种方式。

### 5.1 静态数据集的轨迹异常检测

2008 年,Lee 等人[10]提出了基于划分和检测的框架用于异常轨迹检测,并基于该框架设计了异常轨迹检测算法(Trajectory outlier detection algorithm, TRAOD)。TRAOD 算法分为两阶段:分割阶段和检测阶段。分割阶段采用最小描述长度方法将轨迹划分成一系列轨迹段;而检测阶段采用基于距离和密度的方法识别离群轨迹段,以此为基础判断轨迹是否异常。在本算法中轨迹段间距离计算采用 Hausdorff 距离。算法 TRAOD 的时间复杂度为 $O(n_t^2)$,$n_t$ 为轨迹段的总数。虽然 TRAOD 算法解决了异常轨迹的发现问题,但是它的缺点也非常明显。当轨迹数量巨大时,轨迹段之间的距离计算非常耗

时,因而算法的运行效率并不理想。

2010 年,Ge 等人[36]提出实时轨迹异常检测算法(Top-k evolving trajectory outlier detection method, TOP-EVE),通过分析移动对象的行为发现 Top-K 的异常轨迹。不同于以往的基于距离的轨迹计算,他们同时考虑了异常轨迹在空间距离上和运动方向上的离群因素。具体步骤:首先为空间区域的建立方格,为每个方格定义方向矩阵,根据轨迹的历史数据生成基于方向的摘要向量,如图 3 所示;然后按照式(2)计算指定轨迹和摘要向量的距离,以此判断该轨迹的异常情况。

$$\mathrm{OScoreDir}=1-\sum_{k=1}^{K}q_k\sum_{i=1}^{8}p_i*\cos\angle(v_k,v_i) \tag{2}$$

2009 年,刘良旭等人[37]提出了基于 R-Tree 的异常轨迹检测算法。该方法通过检测轨迹的局部异常程度来判断两条轨迹是否全局匹配,进而检测异常轨迹。该算法以 $k$ 个连续轨迹点作为基本比较单元表示轨迹的局部特征,提出基于基本比较单元匹配程度度量的距离函数,在此基础上定义了局部匹配、全局匹配和异常轨迹的概念。为了提高算法的执行效率,利用 R-Tree 和轨迹间的距离特征矩阵查找所有可能匹配的基本比较单元对,然后再通过距离计算确定其是否局部匹配,从而消除大量不必要的距离计算任务,提高算法执行效率。

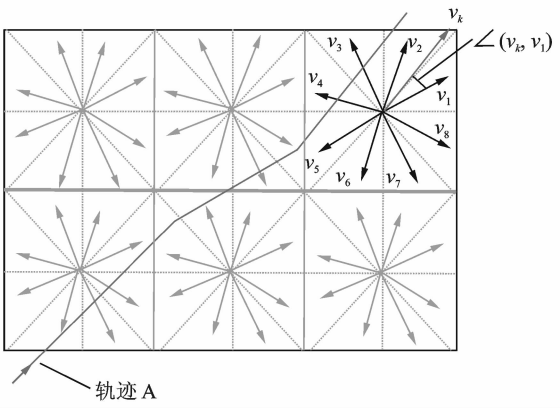


图 3 基于方向的距离度量示例[36]

Fig. 3 Illustration of direction based distance measure

## 5.2 轨迹数据流的异常检测

2009 年,Bu 等人[38]研究轨迹数据流中的异常实时检测问题。不同于数据库中的异常轨迹研究,由于轨迹流的数据量巨大,无法存储后再进行异常检测,所以必须采用实时处理的方式检测其中的异常。该问题定义如下:给定轨迹数据流 $S=\{s_1,s_2,\cdots,s_t,\cdots\}$,$s_i$ 为数据流中在第 $i$ 时刻的实数值,还定义了 3 种滑动窗口,分别是基础窗口 $B=\{s_i,s_{i+1},\cdots,s_{i+w_b-1}\}$,左滑动窗口 $L=\{s_{i-w_l},s_{i-w_l+1},\cdots,s_{i-1}\}$,右滑动窗口 $L=\{s_{i+w_b},s_{i+w_b+1},\cdots,s_{i+w_r-1}\}$,基于上述 3 种滑动窗口定义了轨迹流的异常检测问题。

**定义 15** 距离:给定基础窗口 $X$ 和 $Y$,$X$ 与 $Y$ 之间的距离定义为

$$\mathrm{Distance}(X,Y)=\sqrt{\sum_{i=1}^{w_b}(X_i-Y_i)^2} \tag{3}$$

**定义 16** 邻居:给定待测基础窗口 $B=\{s_j,s_{j+1},\cdots,s_{j+w_b-1}\}$,数据流 $S=\{s_1,s_2,\cdots s_h\}(h>w_b)$和距离阈值 $d$,如果对于 $S$ 中的任意子序列 $S_i=\{s_i,s_{i+1},\cdots,s_{i+w_b-1}\}$,$\mathrm{distance}(B,S_i)<d$,则 $S_i$ 为在 $S$ 中 $B$ 的邻居。

**定义 17** 轨迹流的异常:给定基础窗口 $B$ 和轨迹数据流 $S$,在左滑动窗口 $L$ 中 $B$ 的邻居数量为 $n_1$,

在右滑动窗口 $R$ 中 $B$ 的邻居数量为 $n_2$,如果 $n_1+n_2<d$,则 $B$ 为轨迹流中的异常。

为了加快轨迹流中异常检测的效率,定义了基于局部聚类的方法检测流中的异常。2014 年 Yu 等人[39]提出了海量轨迹流中的异常移动对象检测问题。与文献[38]研究单一移动对象轨迹流中的异常片段不同,Yu 研究的是多个不同移动对象的轨迹流。该问题定义如下:给定移动对象集 $MO=\{o_1,o_2,\cdots,o_n\}$ 及其轨迹流 $S=\{p_1^1p_2^1\cdots p_n^1,p_1^2p_2^2\cdots p_n^2,\cdots,p_1^ip_2^i\cdots p_n^i,\cdots\}$,其中 $p_1^ip_2^i\cdots p_n^i$ 为 $S$ 中落入相同时间间隔 $i$ 的轨迹点。

按照轨迹流中异常粒度的不同,分为基于点邻居的异常检测和基于轨迹邻居的异常检测。点邻居指对于在相同时间间隔 $t_j$ 中的两个轨迹点 $p_m^j$ 和 $p_n^j$,如果 $\mathrm{dist}(p_m^j,p_n^j)\leqslant d$,则 $p_m^j$ 是 $p_n^j$ 的一个点邻居。基于点邻居的轨迹异常(PN-Outlier)定义为:给定距离阈值 $d$,邻居数量阈值 $k$ 和时间间隔阈值 $thr_t$,设 $T=\{t_j\parallel \mathrm{PN}(Tr_i,t_j,d)|\geqslant k,t_j\in[\mathrm{W.T_{Start}},\mathrm{W.T_{End}}]\}$,如果 $|T|<thr_t$,则在时间窗口 $W$ 中的轨迹 $Tr_i$ 为 PN-Outlier。简单地说,判断一条轨迹是否异常取决于该轨迹是否在足够的时间间隔中都具有足够的点邻居。

而轨迹邻居指对于给定的时间间隔数量的阈值 $thr_t$,当且仅当在滑动窗口 $W$ 中至少存在 $thr_t$ 个时间间隔中使得 $p_m^j$ 是 $p_n^j$ 的一个点邻居。基于轨迹邻居的轨迹异常(TN-Outlier)定义为:给定距离阈值 $d$,邻居数量阈值 $k$ 和时间间隔阈值 $thr_t$,如果 $|\mathrm{TN}(Tr_i,d,thr_t)|<k$,则在时间窗口 $W$ 中的轨迹 $Tr_i$ 为 TN-Outlier。简单地说,如果在整个时间窗口中没有足够数量的轨迹邻居,则该轨迹为异常。

为了提高算法执行效率,利用滑动窗口重叠的特性设计增量式的检测算法,为了进一步降低 CPU 的计算开销,提出了基于最小观测优化框架的检测方法,实验结果表明该方法可以大幅提高执行效率。

## 6 时空轨迹大数据模式挖掘技术

对于时空轨迹大数据,时空轨迹模式挖掘除了需要经典的数据挖掘技术[40](关联分析、分类、聚类、异常检测等)以外,还需要其他新技术的支持。这些技术按照体系结构的层次不同分为:云计算技术、时空轨迹数据压缩和消减技术、时空轨迹数据可视化技术。

### 6.1 云计算技术

学术界对于时空轨迹数据的集中式数据管理与分析问题[41]已经深入研究了多年。但是,如何利用并行计算技术分析、处理时空轨迹大数据,仍然需要研究。近年来,社交网络和移动互联网的快速发展,造成数据规模成倍扩大,海量数据增加的速度远远超过现有的处理能力。虽然以 MapReduce[42,43] 和 Hadoop[44] 为代表的大规模并行计算技术的出现,为学术界提供了一条大数据处理的新思路。但是,现有的 MapReduce 计算模型以键值对的形式组织和处理数据,并且中间结果依靠外部磁盘进行转存,所以不适合处理时空轨迹数据。此外,Hadoop 技术无法有效支持数据挖掘中监督学习所常用的迭代式计算方法,因而也无法完全满足时空数据分析的需要。另一方面,时空数据本质上是非结构化的数据,不仅包含时序数据模型,还存在图模型,例如道路网络等。基于图模型的算法时间复杂度通常比较大,对于海量数据而言,即使是 $O(N)$ 的复杂度也无法承受。

近年来,学术界和工业界投入了大量资源研究大数据处理的平台和技术。例如,美国加州伯克利大学 AMP 实验室的团队所开发的 Spark 平台[45]。不同于 Hadoop 依靠磁盘进行数据存储和交换,Spark 采用内存作为数据计算和处理的载体。因而,Spark 非常适用于迭代式的数据挖掘算法。GPU 计算[46]也是一种流行的并行处理技术。它运用 GPU(图形处理器)搭配 CPU 来加速通用科学和工程应用程序。GPU 计算由 NVIDIA 公司首先提出,并已经成为一种行业标准。还有 Twitter 公司开发的分布式、容错的实时处理平台 Storm[47],它可以使用在线方式处理时空轨迹数据[38]。上述平台和技术的不断涌现有助于推动在时空轨迹数据的存储管理和索引技术方面的研究,以应对时空大数据的挑战。

### 6.2 轨迹数据压缩和消减技术

轨迹数据压缩和消减是轨迹数据预处理中的重要内容。现有的轨迹数据大多来自于移动终端中的GPS设备,依靠相对固定的采样率持续生成,这样产生了大量的位置点数据。由于相邻位置点冗余度较大,学术界采用轨迹压缩与消减技术有效降低原始轨迹的数据规模,对原始轨迹进行近似化表示,在不影响挖掘效果的前提下提升了轨迹模式挖掘算法的执行效率。但是,这些方法往往只针对特定挖掘问题进行了优化,并没有从根本上解决轨迹数据的近似化表示。

目前,大量的轨迹数据来自于城市中的移动终端设备。在道路网络的背景下,可以重新考虑轨迹数据的压缩和消减问题。其基本思想是,将原有的轨迹点序列转化成线段序列,同时保留关键性的"拐点"。例如,如果移动对象经过路口,那么此路口的坐标应该作为结点被保留在线段序列中;如果移动对象改变了交通工具,导致行为方式或者移动速度发生变化,那么这样的行为"拐点"也应该被保留下来。总之,在保证一定精度的前提下,采用上述压缩和消减技术可以有效降低轨迹数据规模,提升挖掘算法的执行效率。

### 6.3 时空轨迹数据可视化技术

可视化是一种直观有效的数据展现与分析技术,利用它可以有效展示轨迹模式挖掘的结果。例如,文献[10]采用可视化的方法展现了飓风途径路线中的异常轨迹。文献[20]采用可视化技术展示了轨迹频繁模式挖掘的结果,在城市地图中标识出的热点区域及其关联关系。文献[48]采用密度图技术展现轨迹数据的密度以及帮助识别热点区域。文献[49]在路口交通分析中通过绘制轨迹展现路口交通的总体规律。文献[50]通过颜色、拓扑结构、统计图表等多种可视化的交互手段演示了交通拥堵自动识别跟踪的全过程。总之,以时空轨迹为代表的时空数据可视化分析仍然是一个新兴领域,未来具有广阔的研究与应用前景。

## 7 结束语

随着移动互联网、云计算和大数据技术的快速发展,人们的城市生活方方面面都被时空轨迹数据所包围,从海量时空轨迹数据中挖掘出有价值的规则和知识,可为智慧城市等领域提供决策支持。本文综述了时空轨迹模式挖掘的发展背景、研究进展和关键技术。目前时空轨迹大数据模式挖掘尚处于初步研究阶段,仍然存在许多基础问题尚未解决。期望本文介绍的时空轨迹大数据模式挖掘研究能为同行学者提供一定的参考。

**参考文献:**

[1] 中国互联网络发展状况统计报告[R]. 北京:中国互联网络信息中心,2014.
Statistical reports on the Internet development in China[R]. Beijing: China Internet Network Information Center, 2014.

[2] Gidofalvi G, Pedersen T B. Mining long, sharable patterns in trajectories of moving objects[J]. Geoinformatica, 2009, 13(1): 27-55.

[3] Zheng Y, Xie X. Learning travel recommendations from user-generated GPS traces [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2011, 2(1): 2.

[4] Lee W H, Tseng S S, Tsai S H. Knowledge based real-time travel time prediction system for urban network [J]. Expert Systems with Applications, 2009, 36(3): 4239-4247.

[5] Kitamoto A. Spatio-temporal data mining for typhoon image collection [J]. Journal of Intelligent Information Systems, 2002, 19(1): 25-41.

[6] Zheng Y, Liu F, Hsieh H P. U-Air: When urban air quality inference meets big data[C]//The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Chicago, IL, USA: ACM, 2013: 1436-1444.

[7] Pan G, Qi G, Zhang W S, et al. Trace analysis and mining for smart cities: Issues, methods, and applications[J]. IEEE Communications Magazine, 2013, 51(6): 120-126.

[8] 李宇光.海量低频浮动车数据道路匹配及行程时间估算[D].武汉:武汉大学,2013.
Li Yuguang. Huge-volume low-frequency floating car data map-matching and travel time estimation[D]. Wuhan: Wuhan University, 2013.
[9] Zheng K, Zheng Y, Yuan N J, et al. On discovery of gathering patterns from trajectories[C]//IEEE 29th International Conference on Data Engineering. Brisbane, Queensland, Australia: IEEE Computer Society Press, 2013:242-253.
[10] Lee J G, Han J, Li X. Trajectory outlier detection: A partition-and-detect framework[C]// Proceedings of the 24th International Conference on Data Engineering. Washington: IEEE Computer Society, 2008:140-149.
[11] Zheng Y, Zhou X F. Computing with spatial trajectories[M]. London: Springer, 2011:3-32.
[12] Douglas D, Peucker T. Algorithms for the reduction of the number of points required to represent a line or its caricature[J]. The Canadian Cartographer, 1973,10(2):112-122
[13] Hershberger J, Snoeyink J. Speeding up the Douglas-Peucker line simplification algorithm [C]//The Fifth International Symposium on Spatial Data Handling. Charleston, SC, USA: University of South Carolina, 1992:134-143.
[14] Keogh E, Chu S, Hart D, et al. An on-line algorithm for segmenting time series[C]//International Conference on Data Mining (ICDM). San Jose, California, USA: IEEE Computer Society Press, 2001:289-296.
[15] Maratnia N, de By R. Spatio-temporal compression techniques for moving point objects [C]//International Conference on Extending Database Technology (EDBT). Heraklion, Crete, Greece: Springer LNCS, 2004:765-782.
[16] Benkert M, Gudmundsson J, Hübner F, et al. Reporting flock patterns[J]. Computational Geometry, 2008,41(3):111-125.
[17] Zheng K, Zheng Y, Yuan N, et al. Online discovery of gathering patterns over trajectories[J]. IEEE Transactions on Knowledge and Data Engineering, 2013,8(26):1974-1988.
[18] Zhu D G, Pei J. Sequence data mining[M]. London: Springer, 2007.
[19] Zhang C, Han J, Shou L, et al. Splitter: Mining fine-grained sequential patterns in semantic trajectories [C]//39th International Conference onVery Large Data Bases. Riva del Garda, Trento:ACM, 2014,7(9):769-780.
[20] Giannotti F, Nanni M, Pinelli F, et al. Trajectory pattern mining[C]//Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. San Jose, California, USA: ACM, 2007:330-339.
[21] Luo W, Tan H, Chen L, et al. Finding time period-based most frequent path in big trajectory data[C]//Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2013. New York, NY, USA:ACM, 2013:713-724.
[22] Laube P, Imfeld S. Analyzing relative motion within groups of trackable moving point objects[C]//Geographic Information Science. Berlin Heidelberg: Springer, 2002:132-144.
[23] Laube P, van Kreveld M, Imfeld S. Finding REMO detecting relative motion patterns in geospatial lifelines[C]//Developments in spatial data handling. Berlin Heidelberg: Springer, 2005:201-215.
[24] Jeung H, Shen H T, Zhou X. Convoy queries in spatio-temporal databases[C]//IEEE 24th International Conference on Data Engineering. Cancun, Mexico: ACM, 2008:1457-1459.
[25] Jeung H, Yiu M L, Zhou X, et al. Discovery of convoys in trajectory databases[C]//34th International Conference on Very Large Data Bases. Auckland, New Zealand:ACM,, 2008,1(1):1068-1080.
[26] Aung H H, Tan K L. Discovery of evolving convoys[C]//Scientific and Statistical Database Management, 22nd International Conference, SSDBM 2010. Heidelberg, Germany: Springer LNCS, 2010:196-213.
[27] Li Z, Ding B, Han J, et al. Swarm: Mining relaxed temporal moving object clusters[J]. Proceedings of the VLDB Endowment, 2010,3(1-2):723-734.
[28] Wang Y, Lim E P, Hwang S Y. On mining group patterns of mobile users[J]. Database and Expert Systems Applications, 2003:287-296.
[29] Wang Y, Lim E P, Hwang S Y. Effective group pattern mining using data summarization[C]//9th International Conference on Database Systems for Advanced Application. Seoul, Korea: Springer LNCS, 2004:895-907.
[30] Hwang S Y, Liu Y H, Chiu J K, et al. Mining mobile group patterns: A trajectory-based approach[C]//Advances in Knowledge Discovery and Data Mining. Berlin Heidelberg: Springer, 2005:713-718.
[31] Wang Y, Lim E P, Hwang S Y. Efficient algorithms for mining maximal valid groups [J]. The International Journal on Very Large Data Bases, 2008,17(3):515-535.
[32] EsterM, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise [C]//Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). Portland, Oregon, USA: [s.n.], 1996,96:226-231.

[33] Kalnis P, Mamoulis N, Bakiras S. On discovering moving clusters in spatio-temporal data[C]//Advances in Spatial and Temporal Databases, 9th International Symposium SSTD 2005. Angra dos Reis, Brazil: Springer LNCS, 2005:364-381.

[34] Tang L A, Zheng Y, Yuan J, et al. On discovery of traveling companions from streaming trajectories[C]//IEEE 28th International Conference on Data Engineering. Washington, USA: IEEE Computer Society Press, 2012:186-197.

[35] Meratnia N, Rolf A. Spatiotemporal compression techniques for moving point objects[C]//Advances in Database Technology, EDBT 2004. Berlin Heidelberg: Springer, 2004:765-782.

[36] Ge Y, Xiong H, Zhou Z H, et al. Top-eye: top-k evolving trajectory outlier detection[C]// Proceedings of the 19th ACM International Conference on Information and Knowledge Managemen. Toronto, Ontario, Canada: ACM, 2011:1733-1736.

[37] 刘良旭,乔少杰,刘宾,等.基于 R-Tree 的高效异常轨迹检测算法[J].软件学报,2009,20(9):2426-2435.
Liu Liangxu, Qiao Shaojie, Liu Bin, et al. Efficient trajectory outlier detection algorithm based on R-Tree[J]. Journal of Software, 2009,20(9):2426-2435.

[38] Bu Y, Chen L, Fu A W C, et al. Efficient anomaly monitoring over moving object trajectory streams[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, France: ACM, 2009:159-168.

[39] Yu Y, Cao L, Rundensteiner E A, et al. Detecting moving object outliers in massive-scale trajectory streams[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2014:422-431.

[40] Miller H J, Han J. Geographic data mining and knowledge discovery[M]. USA:CRC Press, 2009.

[41] Cudre-Mauroux P, Wu E, Madden S. Trajstore: An adaptive storage system for very large trajectory data sets[C]// Proceedings of the 26th International Conference on Data Engineering, ICDE 2010. Long Beach, California, USA: IEEE Computer Society Press, 2010:109-120.

[42] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008,51(1):107-113.

[43] DeanJ, Ghemawat S. MapReduce: A flexible data processing tool [J]. Communications of the ACM, 2010,53(1):72-77.

[44] White T. Hadoop: The definitive guide[M] . Sebastopol,CA,USA: O′Reilly Media,2009:1-12.

[45] Zaharia M, Chowdhury M, Franklin M J. Spark: Cluster computing with working sets[C]//The 2th USENIX Workshop on Hot Topics in Cloud Computing. Boston, MA: USENIX Association, 2010:10.

[46] Barlas G. Multicore and GPU Programming: An integrated approach[M]. Waltham, MA, USA: Morgan Kaufmann, 2014:1-26.

[47] Goetz T P, O′Neill B. Storm blueprints: Patterns for distributed real-time computation[M]. Birmingham,UK: Packt Publishing.

[48] Scheepens R, Willems N, van de Wetering H, et al. Interactive visualization of multivariate trajectory data with density maps[C]//IEEE Pacific Visualization Symposium, PacificVis 2011. Hong Kong, China: IEEE Computer Society Press, 2011:147-154.

[49] Guo H, Wang Z, Yu B, et al. TripVista: Triple perspective visual trajectory analytics and its application on microscopic traffic data at a road intersection[C]//IEEE Pacific Visualization Symposium, PacificVis 2011. Hong Kong, China: IEEE Computer Society Press, 2011:163-170.

[50] Wang Z, Lu M, Yuan X, et al. Visual traffic jam analysis based on trajectory data[J]. IEEE Transactions on Visualization and Computer Graphics, 2013,19(12):2159-2168.

作者简介:**吉根林**(1964-),男,教授,博士生导师,研究方向:数据挖掘技术及应用,E-mail:glji@njnu.edu.cn;**赵斌**(1978-),男,讲师,博士,研究方向:Web 数据挖掘。