

文章编号: 1671-8860(2004)12-1106-05

文献标识码: A

一种基于 TIN 的视相关动态多分辨率地形模型

余 明¹ 左小清² 李清泉²

(1 福建工程学院建筑与规划系, 福州市长安路 89 号, 350007)

(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: 提出了一种基于不规则三角网的“回”结构分辨率的 LOD(level of detail) 算法, 真正实现了离视点越近越清楚、越远越模糊的视觉效果, 避免了在模型简化过程中的交叉、重叠和细长的三角形出现。实验结果表明, 该算法能实时动态地生成不规则格网的连续多分辨率模型, 实现地形场景的平滑绘制, 同时具有较快的显示速度。

关键词: 视相关; 多分辨率; 地形模型; 层次细节; 误差控制

中图法分类号: P231.5

目前, 国内外学者对地形 LOD(level of detail) 模型进行了大量的研究, 总的来说分为两种, 即基于规则格网(GRID)的 LOD 模型和基于不规则格网(TIN)的 LOD 模型。对于规则格网地形, 采用四叉树或二叉树等层次结构比较容易实现地形 LOD 模型的动态生成和显示^[1, 2, 3]; 而不规则三角网格网, 由于它本身数据结构的复杂性, 使得其动态 LOD 模型的建立仍然比较困难^[2, 3]。然而, 在很多情况, 如道路网密集的城市区域, 由于道路网模型嵌入到地形模型当中, 使得地形很难以 GRID 结构来表达, 因此, 研究基于 TIN 结构的 LOD 模型是非常有必要的。本文提出了一种新的基于 TIN 结构的 LOD 地形模型。

1 算法设计

针对地形和道路集成的不规则分块的格网模型, 采用动态划分分辨率等级的方法, 分辨率等级在内存中划分好, 在进行模型简化时, 是动态地判断每个三角形属于哪个分辨率等级, 而不是判断整块格网模型属于哪个分辨率等级。这样, 在同一块格网模型中, 可以有多个分辨率等级。

具体的实现思路是以视点为中心, 根据离视点远近依次设置不同分辨率的区域, 区域设置方法如图 1 所示, 其中图 1(a)是在理论上最合理的

划分。然而, 为了判断点落在哪个等级的分辨率区域, 需要先计算点到视点的距离, 然后再判断其在哪个分辨率区域内, 因此需要进行的运算量较大。为了减少运算量, 提高实时简化的速度, 笔者采用图 1(b)所示的以视点为中心的若干个嵌套的正方形来划分分辨率区域, 形状有点像“回”字, 可称为“回”形算法。采用此结构, 可以直接从点的坐标就可以判断其落入哪个分辨率区域, 不需要计算点到视点的距离。当执行简化操作时, 首先判断点是落在哪个分辨率等级内, 并与前一帧所在的分辨率等级进行比较, 如果分辨率等级没有变化, 则不需要进行任何操作; 否则需要进行下一步判断, 并结合当前等级的误差阈值判断是否需要对边进行简化操作还是细化操作。整个过程只要进行简单的判断就可以确定顶点在哪个分辨率区域, 提高了简化的效率。根据算法思想, 当视点不动, 如在原地旋转时, 相邻两帧的格网模型没有变化, 也就是没有执行任何简化或细化操作, 当视点前进或后退时, 格网模型仅在各分辨率区域的边界会发生简化或细化操作, 而其他地方没有变化, 如当视点移动一步, 如 1m, 在 1 级内的所有三角形中, 只有在区域边界的一小部分从 1 级进入到 2 级, 而其他的三角形仍然在 1 级中。因此, 在漫游过程中, 相邻帧之间只有小部分边执行了简化或细化操作, 保证了模型的简化效率。另外,

模型在简化时, 对顶点树的深度进行了控制, 设置其深度在 6 级以内, 因此, 可以高效率地执行恢复(细化)操作。

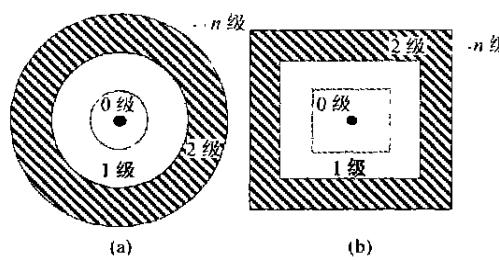


图 1 分辨率区域划分示意图

Fig. 1 Partition of Resolution Regions

2 模型的误差计算

2.1 边的误差计算

在简化过程中, 由于局部区域内的一条边的收缩或顶点分裂, 使原始模型的局部区域在 Z 方向上产生一定的误差大小, 这个误差是一次边收缩所引起的局部范围的实际误差, 以 Δh 表示。由于一条边只能收缩一次, 而边的误差有两个, 即从 c 收缩到 a 与 a 收缩到 c 。本文先计算两个误差, 最后选择较小的误差作为边的误差, 由于边是有方向的, 为了提高算法的执行效率, 这里规定边的误差是从起点收缩到终点的误差, 所以当起点到终点方向的误差大于其反向的误差时, 则要更改边的拓扑关系, 即需要调换边的两端点、边的左右三角形等。从图 2 可以看出, 每次边收缩操作对模型在 Z 方向上所引起的最大变化值, 应该在收缩边的端点处。由于规定了是从起点收缩到终点, 因此, 在 Z 轴上的最大变化发生在 c 点。

c 点的误差计算步骤如下。

† 先假设边已经收缩了, 并且是从 c 点到 a 点收缩;

† 对 c 点的相关三角形组循环(边的左右三角形除外), 将相关三角形中顶点序号为 c 的换为 a 点;

† 判断 c 点落在经变换后的哪个三角形内, 如图 2 落在 $\triangle abd$ 内;

† 求出 c 点在 $\triangle abd$ 平面上的投影点 c' , 则两点的高差 $Hc' - Hc$, 即为边 ca 收缩后在 Z 轴方向上引起的绝对变化值 $\Delta h = Hc' - Hc$ 。

2.2 屏幕空间误差的计算

如图 3 所示, 在 Z 轴方向上误差 Δh 经过透视投影后在投影面上的大小(屏幕误差)不但与视

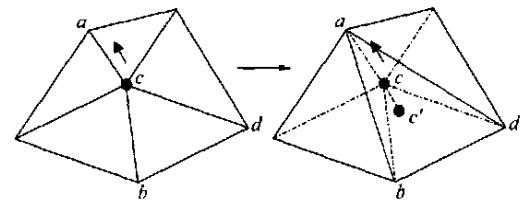


图 2 边的误差计算

Fig. 2 Error Calculation of Edge

线方向有关, 而且与视点的距离以及与视线中心方向的交角均具有一定的关系。因此, 计算公式有三角函数和浮点运算, 如果这些误差都需要实时精确计算, 必然会消耗大量的资源。为了尽量减少实时计算量, 这里对这些误差进行简化计算。如图 3 所示, 笔者通过分析, 同样的 Z 轴误差 Δh , 在到视点平面距离相等的情况下, 当视线中心方向水平时, 即当视点 e 和 ac' 的中点 k 的连线刚好水平时, 在屏幕上产生的投影误差是最大的。不失一般性, 取视线中心方向为水平时的情况来分析, 笔者推导出一个简化的计算公式:

$$\delta = f/d \times \Delta h \quad (1)$$

式中, δ 为屏幕投影误差; f 为焦距; d 为视点到简化点的距离。

式(1)计算出的屏幕误差是最大的 $\max(\delta)$, 可以看出, 屏幕误差 δ 只与焦距、物体离视点的距离和收缩顶点的距离有关, 因此可以事先求出。本文以此最大误差作为衡量不同分辨率 LOD 模型的图像质量的指标。对于特定的情况, 如果投影面与屏幕的比例是 1:1, 那么该误差的大小即是屏幕空间像素的误差。而该误差的大小是由构造 LOD 模型过程中顶点的合并或分裂所引起, 因此, 可以通过控制屏幕投影误差的大小去控制 LOD 模型的简化, 从而得到不同精度的 LOD 模型。从式(1)可以反算出边的空间误差的限差 Δh , 如果以一个像素作为投影误差阈值, 即取 δ 为 1, 则 $\Delta h = d/f$, d 取分辨率等级划分的区域的平均距离, f 在三维环境设置时就已经确定了, 这

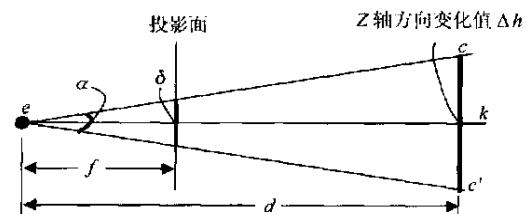


图 3 最大屏幕投影误差计算示意图

Fig. 3 Maximal Error Calculation of Screen Space Projection

样在执行 LOD 算法之前, 可以将各等级分辨率的屏幕投影限差计算出来, 在动态简化过程中, 只需要进行空间误差判断就可以了, 减少了模型在动态过程中的计算量。

3 算法实现

3.1 模型的简化

本文简化算法的主要操作对象仍然是边, 同样采用边收缩和顶点劈开两个互逆的操作来完成对格网的简化和细化。算法没有设计专门的顶点二叉树来记录简化过程, 而是重新设计了数据结构, 将所有的简化信息直接存储在点、边、三角形的拓扑关系中。当边收缩时, 将收缩边标记为删除和收缩状态, 收缩边的左右三角形标记为删除, 收缩边的起始顶点标记为删除, 收缩边的左右三角形的一条边也标记为删除, 并且, 更新收缩边的两个顶点的相关三角形组的拓扑关系, 更新收缩边的起始顶点的相关边组拓扑关系。这样, 从结果上看, 在一次边收缩操作中, 减少了两个三角形。算法的构造主要分为两步, 一是确定边是否需要进行收缩还是劈开操作, 二是更新收缩、劈开过程中顶点、边、三角形的拓扑关系的变化信息。确定边是进行收缩还是劈开操作, 要根据边的误差及所在的分辨率等级来判断。

地形模型的简化具体步骤如下。

输入: 具有 N 个三角形的最细节格网 M_n 。

输出: 具有 $M (M < N)$ 个三角形, 多级依次递减的分辨率格网 M 。

1) 读取原始精度的格网数据。

2) 建立顶点、边、三角形之间的拓扑关系。

3) 划分分辨率等级区域, 并设置各等级的误差限差。

4) 计算边的折叠误差, 注意边的折叠误差有两个, 选择较小边误差进行存储, 将边的分辨率等级设为 0, 并保证边的方向是正向。

5) 对任意边 e_i , 依次对边进行如下判断。

1) 根据分辨率等级区域划分, 获得边的分辨率等级, 并进行边的误差判断, 若小于规定的限差, 则执行步骤 6), 否则转步骤 7);

1) 判断边的起点的相关三角形组中是否有边界三角形, 如果有, 转步骤 7);

1) 判断边是否为活跃状态, 如果是, 执行步骤 6), 否则转步骤 7)。

6) 对边进行收缩操作, 更新顶点、边、三角形

的拓扑关系, 重新计算边的折叠价值。

7) $i = i + 1$; 如果 $i < K$, 转步骤 5) (i 表示边的序号, K 表示边的总数)。

8) 结束。

3.2 模型的细化

模型的细化是简化的相反操作, 即边的劈开操作, 但相对要复杂些。由于顶点的劈开操作的对象是已经执行过收缩操作的边, 而边的收缩对整个顶点树是连续的, 与收缩顶点相关联的顶点、边、三角形等的拓扑关系都发生了变化。虽然没有设计专门的顶点二叉树来记录简化过程, 但边的收缩实际上是构成一个二叉树, 那么当劈开树的中间任意一点时, 都会影响树底端的其他点, 而往往树底端的点又不需要劈开时就存在矛盾。有两种方法解决这个问题, 一是分两步进行, 先把整个网全部细化到原始格网 M , 然后根据分辨率的不同分别进行简化操作, 这种方法实现相对简单, 但需要重复简化, 运算量比较大, 难以保证实时的简化效率; 另一种方法是控制顶点树的深度, 笔者进行了相关实验, 当顶点树的级数超过 6 时, 出现大量的细长三角形。因此, 将顶点树的级数控制在 4~6 比较合适, 既保证了三角形的形状, 又较好地解决了上面的问题。

假设在 t_1 时刻, 顶点树 T_1 中的顶点全部在 2 级分辨率区域, 当视点移动, 在 t_2 时刻, 顶点树 T_2 中的顶点所属分辨率区域变化有以下几种情况: ① 整个树仍然在 2 级区域, 或者全部在 1 级区域; ② 树的上部分在 2 级区域, 下部分在 1 级区域, 如图 4(b); ③ 树的两头仍然在 2 级区域, 而树中间个别点在 1 级区域, 如图 4(c)。对于第①种情况不需要处理, 或直接对整个树进行简化; 对于第②、③两种情况, 由于中间点的收缩会影响树中其他的点, 因此不能直接进行简化。由于算法将树的深度进行了控制, 本文对于第②、③两种情况处理的方式是: 先对树实行归零操作, 即将整个树重新细化到原始精度, 然后再根据不同的分辨率对各点进行简化。由于树的深度控制在 5 级以内, 而要实行归零操作的树是很少的比例, 如图 4(a), 当视点移动时, 只有一小部分点的分辨率等级发生改变, 即处在区域边缘的点的分辨率会发生变化, 而其他点保持不变, 如果分辨率等级划分为 5 级, 则需要归零的顶点数一般在 8% 左右, 所以, 增加的计算量不是很大。同时由于进行了顶点树深度控制, 则有一部分边的精度符合简化条件而没有简化, 这部分边大约占 10%。

模型细化的步骤 1)~步骤 4) 与模型简化的

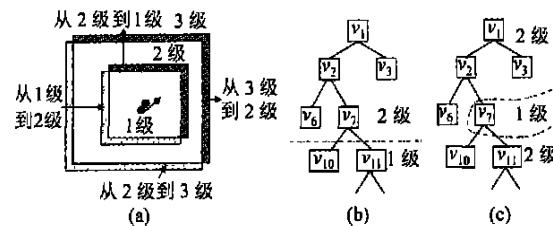


图 4 视点移动过程中顶点树的变化

Fig. 4 Changes of Node-tree When Viewpoint Moving

步骤 1) ~ 步骤 4) 相同, 其他步骤如下。

5) 对任意边 e_i , 依次对边进行以下判断:

1 根据分辨率等级区域划分, 获得边的分辨率等级 L_a , 并与以前的分辨率等级 L_b 比较, 如果 $L_a = L_b$, 转步骤 7);

1 判断边的起点的相关三角形组中是否有边界三角形, 如果有, 转步骤 7);

1 判断边是否为删除状态, 如果不是, 转步骤 7);

1 边的误差判断, 若小于或等于规定的限差, 转步骤 7)。

6) 首先对整个顶点树进行归零, 然后整个顶点树进行简化操作, 同时更新顶点、边、三角形的拓扑关系, 重新计算边的折叠价值。

7) $i = i + 1$; 如果 $i < K$, 转步骤 5) (i 表示边的序号, K 表示边的总数)。

8) 结束。

4 实验

实验数据是在规则格网基础上生成的, 然后进行 TIN 的建立, 所以点的分布很均匀, 地势比较平坦。为了进行比较, 笔者分别以四块数据量不同的地形数据进行实验, 最大范围的地形大约是 $2.6\text{km} \times 2.6\text{km}$, 最小范围大约为 $1\text{km} \times 1\text{km}$, 点的间隔是 10m。

实验首先利用这些分布均匀的离散点进行 Delaunay 三角网的构建, 如图 5(a) 是采用规则点构建的 TIN 的二维平面图显示。为了减少简化过程的计算量, 笔者在简化前事先进行点、边、三角形的拓扑关系建立、顶点的法向量计算、边 Z 轴方向的误差计算, 并按设计好的格式存储到数据库中。数据经过预处理后, 接下来可以对模型进行分辨率的设置, 包括分辨率等级数的设置和控制误差的设置。根据算法的思想, 分辨率可以动态地在任何时刻进行设置, 分辨率的等级数也可以任意设置。分辨率设置好后, 系统自动将各等级的分辨率的控制误差计算出来, 表 1 列出了三角形数量为 27 848 时的分辨率等级划分和控制误差情况。表 2 列出了不同三角形数量的 LOD 模型简化结果比较。图 5(b) 是简化后的格网模型的平面显示, 黑点表示视点位置。从图 5

表 1 分辨率等级划分情况

Tab. 1 Division Result of Resolution Levels

分辨率等级	离视点距离	不同的分辨率等级按不同的屏幕误差控制		不同的分辨率等级按相同的屏幕误差控制	
		Z 轴误差/m	最大屏幕误差/pixel	Z 轴误差/m	最大屏幕误差/pixel
0	100	0	0	0	0
1	250	0.46	1.0	0.46	1.0
2	500	1.17	1.2	0.98	1.0
3	1 000	2.54	1.3	1.95	1.0
4	2 000	5.86	1.5	3.91	1.0

表 2 不同三角形数量的格网模型简化结果比较

Tab. 2 Comparison of Simplification Result Among Meshes With Variant Triangle Number

原始三角形数量	简化后三角形数量	分辨率等级数	平均帧数(前/后)/s	最大屏幕误差
19 705	9 523	4	100/120	1.0
27 848	12 704	5	66/87	1.3
53 161	26 307	5	32/45	1.5
132 849	60 231	5	15/19	1.5

可以看出, 格网模型离视点越近, 简化程度越低, 而分辨率越高, 这种简化效果符合人眼的视觉规律。

实验结果表明, 显示的速度和原始地形三角形数量以及简化的程度(控制误差大小)有关。算法能很好地动态建立不规则分块的格网模型的多

分辨率 LOD 模型。从实验可以看出, 当三角形数量在 20 000 ~ 50 000 之间, 效果比较好。分辨率等级数量对简化效率没有多大影响, 但会影响简化的效果, 尤其是等级之间的距离不宜太大, 相邻两级的距离在 500m 以内比较合适。

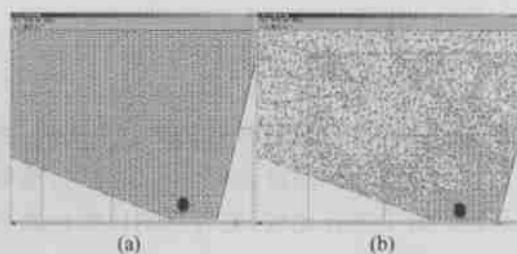


图 5 原始格网和简化后模型比较(二维显示)

Fig. 5 Comparison of Original Mesh and Simplification Mesh

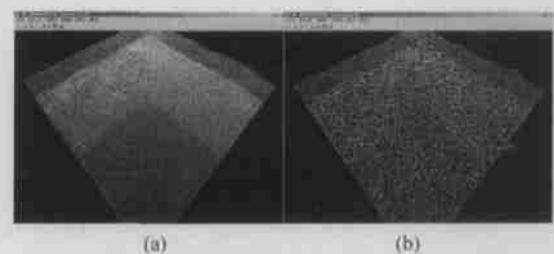


图 6 原始格网和简化后模型比

Fig. 6 Comparison of Original Mesh and Simplification Mesh

参 考 文 献

- 1 王宏武, 董士海. 一个与视点相关的动态多分辨率地形模型. *计算机辅助设计与图形学学报*, 2000, 12(8): 575~579
- 2 许妙忠, 李德仁. 基于点删除的地形 TIN 连续 LOD 模型的建立和实时动态显示. *武汉大学学报·信息科学版*, 2003, 28(3): 321~325
- 3 陈刚, 杨明果, 王科伟. 地形 TIN 模型的实时连续 LOD 算法设计与实现. *测绘学院学报*, 2003, 20(4): 286~289
- 4 何晖光, 田捷, 张晓鹏, 等. 网格模型化简综述. *软件学报*, 2002, 13(12): 2215~2224
- 5 Hoppe H. Progressive Meshes. *Computer Graphics (SIGGRAPH' 96)*, New Orleans, Louisiana, USA, 1996

- 6 Hoppe H. Smooth View-dependent Level of detail Control and Its Application to Terrain Rendering. *IEEE Visualization' 98*, North Carolina, USA, 1998
- 7 Lindstrom P, Koller D, Ribarsky W, et al. Real-time, Continuous Level of Detail Rendering of Height Fields. *Computer Graphics (SIGGRAPH' 96)*, New Orleans, Louisiana, USA, 1996
- 8 Danovaro E, Floriani L D, Magillo P, et al. Morphology Driven Simplification and Multi-resolution Modeling of Terrains. *Proceedings ACM-GIS 2003*, New Orleans, Louisiana USA, 2003. 60~70

第一作者简介: 余明, 副教授. 现从事 3S 方面的应用和研究.
E-mail: yuming@fjut.edu.cn

A View-dependent Dynamic Multi-Resolution Terrain Model of TIN

YU Ming¹ ZUO Xiaoqing² LI Qingquan²

(1 Department of Architecture and Planning, Fujian University of Technology, 89 Chang'an Road
Fuzhou 350007, China)

(2 State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing,
Wuhan University, 129 Luoyu Road, Wuhan 430079, China)

Abstract: A “hui” structure LOD (level of detail) model TIN-based is presented in this paper. The algorithm achieves the experiment effect: the nearer the terrain is away from the viewpoint, the clearer it is. It avoids the triangles crossing or overlapping, and creating some thin triangles. The experiment shows that with this algorithm, the dynamic continuous terrain multi-resolution model will come true.

Key words: view-dependent; multi-resolution; terrain model; LOD; error control

About the first author: YU Ming, associate professor, his major research orientation is 3S.

E-mail: yuming@fjut.edu.cn

(责任编辑: 平子)