浙 江 林 学 院 学 报 2007，**24**(3)：363—368
Journal of Zhejiang Forestry College

文章编号：1000-5692(2007)03-0363-06

空间分析中的空间数据结构

汪杭军，方陆明，张广群
（浙江林学院 信息工程学院，浙江 临安 311300）

摘要：空间分析是基于地理对象位置和形态特征的空间数据分析技术，它在众多领域中应用非常广泛。空间分析的效率由空间数据结构和主要空间操作所决定。阐述了空间数据及其特征，从散列访问方法到四叉树、*k*-d 树、R 树、R^+ 树、R^* 树、S 树和两阶段树等各种树型结构，综述了主要空间数据结构，并通过不同结构之间的比较，得到不同空间结构的优缺点。熟悉空间数据结构及其特点可以指导选择空间分析的方法。图 6 表 1 参 22

关键词：林业工程；空间分析；空间数据结构；空间关系；拓扑分析；地理信息系统
中图分类号：S757　　文献标志码：A

地理信息系统(GIS)是为了解决资源与环境等全球性问题而发展起来的技术与产业。空间分析是基于地理对象的位置和形态特征的空间数据分析技术，其目的在于提取和传输空间信息[1]，从而达到认知、解释、预报和调控[2,3]。它是 GIS 区别于其他信息系统主要标志，体现了 GIS 的本质特征。空间分析的具体应用领域非常广泛，包括水污染监测、城市规划与管理、地震灾害和损失估计、洪水灾害分析、矿产资源评价、道路交通管理、地形地貌分析、医疗卫生、军事领域和图像检索等[4]。GIS 虽然具有管理地理信息和处理图形的能力，也具备一定的空间分析能力，但随着 GIS 应用的深入，GIS 还依赖于空间分析模型的研究以及与 GIS 的结合来解决实际中的一些复杂问题[5]。文章讨论在空间分析中所采用的空间数据结构，介绍了在空间数据挖掘中所采用的数据结构。

1 空间数据及其特征

空间数据具有复杂的结构，因为它涉及空间特征、属性特征和时间特征等 3 个方面的基本特征。空间数据在计算机中的存储方式有矢量模型和栅格模型 2 种。不同的存储方式在空间数据结构的采用上有些差异，以下所讨论的空间数据结构主要是针对矢量模型的。

在进行空间分析时，需要使用空间数据挖掘来提高分析的效率。而在空间数据挖掘中，则需要使用有效的空间数据结构来加快数据挖掘的过程。空间数据结构与空间操作是密不可分的，不同的空间操作需要采用不同的空间数据结构。空间操作包括基本操作(如获得对象的类型、边界、面积和长度等信息)、空间关系运算(如判断 2 个对象是否相等、相离、相交、包含和重叠等)和空间分析操作(如 2 个对象间的距离、对象间的交和并、合并等操作)等 3 个大类，它直接关系到空间分析任务的完

收稿日期：2006-05-25；修回日期：2007-02-07
基金项目：浙江省科学技术重点项目(2005C23061)；浙江省教育厅资助项目(20040699)
作者简介：汪杭军，讲师，硕士，从事数据挖掘、智能信息处理研究。whj@zjfc.edu.cn。通信作者：方陆明，教授，博士，从事资源与环境信息系统以及计算机网络应用研究。E-mail：fluming@126.com

成和效率。因此在实际应用中，主要的空间操作就决定了采用什么样的空间数据结构来提高程序的运行效率。

2 空间数据结构概述

2.1 散列访问方法

典型的散列方法有格子方法、格子文件方法[6] 和 EXCELL 方法[7] 等。格子文件是格子方法的变形，主要用于点查询中，是散列方法的代表。在一个格子块里的所有记录都是存储在相同的存储桶中。只要格子块可以合并构成记录空间的 k 维矩形，几个格子块就可以共享一个存储桶。为了响应符合精确匹配的查询，人们首先使用标尺来定位出包含查找点的单元。如果该格子单元不在主存中，需要一次磁盘存取，并装入有可能包含匹配数据的参考页的单元。

为了提高时空效率和利用率，有人又提出了二级格子文件[8] 和孪生格子文件[9] 的方法。

2.2 四叉树及其变形树

四叉树是最早用于处理高维数据的数据结构之一[10]。它是二分查找树在多维上的扩展，现在已经被广泛应用于地理信息系统中。

四叉树具有 1 个根节点，每个中间节点有 4 个孩子。四叉树的每个节点对应 1 个正方形区域。如果一个节点 v 有孩子，那么这个区域就被平均分成 4 个小的正方形区域，分别用 NE，NW，SW 和 SE 命名，代表了 4 个方向。图 1 给出的是四叉树的例子。

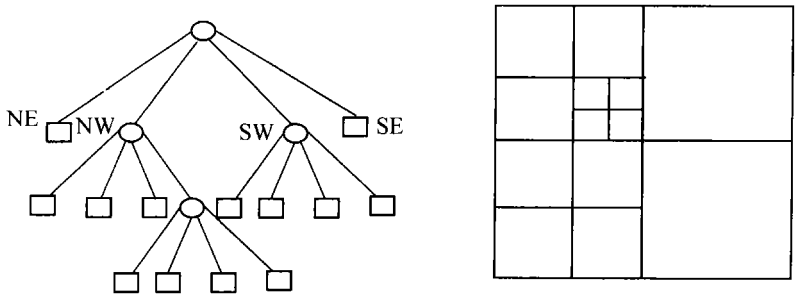

图 1 四叉树和对应的区域划分

Figure 1 Quadtree and the corresponding subdivision

对由 0 和 1 组成的图像阵列区域，区域四叉树[11] 是最重要的四叉树表示方法。它把有边界的图像阵列划分连续的 4 个相同大小的正方形区域。而 PR 四叉树[12] 适合用来存储点数据，它与区域四叉树按同样的方式组织，不同点在于 PR 四叉树的叶子节点要么为空，要么含有 1 个数据点及其坐标。

2.3 k-d 树

k-d 树[13] 也是早期的多维数据结构，它是 k 维的二叉搜索树。树中的每个中间节点把 k 维空间分成 2 个 $k-1$ 维超平面，即顶层节点按一维划分，下一层节点按另一维进行各个维循环往复。当 1 个节点中的点数少于给定的最大点数时结束划分。图 2 显示了 2-d 树的例子。在树的偶数层可以比较 x 轴的值(树根是 0 层)，奇数层比较 y 轴的值。

k-d 树会出现非平衡树，为解决此问题出现了自适应 k-d 树[14]。它在划分时选择把空间分成 2 个拥有相同数量的点的子空间作为超平面。超平面仍然与轴平行，但不包含点，并且无需在各维上进行切换。为了适合于大型的空间数据库，考虑二级存储的分页问题，k-d-B 树[15] 结合了 k -d 树和 B 树的特点。它允许用多个超平面划分 1 个节点，所有的节点都对应磁盘上的分页。

2.4 R 树及其变形树

2.4.1 R 树 R 树是 1984 年 Guttman 提出用于处理地理数据和高维空间数据的结构[16]。它是基于 B^+ 树的分层数据结构，用于动态组织通过最小边界矩形表示的 d 维几何对象。树中每个节点对应于框

住孩子的最小边界矩形，树叶包含指向数据库对象的指针，每个节点作为一个磁盘分页实现。R 树的应用具有很广的范围，从空间和时间到图像和视频(多媒体)数据库。图 3 是 R 树的例子。

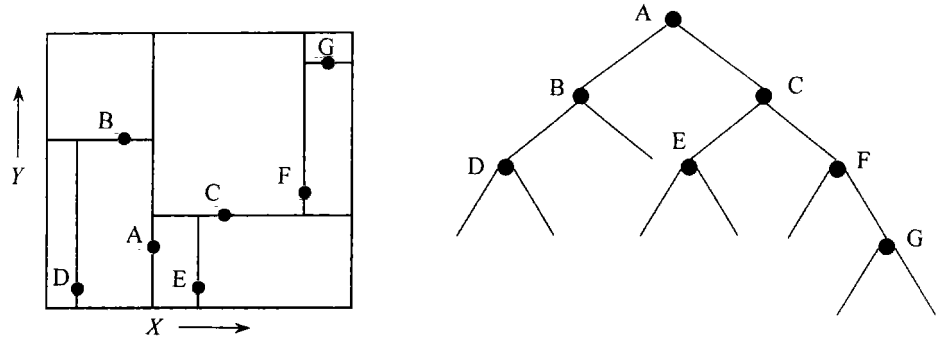

图 2　平面点的分 布及 对应的 2-d 树

Figure 2　A distribution of points in the plane and the correspondent 2-d tree

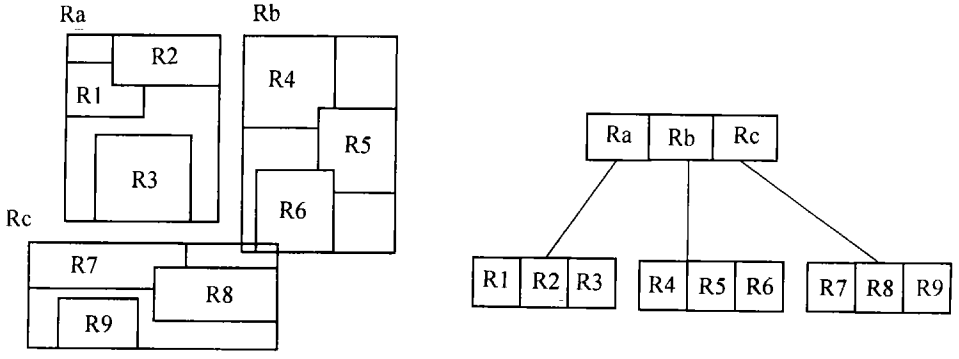

图 3　平面区域及其 R 树

Figure 3　A set of 2-d objects and the correspondent R-tree

2.4.2　R^+ 树　Sellis 等在 1987 年提出了 R^+ 树[17]，它不允许在相同的层上有最小边界矩形重叠的现象发生，这样某些对象可能会被复制并在不同的节点中冗余存放，在窗口查询中可以空间换取存取时间上的高效。但同时也带来了副作用：最小边界矩形增加可能导致一系列链表操作上的更新，另外在一定的条件，该结构可导致死锁发生。图 4 显示了 R^+ 树的例子，其中 R1，R4 和 R7 对象重复存储。

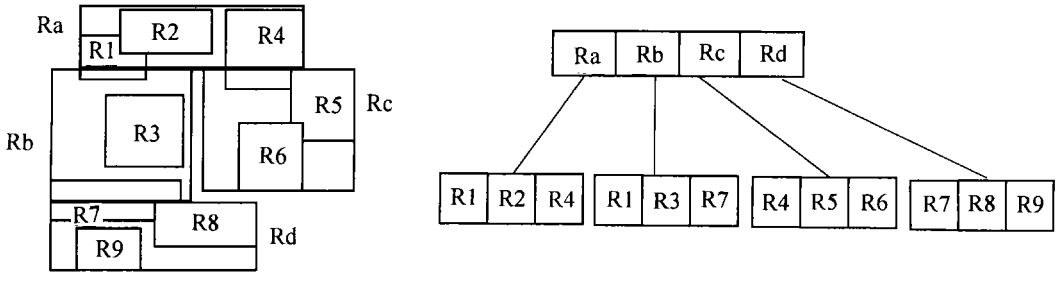

图 4　平面区域及其 R^+ 树

Figure 4　A set of 2-d objects and the correspondent R^+ tree

2.4.3　R^* 树　由于对象的重复存储，R^+ 树的查询要求沿着树的若干路径进行，插入对象和删除对象也可能涉及 1 个以上的中间节点，于是在 1990 年提出了 R^* 树结构[18]。在作性能比较时，现在 R^* 树

仍然是作为较好性能的结构被广泛接受。R* 树不受每个节点最大孩子的限制，并具有复杂的节点分裂技术，特别是应用了“强迫重插”的技术。另外一个显著的特点是它除了考虑使产生的最小边界矩形面积之和最小外，还要使在同层的最小边界矩形间覆盖的面积最小，并且使产生的最小边界矩形的周长最小。这使得 R* 树的插入算法比原始的 R 树要提高很多。

2.5 S 树

Samet[19] 提出了基于二叉树的 S 树，可以显著提高检索效率。对一幅二值图像，它递归地将图像划分成 2 个相同大小的子图像直到子图像都是黑色或白色为止，这可得到二叉树。如图 5 所示例子。

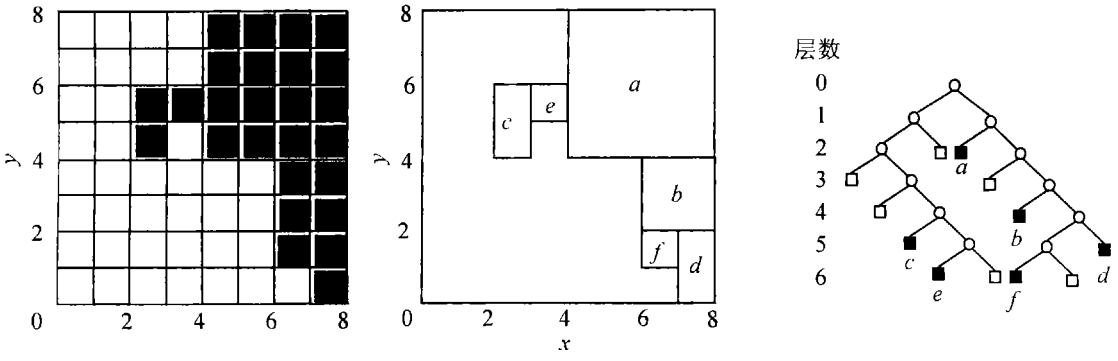

图 5 图像及其二叉树表示

Figure 5 An image and the correspondent binary tree

广度优先遍历该二叉树，将结果存储为 2 个数组 linear-tree 表和 color 表可得 S 树。在遍历二叉树时，当遇到叶子节点(内节点)在 linear-tree 表中存入 1 (0)；同时，当遇到黑色(白色)叶节点，将 1 (0)存入 color 表中。如图 6 的 S 树表示如下：

linear-tree table：0 00 0110 1010 1010 1001 1111

color table：010001111010

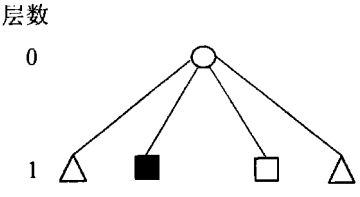

图 6 对图 5 的近似四叉树

Figure 6 Approximate quadtree to figure 5

2.6 两阶段树

Chung 等[20] 在 2003 年提出了两阶段 S 树表示，进一步压缩内存需求，并有较快的计算性能。在第一阶段采用任意一种前面提到的树结构表示二值图像。但并不是将图像划分到全部是黑色或白色的子图为止，而是划分到某层就停止了。在该层树中有 3 类叶子节点：黑叶节点、白叶节点和灰叶节点。其中每个黑叶(白叶)节点代表了整个黑色(白色)的子图；灰叶节点是所采用树结构的一个子树。由于灰叶节点是耗内存部分，为了减少内存的需求，在第二阶段，使用连接部件串(connected component string)对灰叶节点进行编码。连接部件串是能有效表示子图中连接部件的位流。通过在连接部件分析中提出的形态技术可以获得灰叶节点所表示子图的连接部件串。

3 空间数据结构比较

前面提到的空间数据结构与空间操作是密不可分的，针对空间数据的特定操作往往需要选择特定的空间数据结构，现在没有哪一种空间数据结构被公认为是最好的。

评价一个空间数据结构的性能依赖于许多因素和参数，例如数据分布是否均衡、模型是否适合、空间存取方法、数据的数量、数据空间的分布密度、聚类的程度等。同时存在一些不可预知的影响，包括所用硬件、操作系统组件、缓冲区大小、页的大小以及数据集合等。另外，性能通常是根据磁盘存取数量、查找时间和删除时间等进行衡量。表 1 给出了上面讨论的空间数据结构的特点比较。

表 1　空间数据结构性能比较

Table 1　Performance comparison on spatial data structure

空间数据结构	优点	缺点
散列访问方法	在较小查询区域下，针对交集查询和包含查询，具有较好的性能[21]	需要针对应用构造散列函数(方法)
四叉树	容易且有效计算多边形的数量特征；阵列各部分的分辨率可变，以较少存储量精确表示图形	转换的不定性，不利于形状的分析和模式识别
k-d 树	对于查询和插入新节点的操作是比较方便的	删除操作很复杂(会导致删除节点以下的树重组)树结构依赖于插入顺序，导致非平衡树
R 树	使用最小边界矩形代表对象，极大地改善了查询、插入和删除等操作的性能	存在兄弟节点出现交叉的问题，从而使得搜索效率低下
R^+ 树	在点查询中避免访问多重路径[22]	不允许在相同的层上有最小边界矩形重叠，某些对象可能会重复存放
R^* 树	不受每个节点最大孩子的限制；在同层的最小边界矩形间覆盖的面积最小，产生的最小边界矩形的周长最小	构造 R^* 树的时间比较长
S 树	基于二叉树的表示，其空间需求比线性四叉树要好	需要有大量的空间存储叶子和内部节点
两阶段树	通过表示和编码 2 个阶段，通过高效的连接部件串编码，进一步压缩内存需求，并有较快的计算性能	第一阶段只能针对已提出的树型结构，可能不能很好地与其它高效的空间结构进行结合

4　结论

空间分析在现代的地理信息系统和空间数据挖掘等领域中显得越来越重要了，而空间分析的效率问题始终是实际应用中，特别是针对大型数据分析最为关心的问题。空间分析的效率又是由主要空间操作和所采用的空间数据结构决定的。由于空间数据的复杂性，对于大型空间数据库，图结构方式往往需要耗费大量的空间开销，因此空间数据结构以线性结构和树结构 2 个大类为主，特别是树结构在 GIS 中的应用是相当广泛的。

参考文献：

[1] 郭仁忠. 空间分析[M]. 北京：高等教育出版社，2001.

[2] 姜亚莉，张延辉. GIS 空间分析的应用领域[J]. 四川测绘，2004，**27**(3)：99－102.

[3] 郝成元，吴绍洪，杨勤业. 空间分析与自然地理综合研究[J]. 地理与地理信息科学，2005，**21**(1)：52－55.

[4] 余柏蒗，吴健平，魏晓峰，等. 空间分析 GIS 软件开发研究[J]. 测绘与空间地理信息，2004，**27**(5)：14－17，23.

[5] 吴元洪. 空间数据结构分析[J]. 计算机应用研究，2004，**3**：39－41，84.

[6] NIEVERGELT J，HINTERBERGER H，SEVCIK K C. The grid file：an adaptable，symmetric multikey file structure[J]. *ACM Trans Database Syst*，1984，**9**(1)：38－71.

[7] TAMMINEN M，SULONEN R. The EXCELL method for efficient geometric access to data[C] // *Proceedings of the* 19*th Conference on Design Automation Annual ACM IEEE Design Automation Conference*. Piscataway：IEEE Press，1982：345－351.

[8] HINRICHS K. Implementation of the grid file：Design concepts and experience[J]. *BIT*，1985，**25**(4)：569－592.

[9] HUTFLESZ A，SIX H W，WIDMAYER. Twin grid files：Space optimizing access schemes[C] // *Proceedings of the* 1988 *ACM SIGMOD international Conference on Management of Data*. New York：ACM Press，1988：183－190.

[10] FINKEL R，BENTLEY J L. Quad trees：a data structure for retrieval of composite keys[J]. *Acta Information*，1974，**4**：1－9.

[11] SAMET H. The quadtree and related hierarchical data structure[J]. *ACM Comput Surv*，1984，**16**(2)：187－260.

[12] ORENSTEIN J A. Multidimensional tries used for associative searching[J]. *Inf Process Letters*，1982，**14**(4)：150－157.

[13] BENTLEY J L. Multidimensional binary search trees used for associative searching[J]. *Commun ACM*，1975，**18**(9)：509－517.

[14] FRIEDMAN J H, BENTLEY J L, FINKEL R A. An algorithm for finding best matches in logarithmic expected time [J]. *ACM Trans Math Software*, 1977, **3**(3): 209—226.

[15] ROBINSON J T. The *k*-D-B-tree: A search structure for large multidimensional dynamic indexes [C] // *Proceedings ACM SIGMOD Conference on Management of Data*. New York: ACM Press, 1981: 10—18.

[16] GUTTMAN A. R-trees: A dynamic indx structure for spatial searching [C] // *Proceedings ACM SIGMOD Conference on Management of Data*. New York: ACM Press, 1984: 47—57.

[17] SELLIS T, ROUSSOPOULOS N, FALOUTSOS C. The R^{+}-tree: a dynamic index for multidimensional objects [C] // *Proceedings 13th VLDB Conference*. Brighton: Morgan Kaufmann, 1987: 507—518.

[18] BECKMANN N, KRIEGEL H P, SCHNEIDER R, *et al*. The R^{*} tree: an efficient and robust method for points and rectangles [C] // *Proceedings ACM SIGMOD conference*. New York: ACM Press, 1990: 322—331.

[19] SAMET H. *The Design and Analysis of Spatial Data Structures* [M]. New York: Addison-Wesley, 1990.

[20] CHUNG K L, HWANG S L, ChEN I C. New two-phase spatial data structures with application to Binary images [J]. *J Visual Commun Image Represent*, 2003, **14**: 97—113.

[21] GAEDE V, GNTHER O. Multidimensional access methods [J]. *ACM Comput Surv*, 1998, **30**(2): 170—231.

[22] STONEBRAKER M, SELLIS T, HANSON E. An analysis of rule indexing implementations in data base systems [C] // *Proceedings 1st Conference on Expert Database Systems*. South Carolina: Benjamin Cummings, 1986: 465—476.

Spatial data structure in spatial analysis

WANG Hang-jun, FANG Lu-ming, ZHANG Guang-qun

(School of Information Engineering, Zhejiang Forestry College, Lin'an 311300, Zhejiang, China)

Abstract: Spatial analysis is a data analysis technique based on the geographical location and morphological characters, which has been widely used in many fields. Its efficiency is directly decided by the main operation and spatial data structure. From hashing access methods to different tree structures, provided a survey on spatial data structure and presented their characteristic through comparison between all these spatial data structures. [Ch, 6 fig. 1 tab. 22 ref.]

Key words: forest engineering; spatial analysis; spatial data structure; spatial relation; topology analysis; geographic information system (GIS)