

电缆三维模型可视化及数据高效索引研究^①



孙小虎¹, 李 扬², 许 刚²

¹(国网经济技术研究院有限公司, 北京 102209)

²(华北电力大学 电气与电子工程学院, 北京 102206)

通讯作者: 李 扬, E-mail: 2277257354@qq.com

摘 要: 针对电缆三维可视化场景的加载速度慢问题, 提出一种用于电缆工程场景下的三维模型外表面提取简化算法与多细节层次 R-树索引数据调度组织方法. 首先对占据三维场景中大量内存的电缆井和管沟模型进行 LOD 层级简化, 实验结果显示数据量大幅度减小; 然后根据多细节层次的 R-树索引结构对简化后的数据进行组织调度, 与传统 R-树相比, 该方法构建的 R-树在节点筛选和节点分裂时构造了更优的树形, 使得在进行数据的索引和调度时, 对电缆工程三维场景加载速度的提高有明显效果, 有效地实现了电缆工程中三维模型的流畅展示.

关键词: 多细节层次; 外表面提取算法; 模型简化; R-树索引; 三维场景

引用格式: 孙小虎, 李扬, 许刚. 电缆三维模型可视化及数据高效索引研究. 计算机系统应用, 2020, 29(7): 40-47. <http://www.c-s-a.org.cn/1003-3254/7458.html>

Research on Cable 3D Model Visualization and Data Efficient Index

SUN Xiao-Hu¹, LI Yang², XU Gang²

¹(State Grid Economic and Technological Research Institute Co. Ltd., Beijing 102209, China)

²(School of Electrical and Electronic Engineering, North China Electric Power University, Beijing 102206, China)

Abstract: In view of the slow loading speed of cable 3D visualization scene, a simplified algorithm for extracting the outer surface of 3D model and a multi-level of detail R-tree index data scheduling organization method are proposed. Firstly, the LOD level simplification of cable well and pipe trench models occupying a large amount of memory in three-dimensional scene is carried out, and the experimental results show that the amount of data is greatly reduced. Then, the simplified data is organized and scheduled according to the R-tree index structure of multi-level of detail. Compared with the traditional R-tree, the R-tree constructed by this method constructs a better tree shape in node selection and node splitting, which makes the data in progress. In the process of indexing and scheduling, it can improve the loading speed of 3D scene of cable engineering, and effectively realize the smooth display of 3D model in cable engineering.

Key words: multi-level of detail; external surface extraction algorithms; model simplification; R-tree index; 3D scene

目前电缆三维设计成果体系已基本建立, 数字化设计成果的数据量越来越大, 三维可视化作为数字化成果的一部分, 其数据量也变得越来越大. 尤其在电缆工程的三维可视化中, 由于电缆和支架的几何结构简单, 并不会占据大量的内存空间, 而大量结构复杂的电缆井和管沟的三维模型则占据了接近 80% 的内存空

间, 因此对电缆井和管沟进行模型简化, 减少存储的数据量是当前研究工作的重点与难点所在. 另外受限于软硬件技术的发展水平, 计算机无法实现短时间内加载大量数据的要求, 而且由于电缆铺设过程中具有线路繁复、模型复杂而且地形不确定性大等特点, 使得电缆铺设场景中三维模型的数据组织更加困难. 因此

① 收稿时间: 2019-11-22; 修改时间: 2019-12-16; 采用时间: 2019-12-20; csa 在线出版时间: 2020-07-03

必须寻求一种适用于电缆工程的三维数据简化和组织调度方法,来实现三维场景的快速显示和交互。

当前对电缆工程可视化的研究主要集中在三维管线可视化的研究,文献[1]在原有的平面二维系统基础上引入了高程信息,实现了由二维到三维的转变.文献[2]主要利用近景测量技术与结构实体法对管线进行了三维建模,然而该研究主要对地上管线有效,对于重建大范围地下管线及管理具有一定的难度.在文献[3]提出了一种去除三维模型冗余结构的简化方法,但对于结构复杂的模型会存在过度简化的情况,无法保留模型的特征信息.同时在文献[4]中介绍了一种基于 SQL Server 和 OpenGL 的管线模型构建方法.在文献[5]中介绍了基于 ArcGIS Engine 平台进行综合三维管线可视化的方法.文献[6]为了运用 GRASS 实现三维可视化,提出了基于 PostgreSQL 和 PostGIS 设计并构建地下管线综合数据库,利用 AutoCAD 进行管线三维建模的方法.在文献[7]中建立一种计算模型顶点曲率,简化三角面片模型,由曲率来决定权重比重的方法,并将权重最小的三角形去除.文献[8]对网格优化算法进行了改进,引入了递进网格的相关概念.而文献[9]为了提高大量数据条件下的地形渲染的效率,提出了一种六边形的四剖分算法,提升了 LOD 模型简化的精度,提高了模型渲染的加载速度。

当前研究虽然对三维数据加载速度的提高有一定效果,但数据存储量仍然较大.针对目前存在的问题,提出一种三维模型外表面提取算法来对电缆工程中的模型进行简化,减小数据量.因为电缆工程所具有的特殊性,在三维场景中,较低的 LOD 层级 LOD1-2 的情况下,电缆线和支架占据内存量较大,电缆井和管沟所占内存较小;而在实际工程中,大量应用的则是包含内外结构的更高等级的 LOD,即 LOD3-4.在 LOD3-4 中,电缆井和管沟则占据了大量的内存,电缆线和支架则对整个工程的数据量影响很小,如图 1 所示.所以将研究重点置于对电缆井和管沟进行简化的研究上^[8,9].

1 基于外表面提取算法的电缆场景三维模型简化设计

1.1 三维场景下的电缆工程细节层次模型

针对电缆场景下的电缆和支架模型结构较为简单的情况,在实际工程中,只需对其进行 LOD1-2 的两级定义.电缆的两级 LOD 组织如下: LOD1 是矢量点、线,即显示的是电缆管线整体特征; LOD2 是电缆管线

的粗模,是细节层次较为丰富的模型,包含了电缆的外层结构^[10-12].而支架的两级 LOD 组织则是: LOD1 是不具有厚度信息的矩形模型; LOD2 是长方体的支架模型,如图 2 所示。

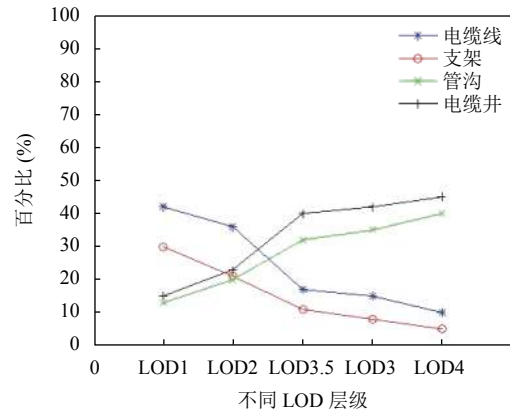


图 1 不同模型在不同 LOD 层级下所占内存比例

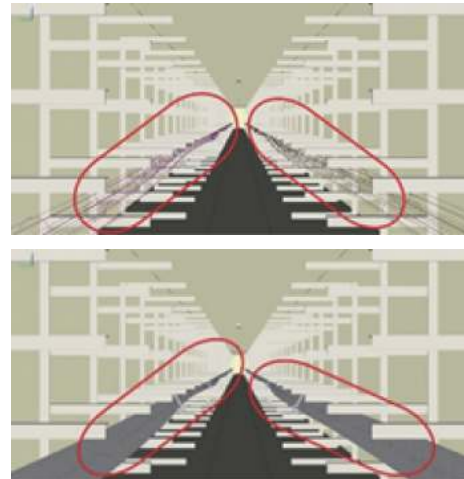


图 2 工程中的电缆 LOD 等级对比

相对于工程中结构较复杂的模型,如电缆井、管沟而言,则需要 LOD 层级更高更丰富的 LOD 模型来进行描述. LOD4 层级的模型通常包含大量且较为详细的几何和语义的信息^[13].与 LOD3 模型相比, LOD4 模型包含了模型的具体内部结构和特征. LOD3 与 LOD4 之间的差异是巨大的.因为 LOD4 模型不再是整体的块模型,而是包含了对所描述物体的内部结构表示.正是因为有了这样详细的信息, LOD4 模型可以支持各种三维场景下的应用展示,但同时由于 CityGML 中的 LOD4 模型在文件大小方面过于庞大,无法在时效性要求较高的应用场景中有效地呈现和传输数据.虽然与

LOD4 相比, 转化为 LOD3 模型后数据大小减小了很多, 但模型细节简化不够精细, 仍具有较多冗余部分而且有着较大的数据量^[14,15]. 为此提出了一个新的子 LOD, 它具有更小的文件大小, 更加简化的几何结构. 由于新提出的 LOD 包含内部结构特征, 即在 LOD4 和 LOD3 之间, 所以命名为 LOD3.5. LOD3.5 模型既具有丰富的数据, 足以进行可视化; 又能够支撑广泛的应用程序, 且需要比 LOD3 模型小得多的存储空间. LOD3.5 和 LOD3 模型差距大的原因如下:

(1) 在 LOD3 模型中, 一些模型表面的结构, 如电缆井的管线入口和出口具有较为复杂的形状, 而在 LOD3.5 中, 则只用一个表面来代表这些结构.

(2) 在 LOD3 模型中, 包含着模型外表面的厚度信息, 而在 LOD3.5 中则不存在.

因此, LOD3.5 模型可以保留三维模型的重要特性, 同时具有更小的数据大小.

如图 3 列举了电缆井中的转角井、四通井以及管沟进行 LOD 模型简化后的效果示意图.

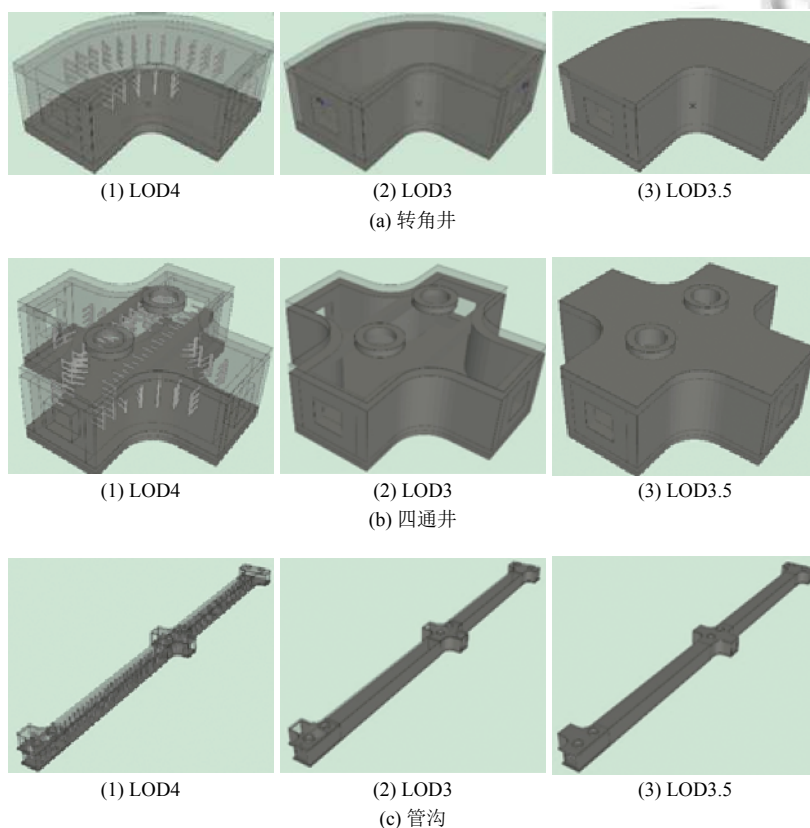


图 3 3 种不同 LOD 等级下的模型对比图

1.2 模型外表面提取算法

在对电缆工程三维模型进行简化时, 提取模型的外部表面是必须进行的步骤. 虽然外部特征和内部特征可以在 CityGML 中定义, 但一些输入模型可能并不一定能够将这些特征区分为外部和内部. 例如, 从 IFC 或 KML 等其他数据格式转换而来的 CityGML 模型可能不会定义每个表面是内部还是外部. 因此, 我们需要为一些输入的模型找到外表面, 同时, 这也是 LOD4 到 LOD3 转换的一个重要步骤. 在三维环境中, 电缆井和管沟的外表面难以提取, 来形成 CityGML 的 LOD3 模

型. 首先, 一个模型可能有不同的布局以及不同的高度, 这样不能通过简单地挤压来代表模型本身. 其次, 计算机程序很难确定电缆井和管沟的内部结构, 也很难找到它们的外表面, 而且在建模过程中还可能会出现大量的曲面结构, 增加了实现模型高精度简化的难度. 针对这些存在的问题, 对光线跟踪算法进行改进, 提出了一种模型外表面提取算法^[16-18]. 经过数据处理, CityGML 模型被分解成多个具有相应语义信息的块模型, 通过检查这些块模型的可见性来找到外表面. 我们的算法如下: 第一步是找到外部, 它是建筑的一个包围

球体,如图4所示.接下来,检查每个表面对边界球上的点的可见性.表面的可见性是由光线跟踪算法确定的,假设观察光线从球面上的点发射.例如,如图4所示,点P11在包围球上的任何一点上都不可见,因此包含该点的任何表面都不能位于建筑模型的外部外壳上.如果从表面随机生成的3个点在边界球上都是可见的,那么这些表面就被认为是可见的,因此在建筑的外壳上也是可见的.为了避免一些外部表面在边界球中不可见的故障判断,程序还会检查生成的外部壳体的连续性.如果在表面上存在孔洞,程序将再次检查孔洞内的表面,看看它们是否是外部的.算法的流程如算法1.

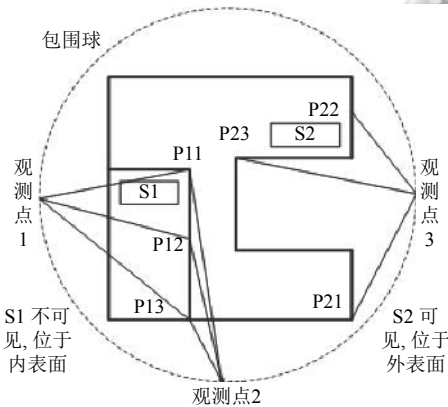


图4 外表面提取算法示意图

算法1. 模型外表面提取算法

- 1) 输入: 电缆工程中的电缆井模型数据集;
- 2) 找出建筑物的边界球 S ;
- 3) 将初始观测点阵列 $P[]$ 分别作为边界球 S 的上、下、左、右端点.
- 4) 进行循环.
- 5) 从输入中选取一个曲面 s , 并生成 s 的数学表达式.
- 6) 从 s 中随机选取3个点, 每一个点 p 都进行如下操作.
- 7) 检查 p 对 $P[]$ 中所有点的可见性.
- 8) 如果被某些表面遮挡, 将每个边界的中心加到 $P[]$ 中, 重复上一步.
- 9) 直到将所有的表面进行检查或返回得到一个点存在为止.
- 10) 如果 $P[]$ 中的任何点没有被遮挡, 则报告 p 为可见; 否则, 报告 p 为不可见.
- 11) 如果这3个点都是可见的, 则返回 s 为可见; 否则, 返回 s 为不可见.
- 12) 将返回得到的所有可见表面作为外部表面.
- 13) 输出: 电缆井模型的 LOD3.5.

使用光线跟踪算法对建筑进行可见性检验, 对于从边界球上的给定点 s 射向目标表面上的点 p 的每一条光线, 可以表示为:

$$I_s + (I_p - I_s)t, t \in R \quad (1)$$

其中, $I_s = (x_s, y_s, z_s)$ 和 $I_p = (x_p, y_p, z_p)$ 表示点 s 和点 p 的坐标. 对于模型中其他每一个点 p , p 所在的平面可以表示为:

$$p_0 + (p_1 - p_0)u + (p_2 - p_0)v, uv \in R \quad (2)$$

其中, $p_k = (x_k, y_k, z_k) k=0,1,2$ 是曲面 p 上3个非共线点的坐标. 光线与平面的交点 p' 计算公式为:

$$I_s + (I_p - I_s)t = p_0 + (p_1 - p_0)u + (p_2 - p_0)v \quad (3)$$

可以解得 (t', u', v') 为:

$$\begin{bmatrix} t' \\ v' \\ u' \end{bmatrix} = \begin{bmatrix} x_s - x_p & x_1 - x_0 & x_2 - x_0 \\ y_s - y_p & y_1 - y_0 & y_2 - y_0 \\ z_s - z_p & z_1 - z_0 & z_2 - z_0 \end{bmatrix}^{-1} \begin{bmatrix} x_s - x_0 \\ y_s - y_0 \\ z_s - z_0 \end{bmatrix} \quad (4)$$

如果 $p' = I_s + (I_p - I_s)t'$ 在表面 P 上, 则对 p' 进行检测, 如果在 P 上, 则 p 在 s 上是不可见的. 完成外部提取算法后, 将外部壳体表面转化为 LOD3 建筑构件, 生成 LOD3 模型. 其他表面将在转换过程中被移除.

在实际的电缆工程应用中, 与 LOD4 模型相比, LOD3.5 模型减少对三维模型厚度的渲染并不会影响实际的电缆模型展示效果和可视分析.

2 基于多细节层次的三维 R-树索引数据调度方法

针对电缆工程三维数据量大, 存储路径多且复杂等特点以及其特有的查询需求, 特提出了一种基于多细节层次的三维 R-树索引数据调度方法, 来对上文简化后的电缆井和管沟的三维 LOD 数据进行组织调度^[14]. 该方法的实施分为以下3个步骤.

(1) 节点筛选. 根据节点筛选算法的原则: 先从底层节点向上进行筛选, 再从最高层节点向下进行一遍筛选. 接着将目标模型的数据插入. 由此可以避免节点重叠的情况出现, 也就可以避免由此带来的选择错误问题, 进而可以选出最优叶节点.

(2) 节点分裂. 对于兄弟节点中所出现的各种情况, 如上溢现象、重叠等, 采用二分为三的算法进行分裂操作, 将出现上述现象的节点进行重组, 得到3个小节点. 在分裂后想要得到形状与尺寸均最优的节点, 就需要对发生重叠和覆盖的情况时, 节点的评价标准进行综合分析.

(3) 多细节层次索引结构的建立. 经过前两步的节

点筛选、节点分裂的操作,已经使得生成的R-树具有了良好的树形,可以使得重要模型的数据自动地分配到R-树的高层节点,从而利用R-树来对数据进行高效的调用。

2.1 节点筛选算法

评判本算法是否最优,取决于将目标模型的数据插入已选节点后,将其上溯至根节点的路径上,对于在路径上的各层节点的影响是否最小.对于R-树结构的特殊性,当原节点范围内出现新的模型目标时,插入新数据的操作不会对节点和父节点的范围产生影响.如果目标不再被包含在集合的所有子节点中,那么就在集合中选择一个子节点,当把目标模型的数据插入其中时,使得该子节点的效果达到最优,再依次向下进行遍历操作,寻找最优节点,直至叶节点所在,即可停止^[19].

通常把覆盖范围和重叠范围作为R-树树形质量的评价标准,但考虑到本索引方法对于多细节层次数据的调用功能的特殊性,故对R-树的各层节点形状也有要求,形状更接近于立方体为最佳,即在三维空间中各个坐标轴上的长度要尽可能保证相等,故定义了如式(5)所示的三维柯西值作为评价标准。

$$\forall X, Y, Z > 0, \exists \left(\frac{X+Y+Z}{3} \right)^3 \geq X \times Y \times Z \quad (5)$$

只有在X、Y、Z相等时,等式才成立.假设 $X \times Y \times Z$ 的值一定,当三者相等时, $\left(\frac{X+Y+Z}{3} \right)^3$ 的值最小,此时节点在三维空间中的形状就更接近立方体.所以,将X、Y、Z三个赋予相同大小的权重值所得到的R-树为最优.综合节点重叠、节点覆盖和节点形状这3个评价因子,将评价标准定义为式(6)所示:

$$Metric = \frac{1}{3}Overlap + \frac{1}{3}Overlay + \frac{1}{3}Shape \quad (6)$$

定义节点的边长分别为X、Y、Z,Overlap代表模型数据插入子节点后,该节点与相邻节点之间重叠部分的体积变化值;Overlay代表着该节点插入模型数据前后的体积变化值;Shape是由式(5)定义的三维柯西值的变化值.具体的算法明细如算法2.

算法入口,R-树,待插目标T:简化后的电缆井和管沟三维模型子集.

算法出口,经过选择的电缆井和管沟模型子集,并命名为叶节点L.

算法2. 节点筛选子算法

- 1) 假设目前R-树有N层,其中,根节点和叶节点分别位于第1和N层.从叶节点所在的第N层开始,查找其中完全包含T的集合,若非空,令 $I=N$,并进入下一步骤;若为空,则向上一层进行查找操作,直到第1层,将根节点加入集合.如果进行上溯操作时,在第J层出现了非空的情况,则令 $I=J$,再进入下一步骤;
- 2) 若第I层的节点集合非空,则将其中所有节点的子节点作为一个集合C;
- 3) 将T插入C中的节点,并计算节点所对应的Metric值,选出该值最小的节点,记为M;
- 4) 新设立的根节点由步骤3)中选中的节点M来担当,在M的子节点中选出合适的节点,将待插目标T插入.选出一个子节点中Metric值最小的,记为 M' ,将其作为新的根节点,重复此步骤直至叶节点;
- 5) 将步骤4)中选出的所有 M' 的Metric值进行比较,选出其中最小的节点作为最终结果;
- 6) 退出程序.

2.2 节点分裂算法

R-树的子元组的数目是有限的.当插入操作使得数目超过最大值时,上溢的节点需要分裂为小节点.这个过程是一种有约束的条件下在三维空间中进行分裂的过程.在执行算法时,一直坚持二分为三的原则.当一个节点出现上溢现象时,在其兄弟节点中寻找二者互相重叠最严重的节点,把包含在这两个节点中的子元素分为三个子节点,这样可以在减少重叠的同时,对节点形状进行改进优化^[20,21].对于没有重叠部分的兄弟节点,继续沿用一分为二的方式.最后,对于分裂后的节点,为其定义评价标准,如式(7):

$$Metric = \frac{1}{3}Overlap + \frac{1}{3}Overlay + \frac{1}{3}Shape \quad (7)$$

为达到最优的算法效果,将式(7)中的3个因子赋予相同的权重.算法流程如算法3所示.

算法入口,出现上溢现象的叶节点L1.

算法出口,调整后的R-树.

算法3. 节点分裂算法

- 1) 在L1的兄弟节点中寻找L2,L2是符合与之重叠体积最大的条件的节点;
- 2) 若不存在L2,就把L1中所包含的全部子元素作为一个集合C1,进入步骤3),准备将其分为两个子集合;若存在L2,就把L1和L2中所包含的全部子元素作为集合C1,进入步骤5),准备将其分为3个子集合.
- 3) 从C1中选择两个子元素,如果这二者的(Overlay+Shape)值最大,就将其作为两个子集合C11和C12的备选元素.接着在剩下的元素中选出一个元素,当其分别插入C11和C12时,Metric值能达到最大,则将其放入到C11和C12中Metric值较小的中.

- 4) 沿用步骤3)中方法,将其余元素插入到合适的子集合中.插入过程进行到一定程度时,子集合中的元素数目恰好满足某个子集合的下限最小值,就将其全部赋给该集合,并进入步骤12).
- 5) 在C1中寻找两个子元素,若二者的(Overlay+Shape)值最大,就将其作为子集合C11、C12和C13中两个子集合的备选元素.接着在剩余的元素中选出一个,当其插入三者中的某一个时,若Metric值最小,就将该元素归入该子集合中.
- 6) 由于步骤5)的约束,集合元素的数目受限,当其恰好与R树节点的下限值相等时即可.
- 7) 将剩余的元素记做集合 $\{P_1, P_2, \dots, P_i, \dots, P_n\}$,其中,对于 P_i 的最小包围体记为 $(X_{\min}, Y_{\min}, Z_{\min}, X_{\max}, Y_{\max}, Z_{\max})$,这些参数分别代表着该元素在3个坐标方向上的坐标最值.接着计算剩余元素的重心.定义元素个数为A, R-树节点最小值定义为B,若 $A > B$,则删除离质心最远的元素,接着返回步骤7);若 $A = B$,则从中选出距质心最近的元素作为第3个集合的备选元素,进入步骤8).
- 8) 对于未被列入上述步骤的元素,从中直接挑选出一个,插入第3个子集合中.将W1定义为评价指标的变化值;再将该元素分别插入另外两个集合中,得到两个Metric值,记W2为其中的较小值.若 $(W2 - W1)$ 的值最大,则将该元素插入第3个子集合.
- 9) 经过步骤8)的筛选,再从剩余的集合中找出合适的元素插入第3子集中,使得该集合的元素数目与R-树的下限值相等即可.
- 10) 寻找一个剩余元素,将其分别插入3个子集合中,对应的有3个Metric值,选择值最小的子集合,并将其归入该集合中.
- 11) 由上一步骤,将剩余的所有元素依次分配到子集合中.
- 12) 对于特殊情况,分裂操作导致父节点出现上溢现象,则将该父节点记做L1,并进入步骤1);若不上溢,则进入步骤13).
- 13) 退出算法流程.

2.3 多细节层次索引结构的建立

对于三维空间中的目标模型而言,需要建立多尺度的LOD模型描述机制,来对其内外部空间特征进行详细描述.此外,对于不同层级下的LOD模型进行简化,可以实现对模型有选择性的筛选保留.之前的算法所建立的R-树结构,有着良好的树形,基于此提出多细节层次三维R-树索引结构,来对三维空间中的模型数据进行快速索引调用.

本方法与传统方法的不同点在于,可以实现对重要目标模型数据的管理和调用.对于子节点所管理的重要模型数据,父节点可从中选择与子节点数目相同的重要模型数据,此操作对于整个R-树的数据量影响有限.

从R-树的根节点开始,向下进行遍历操作,若对于其中的某节点,视点距其的最小距离超过了该节点层次所描述的最远距离,就不需要考虑其所包含的目标;相反,则需要考虑,并对其进行描述.在实际情况中,要结合视距来选择合适的LOD模型.

3 实验结果与分析

3.1 实验环境设置

实验仿真基于北京市某区电缆设计工程,运用本文方法对电缆三维场景的数据进行实验,并测试可视化效果.三维电缆井和管沟模型的数量为175个,原始场景中总的的数据量为2GB.算法的性能取决于两个因素:1)存储模型数据量的大小;2)查询相应时间.数据量越小,查询响应时间越短代表着算法越优.

具体的软硬件配置如表1所示.

操作系统	处理器	内存	开发环境	数据库
Windows7	Inter(R) Core(TM)i5 2.00 GB	2.00 GB	Linux Golang 1.3	LevelDB

3.2 结果分析

共进行3个实验,对不同种类电缆井模型进行简化实验;接着对简化后的模型进行响应时间与加载帧率的对比测试;最后对整体三维场景下的漫游效果进行测试.

三维模型的结构简化与多细节层次索引结构调度的整体思想和流程如图5所示.

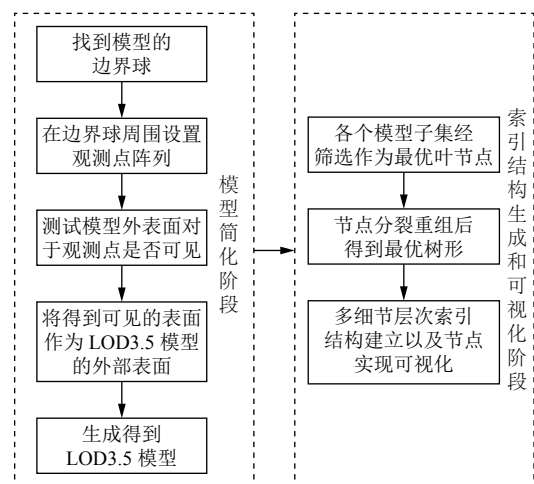


图5 三维场景模型简化和可视化过程

实验1. 电缆三维场景模型简化测试

在实验1中,针对原始三维场景中占据大量内存的LOD4和LOD3层级的电缆井模型进行简化,得到LOD3.5模型.对比3种不同层级下的模型数据量,选取了三种常见的电缆井模型,分别是三通井、四通井和转角井.可以看出,简化后的模型不仅保留了原有模

型的结构特征,更大幅减小了数据量,达到了预期实验目的.结果如图6所示.

实验2.对使用文中算法和使用传统R-树三维场景对比测试

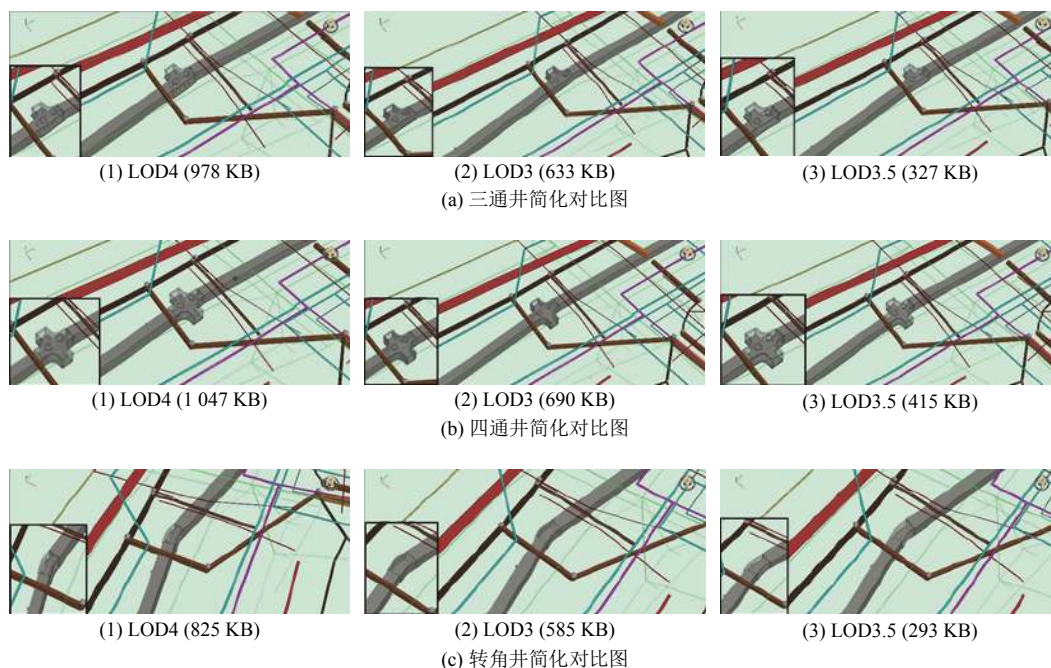


图6 不同LOD层级下的3种电缆井模型

1) 查询响应时间的对比测试

首先用外表面提取算法对电缆三维场景中的模型进行简化,使得整个三维场景的数据量大幅减小,接着对简化后的数据分别应用本文算法和传统R-树索引算法进行索引组织调度.实验在不同的尺寸比例下进行查询响应测试,结果如图7所示,随着比例的增加,所需的查询时间也相应地增加.在相同的比例下,本文算法所需的时间要明显比传统R-树方法的短,特别是随着比例的增加,这种差别越来越明显,本方法的优势也更加突出.

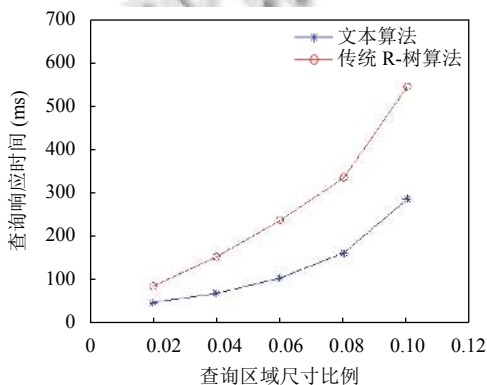


图7 不同查询区域尺寸比例下的查询响应时间

2) 加载帧率的对比测试

通过 Chrome Dev tools 对电缆三维场景加载时间进行记录,对使用本文算法和只使用传统R-树方法的三维电缆场景漫游过程分别记录加载帧率,结果如图8所示.先使用外表面提取算法对模型进行简化,再用多细节层次的三维R-树索引数据调度方法实现的三维场景的加载帧率总体上要比只使用传统R-树方法实现的三维场景的帧率要高,基本保持在40fps左右.

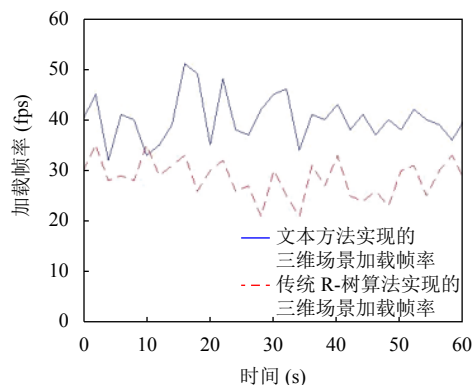


图8 电缆三维场景漫游帧率对比测试

实验3. 电缆工程三维场景图漫游效果测试

图9(a)为电缆三维场景中某一视点的三维场景全

貌图,图 9(b)、图 9(c)、图 9(d) 为该三维场景下的细节图,在整个漫游缩放的过程中,电缆工程中的模型显示流畅,能够实现与用户的良好交互。

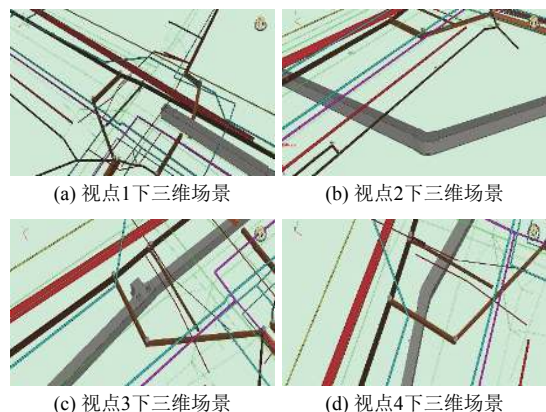


图 9 电缆三维场景效果测试图

4 结论与展望

针对电缆三维场景下加载速度慢的问题,本文提出了一种基于外表面提取算法的三维模型简化与多细节层次的三维 R-树索引数据调度方法。通过对占据大量内存的电缆井和管沟模型进行简化,即保留了电缆三维场景的结构特征,又大幅减小了数据存储量。然后根据多细节层次的 R-树索引结构组织调度简化后的模型数据,最终能够实现提高电缆模型三维场景的加载速度,有效实现了电缆工程中三维模型的流畅展示以及与用户的良好交互的目标。为电网工程数字化移交中的三维场景的交互可视化提供了支撑方法。下一步将电缆工程三维可视化技术应用到运检中,进一步补充和完善相关技术,实现对电缆工程的有效保障。

参考文献

- Deng YC, Cheng JCP. Automatic transformation of different levels of detail in 3D GIS city models in CityGML. *International Journal of 3-D Information Modeling*, 2015, 4(3): 1–21. [doi: 10.4018/IJ3DIM.2015070101]
- Sharkawi KH, Abdul-Rahman A. Improving semantic updating method on 3D city models using hybrid semantic-geometric 3D segmentation technique. *International Society for Photogrammetry and Remote Sensing*, 2013, II-2/W1: 261–268. [doi: 10.5194/isprsannals-II-2-W1-261-2013]
- 王星捷, 卫守林. 基于 WebGL 的三维 GIS 空间算法的研究与实现. *计算机应用与软件*, 2019, 36(4): 63–68, 85. [doi: 10.3969/j.issn.1000-386x.2019.04.008]
- 刘浩翰, 杨佳倩, 贺怀清, 等. 基于突变策略改进的

- Metropolis 光线追踪算法. *计算机应用研究*, 2017, 34(5): 1594–1596. [doi: 10.3969/j.issn.1001-3695.2017.05.071]
- 潘飞, 张社荣. 基于 3D WebGIS 的土木水利工程 BIM 集成和管理研究. *计算机应用与软件*, 2018, 35(4): 69–74, 136. [doi: 10.3969/j.issn.1000-386x.2018.04.013]
- 徐明. 基于节点分裂优化的 R-树索引结构. *计算机应用研究*, 2016, 33(12): 3530–3534. [doi: 10.3969/j.issn.1001-3695.2016.12.003]
- 马贤迪, 常原飞, 乔彦友. 矢量金字塔与 R 树融合模型研究. *测绘通报*, 2016, (2): 59–63.
- 张振鑫, 邓浩, 寇一丹, 等. 基于 ϵ -Voronoi 图的矢量数据自适应简化方法. *地理与地理信息科学*, 2016, 32(1): 29–33. [doi: 10.3969/j.issn.1672-0504.2016.01.006]
- 王卫红, 张鹏灵. 基于移动 GIS 的地块采集系统设计与实现. *计算机应用与软件*, 2017, 34(5): 200–204, 267. [doi: 10.3969/j.issn.1000-386x.2017.05.035]
- 魏子衿, 肖丽. 基于区域分解的并行动态 LOD 构建算法. *计算机工程与应用*, 2018, 54(6): 168–177, 197. [doi: 10.3778/j.issn.1002-8331.1610-0260]
- 汤延辰, 郭星, 张功营, 等. 一种多控制因子 LOD 大规模地形绘制算法. *微电子学与计算机*, 2019, 36(4): 99–104.
- 李鹏飞, 邓秀剑. 三维模型的数据处理与显示技术的设计与实现. *航空电子技术*, 2018, 49(3): 36–40.
- 廖晓苏. 三维 GIS 技术下的输电线路地理信息系统设计探究. *电子测试*, 2016, (18): 119–120. [doi: 10.3969/j.issn.1000-8519.2016.18.062]
- Biljecki F, Ledoux H, Stoter J. An improved LOD specification for 3D building models. *Computers, Environment and Urban Systems*, 2016, 59: 25–37. [doi: 10.1016/j.compenvurbsys.2016.04.005]
- 王汉武, 于涛. 一种改进的自适应多叉树防碰撞算法. *计算机科学*, 2018, 45(11): 66–69. [doi: 10.11896/j.issn.1002-137X.2018.11.008]
- 龚俊, 柯胜男, 朱庆, 等. 一种集成 R 树、哈希表和 B* 树的高效轨迹数据索引方法. *测绘学报*, 2015, 44(5): 570–577.
- Mao B, Harrie L, Ban YF. Detection and typification of linear structures for dynamic visualization of 3D city models. *Computers, Environment and Urban Systems*, 2012, 36(3): 233–244. [doi: 10.1016/j.compenvurbsys.2011.10.001]
- He Q, Chen YT, Dong QH, *et al.* Mining moving object gathering pattern based on resilient distributed datasets and R-tree index. *Neurocomputing*, 2019.
- 曾梦琪, 马蔚吟, 李力. 基于混合相似度的高效图像检索方案. *计算机工程*, 2019, 45(11): 262–268.
- 侯海耀, 钱育蓉, 英昌甜, 等. 基于 Hilbert-R 树分级索引的时空查询算法. *计算机应用*, 2018, 38(10): 2869–2874, 2885. [doi: 10.11772/j.issn.1001-9081.2018040749]
- 王波涛, 梁伟, 赵凯利, 等. 基于 HBase 的支持频繁更新与多用户并发的 R 树. *计算机科学*, 2018, 45(7): 42–52.