ISSN 1000-9825 Journal of Software 软件学报 2000, 11(9): 1181～1188

基于分形维数的地表模型多分辨率动态绘制*

王璐锦 唐泽圣

(清华大学计算机科学与技术系 北京 100084)

E-mail: wlj@cad.cs.tsinghua.edu.cn

摘要 以基于分形维数的树状结构组织三维地表网格模型, 实现了与视点相关的连续多分辨率地表模型简化及实时绘制. 算法分为预处理和实时绘制两个阶段. 在预处理阶段, 通过分形维数评价地表的复杂度, 建立自适应的树状结构, 计算出所有顶点的误差值. 在绘制阶段, 则根据视距、视角等因素动态地确定需保留的顶点集, 并采用受限四叉树方法实时三角化得到所需分辨率下的三角网格近似模型. 该算法具有两个优点: 一个是地表模型的分层区域划分考虑了地形本身的复杂度; 另一个是建立了视点相关各参数与所采用的分辨率表示的直接关系. 实验表明, 此算法简单、有效, 支持对地表模型的交互式实时动态绘制.

关键词 多分辨率简化, 分形维数, 受限四叉树, 地表模型, 动态绘制.

中图法分类号 TP391

实时动态显示是虚拟现实、交互式可视化等技术的关键问题. 当今许多高档图形工作站已能对上万个三角形进行实时绘制, 但随着虚拟现实中的实时漫游、飞行模拟以及科学计算可视化中数据量越来越大, 在绘制中所使用的模型也越来越复杂, 使得三维模型的复杂度大大超出了一般图形硬件的实时绘制能力. 目前, 解决这一问题的方法主要有:

(1) 预先进行可见性判定;

(2) 基于图像的绘制(image based rendering);

(3) 并行计算和分布式技术;

(4) 多分辨率模型(multiresolution modeling 或 level of detail modeling)简化技术.

前两种方法主要是利用在交互和漫游的过程中视点变化的连贯性以及场景和图象之间的连贯性, 但目前只适用于静态场景, 且需要较长的预处理时间, 第2种方法还要耗费很大的内存空间来存放大量的预制图像. 采用并行计算等手段提高计算、绘制速度, 需要有高档并行机或机群系统支持, 成本较高并且不易推广. 多分辨率模型简化技术是对物体的几何性质、表面性质、纹理等进行多分辨率的分析和造型, 根据物体在屏幕上所覆盖面积的大小选择相应分辨率下该物体的简化模型, 尽量减少三角形的数量, 使得在给定视点下获得的图像效果与用最精确的模型画出的效果完全相同或差距在给定范围内, 从而大大提高绘制效率. 这种方法所占用的内存空间较小, 适用于动态场景, 近几年来已受到图形学界的高度重视.

数字高程模型(digital elevation model, 简称 DEM)是地理信息系统(geographic information system, 简称 GIS)中反映地形信息的数据, 通常以计算机中的二维数组形式给出, 在规则矩形网格点上给出该点的地表高程值, 也称为高度场(height field). 数据在网格中的存储位置能够反映实际数据的空间位置, 便于对数据进行空间操作. 大规模数字地表模型(digital terrain model, 简称 DTM)数据量大, 对全分辨率的可见网格直接绘制难以达到实时显示的要求. 若采用多分辨率(Multiresolution 或 LOD)简化算法则可以在保证图像质量的前提下加快显示速度, 使得生成的地形画面在一定的视觉误差范围内; 同时, 只考虑处于视野范围内的一小部分地表模型, 且距视点距离不同的区域采用不同分辨率的模型.

* 本文研究得到国家自然科学基金(No. 69833010)资助. 作者王璐锦, 女, 1973年生, 博士生, 主要研究领域为计算机图形学, 科学计算可视化. 唐泽圣, 1932年生, 教授, 博士生导师, 主要研究领域为计算机图形学, 科学计算可视化.

本文通讯联系人: 王璐锦, 北京 100084, 清华大学计算机科学与技术系软件所

本文 2000-02-28 收到原稿, 2000-04-18 收到修改稿

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

本文针对三维地表模型的实时动态绘制问题,提出一种基于分形维数的树状结构组织地表模型,实现了与视点相关的连续多分辨率地表模型简化算法,并在此基础上实现了一个地表漫游演示系统:

(1) 计算地形表面的分形维数(fractal dimension),用分形维数的大小度量地形的平坦度.

(2) 根据分形维数的大小对地表模型进行区域层次划分,形成自适应的树状结构.

(3) 用受限四叉树(restricted quadtree)方法对地表模型原始数据点进行误差估计.

(4) 绘制时对各子结点区域进行动态多分辨率简化,使得地形表面上的不同区域在同一时刻呈现不同的分辨率,不同分辨率的各部分之间光滑过渡、无裂缝.

(5) 实现与视点相关(view-dependent)的简化,简化结果受视点位置、视线方向、视窗大小等因素的影响.

(6) 加入影像图数据,作为地表模型纹理,实现地表漫游演示系统.

1 相关的工作

目前对于规则网格或不规则三角网格(triangulated irregular network,简称 TIN)模型都进行了大量的表面简化算法研究,包括保持拓扑结构(preserve topology)和不保持拓扑结构方法、误差全局受限和误差不受限方法、视点相关和视点无关(view-independent)方法、静态(static)多分辨率和动态(dynamic)多分辨率简化方法.

各种简化算法采用不同的策略,最简单的方法是采样滤波(sampling and filtering)方法,如金字塔(pyramid)方法[1],一般处理描述地形的规则网格或参数网格.在网格上等间距或周期性地采样得到低分辨率的简化网格,以其简单和压缩性好成为在仿真环境及可视化、动画中实现多分辨率地表模型时最常用的方法.但算法会删除掉重要顶点,无自适应性.层次细分(hierarchical subdivision)方法采用分而治之(divide and conquer)的策略进行层次三角化,将表面按其复杂程度层次化地组织成一棵树,是金字塔方法的自适应形式,如采用自适应的四叉树细分法、k-d 树细分法、三角形四分法(quaternary triangulation)、三角形三分法(ternary triangulation)等[2~5].这种方法快速、简单,但是产生的近似网格一般质量较差.特征(feature)方法用于高度场简化,按某种"重要度"对输入数据点进行排序,选择最重要的点组成顶点集合,构造该顶点集的三角剖分(如 Delaunay 三角剖分)形成简化网格."特征"是指重要点以及连接这些点的边,反映如峰、洞、脊、谷等地形特征[6],在制图学(cartography)中常用,然而质量较差.

另外,求精(refinement)方法与简化(decimation)方法是两个相反的过程.前者是从最粗糙的近似网格开始,每次插入一个或几个在原模型上被认为能反映重要特征的顶点,并重新进行局部三角化,重复这一过程直到达到限定的误差或使用了期望数量的顶点为止,即"贪婪插入"(greedy unsertion)算法及其变体[7].美国环境系统研究院(ESRI)开发的地理信息系统软件 Arc/Info 近似地形网格就是采用了一种特征/求精混合的方法,从一个等边三角形的规则网格开始,用并行贪婪插入算法逐步求精得到简化后的地形网格.后者是从最复杂的模型开始,自下而上地不断删掉不重要的顶点、三角形或其他几何特征,直到得到所期望的近似网格为止.目前的绝大部分简化方法都可以归于顶点删除(vertex decimation)、边删除(edge decimation)[8~12]、三角形删除(triangle decimation)、面片删除(patch decimation)[7]这 4 类方法,主要用于不规则三角网格模型的多分辨率简化,而在高度场中则用得较少.

针对数字高程模型(DEM)表示的均匀地表网格模型简化问题,Lindstrom 在 Siggraph'96 上第一次将自适应的受限四叉树三角剖分(restricted quadtree triangulation,简称 RQT)方法用于均匀地表网格的实时连续多分辨率显示,采用基于屏幕空间(screen-space)误差度量算法、基于特定误差估计的顶点筛选方法和有效的场景剔除方法,地表模型的简化效果明显提高[13].在此基础上,Pajarola 在 Visualization'98 上对 Lindstrom 的 RQT 方法进行改进,提出两种顶点选取算法、更高效的三角形条带结构组织方法和更精确的近似误差计算方法,适用于大规模地表模型可视化[14].文献[15]采用三角形合并的方法,提出一种具有聚焦因子的基于图像空间的误差评判标准.这些方法都未考虑地形的复杂程度,只是将地形数据直接分为大小相等的块进行处理.文献[12]提出了一种基于分形维数的层次多分辨率简化算法,用于三维复杂模型的简化.算法利用分形维数来度量模型表面的平坦度,按表面各区域平坦度的不同将模型划分成层次结构,探索了与视点相关的连续多分辨率简化算法.

本算法利用分形维数来度量地形表面的复杂度,将地形表面的不同区域划分成层次不同的树结构,使各叶子结点区域的复杂度较低,在各区域内部采用受限四叉树方法进行动态简化.采用基于分形维数的地表模型树状结构对表面较平坦的地形不再分层,减少叶子结点数,从而减少了实时绘制时需要遍历的结点数目.另外,当视点距

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

地形较远时, 简化程度较高.

2 基本原理及算法

算法的基本思想是, 在预处理阶段,先根据分形维数大小将地表模型划分成自适应的树结构,然后利用树中的叶子结点计算出所有顶点的误差值;在绘制阶段, 根据顶点的误差和视距、视线、视角等因素实时确定需要保留的顶点集, 再对该顶点集进行受限四叉树三角剖分, 动态生成连续的多分辨率简化网格.

2.1 基于分形维数的地表模型区域层次划分

1975年, Mandelbrot 创建了分形几何学, 用来处理自然界中大量的不能被经典的欧氏几何所描述的极不规则的自然几何形状,分形维数的概念也被提出.通常提到的维数都是非负整数, 称为拓扑维数.自然界的任何形状的分形维数一般都严格大于其拓扑维数. 非整数的分形维数的大小在一定程度上反映了所描述对象的不规则程度以及填充空间的能力[16].

2.1.1 分形维数的计算

计盒维数又称盒维数(box counting)[17],由于数学计算及经验估计而相对容易成为应用最广泛的分形维数之一. 其定义为: 设 X 表示一个完全度量空间, F 为 X 上的非空紧子集, $B(x,\epsilon)$ 表示以 $\epsilon(\epsilon>0)$ 为半径、$x(x\in X)$ 为中心的闭球, 定义 $N(\epsilon)$ 为覆盖 F 所需要的半径为 ϵ 的最少闭球个数,如果 $D_B(F)=\lim\limits_{\epsilon\to 0}\frac{1}{c}\cdot\frac{\ln(N(\epsilon))}{\ln(1/\epsilon)}$ 存在,则称 $D_B(F)$ 为集合 F 的分形维数.

对于三维空间中的三角面片集 F,取边长为 ϵ的相邻立方体来覆盖.因为任一边长为 ϵ 的立方体的直径为 ϵ, 并且任意的直径最大为 ϵ 的闭球一定包含在一个边长为 $\sqrt{3}\epsilon$ 的立方体内,所以采用立方体覆盖与上述定义等价. 取该面片集 F 的包围盒,并沿 x, y, z 轴按不同的步长 ϵ 划分成小的单元盒(通常划分 3～5次),统计不同 ϵ 值的盒子与 F 相交的个数 $N(\epsilon)$. 真实世界集的几何特性都是在一定的尺度范围内成立的,若超出范围, 用于描述某现象的分形几何模型就不再有效, 因此, ϵ不可能为零. 可以对统计出的 $(\ln(1/\epsilon),\ln(N(\epsilon)))_i$ 数据对 $(i=1,\ldots,n)$ 作线性回归(linear regression)计算, 用样本回归直线的斜率 b(即样本回归系数)近似估计计盒维数 $D_B(F)$ 的大小.

$$D_B(F)=\frac{1}{c}\cdot b=-\frac{1}{c}\cdot\frac{n\cdot\sum_{i=1}^{n}(\ln\epsilon\cdot\ln(N(\epsilon)))-\sum_{i=1}^{n}\ln\epsilon\cdot\sum_{i=1}^{n}\ln(N(\epsilon))}{n\cdot\sum_{i=1}^{n}(\ln\epsilon)^2-\left(\sum_{i=1}^{n}\ln\epsilon\right)^2},$$

式中 C 为一个正常数.这样就可以较快地估计出该面片集的分形维数,从宏观上度量出一张三维表面的平坦度.

2.1.2 地表模型层次划分

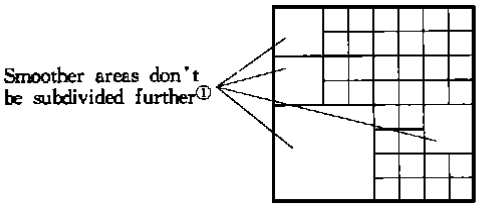
基于分形维数, 我们将三维地表模型组织成自适应的树状结构. 基本过程是从整个地形网格开始, 若地表模型表面起伏变化剧烈, 就将其分成多个子区域.在每个子区域内部再同样递归划分为更小的子区域,直到地形起伏较小或区域相对较小时停止划分, 形成自适应的树状结构.这样, 对平坦的部分少分层,而对起伏大的部分多分层,在最终的叶子结点中保留了较为平坦的大块区域.

在算法实现中,在数据预处理阶段进行地表模型的层次划分, 采用计盒维数估计方法度量出地形表面的平坦度, 分形维数接近于 2.0 的表面区域可以被认为是较平坦,而接近于 2.3 的表面区域则需要进一步进行分层.

具体对地形表面划分子区域的方法可以有多种选择, 导致各层子区域的数量也不同,如采用基于面片生长方法[18,19]、二分法、四分法和不等分的方法. 基于面片生长的方法的优点是, 算法能搜索出周围较平坦的网格并入当前面片集, 导致树中每个结点都包含较多的顶点和面片信息,树的广度和深度较小, 便于实时地进行与观察方向相关的简化. 但是,这种从底向上的方法对于地表网格的大数据量而言,速度较慢.考虑到树结构的特性、将要对树结点进行的简化操作、算法效率等问题, 我们采用了四等分法, 将地形表面划分成自适应的四叉树结构(adaptive quadtree). 在地形网格的动态简化和实时绘制过程中,利用树中所有叶子结点, 实现与视点相关的多分辨率简化, 在不影响视觉效果的前提下大幅度地减少了需要绘制的三角面片数量.区域划分的结果如图 1所示,叶子结点数为 40, 若不采用分形维数,将地表模型自顶向下分为完全四叉树,叶子结点数为 64, 这样绘制时遍历的结点数量明

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

显减少.



①较平坦区域不再进一步分层.

Fig. 1 Leaf nodes of adaptive tree structure based on fractal dimension of terrain model
图1 基于分形维数的地表模型自适应树结构的叶子结点区域

2.2 基于受限四叉树的顶点误差计算

基于四叉树(quadtree)结构的层次细分简化方法用于高度场规则网格的细分很自然, 但是, 这种静态方法会导致出现不连续的分辨率, 且不同分辨率表示之间会出现边界不一致的情况, 需用一些小三角形来填充或者在分辨率低的部分加入新的边界点并重新三角化来解决边界不一致问题. 针对这些缺陷, 近年来提出的受限四叉树(restricted quadtree, 简称 RQ)结构用于高度场的简化.

RQ 的规则是指相邻的四叉树块之间最多相差一层(如图 2 所示). 该规则防止一个表面上采样频率的突然变化, 将与采样频率变化相关的失真降到最小, 确保不同区域之间不会出现裂缝, 以实现高度场的连续多分辨率简化.

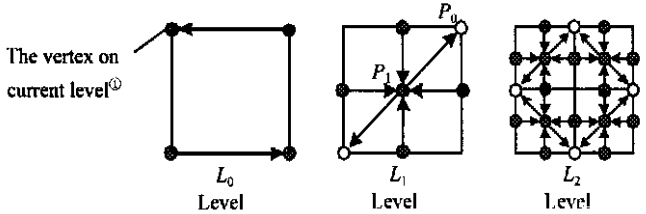


(a) Restricted quadtree (b) Triangulated mesh of restricted quadtree
(a) 受限四叉树 (b) 受限四叉树三角化网格

Fig. 2
图2

2.2.1 顶点的层次及依赖关系

将 RQ 表示成网格上顶点的层次及依赖关系, 如图 3 所示, 将顶点分成 L_i 层, 设原始网格大小为 $m \times m$, 其中 $m = 2^n + 1$, 则 $0 \le i \le \ln(m-1) = n$. 各层顶点的集合包括了网格中的所有顶点. 顶点之间的依赖关系如图 3 所示, $P_1 \rightarrow P_0$ 表示 P_1 依赖于 P_0, 记为 $P_1.dep = P_0$. 依赖关系表示当下层顶点 P_1 在简化后的网格上出现时, 上层顶点 P_0 必然也在该简化网格上出现, 即 P_0 的误差值 $E_{P_0} \ge P_1$ 的误差值 E_{P_1}. 同时, 依赖关系可以传递, 将最初定义的和经传递产生的依赖关系分别称为直接依赖和间接依赖. 通过顶点的层次关系及依赖准则, 可以在算法预处理阶段计算出网格上所有顶点在简化操作后将会导致的误差值, 另外, 在绘制时可以将任意给定顶点集确定地转换成受限四叉树三角剖分.



→ presents the dependent relationship between vertexes②
①当前层上的顶点, ②→表示顶点之间的依赖关系.

Fig. 3 The level and dependent relationship of vertexes on RQ
图 3 RQ 顶点层次及依赖关系

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

2.2.2 顶点误差的计算

顶点误差的计算利用的是顶点的层次和依赖关系. 由于依赖关系的可传递性, 上层顶点的误差会受到下层顶点误差的影响, 误差计算过程应该是自底向上的. 但在算法实现过程中, 我们先自顶向下同时进行顶点误差的计算和顶点集的受限四叉树层次细分, 然后再自底向上地根据顶点依赖关系对误差进行调整, 提高算法效率.

网格某顶点 A 的误差由两部分因素共同决定: 一个是简化过程中忽略该点, 导致三角形不再进一步细分所得到的网格与保留该点得到的网格之间的误差值 E_0; 另一个是直接依赖于该顶点的其他下层顶点 B_i 的误差值 $E_{B_i}(i = 1, \ldots, k)$. 因此顶点 A 的误差 E_A 为

$$E_A = \max\{E_{A_{\text{old}}}, E_0, E_{B_1}, \cdots, E_{B_k}\},$$

其中 k 为直接依赖于 A 点的顶点数目; $E_{A_{\text{old}}}$ 为顶点已计算出的误差值, 初始值可以定为无穷小. 通过对顶点的误差值进行调整, 误差值就通过依赖关系进行了传递. 当自顶向下搜索顶点时, 若某顶点误差低于阈值 E 而不必保留, 则说明那些直接或间接依赖于它的下层顶点也可以忽略, 这时, 对该树支的搜索就可停止.

算法中误差值 E_0 的计算方法如下: (1) 对于受限四叉树中每个结点的中心顶点 C, 它依赖的顶点为 V_0, V_2(或 V_1, V_3), 则 C 点的误差 E_0 = 点 C 到 Line(V_0, V_2) 的距离; (2) 对于每个结点四周边上的中点 M_i, 结点上与它相邻的角点为 V_i, V_{i+1}, 则 M_i 点的误差 E_0 = 点 M_i 到 Line(V_i, V_{i+1}) 的距离(如图 4 所示). 在算法预处理阶段将计算出所有顶点的误差值并加以保存, 为实时绘制时动态网格的生成提供判据.

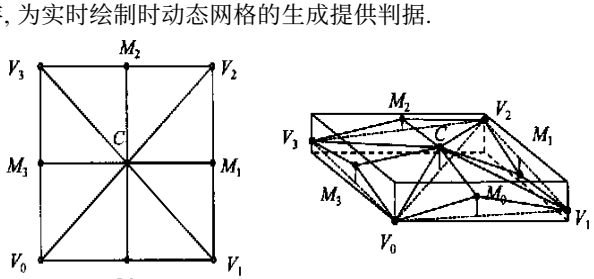


Fig. 4 Calculating vertexes´ error E_0

图 4 计算顶点误差 E_0

2.3 动态网格实时生成

在地表漫游的过程中, 算法综合考虑多方面的因素, 包括地形网格顶点的误差值、视点位置、观察方向、视角大小等因素, 实时判断出在视野范围内的地形区域以及对该区域简化时所需保留的顶点集, 然后动态地生成该顶点集的三角化网格, 在保证图像质量的前提下大大减少了实际绘制三角形的数量, 加快了绘制速度, 实现了与视点相关的连续的多分辨率简化.

2.3.1 顶点集的确定

· 判断可见区域

当视点位置、观察方向、视角大小等因素确定时, 可以计算出视野范围内的地表区域. 视点 p 和视角如图 5 所示, 处于视野范围内的区域结点有两种情况: 一种是结点完全处于视野内(标号为 1 的结点); 另一种是结点部分处于视野内(标号为 2, 3 的结点). 在算法实现中采用两种判据证明结点可见: 一种是结点包围盒的四个角点中至少有一个在视野范围内(对标号为 1, 2 的结点); 另一种是结点包围盒的四条边中至少有一条与视角的边界相交(对标号为 3 的结点). 凡是上述两个条件都不满足的结点则是不可见的, 在本次简化和绘制操作中完全不予考虑.

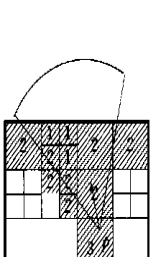


Fig. 5 The visible areas of the model

图 5 模型中的可见区域

· 确定需保留的顶点集

物体空间上一条线段在屏幕空间的投影长度随着线段距视点的远近不同而变化, 当距离足够远时, 线段在屏幕上的投影长度会小于一个像素值, 这时, 线段对屏幕上的最终图像的质量没有贡献, 可以被忽略. 利用文献[12]建立的模型上顶点重要度与视距的关系, 我们可以在视窗、视角大小确定的前提下, 根据顶点的误差值计算出该误差可以被忽略的最近距离 d_0(即当该顶点距视点比 d_0

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

远时, 都可以被忽略), 这一工作可以在算法预处理阶段完成, 以减少每次实时绘制前的计算量.

在视点移动的过程中, 实时计算出视点与可见区域中的顶点 V 的距离 d, 若 d_0 小于或等于 d, 该顶点就可以被认为是不重要的细节, 否则要保留该顶点, 标记为被使用, 即 V.use= TRUE. 然后根据顶点之间的依赖关系将所有直接或间接被依赖的顶点的 use 标记为被使用, 已被标记的顶点在本次绘制中不需再进行上述判别计算, 这样能够减少需要计算到视点距离的顶点数. 对所有的可见区域中的顶点进行上述判别后, 所有 use 标记为 TRUE 的顶点组成了需要保留的顶点集, 下一步可以进行网格动态生成的工作.

2.3.2 实时三角化

对于每个可见区域实时三角化的递归算法如下(如图 6 所示).

```
Triangulation_RQ(可见区域结点 F){
    if (结点的中心点 C.use= FALSE)      // C 可以被忽略
        将 F 直接分成两个三角形绘制:
    else{      // C 不能被忽略
        for (F 每条边界的中点 Mi){
            if (Mi.use= FALSE) 绘制 Δ(Vi, C, V(i+1)%4);
            else{
                if (Ci.use= FLASE)          绘制 Δ(Vi, C, Mi);
                if (C(i+1)%4.use= FALSE)    绘制 Δ(V(i+1)%4, C, Mi)
                else Trangulation_RQ(子结点区域 F(i+1)%4);
            }
        }      // for
    }          // else
}              // Triangulation_RQ
```

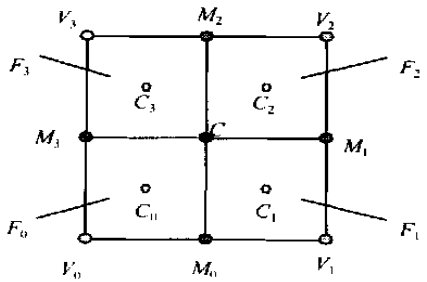


Fig. 6 Visible areas F and its inner vertex

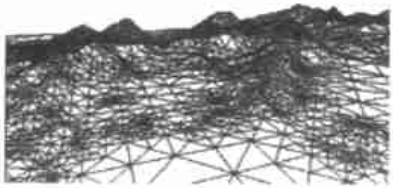
图 6 可见区域 F 及其内部顶点

在采用基于受限四叉树的方法三角化顶点集的过程中, 决定细分某块的一个条件是其相邻的块已因某种原因进行了细分, 这样就将细分的信息传播给相邻区域, 防止出现相邻区域简化网格边界不一致的情况, 保证不同分辨率连续、光滑无缝的过渡.

3 实验结果

基于以上算法, 我们采用 VC6.0 语言和 OpenGL 在 PIII 600 微机上实现了一个地表漫游演示系统, 图 7 和图 8 为地表模型的简化结果, 能够实时地产生连续分辨率的近似地表模型, 同时加入影像图, 作为地表模型的纹理, 实现带纹理的地表漫游. 图 7 中原始数据为 1025×1025, 三角形数量为 2 097 152 个, 某时刻下简化网格如图 7(a)所示, 实际绘制的三角形数量为 13 837 个(约占原三角形总数的 0.66%), 视点移动后如图 7(d)绘制的三角形数量为 15 562 个(约占 0.74%), 不带纹理的图像可达到实时要求, 但加入纹理, 由于目前尚未有纹理硬件支持, 还不能达到实时, 带 2048×2048 大小纹理的原始网格绘制一帧平均要 12 秒, 简化后平均为 0.8 秒, 速度有很大提高.

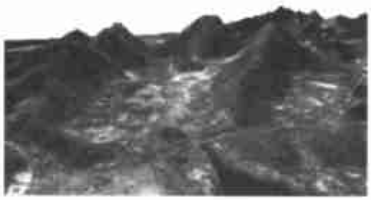
© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c



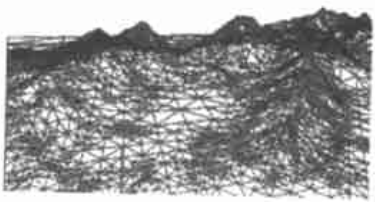
(a) Mesh image
(a) 网格图



(b) Reality image with light
(b) 带光照的真实感图像

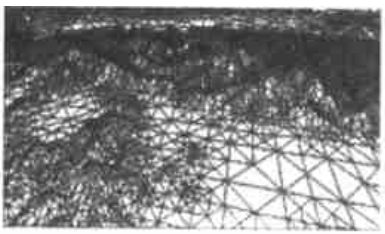


(c) Texture image
(c) 带纹理的图像



(d) Changed mesh after view point moved
(d) 相对于(a)视点移动后生成的另一网格图

Fig. 7 The simplified result of 1025×1025 terrain mesh model
图 7 1025×1025 地表网格模型的简化结果



(a) Mesh image
(a) 网格图



(b) Reality image with light
(b) 带光照的真实感图像

Fig. 8 The simplified result of another terrain model
图 8 另一地表模型的简化结果

4 结 论

本文给出了一个对地表模型进行实时多分辨率表示以及动态绘制的算法,适用于基于 DEM 数据的规则网格地表模型. 算法最重要的特点是利用分形维数度量地表的复杂度,将复杂地形划分为更小的区域进行处理,并能根据当前视点到地表网格各顶点的距离、观察方向、视角大小等因素实时产生连续的多分辨率近似网格,使得被绘制网格的三角形数量随视点的改变而变化. 采用基于分形维数的树状结构能够减少实时绘制时所需遍历的叶子结点数,且当视点距地形较远时,简化程度较高. 实验证明,该算法适于应用到三维地表模型的实时动态绘制中. 如何进一步提高算法的速度、平衡绘制速度和图像质量、解决数据的动态内存管理等问题还需进一步的研究.

参考文献

1 Williams L. Pyramidal parametrics. Computer Graphics, 1983, 17(3): 1～11
2 Von Herzen Brian, Barr Alan H. Accurate triangulations of deformed, in tersecting surfaces. Computer Graphics, 1987, 21(4): 103～110
3 Jr. DeHaemer M, Zyda M J. Simplification of objects rendered by polygonal approximations. Computers and Graphics, 1991, 15(2): 175～184

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

4 Gross M H, Gatti R, Staadt O. Fast multiresolution surface meshing. In: Proceedings of the Visualization'95. Los Alamitos, CA: IEEE Computer Society Press, 1995. http://www.inf.ethz.ch/publications/tr.html

5 De Floriani L, Falcidieno B, Puppo E. A hierarchical triangle-based model for terrain description. In: Frank A U, Campari I, Formentini U eds. International Conference GIS——From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning. New York: Springer-Verlag, 1992. 236～251

6 Scarlatos L, Pavlidis T. Hierarchical triangulation using cartographic coherence. Graphical Models and Image Processing. 1992, 54(2): 147～161

7 Alan D Kalvin, Taylor R H. Superfaces: polygonal mesh simplification with bounded error. IEEE Computer Graphics and Applications, 1996, 16(3): 64～77

8 Hoppe H. Efficient implementation of progressive meshes. Technical Report, MSR-TR-98-02, 1998

9 Hoppe H. Smooth view-dependent level-of-detail control and its application to terrain rendering. In: Ebert D, Rushmeier H, Hagen H eds. Proceedings of the Visualization'98. Los Alamitos, CA: IEEE Computer Society Press, 1998. 35～42

10 Xia J C, Varshney A. Dynamic view-dependent simplification for polygonal models. In: Proceedings of the Visualization'96. Los Alamitos, CA: IEEE Computer Society Press, 1996. 327～334

11 Michael Garland, Heckbert P S. Fast triangular application of terrains and height fields. Multiresolution Surface Modeling Course. In: Proceedings of the SIGGRAPH'97. 1997. http://www.cs.cmu.edu/～garland

12 Li Jie. Model simplification and multiresolution representation for 3D triangular mesh [Ph.D. Thesis]. Beijing: Tsinghua University, 1998
(李捷. 三角网格模型的简化及多分辨率表示[博士学位论文]. 北京: 清华大学, 1998)

13 Lindstrom P, Koller D, Ribarsky W *et al*. Real-Time, continuous level of detail rendering of height fields. Computer Graphics (SIGGRAPH'96), Annual Conference Series, 1996. 109～118

14 Pajarola R. Large scale terrain visualization using the restricted quadtree triangulation. In: Ebert D, Rushmeier H, Hagen H eds. Proceedings of the Visualization'98. Los Alamitos, CA: IEEE Computer Society Press, 1998. 19～26

15 Liu Xue-hui, Wu En-hua. Accelerating techniques in rendering terrain surface based on image-space error. Journal of Software, 1998, 9(7): 481～486
(刘学慧, 吴恩华. 基于图像空间判据的地表模型加速绘制技术. 软件学报, 1998, 9(7): 481～486)

16 Qi Dong-xu. Fractal and Its Computer Generation. Beijing: Science and Technology Press, 1994. 188～195
(齐东旭. 分形及其计算机生成. 北京: 科技出版社, 1994. 188～195)

17 Brivio P A, Marini D. A fractal method for digital elevation model construction and its application to a mountain region. Computer Graphics forum, 1993, 12(5): 297～309

Level of Detail Dynamic Rendering of Terrain Model Based on Fractal Dimension

WANG Lu-jin TANG Ze-sheng

(*Department of Computer Science and Technology Tsinghua University Beijing* 100084)

Abstract In this paper, the authors use a tree structure based on fractal dimension to organize the 3D-terrain mesh model and realize real time continuous view-dependent level of detail rendering of terrain data. The total algorithm can be divided into two stages: preprocessing and rendering. In preprocessing, it builds an adaptive tree structure by considering the complexity of terrain area which was evaluated by the fractal dimension. Then it generates triangular mesh approximation by dynamically determining reserved data points according to the view related factors and performing a real time restricted quadtree triangulation. Experimental results show that this algorithm is simple and efficient, and supports the interactive real time dynamic rendering of terrain model.

Key words Multiresolution simplification, fractal dimension, restricted quadtree, terrain model, dynamic rendering.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c