

基于细节迁移的快速外存模型简化方法*

费广正¹, 蔡康颖¹, 吴恩华^{1,2}

¹(中国科学院 软件研究所 计算机科学开放研究实验室, 北京 100080);

²(澳门大学 科技学院, 澳门)

E-mail: {feigz, cky}@ios.ac.cn; fstehw@umac.mo

http://www.ios.ac.cn

摘要: 现有的快速外存模型简化方法一般是对模型进行近似均匀采样, 无法根据细节分布对模型进行不同程度的简化, 从而对于细节分布非均匀的模型失真较大. 给出一种快速的外存模型简化方法, 用于对无法一次装入内存的大型几何模型进行简化, 大大改善了细节分布非均匀的模型的简化结果. 该方法首先均匀采样, 对外存模型快速生成初始简化模型, 生成的同时计算原模型特征的统计信息, 然后依此对初始简化模型进行细节迁移和局部精细采样. 该方法在保持近似线性处理速度的同时, 大大提高了简化模型的质量, 所需的内存空间也非常小.

关键词: 外存模型简化; 细节定位; 细节迁移; 非均匀采样; 边折叠

中图法分类号: TP391 文献标识码: A

在过去的 10 年中, 模型简化一直是计算机图形学中一项非常重要的研究内容. 人们提出了许多高效的模型简化方法, 然而这些方法通常只能处理中等规模的数据模型. 这些模型可以一次装入内存, 从而可以利用诸如顶点重要度排序^[1]、模型频域分析^[2]等细节定位方法进行处理. 随着三维扫描及 CT/MRI 设备精度的逐步提高, 计算机获取的三维模型也越来越复杂. 尤其是在最近几年, 模型的规模更是突飞猛进地增长, 例如, IBM 及斯坦福大学的研究小组对 Michelangelo 雕塑作品扫描后生成的几何数据就高达数十亿三角形之巨. 对模型进行精细扫描有时是非常必要的, 尤其是对于那些已经出土但目前的技术尚无法予以保护的珍贵文物, 这种需求就更为迫切了. 但是如此巨量的数据量对图形的绘制、编辑、压缩等无疑都是一个巨大的挑战, 即使对于期望缓解上述压力的模型简化方法来说也是一个不小的难题.

在对上述巨型数据量的模型进行简化时, 除了巨大的内存需求无法在现有条件下满足之外, 其过于漫长的处理时间也是人们难以接受的. 例如, 利用目前较为流行的基于边折叠的方法对模型进行简化, 对于一个由 n 个顶点构成的模型, 其内存需求理论值为 $m \times n$ (m 为几十到几百的常数), 上述由数十亿个三角形构成的模型则需要几百 Gigabytes 的内存, 即便拥有如此巨大容量的内存, 对于一个高档图形工作站来说, 其简化处理时间也将达几个星期, 这是我们无法忍受的.

目前还没有多少方法可以实现高质量的外存模型简化, 这是由于以往快速模型简化方法通常需要考虑对模型进行一些局部操作, 而这些局部操作是依赖于模型顶点之间的连接关系的. 例如, 以顶点删除或边折叠为基本操作的简化算法通常需要给出顶点或边周围的三角形信息. 可想而

* 收稿日期: 2000-11-10; 修改日期: 2001-03-15

基金项目: 国家自然科学基金资助项目(69873044); Research Grant of University of Macau 基金资助项目(RG009/99-005/WEH/FST)

作者简介: 费广正(1973-), 男, 安徽合肥人, 博士, 主要研究领域为计算机图形学; 蔡康颖(1977-), 女, 浙江宁波人, 硕士生, 主要研究领域为计算机图形学, 虚拟现实; 吴恩华(1947-), 男, 江苏南通人, 教授, 博士生导师, 主要研究领域为计算机图形学, 虚拟现实, 科学计算可视化, 计算机动画.

知,对于存储在外存中的大型几何模型,获得这些邻域数据是如何的困难.因此,以往的外存模型简化方法不外乎以下两种:将模型分成多块进行简化,或者仅仅利用有限的连接关系,对模型进行简化.后者由于是一种近似的均匀采样方法,因此处理速度非常快(接近线性处理速度).然而,这两种方法都不可避免地以损失模型简化质量的代价来换取可以接受的简化时间和空间需求.

为了克服上述缺陷,本文的方法是,首先对外存模型用均匀采样的方法快速生成一个初始简化模型,在均匀采样的同时记录下原模型在细节分布的统计信息.通过对该统计信息的分析,我们可以得出原模型细节信息的全局分布特征,从而在以上得到的初始简化模型上,进行细节迁移和局部精细采样.该方法保持了近似线性的处理速度,大大提高了简化模型的质量,同时内存需求量也非常小.

1 前人的工作

目前,已有的外存模型简化方法主要有两类:分块模型简化方法和近似均匀采样的简化方法.后者的处理速度更快.

Bernardini 等人的方法^[3]属于第 1 类,专门用于进行外存模型简化.该方法首先将模型分成多个足以在内存中进行处理的小块,再用常规的方法对每个块分别进行简化.在下次重复简化的过程中再形成更大的块,并允许对前一次分块的接缝进行简化.为了保持模型的连续性,避免在块与块的连接处出现裂缝,算法中限制每个分块的边界不允许修改.Hoppe^[4]在创建多层次 LoD 地形时也使用了类似方法.虽然这种方法在概念上非常简单,但是它需要对模型先进行分块处理再进行拼接,这无疑增加了模型简化的时间,因此并不适合处理规模非常大的几何模型.

Rossignac 和 Borrel^[5]的模型简化方法属于第 2 类.他们首先用许多 Cell 对模型进行均匀划分,然后用一种启发式的方法,在与模型相交的 Cell 中选择一个视觉重要性最大的点作为代表点,从而对模型进行简化.在这种方法中,顶点的重要度等级被用来决定 Cell 中的代表点.由于上述计算并不依赖于模型的精确连接关系,因此,虽然作者并未在文中明确指出,但它确实是一种可用于外存模型简化的方法.不过,虽然该方法实现简单,但简化效果并不好.注意到该方法与其网格划分位置的紧密相关性,Low 和 Tan^[6]给出了一种“Floating Cell”的方法,即 Cell 的划分并不是一成不变的,而是随着简化过程的进行,Cell 中心逐次选择最重要的顶点,并用于吸引它周围任意形状的 Cell 内的所有顶点.这种方法在提高简化质量的同时,需要对顶点依重要度排序,因此需要 $O(n \log n)$ 的处理时间,当 n 值很大时,上述处理时间可能是线性处理时间的几十倍.

受 Rossignac 和 Borrel 方法的启发,Lindstrom^[7]给出了一种求解 Cell 中最优代表点的简化方法.在他的方法中,模型与 Cell 相交的所有三角形构成模型在 Cell 内的三角形集 S ,最优代表点使得与 S 中每个三角形构成的小四面体体积的平方和最小.为了求解该最优代表点,Lindstrom 改进了 Garland 和 Heckbert^[8]提出的用二次误差(quadric error)记录邻域信息的方法.Garland 等人用一个 4×4 的矩阵 Q 记录每个顶点周围平面与空间任意一点距离的平方和二次方程的各项系数,而 Lindstrom 则通过在式中引入面积加权使该矩阵可用于记录体积的平方和.对于每个输入三角形 $t = (v_1, v_2, v_3)$,相应的 Q 矩阵定义如下:

$$Q = \begin{pmatrix} A & b \\ b^T & c \end{pmatrix} \rightarrow \tilde{n} \tilde{n}^T, \quad (1)$$

$$\tilde{n} = \begin{pmatrix} v_1 \times v_2 + v_2 \times v_3 + v_3 \times v_1 \\ [v_1, v_2, v_3] \end{pmatrix}, \quad (2)$$

$$Q(v) = \left(\frac{1}{3} S h_f \right)^2 = \frac{1}{9} [S(n^T v + d)]^2 = \frac{1}{9} (v^T A v + 2b^T v + c). \quad (3)$$

式(2)中的前3维向量 $v_1 \times v_2 + v_2 \times v_3 + v_3 \times v_1 - (v_2 - v_1) \times (v_3 - v_2)$ 实际上是该三角形面积乘以其法向量,第4维的 $[v_1, v_2, v_3]$ 表示3个顶点向量的混合积,式(3)定义了空间一点 v 与三角形 t 所构成的小四面体体积的平方. 考察式(3)可知,该运算具有线性特性,即 $Q_1(v) + Q_2(v) - (Q_1 + Q_2)(v)$. 因此,如果我们对每个 Cell 都记录一个 $\sum Q$,则该矩阵可用来计算空间一点与 Cell 中所有三角形构成的小四面体体积的平方和,计算方法仍如式(3)所示. Lindstrom 在算法中所求的 Cell 的最优代表点就是使该体积平方和达到最小值的点,由于式(3)是一个二次式,求其最小值点实际上是一个线性问题,即求解线性方程组 $Ax + b = 0$.

对于每个输入三角形,算法首先计算与此三角形相应的 Q 矩阵,然后判断每个顶点属于哪一个 Cell,再将得到的 Q 矩阵累加到该 Cell 的 Q 矩阵上. 如果三角形的两个或3个顶点同属于一个 Cell,则该 Q 矩阵被累加了两次或3次. 求解每个 Cell 最终的 $\sum Q$ 所决定的方程就得到了各自的最优代表点. 对于可能存在的退化方程组,算法中还给出了利用奇异值分解进行求解的方法.

考察 Lindstrom 的算法可知,该方法实际上是一种近似的均匀采样方法:每个 Cell 是对模型的均匀划分,且每个 Cell 中仅输出一个顶点,尽管代表点的位置是通过对该 Cell 中的所有三角形求解一个优化方程组得到的. 因此,该简化算法并不具备细节定位能力,也就是说,不能根据模型的细节分布情况决定简化的多少. 对于模型细节分布非均匀的模型,其简化结果往往不甚理想.

2 基于细节迁移的简化方法

为了获得更高的模型质量,我们考虑对模型进行基于尖锐特征(sharp feature)的细节增强,使模型的尖锐特征在简化模型中得到更好的保持.

若希望模型经细节增强后仍保持与均匀采样模型相当的规模,则需要对模型进行细节迁移. 所谓细节迁移是指降低模型某些部位的细节,同时提高其他部位的细节,细节的变化在量上保持平衡,即删除的顶点(或三角形)的数目等于增加顶点(或三角形)的数目. 显然,对原模型的采样率越高,统计信息对原模型细节分布的反映能力也就越好. 因此,首先选择与期望输出规模相当的初始均匀采样率进行简化,再对该模型进行基于尖锐特征增强的细节迁移,比先进行较低分辨率采样再单纯地增加细节的方法效果要好. 细节迁移可以按以下过程进行:

(1) 根据均匀采样时记录的原模型统计信息对尖锐特征进行定量计算,并依此对顶点进行排序.

模型的尖锐特征有以下一些类别:折痕(crease)、角点(corner)、尖刺(dart)、锥(cone)、公共尖刺等. 对这些特征的保持和增强可显著地提高模型的视觉质量.

(2) 删除重要度较低的一些顶点.

由于删除一个顶点将影响其相邻顶点的重要度,因而顶点的删除过程实际上是顶点重要度列表的动态排序过程. 若把顶点删除看做是细节资源的释放,则以后的工作可以看做是所释放资源在模型细节分布较多处的重新分配.

(3) 局部精细采样.

对于模型中未被删除的顶点,若不含尖锐特征,则在重新分配细节时保持不变,否则对其进行子分以增强其尖锐特征. 子分时可以利用原模型信息,以得到更加精确的结果.

(4) 剩余细节处置.

由于事先无法知道增强每处尖锐特征最终耗费的顶点(三角形)数目,因此很难保证删除顶点后释放的所有细节资源恰好用完,这就产生了所谓的冗余细节资源.如果期望保证特征增强后的简化模型在顶点(三角形)数目上与均匀采样的结果相同,则需要考虑如何处理冗余细节资源的问题.一般来说,处理这些冗余细节资源可以有两种办法:一种是“就地处置”,也就是说,在一些重要的尖锐特征处用插值方法生成一些新顶点进行子分;另一种是“原物归还”,也就是说,将这些冗余细节资源按顶点删除重要度逆顺序归还,由于边折叠方法是一种可逆操作,因此细节归还操作也非常容易实现.

总之,我们的算法将经过均匀采样、尖锐特征分析和细节资源删除、局部精细采样、细节资源归还、模型输出等几个过程.下面我们分别对上述过程作进一步的阐述.

2.1 均匀采样

对三维模型进行均匀采样事实上可以看成是平行的两个过程,即顶点的均匀采样和拓扑连接信息的重构.对于用三维扫描仪等数字化方法产生的超大规模三维模型,通常都会给出其包围盒信息,我们将该包围盒均匀划分成大小相同的 Cell,从每个与模型相交的 Cell 中求取一个顶点就得到了顶点均匀采样的结果.对于拓扑连接关系,图 1 演示了其重构方法.图 1(a)~(c)的左侧是原模型的一个三角形与 Cell 可能的 3 种位置关系,其右侧则是与之对应的不同的简化结果.原模型中 3 个顶点同在一个 Cell 中的三角形退化为简化模型中一个顶点,跨越两个 Cell 的三角形退化为一条线段,而跨越 3 个以上 Cell 的三角形则仍是一个三角形.由此可见,只需保留原模型中 3 个顶点分属不同 Cell 的三角形即可重构原模型的拓扑连接信息,不难证明,这种方法构造的拓扑连接信息是完备的,即它可以保持原模型在空间上的连续性.

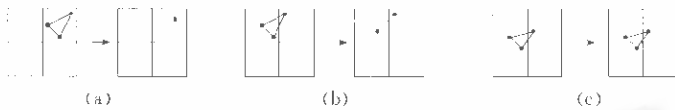


Fig. 1 The relation of the triangles of the original model and the cells
图1 三角形与Cell的位置关系

然而,上述重构三角形连接信息的方法并不能保证不出现三角形翻转现象.例如,若从原模型上选择的三角形 ABC 位置关系如图 2 中的左图所示,每个 Cell 求解出相应的新点后,三角形 $A'B'C'$ 位置关系如图 2 中的右图所示,显然,三角形出现了翻转.由于在选择用于建立模型拓扑关系的三角形时,无法事先确定最终顶点的位置,因此这种由顶点位置变化导致的三角形翻转是无法避免的.当出现这种情况时,我们只需对生成的三角形进行翻转,即将三角形 $A'B'C'$ 输出为 $C'B'A'$ 即可.具体的做法是,在最初从原模型中选择三角形时,计算其法向量并保存在待输出三角形结构中,最终输出三角形前再计算结果三角形的法向量,若前后两个法向量乘积的结果小于 0,则出现翻转,需将三角形逆序后再输出.



Fig. 2 Presence of flip triangle
图2 输出三角形出现翻转

2.2 尖锐特征分析

Hoppe 等人^[9]把模型的尖锐特征分成以下类别:

- 折痕:一条本身光滑的曲线,曲面在该曲线的两侧分别光滑,而在该曲线处不光滑;
- 公共尖刺:两条折痕的交点;
- 角点:3条或3条以上折痕的交点;

尖刺: 曲面内部折痕的终止点;

锥: 折痕退化为一个点, 即除这一点以外, 周围其他各处均光滑。

从上面的尖锐特征分类中可以看出, 折痕是曲面中最基本的尖锐特征, 其他的尖锐特征都可以看作由折痕派生或退化而成。折痕是有一定走向的, 从几何上理解, 折痕类似于一条空间曲线, 其中, 法向量产生突变的方向是垂直于折痕的方向, 而法向量变化较为缓慢的方向则是沿着折痕的方向, 折痕将原模型在该处划分成各自光滑的两侧。

在式(2)中记 $v_1 \times v_2 + v_2 \times v_3 + v_3 \times v_1$ 为 n , 记 $[v_1, v_2, v_3]$ 为 r , 则式(1)转化为式(4):

$$Q = \begin{pmatrix} A & b \\ b^T & c \end{pmatrix} = \begin{pmatrix} nm^T & -rn \\ -rm^T & r^2 \end{pmatrix}. \quad (4)$$

在上式中, $A = nm^T$ 。由代数知识我们知道, A 是一个实对称的半正定矩阵, 即它的所有特征值均为非负实数, 因此 $Q(v) - \varepsilon$ 实际上定义了一个椭球方程(也可能退化为一个圆柱面或两个平行平面), 矩阵 A 的特征值、特征向量分别定义了椭球的半径和坐标轴。若 A 的特征值分别为 $\lambda_1, \lambda_2, \lambda_3$, 则 3 个轴向半径分别为 $1/\sqrt{\lambda_1}, 1/\sqrt{\lambda_2}, 1/\sqrt{\lambda_3}$ 。另外, 由于矩阵 A 是 Cell 内一系列三角形法向量按面积加权的向量与其转置向量乘积的累加和, 因此从统计理论上来看, A 实际上也是 Cell 内三角形法向量面积加权的二阶中心矩。椭球的形状特征反映了模型法向量的变化情况, 若椭球形状近乎一个托盘, 则表明法向量沿着模型表面各方向变化率近似均等; 若椭球形状细长, 则法向量沿着椭球半径较窄的方向变化迅速, 沿着椭球半径较宽的方向变化缓慢。

假如 A 的特征值 $\lambda_1, \lambda_2, \lambda_3$ 已经按升序排列, 也即椭球半径 $1/\sqrt{\lambda_1} \geq 1/\sqrt{\lambda_2} \geq 1/\sqrt{\lambda_3}$, 我们定义椭球半径比为 $1/\sqrt{\lambda_1} - 1/\sqrt{\lambda_2} = \sqrt{\lambda_2}/\sqrt{\lambda_1}$, 椭球半径最小 ($1/\sqrt{\lambda_3}$) 的方向实际上是区分模型内外的主方向, 与模型的形状特征无关。椭球半径比越大则表明模型在该区域变化越剧烈, 反之则表明变化缓慢。若椭球半径比大, 不失一般性, 我们可以假设有一条折痕经过该处。

定位了初始简化模型上的尖锐特征之后, 模型的其他部分则看成是平滑的, 降低这些部位的细节对模型绘制效果影响不大。为了给出一个合理的细节删除顺序, 我们对上述尖锐特征的定量计算结果进行排序, 其正向顺序看成是需要增加细节的顺序, 而其逆向顺序则看成是删除细节的顺序。

由于上述算法在判定折痕时实际上利用了某种统计特性, 这就不可避免地将造成统计噪声, 例如, 可能将一些变化较大的随机散点也看成了折痕, 这显然是不合理的。对随机散点进行局部精细采样, 是不会有太大质量改善的, 相反地, 还将造成不必要的细节资源浪费。为了避免这种现象的发生, 算法在标记为折痕的顶点集中利用三角形的连接关系建立连通图, 每个连通图看成是一条折痕, 当折痕的长度(通过连通图上的最大最短路径定义)过短时, 则认为它是一条假折痕, 应予以剔除。但在剔除过程中, 注意不能破坏周围折痕的完整性和连通性。剔除这些顶点的处理过程实际上是首先将这些顶点从待增加细节的顶点列表中删除, 再根据其周围顶点的信息对其进行中值滤波。我们假定该顶点与周围顶点产生较大不同是由于均匀采样求解最优代表点时的结果误差导致的, 而并非来源于模型本身在该处的不连续性。

2.3 细节资源删除

确定细节的删除(边折叠)过程有两种方法。

第 1 种方法是在每折叠一条边时即计算影响区域的代价并重新排序, 因而细节删除过程实际上是一个动态排序过程。我们可以通过使用一个候选队列和两个交替切换的活动队列来提高排序

速度.具体实现方法是:从当前活动队列中选取最小代价顶点进行边折叠,修改影响区域内的顶点代价并将其加上“被移走”标志,然后放入候选队列,注意保持候选队列中最小代价顶点居于队首.下一个最小代价顶点在当前活动队列的下一个没有“被移走”元素和候选队列首元素之间选择,重复上述过程,直到满足细节删除需求为止.在处理过程中,若候选队列的元素个数超过其数组长度,则将当前活动队列和候选队列通过归并排序放入另一个队列中,切换当前活动队列到归并排序结果队列上,继续处理过程.

第2种方法是在折叠每一条边时,不立即计算影响区域的代价,而是将其冻结,冻结顶点在解冻前不参与细节删除,删除细节仍按原来的顺序寻找下一个活动顶点进行边折叠.若一遍处理完成后尚未满足细节删除的需求量,则一次性将所有冻结顶点解冻,重新计算其代价并排序,然后重复上述细节删除过程,直到满足细节删除需求为止.

显然,第2种方法的处理速度较快,并且可以避免某些部位因细节资源过度删除而产生过大的三角形面.因此,我们在算法中主要使用第2种方法进行细节删除.

2.4 局部精细采样

因为折痕是模型上最基本的尖锐细节特征,因此,下面我们主要讨论通过局部精细采样来增强折痕特征.用第2.2节中的方法定位出模型的折痕之后,对模型的精细采样并不能简单地来进行,否则即使耗费大量顶点或三角形,往往仍不能明显提高其质量.事实上,在上述折痕定位方法中已经给出了增加细节的依据,也就是说在法向量变化最快的方向上增加采样点.

局部精细采样的具体做法是,以折痕经过的顶点和代表法向量变化最快方向的特征向量建立点法式方程作为所属 Cell 的分割平面,将 Cell 一分为二,然后对两个子 Cell 重新运用第1步的方法各选取一个代表点,并建立该位置的三角形连接关系即可.此时,由于两个子 Cell 形状并不规则,不能通过包围盒的方法判定顶点属于哪个子 Cell,只能根据顶点在分割平面的前后向来决定.

从第2.2节对矩阵 A 的特征值和特征向量的分析中可知,除了最大特征值对应的特征向量近似为模型在该处的平均法向量方向以外,另外两个特征向量都和模型近似相切.因此,由其中一个特征向量作为平面法向量决定的分割平面与模型在该区域的曲面近似垂直,大多数情况下,模型分布在该平面的两侧,即子分成功.相反地,若错误地选择最大特征值对应的特征向量作为分割平面的法向量,则在大多数情况下,模型仅在分割平面的一侧分布,从而在多数情况下子分失败,不会对模型质量有任何改善.这也从反面说明了选取合适的子分方法的重要性.由于我们只能保证大多数情况下子分成功,即简化模型中原先的一个顶点用精细采样的两个顶点替换,少量的顶点由于未能使分割平面两侧都有原模型顶点分布而子分失败,并且由于算法是对原模型进行顺序处理的,子分失败直到处理完最后一个三角形才能确定,因此,算法不能保证由第2.3节边折叠删除的顶点资源被完全用完,这时就需要考虑细节资源归还的问题.

2.5 细节资源归还

由于在删除细节资源时采用的是边折叠方法,注意到边折叠是一种可逆操作,也就是说,由折叠目的顶点上记录的信息可以很容易地恢复折叠前的模型信息,因此,由上一节中形成的未用完细节资源可以很容易地经边折叠的逆操作实现细节资源的归还.

但是,为了能够做到允许细节资源归还,顶点信息中需要记载不少额外信息,这无疑增加了系统开销以及处理时间.因此,在没有严格要求模型精确的输出顶点数目时,可以不考虑增加这一步操作,模型的质量不会有太大变化.

2.6 模型输出

为了尽量节省内存开销,我们对模型的均匀划分建立了一张哈希表,每个顶点对应于哈希表上的唯一位置,每个表项用一个指针指向实际的顶点信息结构,于是未存储顶点信息的表项仅保存一个空指针即可,从而避免了因顶点信息较多而引起的内存浪费.同样地,由于并不是每个顶点都进行子分,因此在顶点信息中也用指针指向其子分信息,空指针表示未进行子分.三角形信息记载顶点间的拓扑连接关系,在顶点输出前仅由组成该三角形的3个顶点的指针表示,待确定了顶点的输出顺序之后,三角形才转换成由每个顶点在输出列表中相应的下标来表示.

3 实验结果

图3给出了对斯坦福大学三维模型库中提供的 Dragon 模型的简化结果,从图上可以很容易地看出,Lindstrom 的近似均匀采样的方法在头部等法向量变化剧烈的区域表现出很大的随机性,与原模型误差较大,而经过细节迁移的简化模型在全身各处均很好地保持了原模型的细节特征,尤其是在耳朵、嘴角和龙须等各处很好地保持了折痕的走向.

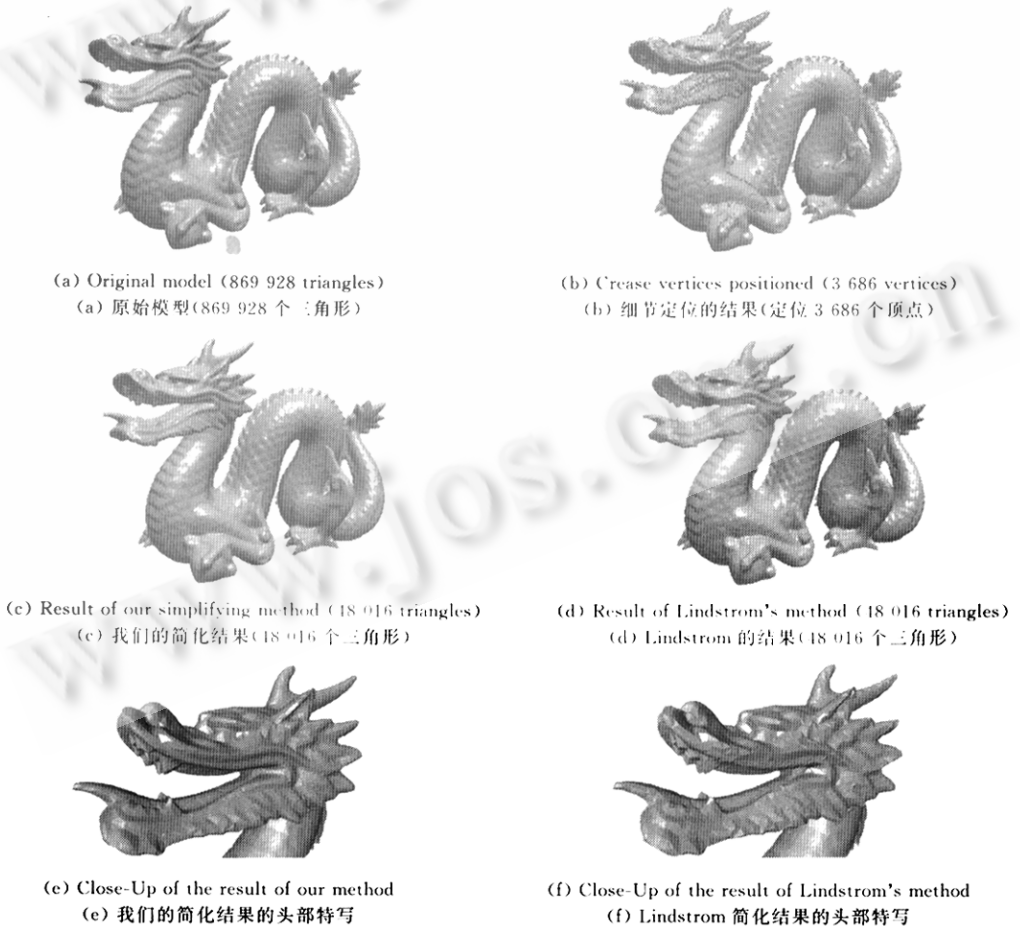


Fig. 3 Comparison of the result of our method and that of Lindstrom's

图3 我们的方法与 Lindstrom 方法的简化结果比较

由于本算法在很大程度上是基于对原模型的统计分析,三角形信息可以逐个处理,并不依赖于顶点或三角形的邻接信息,因此对模型的规模没有任何限制。

我们对一组数据进行了测试,表 1 给出了本文所提出的算法的执行效率、空间开销及其对比情况,所有数据均在 Pentium III 667,256M 内存的微机测得。从表中给出的数据可以看出,基于细节迁移的算法在只增加少许处理时间的基础上,对模型的简化质量有了显著的提高。

Table 1 Time and memory cost of the two simplification methods

表 1 两种简化算法时间及内存开销

Model ^①	$ T_{in} $	$ T_{out} $	DT	Memory cost ^② (MB)		Time cost ^③ (s) ④	
				OoCS	DSA	OoCS	DSA
Dragon ^⑤	871 306	166 280	9 500	15.8	20.2	9.6	13.5
Dragon	871 306	48 016	3 500	2.8	4.0	4.9	6.9
Buddha ^⑥	1 087 716	185 694	15 500	15.9	20.9	11.3	16.8
Buddha	1 087 716	77 692	4 000	4.6	6.6	6.9	9.5
Woman ^⑦	618 600	103 499	6 500	13.1	15.9	9.7	12.5

①模型,②内存开销,③时间开销,④龙模型,⑤佛模型,⑥妇女头像模型。

在表 1 中, $|T_{in}|$ 是指原模型的三角形个数, $|T_{out}|$ 是指简化结果的三角形个数,DT 是指迁移的顶点数目,OoCS 为 Lindstrom 的外存模型简化算法^[7],DSA 是本文所述的简化算法。

4 进一步的工作

由于在本算法中有一个隐含的假设,即认为原模型的法向量变化并不是完全杂乱无章的,而是有一定的走向,且不同的走向之间有一定的间隔,从而不允许在初始均匀采样时一个 Cell 内有多个走向存在,即出现多条折痕。当这种情况发生时,就很难保证能够根据 Cell 内模型法向量变化的统计信息得出模型的正确走向。虽然在本算法中可以通过提高初始均匀采样率解决该问题,但在内存需求或模型输出规模不允许的情况下,该问题仍然得不到解决。因此,进一步的工作将着重于解决原模型法向量变化比较杂乱的情形,即允许在小 Cell 大小范围内出现多个法向走向的变化。另外,在本算法中,迁移的顶点数目还是一个用户给定的参数,在今后的工作中希望能够根据原模型的细节分布情况来自动给出迁移顶点数目。

References:

- [1] Schroeder, W. J., Zarge, J. A., Lorensen, W. E. Decimation of triangle meshes. *Computer Graphics*, 1992, 26(2): 65~70.
- [2] Eck, M., DeRose, T., Duchamp, T., et al. Multiresolution analysis of arbitrary meshes. In: Cook, R., ed. *Proceedings of the ACM SIGGRAPH'95*. New York: ACM Press, 1995. 173~182.
- [3] Bernardini, F., Mittleman, J., Rushmeier, H., et al. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 1999, 5(4): 349~359.
- [4] Hoppe, H. Smooth view-dependent level-of-detail control and its application to terrain rendering. In: Hagen, E. D., Rushmeier, H. H., eds. *Proceedings of the IEEE Visualization'98*. New York: ACM Press, 1998. 167~174.
- [5] Rossignac, J., Borrel, P. Multi-Resolution 3D approximation for rendering complex scenes. In: Falcidieno, B., Kunii, T., eds. *Modeling in Computer Graphics*. New York: Springer-Verlag, 1993. 455~465.
- [6] Low, K.-L., Tar, T. S. Model simplification using vertex-clustering. In: Cohen, M., Zeltzer, D., eds. *Proceedings of the 1997 Symposium on Interactive 3D Graphics*. ACM Press, 1997. 75~82.
- [7] Lindstrom, P. Out-of-Core simplification of large polygonal models. In: Akeley, K., ed. *Proceedings of the SIGGRAPH 2000*. ACM Press, 2000. 259~262.

- [8] Garland, M. , Heckbert, P.S. Surface simplification using quadric error metrics. In: Whitted, T. , ed. Proceedings of the SIGGRAPH'97. ACM Press, 1997. 209~216.
- [9] Hoppe H. , DeRose T. , Duchamp T. , *et al.* Piecewise smooth surface reconstruction. In: Glassner, A. , ed. Proceedings of the SIGGRAPH'94. ACM Press, 1994. 295~302.

Fast Out-of-Core Simplification Based on Detail Shifting*

FEI Guang-zheng¹, CAI Kang-ying¹, WU En-hua^{1,2}

¹(Laboratory of Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100080);

²(Faculty of Science and Technology, University of Macau, Macau, China)

E-mail: {feigz,cky}@seu.edu.cn; fstehw@umac.mo

<http://www.ios.ac.cn>

Abstract: Most existing fast out-of-core simplification methods are based only on uniform or approximated uniform sampling and thus hard to perform adaptive sampling according to the detail distribution of the models. The loss of visual fidelity may be large when processing models with non-uniform distribution of detail. In this paper, an efficient high quality out-of-core simplification method is presented. It can deal with models that are too large to be loaded into main memory. This method greatly improves the simplification result of the models whose detail is non-uniform distributed. In this paper, an initial simplified model is first generated by uniform sampling. A statistical analysis of the original model is also performed at the same time, based on which detail shifting and locally refined sampling to the model can be performed. The algorithm greatly improves the quality of the result model while the processing time is still linear to the model size and the memory cost is also small.

Key words: out-of-core simplification; detail positioning; detail shifting; non-uniform sampling; edge collapse

* Received November 10, 2000; accepted March 15, 2001

Supported by the National Natural Science Foundation of China under Grant No. 69873044; the Research Grant of University of Macau of China under Grant No. RG009/99-00S/WEH/FST