基于 GPS 轨迹的城市拥堵区域挖掘与分析

武兴业,吴 悦,岳晓冬
(上海大学 计算机工程与科学学院,上海 200444)

摘 要:随着世界城市化进程的发展,以及机动车保有量的飞速增长,交通拥堵问题日益严峻,城市道路交通拥堵问题已成为研究热点。传统的拥堵检测手段一般采用视频监控或传感器检测,虽然可以实时有效地反映拥堵状态,但无法挖掘城市拥堵规律,更无法有效分析拥堵原因和拥堵影响。文中提出基于 DENCLUE 的拥堵区域挖掘算法,对车辆 GPS 数据预处理,计算出拥堵点,然后对拥堵点进行 DENCLUE 聚类来确定拥堵区域。实验证明,该算法可以有效找出拥堵区域,并将拥堵划分等级,得到城市区域拥堵状态,反映城市拥堵情况。此外,使用斯皮尔曼等级相关系数计算区域间拥堵程度相关系数,结合实际地理位置分析拥堵区域的拥堵原因及影响。

关键词:拥堵点;拥堵区域挖掘;DENCLUE;相关系数;拥堵分析
中图分类号:TP391.41 **文献标识码**:A **文章编号**:1673-629X(2016)07-0116-06
doi:10.3969/j.issn.1673-629X.2016.07.025

Mining and Analysis of Urban Congestion Region Based on GPS Trajectory

WU Xing-ye, WU Yue, YUE Xiao-dong
(School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China)

Abstract: With the development of world urbanization, and the rapid growth of motor vehicle, traffic congestion problem is increasingly serious and has become a hot research topic. Traditional congestion detection generally uses the video surveillance or sensors, although they can effectively reflect the real-time congestion, but can't mine the law of the urban congestion, even fail to analyze the congestion correlation between urban areas. In this paper, the congestion region mining algorithm is proposed based on DENCLUE, first preprocessing the urban taxi GPS data, calculation of the congestion point, then clustering them by DENCLUE to determine the congestion area. It is shown in the experiment that the algorithm can effectively find out congestion areas and grade them, getting the congestion state of the city, reflection of urban congestion. In addition, Spearman rank correlation coefficient is used to calculate the flow of car correlation coefficient between regions, analysis of the congestion causes and effects of severe congestion regions combined with the actual location.

Key words: congestion point; congestion region mining; DENCLUE; correlation coefficient; congestion analysis

0 引 言

随着世界城市化进程的发展,城市交通问题日益严重和普遍,而城市交通与城市规划互相影响,合理的城市规划会促进交通状况,而错误的城市规划将导致严重的交通问题。因此,需要将城市交通与城市规划有效结合以促进问题的解决[1]。目前我国正处于高速城市化时期,城市和城市交通的发展正处于挑战和机遇并存的关键历史阶段,同时也是交通治理转型的关键时期。像北京这种大城市的交通问题是不可避免的,虽然修建公路和地铁可以缓解,但并不是长久之计[2]。智能交通系统(Intelligent Transportation System,ITS)[3]是将先进的科学技术(信息技术、数据通信技术、传感器技术、电子控制技术、数据处理技术等)有效地综合运用于地面运输管理体系,加强车辆、道路、使用者三者之间的联系,从而形成一种保障安全、提高效率、改善环境、节约能源的综合运输系统,是当前研究与应用的热点。所以大城市都大力发展 ITS 作为城市交通问题的重要解决途径之一。其中,大规模交通数据管理、整合和挖掘是一项关键技术。大数据时代,智能交通系统已经积累了巨量而复杂的道路交通数据信息,比如车辆的 GPS 信息,这些交通数据信息为智能交通的研究提供了重要的数据基础。

收稿日期:2015-09-14 **修回日期**:2015-12-17 **网络出版时间**:2016-06-22
基金项目:国家自然科学基金面上项目(61573235)
作者简介:武兴业(1991-),男,硕士,研究方向为数据挖掘;吴 悦,博士,教授,研究方向为智能信息处理。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20160622.0842.002.html

数据挖掘作为目前最强有力的一种数据分析工具,为道路交通数据的处理提供了新的分析手段,如何设计有效的数据挖掘算法将特定的交通规律挖掘出来是当前智能交通数据挖掘研究的关键。城市拥堵区域挖掘是智能交通领域研究的一个关键问题,具有非常重要的现实意义。通过空间数据挖掘技术挖掘出拥堵区域,分析拥堵区域之间的相关性,就能发现拥堵原因,给城市建设提供科学依据,解决城市发展中的很多问题。

文中使用北京 10 357 辆出租车一周的 GPS 轨迹数据做实验,提出一种基于 DENCLUE 拥堵区域挖掘算法,和一套用于分析拥堵区域原因及影响的方法。

1　相关工作

1.1　城市规划

微软研究院郑宇博士等,通过分析和融合城市中的各种大数据,实现了一系列关于智能交通、城市规划的实际案例。比如文献[4]利用高速和环路等主干道将城市分割成区域,然后分析大规模车流轨迹数据在不同区域之间行驶的一些特征,便可找到连通性较差的区域对,从而发掘现有城市道路网的不足之处。通过对比连续两年的检测结果,可以验证一些已经设施的规划(如新建道路和地铁)是否真的有效。

1.2　交通异常分析

文献[5-8]通过分析北京 3 万多辆出租车的轨迹来发现城市中的异常事件。其主要思想是当异常事件发生时,附近的交通流将出现一定程度的紊乱。文献[6]试图用具体的交通线路来进一步解释异常出现的原因。有时候,两个区域之间出现了交通流异常,但问题本身可能并不在这两个区域,而在于远处的车流必须通过这两个区域前往另一个目的地,这些车流才是问题的根源。文献[7]根据司机们路线选择方式的改变来捕捉交通异常,并进一步从相关的微博中提取关键词来解释异常的原因,如婚博会、道路坍塌等。

2　基于 DENCLUE 的拥堵区域挖掘算法

2.1　拥堵点

2.1.1　拥堵点的基本定义

拥堵点[9] c_{point} 是一个地理位置区域,如果车辆经过拥堵点,它会在一段时间间隔中处于拥堵点附近。拥堵点的判断依据是车辆经过拥堵点附近时速度会变慢并且速度不为零(速度为零表示停车,不一定拥堵,故排除)。计算拥堵点需要三个阈值,首尾 GPS 数据之间的距离间隔阈值 $S_{\text{threshold}}$ 和时间间隔阈值 $t_{\text{threshold}}$,以及速度阈值 $v_{\text{threshold}}$。另外还需要规定确定拥堵点所需的 GPS 数据的个数,即限定数量 n,它与数据量呈正比,但是限定数量越大,造成的误差越大。在 $S_{\text{threshold}}$ 和 $t_{\text{threshold}}$ 条件下,当限定数量 n 个 GPS 数据得到的平均速度 $\bar{v}$ 小于 $v_{\text{threshold}}$ 时,构成一个拥堵点。

2.1.2　拥堵点的计算方法

(1)根据 GPS 数据集和城市交通状况确定 $S_{\text{threshold}}$、$t_{\text{threshold}}$、$v_{\text{threshold}}$ 和 n。

(2)选取 n 个 GPS 数据点的数据子集 $P\{p_1,p_2,\cdots,p_n\}$,如果超出 $S_{\text{threshold}}$ 或 $t_{\text{threshold}}$ 则舍弃,否则计算包含 n 个 GPS 数据点的平均速度 $\bar{v}=\dfrac{\sum_{i=1}^{n-1} d(p_i,p_{i+1})}{t(p_1,p_n)}$。其中,$d(p_i,p_{i+1})$ 为两个相邻 GPS 数据之间的距离;$t(p_1,p_n)$ 为首尾两个 GPS 数据之间的时间间隔。

(3)当 $0<\bar{v}\leqslant v_{\text{threshold}}$,数据子集 $P\{p_1,p_2,\cdots,p_n\}$ 确定一个拥堵点,转到(4);当 $\bar{v}>v_{\text{threshold}}$,判断下一个 GPS 数据 p_{n+1} 是否存在。如果 p_{n+1} 存在,将数据子集 $P\{p_1,p_2,\cdots,p_n\}$ 删除 p_1 并添加 p_{n+1},转到(2);如果 p_{n+1} 不存在,则输出求得的拥堵点 GPS 数据集,并结束算法。

(4)计算拥堵点 $c_{\text{point}}=(\text{Lat},\text{Lngt},\bar{v},\text{arvT},\text{levT})$。以数据子集 $P\{p_1,p_2,\cdots,p_n\}$ 为例,c_{point} 的纬度为 $\text{Lat}=\sum_{i=1}^{n} p_i.\text{Lat}/n$,$p_i.\text{Lat}$ 为第 i 个 GPS 数据的纬度;c_{point} 的经度为 $\text{Lngt}=\sum_{i=1}^{n} p_i.\text{Lngt}/n$,$p_i.\text{Lngt}$ 为第 i 个 GPS 数据的经度;c_{point} 的到达时间为 $\text{arvT}=p_1.T$,$p_1.T$ 为 p_1 点的时间记录;c_{point} 的离开时间为 $\text{levT}=p_n.T$,$p_n.T$ 为 p_n 点的时间记录。记录拥堵点 c_{point} 的信息以构成拥堵点 GPS 数据集,并以 p_{n+1} 开始的 n 个 GPS 数据即 $P\{p_{n+1},p_{n+2},\cdots,p_{2n}\}$ 作为数据子集转到(2),如果不足 n 个,则输出求得的拥堵点 GPS 数据集,并结束算法。

2.2　DENCLUE 聚类算法

2.2.1　密度聚类算法

聚类是按照属性值把一组对象划分成一系列有意义的子集的描述性任务。它的目的是揭示样本点之间最本质的“抱团”性质。密度聚类算法根据数据周围密度的不断增长聚类,将密度足够高的区域内数据对象划分为簇,具有快速识别任意形状簇和处理数据对象中的噪声点的优点[10]。DENCLUE[11] 就是一种典型的密度聚类算法。

2.2.2　DENCLUE

密度估计是基于密度的聚类算法的核心问题,DENCLUE 就是一种泛化的基于核密度估计的聚类算法。DENCLUE(DENsity-based CLUstEring,基于密度的聚类)是 Hinneburg 等提出的,是一种基于一组密度

分布函数的聚类算法。其核心思想是每一个空间数据点通过事先影响函数对空间产生影响,影响值可以叠加,从而在空间形成一曲面,曲面的局部极大值点为一密度吸引子,该吸引子的吸引域形成一类[12]。将 DENCLUE 应用于交通数据挖掘,密度吸引子为拥堵区域的中心,吸引域为拥堵区域。

2.2.3 DENCLUE 的一些基本定义

定义 1:基本影响函数。

(1)方波影响函数。

$$f_{\text{Square}(x,y)}=\begin{cases}0 & d(x,y)>\sigma\\ 1 & \text{otherwise}\end{cases}$$

(2)高斯影响函数。

$$f_{\text{Gauss}}(x,y)=\mathrm{e}^{-\frac{d(x,y)^2}{2\sigma^2}}$$

定义 2:局部密度函数。

已知给定的数据集 $D=\{x_1,x_2,\cdots,x_n\}$,D 中的任意一点 x 的局部密度函数定义为 $f^D_{\text{gauss}}(x)=\sum_{i=1}^{n}\mathrm{e}^{-\frac{d(x,x_1)^2}{2\sigma^2}}$。其中,$d(x,x_i)$ 为点 x 和点 x_i 之间的广义距离(一般指欧氏距离);σ 为影响函数的宽度,反映了该点数据对周围影响的能力。

定义 3(梯度):局部密度函数的梯度定义为 $\nabla f^D_{\text{gauss}}(x)=\sum_{i=1}^{n}(x_i-x)\mathrm{e}^{-\frac{d^2(x,x_i)}{2\sigma^2}}$,以计算密度吸引子。

定义 4:密度吸引子。

称 $x^*\in D$ 为一个密度吸引子,当且仅当 x^* 是密度函数的局部最大值;称点 $x\in D$ 被密度吸引子 x^* 密度吸引,如果 $\exists k\in n$,$d(x^k,x^*)\leqslant\varepsilon$,使得 $x=x^0,x^{i+1}=x^i+\delta\frac{\nabla f^D_{\text{gauss}}(x^i)}{\|\nabla f^D_{\text{gauss}}(x^i)\|}$。

定义 5:中心确定的聚类。

对于密度吸引子 x^*,如存在子集 $C\subseteq D$,使得 $\forall x\in C$,x 都被 x^* 密度吸引且 $f^D_{\text{gauss}}(x^*)\geqslant\xi$($\xi$ 为预定义的密度阈值),则称 C 为以 x^* 为中心(关于 ξ,σ 的)确定的聚类。如 $x\in D$ 被局部极大值点 x^* 密度吸引,但 $f^D_{\text{gauss}}(x^*)<\xi$,则称 x 为(关于 ξ,σ 的)噪声点。

定义 6:任意形状的聚类。

对于密度吸引子集合 X,如果存在子集 $C\subseteq D$,使得:

(1) $\forall x\in C$,$\exists x^*\in X$ 使 x 被 x^* 密度吸引,且 $f^D_{\text{gauss}}(x^*)\geqslant\xi$;

(2) $\forall x_1^*,x_2^*\in X$,总存在从 x_1^* 到 x_2^* 的路径 P,满足 $\forall p\in P$ 有 $f^D_{\text{gauss}}(p)\geqslant\xi$。

则称 C 为由 X 确定的(关于 ξ,σ 的)任意形状聚类。

2.2.4 爬山算法

在对数据进行 DENCLUE 时,需要使用爬山算法计算密度吸引子 x^* 和吸引域。在 Hinneburg 的论文[11]中,密度吸引子和吸引域根据局部密度函数 $f^D_{\text{gauss}}(x)$ 和对应的梯度 $\nabla f^D_{\text{gauss}}(x)$,以公式 $x=x^0,x^{i+1}=x^i+\delta\frac{\nabla f^D_{\text{gauss}}(x^i)}{\|\nabla f^D_{\text{gauss}}(x^i)\|}$,使用爬山算法求得。由于 GPS 数据是一种含有经纬度的可排序网格数据,可以对相连的高密度网格 C_r 中的拥堵点 $c_{\text{point}}=(\text{Lat},\text{Lngt},\bar{v},\text{arvT},\text{levT},\hat{f}^D_{\text{gauss}}(x))$ 按照经纬度进行排序,从而简化爬山算法。

首先以经度大小对拥堵点排序,如果经度在同一范围(设定阈值),则按纬度大小排序。对排序后的拥堵点数据运用爬山算法,极大值点为密度吸引子,即当 $f^D_{\text{gauss}}(x^{k+1})<f^D_{\text{gauss}}(x^k)<f^D_{\text{gauss}}(x^{k-1})$,其中 $k\in n$,则记 $x^*=x^k$ 为密度吸引点,处在两个极小值之间的数据构成被密度吸引子吸引的吸引域,归于 x^* 所在的类;用此启发式方法,运用两次爬山算法(一次为纬度方向,一次为经度方向),所有的点都会归类于对应的吸引域。

2.3 拥堵区域挖掘算法步骤

1)轨迹预处理[13]。

由于采集到的 GPS 数据极易受到噪声、缺失值和不一致数据的侵扰,并且低质量的数据将导致低质量的挖掘结果,所以必须对 GPS 数据进行轨迹预处理。数据预处理的技术有很多,比如数据清理、数据过滤、数据集成、数据变换等,文中主要采用以下两种数据预处理技术:

(1)数据清理:GPS 设备刚启动或故障等原因会造成采集到大量为 0 的数据,而且 GPS 定位的误差会导致在某一时刻定位错误后,在接下来的整个时间段采集的数据都是错误的。对于这两种数据完全删除。

(2)数据过滤:GPS 传感器的噪声会造成采集到的个别数据存在误差,称为异常值(outliers)。对于个别异常值采用中值滤波器(Median Filters)进行过滤(或者纠正),即对于检测到的异常值,取其附近 n 个点的中值替换该异常值。

2)计算拥堵点。

根据过滤后的数据计算拥堵点,得到候选拥堵点 GPS 数据。

3)拥堵点聚类。

(1)对候选拥堵点 GPS 数据 D 以 2σ(σ 为设定的宽度阈值)为宽度进行网格划分,决定非空的网格集 C_p,每个网格 c 中数据数记为 N_c。

(2)设 ξ_C 为预定义的网格密度阈值,称 $C_{sp}=\{c\in C_p\mid N_C\geqslant\xi_C\}$ 为高密度网格,将相邻的高密度网格连接起来作为 C_p 的子集,记为 $C_r=C_{sp}\cup\{c\in C_p$

$|\exists c_s \in C_{sp} \text{and} \exists \text{connection}(c_s, c)\}$。其中 c_s 为与 c 相连的高密度网格,以备计算局部密度函数。

(3)用高斯密度函数 $f^D_{\text{gauss}}(x) = \sum_{i=1}^{N} e^{-\frac{d^2(x,x_i)}{2\sigma^2}}$ 计算相连高密度网格的局部密度函数。

(4)根据局部密度函数 $f^D_{\text{gauss}}(x)$,用爬山算法确定密度吸引子 x^* 以及被密度吸引子所吸引的吸引域作为标记类,密度吸引子 x^* 为拥堵区域的中心,标记类为拥堵区域,记为 c_{region},并根据吸引域中的 GPS 数据计算平均速度 v 作为此拥堵区域的平均速度。

4)拥堵区域后处理。

将上述步骤得到的拥堵区域 c_{region}(具体位置由密度吸引子的经纬度和吸引域数据确定),以及相应的密度吸引子的密度 $f^D_{\text{gauss}}(x^*)$、拥堵区域数据量、拥堵区域平均速度 v,作为城市拥堵状态信息,评价拥堵状态,以做拥堵分析。

3　拥堵区域相关性分析

3.1　相关系数计算方法的选择

相关性指标的选择对相关性分析的结果有重要影响[14-15]。线性相关系数有皮尔逊积矩相关系数、斯皮尔曼等级相关系数、肯德尔等级相关系数等几种,其中斯皮尔曼等级相关系数适用范围最广。

文中分析的是两个区域中车流量随时间变化的相关性,对 GPS 数据以 10 min 为间隔进行抽样,得到的是离散数据。当交通出现拥堵时采集到的数据量会出现尖峰,但区域间可能存在轨迹模式[16],并且车流量也是相关的。如果使用皮尔逊积矩相关系数在交通拥堵时会因尖峰导致相关性减弱,从而削弱可能存在的轨迹模式。而斯皮尔曼等级相关系数对数据条件的要求不是很严格,只要两个变量的观测值是成对的等级评定数据,或者是由连续变量观测数据转化得到的等级数据,不论两个变量的总体分布形态、样本容量的大小如何,都可以用斯皮尔曼等级相关来进行研究。所以对于文中的拥堵区域相关程度的相关系数计算,应当使用斯皮尔曼等级相关系数。

3.2　斯皮尔曼等级相关系数

斯皮尔曼等级相关(Spearman's correlation coefficient for ranked data)主要用于解决称名数据和顺序数据相关的问题。适用于两列变量,而且具有等级变量性质具有线性关系的资料。由英国心理学家、统计学家斯皮尔曼根据积差相关的概念推导而来,一些人把斯皮尔曼等级相关看作积差相关的特殊形式。

在统计学中,斯皮尔曼等级相关系数以 Charles Spearman 命名,并经常用希腊字母 ρ 表示。斯皮尔曼等级相关系数用来估计两个变量 X, Y 之间的相关性。假设两个随机变量分别为 X, Y,它们的元素个数均为 n,取两个随机变量的第 $i(1 \leqslant i \leqslant n)$ 个值,分别用 x_i, y_i 表示。对 X, Y 进行排序(同时为升序或降序),得到有序集合 X, Y,将有序集合 X, Y 中的元素对应相减得到一个差分集合 D,其中 $d_i = x_i - y_i, 1 \leqslant i \leqslant n$,从而可以使用 D 计算随机变量 X, Y 之间的斯皮尔曼等级相关系数,公式为:

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2 - 1)}$$

3.3　相关性分析方法

在城市交通中,当一个区域为严重拥堵区域时,必定会对周围区域产生影响。对于相关性较大的拥堵区域与非拥堵区域,应当解决拥堵区域的问题,以避免或缓解其对周围非拥堵区域的影响。而拥堵区域拥堵的原因可能是两个拥堵区域互相影响的结果,此时应根据它们的拥堵程度相关性来分析拥堵原因,并提出解决拥堵的方案。

根据斯皮尔曼等级相关系数公式计算出拥堵区域的拥堵程度相关系数;按照三个等级评价拥堵区域之间的相关性,最后集合拥堵区域的实际地理位置做拥堵分析。

(1) $0 \leqslant |r| < 0.4$ 为低度线性相关;

(2) $0.4 \leqslant |r| < 0.7$ 为中度线性相关;

(3) $0.7 \leqslant |r| \leqslant 1$ 为高度线性相关。

4　实验结果与分析

4.1　拥堵区域挖掘实验

为了验证基于 DENCLUE 的拥堵区域挖掘算法的效果,使用北京市 10 357 辆出租车一周的 GPS 轨迹数据作为实验数据。实验代码使用 Java 语言进行编写。由于需要使用百度地图对结果进行展示,实验需要将原始数据转换为百度地图格式的数据。

当车辆经过一个区域的平均速度较小时,认为此区域拥堵,以拥堵点记录。每个拥堵点对附近拥堵点都有一定的影响,拥堵点的影响函数使用高斯函数进行计算,因此 DENCLUE 算法中所研究的点可以描述为拥堵点,维数是二维的,即经度和纬度。在寻找高密度网格时,ξ 作为拥堵点的网格密度阈值,当网格中拥堵点数量 $N_c \geqslant \xi$ 时,形成高密度网格,说明附近拥堵点多,可聚类为拥堵区域;如果 $N_c < \xi$,说明拥堵点稀少,不足以聚类构成拥堵区域。将高密度网格连接起来聚类,以爬山算法确定拥堵区域中心点,就得到了如图 1 所示的拥堵区域分布图。

图 1 使用百度地图提供的添加热力图 API 绘制,热力区域标记黑色的强度区分拥堵程度。对于挖掘出

来的拥堵区域,根据拥堵区域的拥堵点数目对拥堵程度进行划分,越拥堵的区域,拥堵区域的拥堵点数目越多。图 1 中根据黑色标记强度的减弱,区域拥堵程度依次递减,没标记的地方代表不拥堵或者没有相关数据。可以看到有少量区域拥堵等级高,为出租车经常发生拥堵的区域。对于这些出租车经常发生拥堵的区域,找出它们的拥堵原因很重要,所以要做区域相关性分析。

图 1　基于 DENCLUE 的拥堵区域挖掘算法结果

4.2　相关性和拥堵影响分析实验

一般车辆的活动时间从早上 6 点到晚上 23 点为高峰,而晚上 23 点到早上 6 点的活动较少,这也符合人们的生活规律,所以实验选取早上 6 点到晚上 23 点的数据作为实验数据。为了分析区域之间的相关性,以区域之间数据量、平均速度和拥堵密度作为特征进行相关性分析。实验以 10 min 为时间间隔(每小时 6 次抽样)进行抽样,样本为该区域每个时间段计算数据量、平均速度和拥堵密度的加权平均值得到的拥堵程度。根据抽样得到的拥堵程度样本数据,求得拥堵区域之间拥堵程度的斯皮尔曼等级相关系数,然后结合区域的实际位置进行分析,实验如下:

这组实验选取的是地铁 4 号线辟才胡同与灵境胡同附近区域。经过拥堵区域挖掘后,可以得到如图 2 所示的热力图,位置详情如图 3 所示。图 2 标记了 1、2 两个区域,1 号区域为拥堵严重且拥堵范围大的区域,2 号区域为十字路口拥堵区域。

图 2　拥堵区域热力图

图 3　拥堵区域放大详情图

根据拥堵区域挖掘结果,可以得到两个区域的拥堵状态,其中经度和纬度为密度吸引子的位置,平均速度由车辆总体采样数据计算得到,密度吸引子密度由高斯密度函数计算得出,拥堵密度由区域的拥堵点计算得到。

为了分析两个区域的拥堵原因和拥堵影响,对每个区域数据以 10 min 为时间间隔进行抽样,将不同时间段的数据量、拥堵密度、平均速度作为拥堵程度的判定条件绘制图 4 和 5。实验中的拥堵程度主要由数据量决定,当拥堵密度非常大、平均速度非常小时才考虑其影响。

由图 4 和图 5 可以明显看出,从 13:00 开始,直到 20:00,两个区域的拥堵程度都是比其他时间段严重

的。经过计算,两区域整体的斯皮尔曼等级相关系数高达 0.821 6,说明两区域具有很强的拥堵相关性;两区域从 13:00 至 20:00 拥堵程度的斯皮尔曼等级相关系数更是高达 0.953 7,说明两区域在高度拥堵时的拥堵程度相关性极强,两区域的拥堵会互相影响。由图 3 可知,2 号区域有很多饭店,此区域是个娱乐区域,这造成了车辆在这里停留过多,导致严重拥堵。而两区域的拥堵程度具有高度相关性,尤其是拥堵时间段,也就是说两区域拥堵规律高度相关,所以 1 号拥堵区域的拥堵会直接导致 2 号十字路口的拥堵。为防止交通恶化,应当在拥堵严重的时间段加强交通管制,或者对异常拥堵区域进行重新规划,将拥堵代价降到最低。

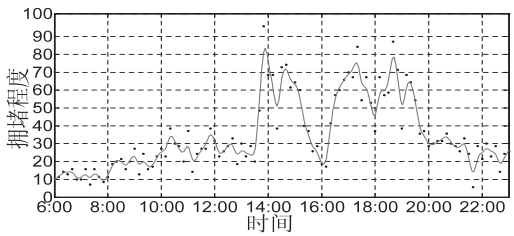

图 4　1 号区域分时段拥堵程度抽样图

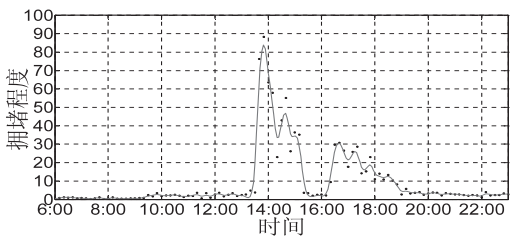

图 5　2 号区域分时段拥堵程度抽样图

5　结束语

文中提出了基于 DENCLUE 的拥堵区域挖掘算法。首先将 GPS 轨迹数据进行预处理,以最小化实验的误差;然后通过计算拥堵点,得到候选拥堵点 GPS 数据;最后使用 DENCLUE 聚类算法对候选拥堵点 GPS 数据进行聚类。此算法可以有效找出交通拥堵区域,并对拥堵状态进行分级。再对严重拥堵区域进行了相关性和拥堵原因及影响分析。实验结果表明,文中提出的拥堵区域挖掘与分析方法可以有效检测拥堵区域,查明拥堵原因和影响,为城市规划提供建议。

参考文献:

[1] 王　荻,张冠增. 城市轨道交通规划与城市规划的互动关系[J]. 城市轨道交通研究,2007,10(2):1-4.

[2] Li Fengyi, Wang Bing. The death and the life of Beijing-the history, present situation and strategy of urban planning in Beijing[J]. Canadian Social Science, 2011, 7(3): 198-201.

[3] Luo Qi. Research on intelligent transportation system technologies and applications[C]//Proc of the workshop on power electronics & intelligent transportation system. [s. l.]: IEEE, 2008: 529-531.

[4] Zheng Y, Liu Y, Yuan J, et al. Urban computing with taxicabs [C]//Proceedings of the 13th international conference on ubiquitous computing. [s. l.]: ACM, 2011: 89-98.

[5] Liu Wei, Zheng Yu, Chawla S, et al. Discovering spatio-temporal causal interactions in traffic data streams[C]//Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining. San Diego, CA, USA: ACM, 2011: 1010-1018.

[6] Pang L X, Chawla S, Liu W, et al. On mining anomalous patterns in road traffic streams[C]//Proceedings of international conference on advanced data mining and applications. Beijing, China: [s. n.], 2011: 110-118.

[7] Chawla S, Zheng Y, Hu J. Inferring the root cause in road traffic anomalies[C]//Proc of 12th international conference on data mining. [s. l.]: IEEE, 2012: 141-150.

[8] Pan B, Zheng Y, Wilkie D, et al. Crowd sensing of traffic anomalies based on human mobility and social media[C]// Proceedings of the 21st ACM SIGSPATIAL international conference on advances in geographic information systems. [s. l.]: ACM, 2013: 344-353.

[9] Zheng Y, Zhang L, Xie X, et al. Mining interesting locations and travel sequences from GPS trajectories[C]//Proc of international conference on world wide web. [s. l.]: [s. n.], 2009: 791-800.

[10] Tran T N, Drab K, Daszykowski M. Revised DBSCAN algorithm to cluster data with dense adjacent clusters[J]. Chemometrics & Intelligent Laboratory Systems, 2013, 120(2): 92-96.

[11] Hinneburg A, Keim D A. An efficient approach to clustering in large multimedia databases with noise[C]//Proceedings of the 4th international conference on knowledge discovery and data mining. New York, USA: [s. n.], 1998: 58-65.

[12] 张志兵. 空间数据挖掘关键技术研究[D]. 武汉:华中科技大学,2004.

[13] Zheng Y. Computing with spatial trajectories[M]//Computing with spatial trajectories. [s. l.]: Springer, 2011.

[14] 张尧庭. 我们应该选用什么样的相关性指标?[J]. 统计研究,2002(9):41-44.

[15] 李秀敏,江卫华. 相关系数与相关性度量[J]. 数学的实践与认识,2006,36(12):188-192.

[16] 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. [s. l.]: ACM, 2007: 330-339.