路网中空间关键字连续范围查询算法研究

李艳红[1] 黄 群[2] 蒋 宏[3] 李国徽[4]

(中南民族大学计算机科学学院 武汉 430074)[1] (武汉数字工程研究所 武汉 430074)[2]
(海军工程大学 武汉 430033)[3] (华中科技大学计算机科学与技术学院 武汉 430074)[4]

摘 要 空间关键字查询相对传统的位置相关查询而言更能满足实际查询处理的需要。着重探讨路网中结合距离和关键字相似度两个因素的空间关键字查询处理问题,提出解决路网中空间关键字连续范围查询(CRSKQ)的有效方法。提出了一个综合考虑了路网上的道路、对象和路网的连通性的路网模型以支持CRSKQ查询的处理。为了实现连续监控,所提出的算法包括两个阶段,即初始结果获取和查询结果连续监控。初始结果监控阶段,通过路网扩展和关键字匹配寻找满足要求的结果对象;在连续监控阶段,充分利用前面时刻的查询结果来减小连续监控的代价。模拟实验表明,所提出的算法是有效的。

关键词 位置相关查询,空间关键字范围查询,路网,算法

中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.07.048

Research on Processing Continuous Spatial Keyword Range Queries in Road Networks

LI Yan-hong[1] HUANG Qun[2] JIANG Hong[3] LI Guo-hui[4]

(College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China)[1]
(Wuhan Digital Engineering Institute, Wuhan 430074, China)[2] (Naval University of Engineering, Wuhan 430033, China)[3]
(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)[4]

Abstract Compared with traditional location-based queries, spatial keyword queries can better meet the requirement of actual query processing. The paper addressed the issue of processing spatial keyword queries which consider both the distance and the keyword similarity of objects, and presented an efficient method to deal with continuous spatial keyword range queries in road networks(CRSKQ). A network model which considers the road segments, the objects, and the connectivity of the road network was proposed to support CRSKQ query processing. In order to continuously monitor the query result, a method which includes two phases, the initial query result getting phase and the continuous monitoring phase, was proposed. In the first phase, the initial query result objects can be found by road network expansion and keyword matching. In the second phase, the cost of continuous query monitoring can be greatly reduced by taking advantage of the query result of the previous time. Finally, experimental result shows the efficiency of our method.

Keywords Location-based query, Continuous spatial keyword range query, Road network, Algorithm

随着通讯技术、计算机互联网和移动网的飞速发展,移动计算逐渐成为现实。信息对用户是重要的,而信息只有在适当的时间、适当的地点才是真正有用的。正是由于对位置相关信息的需求促进了位置相关服务的产生和发展。作为支持位置相关服务的一项关键技术,位置相关查询(Location Based Queries, LBQs)的处理引起了人们日益普遍的关注,成为数据库领域的一个研究热点[1-5]。近年来,为了满足位置相关服务新的要求,研究者们提出了一种同时考虑待查询对象、查询点的位置信息和它们关键字的匹配程度的新型的查询处理类型,即空间关键字查询(Spatial Keyword Query)。空间关键字查询由于较传统的位置相关查询能更好地满足实际应用的需要,受到了研究者们的广泛关注,他们纷纷投入到这个新的研究领域中。

Zhou等人[6]研究了空间关键字最近邻问题,他们对每个不同关键字构建R*树,然后针对不同的关键字搜索R*树而获得查询结果,该方法在关键字数目多的情况下效率及灵活度太低。Felipe等人[7]也研究了此问题,提出了一种R树的变体结构—IR^{2}树。IR^{2}为R树的每个结点增加了一个标记文件,以标识该结点(或其子孙结点)是否包含各关键字。这种算法适合于考察对象点是否包含某些关键字的情形,但受到了标记文件严重局限性的影响,其错误匹配数与集合的大小呈线性关系。Wu等人[8]讨论了空间关键字连续最近邻查询问题,他们采用加倍带权的Voronoi单元作为查询的安全区域,从而保证当用户的活动局限在此安全区域内时,其查询结果是持续有效的。Lu等人[9]研究了空间关键字反向最近邻查询问题,并提出了一种交-并R树(IUR树)来处理该问题。Li等人[10]提出了一种方向感知的索引结构来处理方向感知的空间关键字查询,所提出的索引结构能够从距离和方

到稿日期:2013-09-27 返修日期:2013-11-04 本文受国家自然科学基金项目(61309002)资助。

李艳红(1973—),女,博士,副教授,主要研究方向为现代数据库,E-mail:anddylee@163.com;黄 群(1966—),女,高级工程师,主要研究方向为信息技术;蒋 宏(1974—),男,讲师,主要研究方向为信息系统;李国徽(1973—),男,博士,教授,主要研究方向为现代数据库、实时系统。

向两方面来消减查询空间，所以有较高的查询处理效率。Wu等人[11]讨论了联合的空间关键字查询，在IR树的基础上提出了W-IR树来方便查询的处理。W-IR树的结构与IR树相同，只是在构建时先考虑单词的划分再考虑对象的位置，因此较IR树有更好的性能。

已有的空间关键字查询算法大都做了简单的假设，即假定对象处在欧氏空间环境中。而现实中对象都是运动在路网空间中，对象之间的距离取决于所在路网的连通性。已有的算法不能完全满足现实应用的需要。位置相关查询主要包括范围查询、(连续)k近邻查询和反向(连续)k近邻查询等。本文着重探讨路网中空间关键字连续范围查询(CRSKQ)问题，提出相应的查询处理算法。首先，提出了一个包含邻接表、路段折线表、对象表和路网R树的路网模型。该模型既考虑了道路的联通性信息，也支持路网距离计算和关键字匹配检查，适合于路网空间关键字查询的处理。基于所提出的路网模型，提出路网空间关键字连续范围查询处理算法。该算法包括两个阶段，即初始结果获取和查询结果连续监控。初始结果监控阶段，首先通过查询路网R树索引结构定位查询点q所在的路段，然后通过路网扩展和关键字匹配寻找满足要求的结果对象构成初始结果集。在连续监控阶段，以前面时刻的查询结果为起点，求解出可能满足查询要求的对象构成候选对象集，并计算候选对象与查询点间的路网距离函数。这些函数都是相对于查询点移动距离的线性函数，从而可以计算出查询结果发生变化的位置点，称为result change point(RCP)，在各个RCP点对查询结果集进行修正，从而保证查询结果的持续有效性。模拟实验表明，所提出的算法是有效的。

1 问题描述及采用的网络道路模型

1.1 问题描述

本文研究路网中空间关键字连续范围查询问题。如图1所示，路网中有一组空间关键字对象(o_1，o_2，o_3，o_4，o_5，o_6)，用黑色圆点标识。对象的关键字信息见括号内。假设有一个移动查询点q(用黑色三角形标识)，它所在的位置为$q.l$。图中，虚线圆圈给出了以$q.l$为中心的查询范围，即以$q.l$为圆心、查询距离r为半径的圆。请注意，由于这里讨论的是路网中空间关键字范围查询，这里所说的距离都是路网距离。也就是说，真正的查询范围是以$q.l$为中心、距离$q.l$的路网距离不超过查询距离r的点所构成的封闭区域。实际的查询范围可能不是一个规整的圆形，这里为了描述的方便用圆形代替。

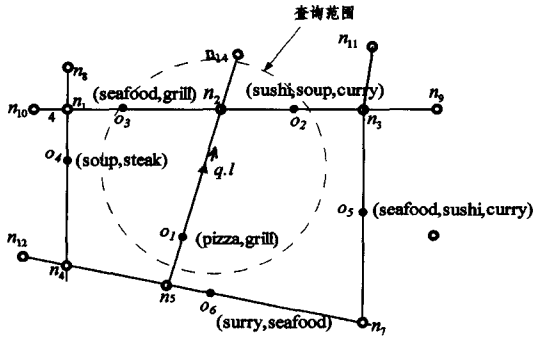

图1 路网中空间关键字连续范围查询示例

假定查询点q希望查找从当前时刻点($t=0$)起未来一段时间内的、包含关键字“sushi”和“curry”、位于查询范围内的所有对象。由图中可知，在$t=0$时刻位于查询范围内的对象有3个，即o_1，o_2和o_3。但这3个对象中，只有o_2包含查询点q的全部两个关键字。因此，综合考虑关键字和距离两个因素，$t=0$时刻q的空间关键字范围查询的结果是对象o_2。由于这里讨论的是连续范围查询，应根据查询点q的位置变化情况，给出及时、准确的查询结果。

1.2 路网模型

路网中位置相关查询与欧氏空间下位置相关查询的最大区别就是对象间的距离计算方式不同：在欧氏空间下，对象间的距离由对象的相对位置决定。给定两个对象的坐标，就可以很容易获得对象间的距离；而在路网中，对象间的距离由对象所在路网的连通性所决定，因此两个看似很接近的对象其路网距离可能很大，这给位置相关查询处理带来了困难。在进行查询处理时，可能不得不考虑大量的对象，并计算它们与查询点间的路网距离才能最后确定查询结果。而空间关键字查询比一般的位置相关查询更加复杂，除了考虑距离之外还需要考虑关键字的相似程度，给查询处理带来更大的难度。因此，如何对路网结构进行合理的划分、组织，综合考虑路网距离与关键字相似度两方面的因素，以支持空间关键字查询的处理，缩小查询处理的搜索范围，是非常重要的。

本文采用的路网模型综合考虑了上述几个方面的要求，具体包括以下4部分。

1. 邻接表。邻接表表示路网中道路的连通性，为路网中每个结点存储以下信息：各邻接点的存储位置、与邻接点形成路段的网络距离以及对应的MBR(Minimum Bounding Rectangle，最小边界矩形)、对应路段在路段折线表中的存储位置。

2. 路段折线表。路段折线表为每个路段存储以下信息：路段的具体形状、路段对应端点在邻接表中的存储位置。

3. 对象表。对象表给出了对象的具体信息：对象所在的路段、对象在路段中的具体位置、对象的关键字列表。

4. 网络R树[12]。利用网络R树对路段折线表的MBR进行索引，以支持对路网某点所在路段的快速查询。

2 路网中空间关键字连续范围查询算法研究

本节讨论路网中空间关键字连续范围查询(CRSKQ)处理算法，具体包括两个阶段：查询初始结果集的获取和查询结果的连续监控。

2.1 获取查询的初始结果集

首先，根据查询点q的位置搜索路网R树，以定位q所在的MBR，进而确定q所在的路段。然后，采用由近至远的顺序进行路网扩展，并按照相遇的先后顺序依次检测各目标，避免不必要的磁盘访问和路网距离的计算代价。算法设置优先队列L，以保存路网扩展中待访问的结点。队列L中元素按距离查询点q的路网距离由小到大的顺序排列。设置候选对象集合$Cand_set$以存放候选结果对象。如图1所示，首先定位查询点q所在的路段n_2n_5，搜索该路段上的对象并将对象o_1加入候选结果集$Cand_set=\{o_1\}$。考察该路段两个端点n_2和n_5到q的路网距离：由于n_2到q的路网距离小于查询距离r，也即n_2位于查询范围之内，因此将n_2连同其到q的

路网距离值加入队列 L。而另一端点 n_5 到 q 的路网距离大于查询距离 r，其位于查询范围之外，则不加入队列 L，该方向的探寻结束。考察队列 L，由于 L 为非空，因此将 L 队头元素 n_2 出队。依次考察与 n_2 相连的其他未被访问的路段 n_2n_9 和 n_2n_{10}，并将其上的、位于查询范围内的对象 o_2 和 o_3 加入候选集 $Cand\_set=\{o_1,o_2,o_3\}$ 中。n_9 和 n_{10} 由于均位于查询范围之外，因此不加入队列 L。此时，队列 L 为空，路网探寻结束。接下来，用查询点 q 的关键字集对候选集中对象进行过滤。由于只有对象 o_2 包含查询点 q 的全部关键字，因此将 o_2 加入空间关键字范围查询结果集 $q.RSK$。该阶段的伪代码见算法 1。

算法 1 GetInitRSK

```
input:query point q(q. l,q. keywords)
output:Return the range spatial keyword query result of q(q. RSK)
Begin{
(1)List L=∅,Set Cand_set,q. RSK=∅;
(2)Search network-R tree to locate the MBR including q;
(3)Locate the road segment e where q is moving on;
(4)Insert all the object o locating on e,supposing that dist(o,q)≤r,
   into Cand_set; // dist(o,q) is the network distance from o to q
(5)Insert the two end points n of e ,supposing that dist(n,q)≤r,into
   L in ascending order of distance
(6)while(L not empty)
(7)  {p=dequeue_head(L) // dequeue the head element of L
(8)    for(each non-visited node n which is connected to p)
(9)     Insert all the objects o on road segment np, supposing that
        dist(o,q)≤r, into Cand_set;
(10)       Insert n into L if dist(n,q)≤r;}
(11)  }
(12)For(each object o in Cand_set)
(13)  { Insert o into q. RSK if o. keywords includes all the keywords
        in q. keywords;}
(14)Return q. RSK;
```

2.2 查询结果的连续监控

为了处理连续位置相关查询，一种直观的方法就是连续不断地调用 2.1 节所介绍的静态查询算法，获得各个时间点的查询结果集。很显然，这种做法没有能够利用之前时刻的查询结果来加速后续查询的处理，因此是低效的。本文所采取的方法充分考虑了查询结果对后续查询处理的有利作用。请注意，查询点 q 到达路网中路段的端点后，可能选择的路段有多条，这里只处理 q 到达路段端点时刻之前的连续时间段内的查询监控。对于 q 到达端点后的时间段，可以作为一个新的查询进行处理。首先，调用 2.1 节的初始结果集获取算法 GetInitRSK 以获得 $t=0$ 时刻的结果集 $q.RSK$。然后，根据 q 的移动方向，用 $q.l$ 所在路段的、靠近 q 移动方向的端点 n，调用 GetInitRSK 算法，并获得 $n.RSK$。依次考虑 $q.RSK$、$n.RSK$ 及位于从 $q.l$ 到端点 n 的路段上的各对象，将其关键字集包含查询点 q 的所有关键字的对象都加至 $Cand\_set$ 集。

由于 q 是连续移动的，候选结果集中各对象相对于 q 的距离会随时发生变化。为了实现连续监控，以 $q.l$ 为起点，将 q 离开 $q.l$ 的距离设为 x，可以求解 $Cand\_set$ 中各对象相对于 q 的距离值，它们都是 x 的一次函数。根据所得的一次函数，求解 $Cand\_set$ 中各对象的距离值达到邻接距离 r 的点（称为查询结果变化点 RCP），这些 RCP 点会将监控的路段分为若干子段，而每个子段内的查询结果集由上个子段的结果集加上在分界点加入的对象、减去因距离增大而离开监控范围的对象构成。本阶段的具体伪代码见算法 2。

算法 2 MonitorRSK

```
Input:The initial query result set(q. RSK)
Output:The final result set(q. Final_RSK_Set)
Begin{
(1)List RCP_List=∅,Set RSK_set=q. RSK, Cand_set,q. Final_
   RSK_Set=∅;
(2)Let n be the end point of the segment e which q moves towards;
(3)Call GetInitRSK using n as the parameter;
(4)For each object o in the union of(q. RSK,n. RSK and the objects
   located on part of segment from q. l to n);
(5)  {insert o into Cand_set if o. keywords includes all the keywords
      in q. keywords}
(6)For(each object o in Cand_set)
(7)  {get the distance function fdist(q,o,x) which is a liner function
      of x;//x is the distance which q leave its original position q. l
(8)    if fdist(q,o,x1)=r at some point x1,insert(x1,o) into RCP_
       List in ascending order of distance;}
(9)x1=q. l;
(10)while(RCP_List is not empty)
(11)  {(x2,o)=dequeue_head(RCP_List)// dequeue the head ele-
      ment of RCP_List
(12)    Insert([x1,x2],RSK_set) into q. Final_RSK_Set;
(13)    if(o is belong to RSK_set)
(14)      delete it from RSK_set;
(15)    else insert it into RSK_set;
(16)  Insert([x2,n],RSK_set) into q. Final_RSK_Set;
(17)Return q. Final_RSK_Set;
```

3 模拟实验

本节通过模拟实验评估所提出的空间关键字连续范围查询处理算法的性能（以下简称 CRSKQ 方法）。这里选择一种直观处理方法作为参考算法（以下称为 LS 方法）。该 LS 方法在监控的时间段内，通过连续不断地发起 2.1 节所介绍的静态查询处理方法来获取查询结果集。为了简便起见，假设查询点 q 在每 10 个时间单位发一个静态查询。通过测量两种方法在处理路网空间关键字连续范围查询时所需的平均运行时间，来评价所提算法的性能。

3.1 实验环境和实验参数

这里采用一个真实的路网来测试算法的性能。为了建模现实生活中的真正路网，采用了美国三藩市地区路网的真实数据，构建一个包含有 20000 条边的子路网。表 1 给出了控制实验的参数，表中加粗的字体给出的是缺省的参数值。实验中，随机发起 1000 次查询，实验结果为查询处理的平均运行时间，用秒表示。

表 1 实验参数表

时间间隔/时间单位	对象数/个	关键字数/个
10	10000	1
30	20000	2
50	**40000**	**3**
70	60000	4
90	80000	5

3.2 实验结果

图 2 评价了查询时间段的长度对 CRSKQ 和 LS 算法运行时间的影响。如图 2 所示，随着监控的查询时间段的增长，这两种算法的运行时间也增大。对于 CRSKQ 而言，这是因为随着查询时间的变长，查询点 q 到达路段端点的可能性增大，则需要发起新的查询的可能性也增大；对于 LS 而言，其原因在于所需发起的静态查询的次数随着查询时间段的增长而增多。

图 3 评估了系统内对象数对算法性能的影响。由图 3 可知，这两种算法的运行时间随着对象数的增加而有所增大。其原因在于，随着对象数的增加，系统内对象的密度增大，位于监控范围内的对象数也会增大。因此，查询处理所需的对象距离值计算和比较工作量也相应增加，从而加大了查询处理的时间代价。

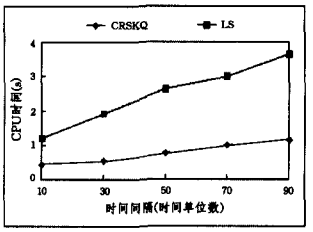

图 2 查询时间段长度对算法运行时间的影响

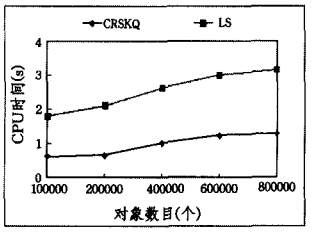

图 3 对象数对算法运行时间的影响

图 4 评估了变化的关键字数目对这两种算法运行时间的影响。如图 4 所示，LS 算法的性能随着关键字数目的增加稍有增大。而 CRSKQ 算法的运行时间随关键字数目的增加而有所减少。对于 CRSKQ 而言，关键字多时，算法处理过程将过滤掉较多的不满足关键字要求的对象，从而减少监控的候选对象数，则相应的候选对象距离值计算和比较的操作也会相应减少，从而减少查询处理所需的 CPU 运行时间。

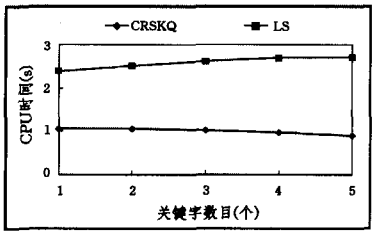

图 4 关键字数目对算法运行时间的影响

结束语 近几年来，研究者开始关注综合考虑距离因素和关键相似性的空间关键字查询处理问题。但现有的研究成果大都是局限于欧氏空间，不适用于路网中空间关键字查询的处理。本文重点讨论了路网中连续空间关键字范围查询问题，提出了有效的查询处理算法。算法分为初始结果获取和查询结果连续监控两阶段，能够充分利用前面时刻的查询结果来减少查询结果连续监控的代价，具有良好的性能。最后，模拟实验验证了所提算法的有效性。

参 考 文 献

[1] Papadias D, Zhang Jun, Mamoulis N, et al. Query processing in spatial network databases[C]// Proc of VLDB. Gerlin: Morgan Kaufmann, 2003: 802-813

[2] Kolahdouzan M, Shahabi C. Voronoi-based k-nearest neighbor search for spatial network databases[C]// Proc of VLDB. Toronto: Morgan Kaufmann, 2004: 840-851

[3] Mouratidis K, Yiu Manlung, Papadias D. et al. Continuous nearest neighbor monitoring in road networks[C]// Proc of VLDB. Seoul: ACM Press, 2006: 43-54

[4] Huang Yuan-ko, Chen Zhi-wei, Lee Chiang. Continuous K-Nearest neighbor query over moving objects in road network[C]// Proc of APWeb-WAIM. Suzhou: Springer, 2009: 27-38

[5] Zheng Bai-hua, Xu Jian-liang, Lee Wang-chien, et al. Grid-partition index: a hybrid method for nearest-neighbor queries in wireless location-based services[J]. The International Journal on Very Large Data Bases, 2006, 15(1): 21-39

[6] Zhou Y, Xie X, Wang C, et al. Hybrid index structures for location-based web search [C]// Proc of ACM CIKM. Bremen: ACM Press, 2005: 155-162

[7] Ed Felipe I, Hristidis V, Rishe N. Keyword search on spatial databases [C]// Proc of ICDE. Cancun: IEEE, 2008: 656-665

[8] Wu D, Yiu M L, Jensen C S, et al. Efficient continuously moving top-k spatial keyword query processing [C]// Proc of ICDE. Hannover: IEEE, 2011: 541-551

[9] Lu J, Lu Y, Cong G. Reverse spatial and textual k nearest neighbor search[C]// Proc of SIGMOD. Athens: ACM Press, 2011: 349-360

[10] Li G, Feng J, Xu J. DESKS: Direction-Aware Spatial Keyword Search[C]// Proc of ICDE. Washington DC: IEEE, 2012: 474-485

[11] Wu D M, Yiu M L, Cong G, et al. Joint Top-k Spatial Keyword Query Processing [J]. IEEE Transaction on KDE, 2012, 24 (10): 1889-1903

[12] Guttman A. R-Tree: A Dynamic Index Structure for Spatial Searching[C]// Proc of ACM-SIGMOD International Conference on Management of Data. San Jose: ACM Press, 1995: 47-57

(上接第 209 页)

[13] Pevny T, Bas P, Fridrich J. Steganalysis by Subtracting Pixel Adjacency Matrix[J]. IEEE Transactions on Information Forensics Security, 2010, 5(2): 215-224

[14] Nash J. Equilibrium Points in n-Person Games[J]. Proceedings of the National Academy of Sciences of the United States of America, 1950, 36: 48-49

[15] Kodovsky J, Fridrich J. On Completeness of Feature Spaces in Blind Steganalysis[C]// Proc of the 10th ACM Multimedia & Security Workshop. New York: ACM Press, 2008: 123-132

[16] Lyu S, Farid H. Steganalysis Using Higher-Order Image Statistics[J]. IEEE Transactions on Information Forensics and Security, 2006, 1(1): 111-119

[17] Orsdemir A, Altun H O, Sharna G, et al. Steganalysis Aware Steganography: Statistical Indistinguishability Despite High Distortion [C]// Proc of Security, Forensics, Steganography, and Watermarking of Multimedia Contents X. Bellingham: SPIE Press, 2008, 6819: 9-18

[18] Schaefer G, Stich M. UCID: An Uncompressed Color Image Database [C]// Proc of SPIE Electronic Imaging, Storage and Retrieval Methods and Applications for Multimedia. Bellingham: SPIE Press, 2003, 5307: 472-480

[19] 陈园园，朱孝成，叶甫谕. 一种改进的 DCT 信息隐藏算法[J]. 重庆理工大学学报：自然科学版，2011，25(12)：100-105