

基于 R 树索引的三维可视化查询方法

龚俊¹ 谢潇²

(1 江西师范大学地理与环境学院, 南昌市紫阳大道 99 号, 330022)

(2 武汉大学测绘遥感信息工程国家重点实验室, 武汉市珞喻路 129 号, 430079)

摘要:针对应用日益广泛的最近邻查询,提出了一种基于 R 树广度遍历和优化排序原理的最近邻查询算法,能适应不同空间分布的目标数据集。同时,提出了多细节层次(LOD)目标查询方法。实验证明,此方法支持多尺度场景逼真描述,查询结果准确,满足当前三维 GIS 的功能需求。

关键词:R 树; 空间索引; 三维空间查询; 细节层次

中图法分类号:P208

三维可视化查询是空间数据库查询的应用延伸,也是人机交互的重要功能^[1]。传统 GIS 应用中,可视化查询基于二维空间,三维可视化查询方式相对于二维应用有着较大差异。数据和计算复杂度的陡增使三维可视化查询的实现难度远大于二维查询。

空间索引是提高空间查询效率的关键技术。为了提高三维空间查询性能,传统二维索引被扩展至三维空间。其中由数据驱动的 R 树索引因具有良好的可扩展性和效率,被认为是最有前途的三维空间索引方法^[2-3]。在通过改进节点的选择和利用节点分裂算法改善节点的形状后,R 树索引因节点重叠导致多路查询激增带来的执行效率问题也得到了大幅优化^[4-6]。在保证查询精度的基础上,本文利用改进的 R 树索引,来提高复杂三维场景中典型可视化查询方法的效率。

1 最近邻查询

在人们对空间对象分布不熟悉的情况下,最近邻查询是极有意义的,但用传统的范围查询需要多次变化查询半径才能成功。

本文设计了一种基于 R 树的最近邻顺序查询算法——以宽度优先遍历 R 树,采用查询点到节点包围盒的最小距离 MinDist 值作为判断条

件。最近邻顺序查询算法采用一个优先队列保存已经访问过的所有节点和目标,队列中的元素根据其 MinDist 值排序存放。该方法是一种递增的最近邻查询算法,使查询点的最近邻能够逐个被找到,对于已知 k 个最近邻,查找第 $k+1$ 个最近邻时,也不需要重复计算。

算法流程:从根节点开始,计算其子要素和查询点的 MinDist,根据该值按照升序原则将子要素逐一加入优先队列中,然后再将队列中第一个要素取出。如果要素是节点,则计算其子要素和查询点的 MinDist,同样按照升序原则将它们逐一加入已有的优先队列;如果要素是目标且没有判断过真实距离,则取出目标的完整模型数据,计算查询点到目标的真实距离,同样加入优先队列;如果要素是目标且已判断真实距离,则该目标即为当前最近邻目标,如果还没有达到最近邻目标个数要求,则继续取出队列中第一个要素进行判断,直到找到指定数目 k 个的最近邻目标。

2 几何查询

2.1 视锥体查询

通常,视锥体内的目标仅占完整场景的较小部分,因此,视锥体裁剪对于优化数据调度和三维可视化效率来说极其关键。

利用三维 R 树优化视锥体查询的关键问题是确定三维 R 树节点与视锥体的空间关系。本文将两者的关系定义为:如果节点的 8 个顶点均在视锥体内,则它完全包含于视锥体内;如果节点的 8 个顶点均在视锥体的某个平面外,则它完全落于视锥体外。实际应用中,其余情况仅占很小一部分,存在相交和相离两种关系,其中相交的可能性更大,可以作为相交处理。解决节点与视锥体的关系问题后,即可从根节点开始自上而下判断目标和视锥体的关系。

2.2 多边形查询

利用三维 R 树优化屏幕多边形查询的关键问题是确定三维 R 树节点与多边形的空间关系。针对这个问题,本文提出一种降维法,其原理是:三维 R 树节点投影到屏幕平面成为二维图元,进而计算二维图元和查询多边形的空间关系。R 树节点形状是轴向包围盒,它在屏幕上的投影形状是凸多边形。三维节点变换为屏幕凸多边形后,调用两个多边形空间关系计算函数解算节点和查询条件的空间关系。

计算节点的屏幕图形需要计算节点各个顶点的屏幕坐标,它可以通过两个步骤获得:第一步实现世界坐标到裁剪坐标的变换,第二步利用公式

$$\begin{aligned} X &= ((x + w)/w) W/2.0 \\ Y &= ((z + w)/w) H/2.0 \end{aligned} \quad (1)$$

实现仿射变换将裁剪坐标变换到屏幕坐标。计算 R 树节点中 8 个顶点的屏幕坐标后,利用凸壳算法推算节点的屏幕投影图形。

需要特别指出的是,如果点落于视锥体的反面即视线后方时, w 为负值,此时不存在屏幕投影。因此,如果节点中某些顶点落于视锥体的反面时,需要调整节点的屏幕图元生成算法,算法描述如下。

算法 1 描述:节点的屏幕图元生成算法。

算法入口:三维节点;算法出口:屏幕图元。

- 1) 生成空的二维点集;
- 2) 判断节点 8 个顶点投影是否落于屏幕中,将屏幕中的二维投影点坐标加入点集合;
- 3) 判断节点的 12 条边同视锥体的侧面是否存在交点,如果存在,将交点的二维投影点坐标加入点集合;
- 4) 判断视锥体的 4 条棱边同节点的 6 个面是否存在交点,如果存在,将对应的屏幕角点加入点集合;
- 5) 如果点集合为空,则无屏幕图元;如果非空,计算点集凸壳,该凸壳即为屏幕图元;
- 6) 退出。

3 多细节层次目标查询

10 多年来,基于 R 树索引出现了多种细节层次目标查询功能,目前关于 R 树和多细节层次模型结合的研究都试图采用 R 树的天然层次结构实现目标查询和细节层次查询的双重功能^[7-9]。但这类方法均采用节点包围盒作为低细节层次描述,忽略了单个目标的 LOD 描述需求。

本文依托 R 树的层次结构,将各个尺度上关注的目标分别存储在对应的 R 树层次上的节点中,使得 R 树具有尺度特性。实际应用时,根据节点的视距判断是否启用节点中目标。因此,本文扩展 R 树结构,允许在中间层上管理目标。每层设置一个适用范围,包括最近距离和最远距离,相邻层的适用范围无缝拼接。同层中所有节点的适用范围相同,若视点和节点的距离落于该范围内,即为访问节点中的目标模型。在全景描绘时,只需访问根节点中的目标。随着视点接近,视域逐渐减小,关注细节逐渐提高,从根节点访问其子节点,根节点中的目标元组仅是其子节点中的重要目标,因此,关注的目标集合有所增加,细节层次增强。上层节点中的目标是从其子节点中选择出来,从其每个子节点中选择一个最重要的目标,如高度最大的目标。从叶节点开始,逐层向上生成所有上层节点中的目标集合。

多细节层次目标生成后,如何实现多细节层次检索成为本文方法实用化的关键问题。多细节层次模型检索是一个与视点密切相关的功能,远目标属于上层节点(甚至根节点)管理,近目标属于下层节点(甚至叶节点)管理,检索过程以根节点为入口,借助三维空间查询技术裁剪视场之外的节点和目标。本文方法生成的全部场景均可见,视点近的区域目标密度高,视点远的区域目标密度低,符合人类的视觉感知规律。随着视点逼近,某层节点的作用范围大于视距时,将进入下层节点检索。

算法 2 描述:多细节层次模型检索算法。

算法入口:视点和目标;算法出口:待绘制的目标集合。

- 1) 设 R 树根节点为 Node;
- 2) 判断 Node(最小包围盒)与视锥体的空间关系,如果相离,终止该分支;如果被视锥体包含,转到步骤 3);如果相交,转到步骤 5);
- 3) 如果视距大于节点的细节层次作用范围,则终止该分支;如果视距落于节点的细节层次作用范围,则将节点中的目标全部加入目标集合,并

终止该分支;如果视距小于节点的细节层次作用范围,则将该节点中的目标全部加入目标集合,并转到步骤4);

4) 循环访问各子节点($Child_0, Child_1, Child_2, \dots$),将子节点逐一设为 Node 并转到步骤3);

5) 如果视距大于节点的细节层次作用范围,则终止该分支;如果落于节点的细节层次作用范围,则逐一判断节点中的目标与视锥体的空间关系,如果非相离则加入目标集合,并终止该分支;如果小于节点的细节层次作用范围,则将该节点中不与视锥体相离的目标加入目标集合,并转到步骤6);

6) 循环判断各子节点($Child_0, Child_1, Child_2, \dots$)与视锥体的空间关系,如果非相离,则将子节点设为 Node 并转到步骤2);

7) 如果所有查询分支均终止,则退出。

4 实验结果分析

本实验测试数据为某城市部分城区的离散建筑数据,目标对象 23 951 个。其中,目标模型均采用目标的最小包围盒表示,便于关注空间索引技术本质,实际单个对象包括几何和纹理数据量可能超过 1 M 数据量,单个对象允许存在多个不同细节层次模型。测试环境配置为:CPU Intel Duo T7500,内存 1 G。

4.1 视锥体查询

视锥体查询的测试内容包括查询准确度和效率,本文采用以下方法进行测试:查询准确度通过三维画面和测试数据说明,查询效率采用测试数据评价。在视锥体外的目标被裁剪,不予显示,如果视锥体内无目标,则无目标绘制,裁剪效果最为明显。图 1 是开启和关闭裁剪效果的对比图,其中黑框是视锥体的透视投影图形,关闭视锥体裁剪时绘制 23 951 个目标,开启视锥体裁剪时绘制 2 839 个目标,与黑框相交和被黑框包含的目标全部保留,与黑框相离的目标全部剔除,说明视锥体查询的准确性。

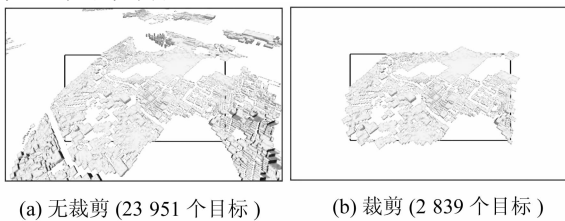


图 1 视锥体裁剪效果

Fig. 1 Effect of View-fructum Culling

三维 R 树索引的视锥体查询操作的时间开销主要耗费在 R 树结构的换层迭代运算、视锥计算(判断视锥体和长方体的关系)和更新查询结果集合等三个方面。测试结果表明,1 万次视锥计算时间大约为 0.000 1 s。表 1 是多次不同查询条件下的查询开销及查询目标数目,其中包括各种典型临界状态。测试数据表明,场景包含于视锥体内时,查询耗时最多,场景完全落于视锥体外时,查询耗时最少。可以得出结论:相对于视锥计算,R 树遍历是相对费时的操作。实时应用要求每帧处理时间在 0.04 s 左右(25 帧/s),查询操作仅耗费 1/30 的时间,因此,对上万级别目标集合实施三维查询操作也能满足实时需求。

表 1 不同查询条件下的测试结果

Tab. 1 Test Results in Different Query Situations

| 查询条件 | 查询时间/s | 目标数目/个 | 视锥计算次数/次 |
|------------------|----------|--------|----------|
| 根节点完全落于视锥体内(最耗时) | 0.001 4 | 23 591 | 1 |
| 根节点完全落于视锥体外(最高效) | 0.000 04 | 0 | 1 |
| 视锥计算次数最多 | 0.000 65 | 12 886 | 1 101 |

4.2 最近邻空间查询

最近邻空间查询的测试项目包括查询准确度和效率。查询准确度通过三维画面和测试数据说明,查询效率采用测试数据评价。图 2 中黑点是查询点,最近邻查询目标数目为 40,深色目标即为查询结果,共有 40 个目标,整个查询过程访问到 61 个节点和 58 个目标,仅占有所有节点和目标的极小部分,而且误访率低。选择其他查询点测试,测试数据相差不大,表明该算法具有很好的稳健性和适用性。



图 2 最近邻查询

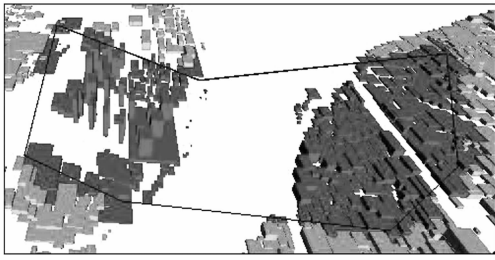
Fig. 2 KNN Query

4.3 屏幕多边形查询

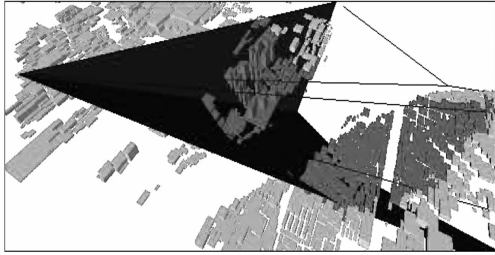
二维屏幕空间查询与传统空间查询相似,查询条件均是屏幕上的二维几何形状,如点、多边形、矩形和折线等,实际上是三维空间中的射线、多棱锥或者折面。图 3 是多边形查询效果。

4.4 多细节层次目标查询

多细节层次目标查询的作用是,在保持三维



(a) 屏幕查询结果



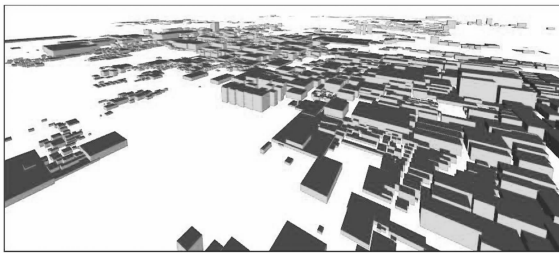
(b) 侧视图

图 3 多边形查询

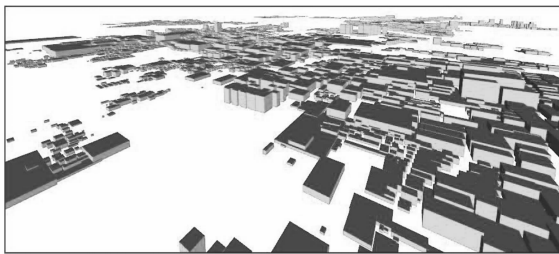
Fig. 3 Polygon Query

场景逼真度的前提下尽可能剔除次要目标和保留重要特征,实验重点是考察细节层次目标查询结果能否保持场景真实性,以及场景简化程度。

图 4 是在平视条件下的对比图,相对于全细节场景,细节层次查询结果中的目标数目减少了 80% 以上,但是视觉效果相差并不明显,达到多细节层次场景描述的要求。图 5 是在俯视条件下的对比图,细节层次查询结果的目标数目减少了 70% 以上,视觉效果虽有一定差异,但是重要特征全部保留,同样满足多细节层次场景描述的要求。



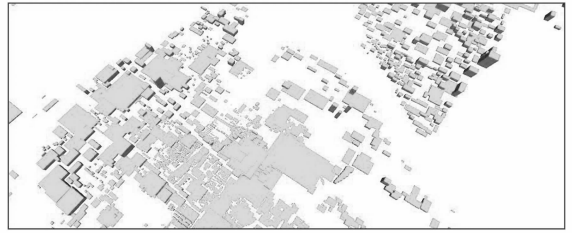
(a) 多细节层次场景 (3 712 个目标)



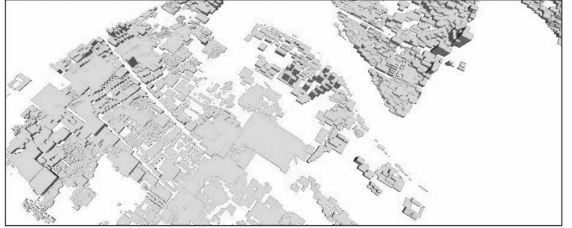
(b) 全细节场景 (22 192 个目标)

图 4 细节层次目标查询结果和全细节场景对比(平视)

Fig. 4 LOD and Non-LOD Scene in Flat View



(a) 多细节层次场景 (3 435 个目标)



(b) 全细节场景 (12 921 个目标)

图 5 细节层次目标查询结果和全细节场景对比(俯视)

Fig. 5 LOD and Non-LOD Scene in Eagle View

致谢:感谢朱庆教授给作者提供的理论和方法指导。

参 考 文 献

- [1] 吴立新,史文中. 地理信息系统原理与算法[M]. 北京:科学出版社, 2003
- [2] Guttman A. R-trees: A Dynamic Index Structure for Spatial Searching[C]. ACM SIGMOD, Massachusetts, Boston, 1984
- [3] Zhu Qing, Gong Jun, Zhang Yeting. An Efficient 3D R-tree Spatial Index Method for Virtual Geographic Environments[J]. ISPRS Journal of Photogrammetry and Remote Sensing, 2007, 62(3):217-224
- [4] 龚俊,朱庆,张叶廷,等. 顾及多细节层次的三维 R 树索引扩展方法[J]. 测绘学报, 2011, 40(2):249-255
- [5] 朱庆,龚俊. 一种改进的真三维 R 树空间索引方法[J]. 武汉大学学报·信息科学版, 2006, 31(4): 340-343
- [6] 陈鹏,孟令奎,宋杨. 三维 GIS 中基于空间拓扑约束条件的 R 树研究[J]. 武汉大学学报·信息科学版, 2007, 32(4):347-349
- [7] Kofler M. R-trees for Visualizing and Organizing Large 3D GIS Databases[D]. Austria: Graz University of Technology, 1998
- [8] Zlatanova S. 3D GIS for Urban Development[D]. The Netherlands: ITC, 2000
- [9] 李军,景宁,孙茂印. 多比例尺下细节层次可视化的实现机制[J]. 软件学报, 2002, 13(10):2 037-2 043

第一作者简介:龚俊,博士,副教授,硕士生导师,现从事多维地理信息系统和三维激光点云处理的理论与方法研究。

E-mail:gongjunbox@gmail.com

(下转第 1153 页)