

体积保持的多分辨率多边形网格的光顺造型*

刘新国¹ 鲍虎军¹ 王平安² 彭群生¹

¹浙江大学CAD&CG国家重点实验室, 浙江杭州 310027

²香港中文大学计算机科学与工程学系, 香港

摘要 本文给出了一个高效的多边形网格的多分辨率光顺造型算法。该算法首先通过引入体积保持约束, 快速地实现稠密多边形网格的多分辨率表示。然后结合一个有效的无收缩光顺算法, 对网格执行高效的多分辨率光顺编辑。与传统编辑方法不同, 本方法利用体积保持约束及优化技术来自动恢复编辑区域中相应的细节, 而无需引入非常损耗资源的局部标架。实验结果表明, 该算法计算稳定、高效, 能产生复杂的模型。

关键词 几何造型, 多分辨率模型, 多边形网格, 光顺

1 引言

基于分片多项式曲面表示, 传统几何造型技术在计算机辅助设计和图形学等领域得到了广泛的应用。但是由于多项式基函数仅能很好地表示光滑的三边或四边曲面片, 欲构造复杂的物体表面, 必须实现复杂曲面片沿其公共边界线的光滑拼接[1]。研究者为此付出了巨大的努力, 尽管如此, 复杂曲面区域间的高阶曲面光滑拼接仍非常困难。目前, 主要有参数多项式和代数多项式曲面两类造型技术。

为避免复杂的分片多项式曲面的光滑拼接, 近几年出现了一些新的曲面造型技术, 如细分曲面造型技术, 势能场造型技术等。势能场造型技术通过光滑地过渡简单隐式势能场来构造复杂势能场, 进而抽取势能场的等值面来生成所需曲面[2]。由于隐式表示方法的优越性, 在曲面的光滑过渡方面, 该技术具有良好的特性。但由于该技术采用骨架(如点、线)来定义所需的隐函数, 因而生成的势能场具有非常复杂的表示, 导致等值面的抽取非常耗时, 难以达到实时交互[2,3]。更致命的是, 隐函数的光滑特性使得该技术无法生成尖锐的边界轮廓, 而其骨架定义方式又难以达到实际应用的精度要求, 因而势能场技术往往应用于无精度要求的领域, 如计算机动画领域。细分曲面造型技术一般化了经典样条曲面表示使得它能处理和生成一般拓扑网格控制的曲面, 从而克服了传统样条曲面要求规范的控制拓扑网格的缺陷[4,5,6,7,8]。应用简单的迭代细分规则, 这种技术可容易地生成任意形状的整体光滑曲面, 它事实上是细分过程生成网格的极限曲面。但在实际应用中, 我们往往只需执行有限的细分步骤即可满足要求。本质上, 细分曲面技术提供了一种多分辨率造型的机制。

与此同时, 随着自动空间激光测距技术的发展, 我们可方便地直接通过测量实物模型得到稠密的三维测量点集, 并采用自动重建技术得到原模型表面的一个稠密三角形网格表示。由于其直观的处理方法, 这一技术在模型的快速设计过程中表现出了极大的优越性, 成为目前应用广泛的表面造型技术之一。但由于庞大的采样数据, 用户往往对得到的网格难以实行进一步的编辑和修改, 为此, 近几年出现了许多网格简化算法[9,10], 以在一定精度要求下简化稠密网格, 便于进一步的应用, 从而导致了稠密网格的多分辨率表示的出现。根据表示方法的不同, 网格的多分辨率表示主要有均匀细分和累进表示两种。均匀细分多分辨率表示完全继承了传统多分辨率表示模式, 它通过对原始网格的重新采样来自动建立从粗糙网格到精细网格的映射[11]。因该表示方法要求原始网格始终拓扑同胚于最粗的基本网格, 且其拓扑具有剖分连通性(subdivision connectivity), 所以这种表示方法显得比较僵硬, 不够灵活。前述的细分曲面造型技术则可归结为这一方法。Eck 等结合小波的多分辨率表示技术给出了任意网格的均匀细分多分辨率分析[12]。而以 Hoppe 的累进网格简化算法为代表的累进多分辨率表示技术通过在简化过程中记录粗糙网格到精细网格的映射来实现始于基本最粗网格的细节恢复[10]。该表示方法

*本项目受香港政府研究资助局(RGC)合作研究中心项目(Project Title: Virtual Reality, Visualization and Imaging Research center)的资助。第二作者还受中国国家自然科学基金杰出青年基金(批准号: 69925204)和高等学校骨干教师资助计划项目的资助。

无需网格的重新采样过程,因而可方便地恢复出初始网格。为提高稠密网格的编辑速度,Zorin等基于均匀细分分辨率表示方法实现了网格的多分辨率编辑[13],而Kobbelt等则基于累进多分辨率表示方法实现了同样的目标[14]。这两个算法的基本思想是相同的,二者均在简化过程中采用局部坐标标架来刻画不同细节顶点间的空间几何关系,一旦某个细节顶点的几何发生变化,它所相连的坐标标架将发生变化,并带动所附着的细节亦随着变化。为此,它们均需化大量的时空耗费去计算和存贮这些坐标标架。

本文首先给出一个快速的体积保持的累进多分辨率表示算法,并采用一个优化的过程来动态实现相邻层次细节间的映射关系,最后结合作者提出的无收缩、无变形的网格光顺算法实现稠密网格的多分辨率编辑造型。

2 基本符号

不失一般性,本文将仅讨论三角形网格,当然它适用于一般的多边形网格。一个三角形网格 M 表示为一三元组 $\langle V, T, X \rangle$,其中 $V = \{1, 2, 3, \dots, N\}$ 为网格 M 的顶点集; $T \subseteq 2^V$ 是 M 上的三角面片,其中任一三角形 $t \in T$ 可表示为顶点元素的有序三元组 $t = \langle i, j, k \rangle$; $X: V \rightarrow R^3$ 是一个将每个顶点映射到它所处的空间位置的映射。记号 $\text{Adj}(i) \subseteq V$ 表示与顶点 i 相邻的顶点集合,相应地, $n(i)$ 表示顶点 i 相邻顶点的个数。在不引起混淆的情况下,我们用 x_i 代表 $X(i)$ 表示顶点 i 的空间位置;并用 X 表示所有顶点坐标组成的向量,即 $X = [x_1 \ x_2 \ \dots \ x_N]^T$ 。

3 体积保持的多分辨率表示

正如前面所述,网格的多分辨率表示可分为两类。一类是基于剖分连通性的表示,按照“一分为四”的剖分规则,模型在不同粗细表示间进行切换;另一类是基于累进网格的表示,通过一些“点分裂”和“边收缩”,模型在不同粗细表示间进行切换。鉴于前者需重新采样原始给定网格,因而不可避免地会带来不必要的误差。所以,本文采用基于累进网格的多分辨率表示。因此我们首先要设计一个快速的简化算法,实时地生成高质量的多分辨率网格。

我们知道,对于基于“边收缩”的这一类简化算法,为了尽量地逼近原始模型,最好的方法是按照误差的顺序选择被收缩的边,即:首先对所有可以收缩的边,计算收缩它所引起的误差,并按照误差进行排序;然后每次选择误差最小的边进行收缩操作。因为彼此接近的边是相互关联的,所以收缩一条边后,还要重新计算相关的边的收缩误差,并重新排序。由于误差计算和排序是非常耗时的,因此对较大的模型不可能做到实时。为此我们采用“两步法”快速选择被收缩的边,同时较好的逼近原始模型,

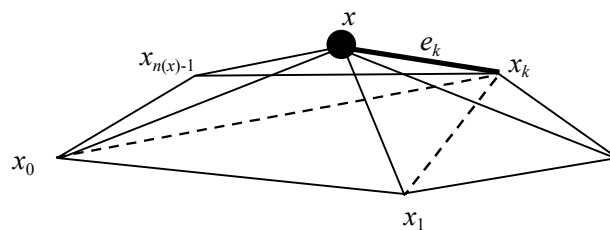


图1 体积保持简化的误差度量

第一步:选择被删除的顶点。我们按照顶点最小入射边 $\min_edge(x)$ 和最大入射边 $\max_edge(x)$ 的加权平均值作为被删除的优先级:

$$\text{priority}(x) = \alpha \cdot \min_edge(x) + \beta \cdot \max_edge(x) \quad (1)$$

其中 $\min_edge(x)$ 和 $\max_edge(x)$ 分别为顶点 x 的入边中最短和最长的边长。参数 α, β 是两个常数。值越小优先级越高。一般来说,顶点周围一圈入射边的大小反映了它所表示的几何特征的大小,边长越大,特征越大;反之亦然。因此我们采用式(1)中所定义的加权平均值作为顶点删除的优先级,这样保证越先删除的顶点所表示的几何特征越小,符合多分辨率的思想。同时注意到不同大小的边长反映特征的不对称性,所以在选择权值时, α 通常取值要远大于 β ,例如在我们的所有实验模型中,都选取: $\alpha = 1.0$, $\beta = 0.01$ 。

第二步：在即将删除的顶点的周围选择被收缩的边。选择被收缩的边的原则是尽量保持几何特征，较好地逼近原始模型。传统算法通常采用人工交互方法，或自动夹角检测方法首先标识出模型特征，再进行相应的简化。显然人工交互的方法对我们不合适，而自动检测的方法本身只对棱角分明的机械零件模型比较合适，否则检测的特征不能完全反应模型特征，从而严重影响模型的简化和多分辨率表示。因此，我们抛弃特征标识的方法，提出如下的局部体积保持标准来同时度量模型表面的特征和评估边收缩所引起的误差。

设 x 为被选择的欲被删除的点， $x_k (k=0,1,\dots,n(x)-1)$ ($n(x)$ 为 x 的入度) 为 x 的周围邻接点，我们利用围绕 x 的锥状体来度量网格由于边收缩引起的局部形状变化。不妨假设边 $e_k = \langle x, x_k \rangle$ 将被收缩(见图 1)，则 x 处被其周围三角形 $\Delta x x_j x_{j+1} (j=0,1,\dots,n(x)-1)$ 和底部三角形 $\Delta x_j x_{j+1} x_k (j=0,1,\dots,n(x)-1)$ 所包围的锥状体的体积可用来度量顶点 x 处的局部形状变化。该锥的体积可表示为

$$\text{collapsed_volume}(e_k) = \sum_j ((x_j - x_k) \otimes (x_{j+1} - x_k)) \cdot (x - x_k) \quad (2)$$

显然，当 x 和其邻接相邻点共面时，其局部邻域锥的体积为 0。容易发现，局部锥的体积与简化网格的质量有着密切的联系。体积越小，简化结果越好。为此，我们定义边收缩的代价函数为：

$$\text{cost}(e_k) = |\text{collapsed_volume}(e_k)|$$

因此， x 的入边中具最小代价的边 $e_{\text{collapsed}}$ 将被收缩(删除)，即

$$\text{cost}(e_{\text{collapsed}}) = \min_{0 \leq k < n(x)} \text{cost}(e_k) \quad (3)$$

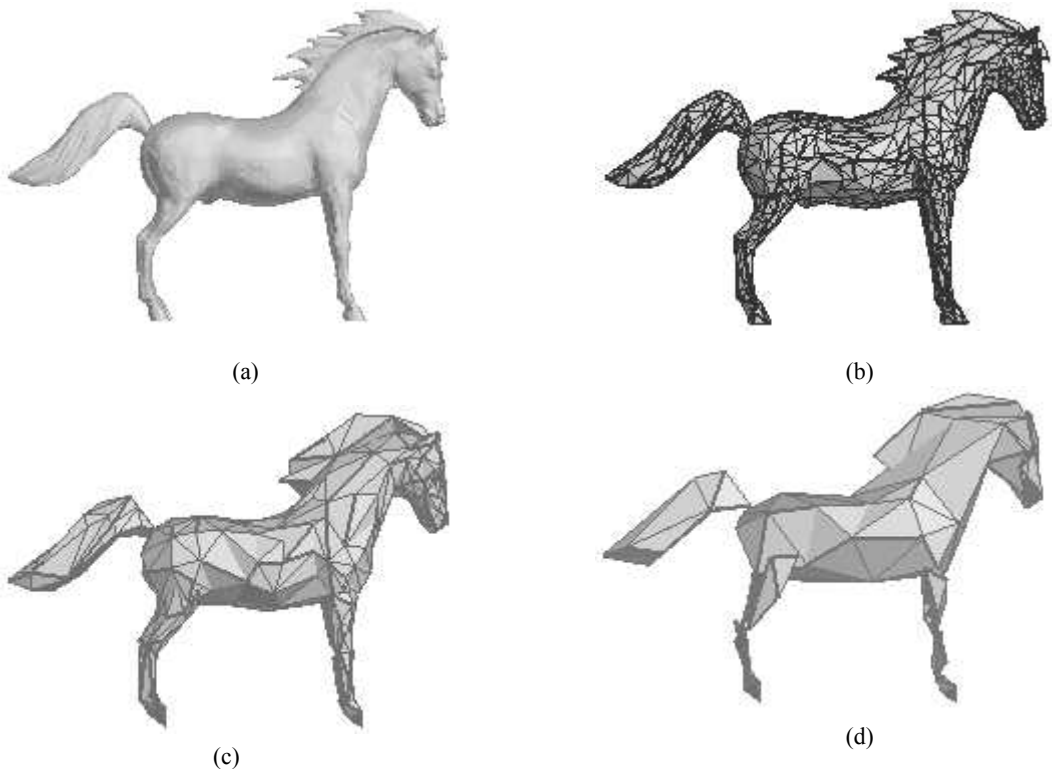


图 2 骏马模型的简化结果。(a) 原始模型(10912 个顶点); (b) 简化模型(610 个顶点); (c) 简化模型(146 个顶点); (d) 简化模型(79 个顶点);

为了避免一次边收缩操作后相邻顶点优先级的计算和顺序的更新，以加快算法的速度，我们还加入了“独立点集”的限制，即：每次删除了一个顶点之后，就将其周围的相邻的顶点暂时定为“不可删除”，直到所有剩下的顶点都是不可删除的时候再将它们全部清除。图 2 给出了采用这一算法生成的骏马模型和兔子模型的简化结果。容易发现，我们的两步法能很好地保持模型的特征。

4 重心约束的光顺算法

注意到多分辨率编辑时, 细节的恢复可能导致曲面的不光顺, 这是由于在细节恢复时, 原来建立的细节映射函数并不一定满足光顺约束条件[13]。其结果则将原来光顺的物体表面编辑为不光顺的表面, 在实际应用中, 这是绝对不允许的。为此, 我们必须在细节恢复过程中, 交替地保持曲面网格的光顺性。

理论上, 连续参数曲面 $S: X = X(u, v)$ 的光顺技术是使得如下的曲面能量达到最小[15, 16]:

$$E(S) = \alpha E_{\text{memb}}(S) + \beta E_{\text{thin}}(S)$$

其中 $E_{\text{memb}}(S) = \frac{1}{2} \int_S (X_u^2 + X_v^2) dudv$, $E_{\text{thin}}(S) = \frac{1}{2} \int_S (X_{uu}^2 + 2X_{uv}^2 + X_{vv}^2) dudv$ 分别为 S 的薄膜(membrane)能量和薄板(thin-plate)能量, 这里 $X_u, X_v, X_{uu}, X_{vv}, X_{uv}$ 分别表示曲面的一、二阶偏导数。 α, β 为位于 0 和 1 间的常数。

应用上述能量于离散三角形网格 M , 在特殊的局部参数化下, 取适当的参数 α, β , 我们可利用 Laplace 算子来线性逼近上述能量[17,18], 即

$$E(M) = \sum_{i \in V} |L(x_i)|^2 = X^T (K^T K) X \quad (4)$$

其中 $L(x_i) = \sum_{j \in n(x_i)} \omega_{ij} (x_j - x_i)$ 为一阶 Laplace 算子。 ω_{ij} 是组合系数, 满足: $\omega_{ij} \geq 0$ 并且

$\sum_{j \in n(x_i)} \omega_{ij} = 1$ 。显然, $K = (\omega_{ij})_{N \times N}$ 是一个稀疏矩阵。

鉴于直接采用优化技术使(4)式达到最小, 即采用 Laplace 算子迭代磨光, 将引起严重的曲面变形和收缩[14,18,19]。在文[19]中, 我们引进了重心约束到方程(4)中, 有效避免了网格的形变。这样, 问题就成为求解如下的条件极值问题:

$$\begin{cases} \text{目标函数} & X^T (K^T K) X \\ \text{约束条件} & HX - HX^0 = 0 \end{cases} \quad (5)$$

其中每个三角形 $t = \langle i, j, k \rangle$ 对应于矩阵 H 的一个行向量 H_t , 且满足 $H_{ts} = \begin{cases} 1 & s = i, j, k \\ 0 & \text{其他} \end{cases}$,

X^0 表示三角形网格 M 中顶点的初始位置。

我们在文[19]中给出了方程(5)的迭代求解算法。其具体步骤如下:

Pold: Input array of the original vertex locations

Pnew: Output array of the new vertex locations

Cnew: Temporary array of the number of new vertex locations

Initialization: *Pnew* = Null; *Cnew* = Null;

for each interior triangle $t = \langle i, j, k \rangle$ of the mesh

Use Equation (7) and *Pold* to compute new locations *Newi*, *Newj*, *Newk* of vertices i, j, k ;

Pnew[i] = *Pnew*[i] + *Newi*; *Cnew*[i] = *Cnew*[i] + 1;

Pnew[j] = *Pnew*[j] + *Newj*; *Cnew*[j] = *Cnew*[j] + 1;

Pnew[k] = *Pnew*[k] + *Newk*; *Cnew*[k] = *Cnew*[k] + 1;

end for

for all vertices v

if *Cnew*[v] > 0 then

Pnew[v] = *Pnew*[v] / *Cnew*[v];

end if

end for

return;

5 体积保持的多分辨率编辑

一旦完成网格的多分辨率构造,我们就可任意选择一个层次模型来对网格进行编辑和修改。不失一般性,假设 $\{M_i, i=0,1,\dots,n\}$ 为采用前述简化算法生成的多分辨率模型,其中 M_0 为给定的初始网格模型。显然, M_i 具有不同的层次细节复杂性,层次越高,复杂性就越低。编辑时,用户首先交互地选择一编辑层次,如 M_k ,然后自由地用传统编辑技术对网格 M_k 的顶点进行编辑修改。编辑时,用户既可以移动一个顶点,亦可选择一组顶点,对它们进行平移,旋转或比例缩放。一旦编辑完毕,算法将根据各个顶点的局部修改情况计算出下一层网格细节顶点的偏移量,最后得到其最终的位置。这样,我们可从 M_k 修改 M_{k-1} ,依此类推,直至到达最细网格 M_0 为止。因此,多分辨率编辑技术的问题就转化为如何快速稳定地根据上一层较粗细节的顶点偏移量计算出更细细节顶点的偏移量。传统多分辨率编辑技术均采用局部坐标标架来建立相邻两层细节顶点间的映射关系[13,14]。由于局部坐标标架需在网格简化过程中建立并预先存贮起来,因而传统算法非常耗空间和时间。为提高编辑过程的计算稳定性,我们将抛弃局部标架结构,代之以直接利用原来的形状信息确定初始位置,进而根据顶点编辑的结果采用最小二乘技术估算出恢复顶点细节的新位置。

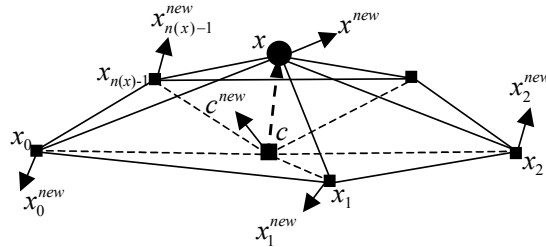


图3 顶点恢复的几何说明

在恢复网格的细分过程中,要将更细一层的顶点细节逐步加进去。由于本算法不改变网格的连接关系,只是编辑顶点的位置,所以需要解决的问题是如何确定被恢复顶点的空间位置。一般地,所确定的新点位置应使得网格尽量保持原来的局部形状,为此,我们再次利用局部体积来度量顶点周围的局部形状。假设 x 为将被恢复的顶点, $x_i (i=0,1,\dots,n(x)-1)$ 为其邻接的周围一圈相邻顶点, c 是这些相邻顶点组成空间多边形的重心,则通过一系列的体积值来衡量 x 点处的局部邻域,即:

$$V_i = (x_i - c) \otimes (x_{i+1} - c) \cdot (x - c) \quad i = 0, 1, \dots, n(x) - 1$$

这样,当 x_i 偏离了原来的位置到达新的位置 x_i^{new} 后,上述计算得到的局部体积序列亦将随之变化。根据多分辨率编辑要求,被恢复顶点细节 x 亦将发生相应的变化。类似于前面的体积保持的简化算法,我们将 x 移动到一个新的位置 x^{new} 使得编辑前后的体积变化最小。因此, x^{new} 的坐标可以通过求解以下的最小二乘问题得到:

$$\begin{cases} (x_0^{new} - c^{new}) \otimes (x_1^{new} - c^{new}) \cdot (x^{new} - c^{new}) = V_0^0 \\ (x_1^{new} - c^{new}) \otimes (x_2^{new} - c^{new}) \cdot (x^{new} - c^{new}) = V_1^0 \\ \dots\dots\dots \\ (x_{n-1}^{new} - c^{new}) \otimes (x_0^{new} - c^{new}) \cdot (x^{new} - c^{new}) = V_{n-1}^0 \end{cases} \quad (6)$$

其中 c^{new} 为 $x_i^{new} (i=0,1,\dots,n(x)-1)$ 构成多边形的重心,方程右边的 V_i^0 是用编辑前的 x 和 x_i 的初始几何数据计算出来的一系列的体积值(如图3所示)。

值得注意的是,当 x_i 保持不变时,上述最小二乘问题的解应是平凡解,即 x 点的位置保持不变。此时,为避免出现数值计算不稳定,我们将 c 点沿 $x_i^{new} (i=0,1,\dots,n(x)-1)$ 确定的近似平面的法向方向移动一小距离。

考虑到上述细节恢复技术并不保证得到光顺的曲面编辑结果,我们将上一节中的网格光顺算法用来进一步调整恢复细节顶点的位置。算法交替地执行细节恢复和光顺操作,直到最细层网格为止。图 4 给出了多分辨率编辑的过程。

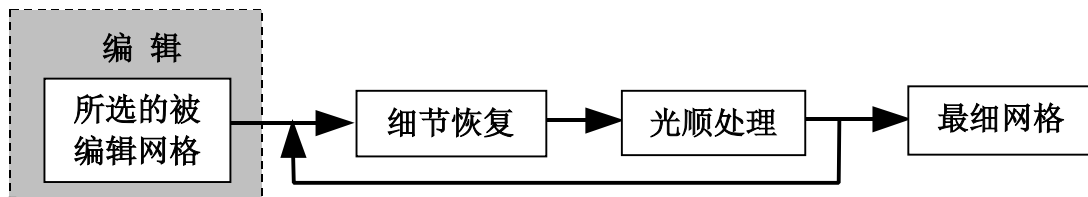


图 4 多分辨率编辑过程

6 执行结果

我们用 C++ 在 Pentium III 500 (128 兆内存) 的个人计算机上实现这个稠密多边形网格的光顺造型原型系统。本系统由四部分组成,即散乱数据的网格重建,网格的累进多分辨率表示,光顺处理和多分辨率编辑模块。本文主要介绍了系统的多分辨率编辑技术的实现方法。图 5 给出了兔子模型的多分辨率表示的几个层次,模型共有 24 层层次细节。容易发现,初始模型的特征在各个层次模型上均得到了很好的保持。生成这一模型只需 4.84 秒。图 6 为在不同层次上编辑同一点的结果。容易发现,在越粗的层次上编辑,其影响区域亦越大,最后的曲面越光顺。反之,影响范围越小,曲面变化越厉害。因而,用户可有选择地在不同层次上进行编辑,最后合成非常复杂的编辑结果。图 7 给出了兔子模型的多分辨率编辑结果。我们在第 11 层上将兔子的嘴巴和尾巴往外拉出了一定距离,通过逐步细节恢复和光顺处理,得到了如图 7(b)、(c)和 (d)所示的编辑效果。在这个例子中,系统在整个编辑过程中基本达到了交互速度。

7 结论

多分辨率造型技术是新近发展起来的一种新的曲面造型技术。该技术充分利用硬件技术来达到快速构造复杂曲面的目的。本文介绍了一个高效的多边形网格的多分辨率光顺造型算法。算法首先通过引入体积保持约束,快速地实现了稠密多边形网格的累进多分辨率表示。结合一个有效的无收缩、无变形光顺算法,我们提出并实现了一个高效的多分辨率光顺编辑方法。与传统编辑方法不同,本算法利用体积保持约束及最小二乘技术来自动恢复编辑区域中相应的细节,而无需引入非常损耗资源的局部标架。实验结果表明,本算法能编辑生成复杂的模型,且具有计算稳定、高效的特点。因而本算法可望在 CAD 和计算机动画等系统中得到应用。

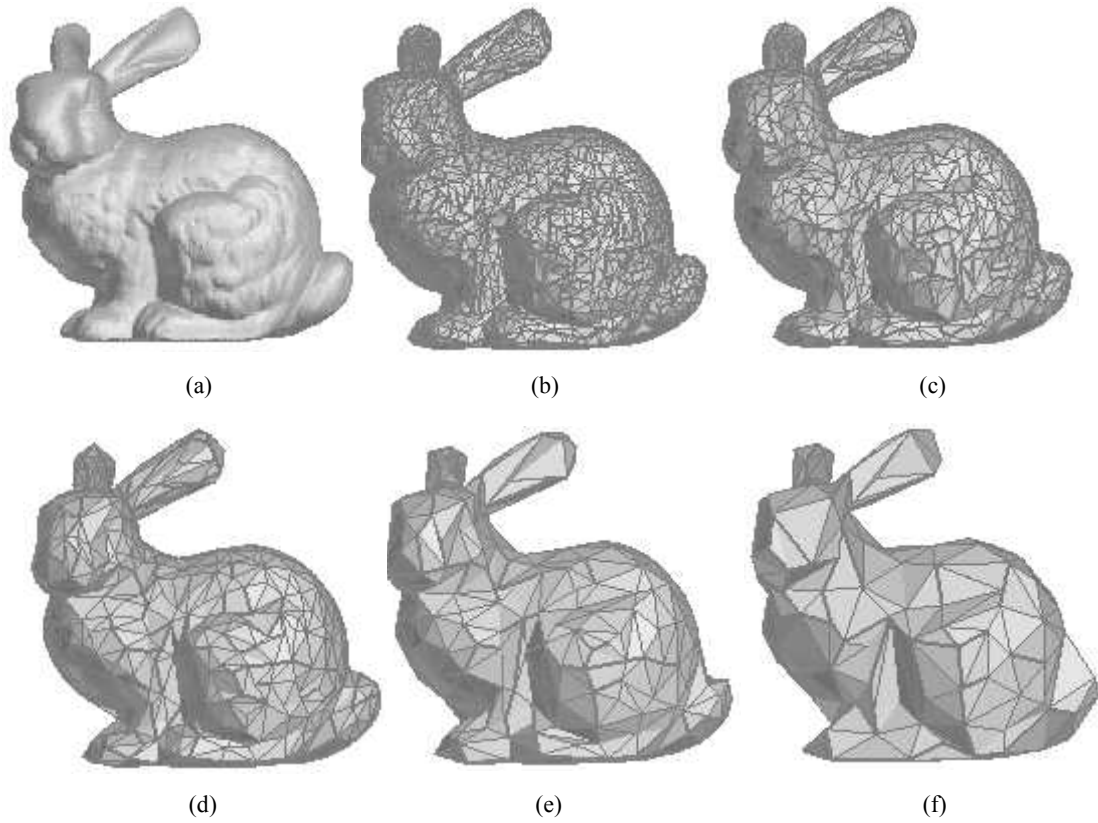


图 5 兔子模型的多分辨率表示(24层)。(a)原始模型(34835个顶点);
(b)简化模型(2625个顶点); (c)简化模型(1420个顶点); (d)简化模型(790个顶点);
(e)简化模型(461个顶点); (f)简化模型(288个顶点);

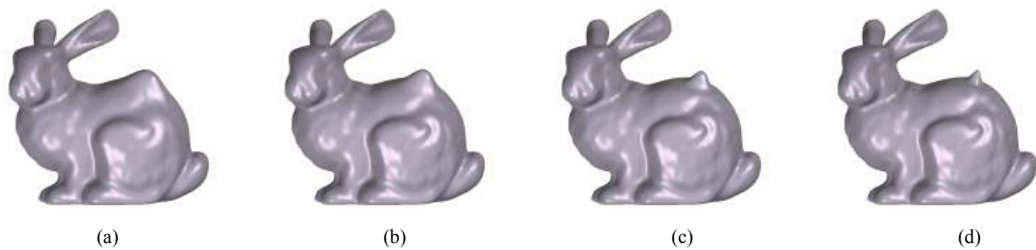


图 6 在不同的简化层次上进行编辑的效果比较图。(a)第十三层编辑结果;
(b)第十一层编辑结果; (c)第九层编辑结果; (d)第七层编辑结果;

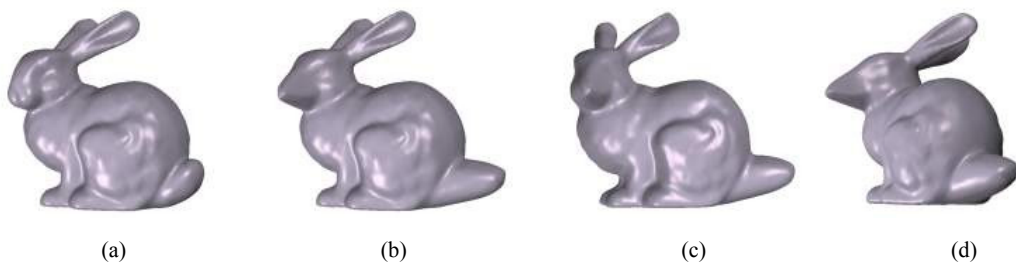


图 7 兔子模型的多分辨率编辑结果。(a)原始模型; (b)、(c)和(d) 在第 13 层处编辑的结果

参考文献

- [1] FARIN, G. Curves and Surfaces for CAGD, 3rd ed. Academic Press, 1993.
- [2] BLOOMENTHAL, J., BAJAJ, C., WYVILL, B., AND WYVILL, G. Introduction to Implicit Surfaces. Morgan Kaufman Publishers, 1997.
- [3] BLOOMENTHAL, J., AND SHOEMAKE, K. Convolution surfaces. Computer Graphics, 1991, Vol.25, No.4, 251-256.
- [4] CATMULL, E., and CLARK, J. Recursively Generated b-spline surfaces on arbitrary topological meshes. Computer Aided Design, 1978, Vol. 10, No. 6, 239-248.
- [5] DYN, N., LEVIN, D., AND GREGORY, J. A. A butterfly subdivision scheme for surface interpolation with tension control. ACM Transactions on Graphics, 1990, Vol. 9, No. 2, 160-169.
- [6] ZORIN, D., SCHRODER, P., AND SWELDENS, W. Interpolating subdivision for meshes with arbitrary topology. SIGGRAPH'96 Proceedings, Edited by Holly Rushmeier etc., New Orleans, Louisiana, 1996, 189-192.
- [7] DEROSE, T., KASS, M., AND Truong, T. Subdivision surfaces in character animation. SIGGRAPH'98 Proceedings, Edited by Michael Cohen etc., Orlando Florida, 1998, 85-94.
- [8] SEDERBERG, T. W., ZHENG, J. M., SWELL, D., AND SABIN, M. Non-uniform recursive subdivision surfaces. SIGGRAPH'98 Proceedings, Edited by Michael Cohen etc., Orlando Florida, 1998, 387-394.
- [9] GARLAND, M., AND HECKBERT, P. S. Surface simplification using quadric error metrics. SIGGRAPH'97 Proceedings, Edited by John M. Snyder, Los Angeles, 1997, 209-218.
- [10] HOPPE, H. Progressive meshes. SIGGRAPH'96 Proceedings, Edited by Holly Rushmeier etc., New Orleans, Louisiana, 1996, 99-108.
- [11] LEE, A. W. F., SWELDENS, W., SCHRODER, P., COWSAR, L., AND DOBKIN, D. MAPS: Multiresolution adaptive parameterization of surfaces. SIGGRAPH'98 Proceedings, Edited by Michael Cohen etc., Orlando Florida, 1998, 95-104.
- [12] ECK, M., DEROSE, T., DUCHAMP, T., HOPPE, H., LOUNSBERY, M., AND STUETZLE, W. Multiresolution analysis of arbitrary meshes. SIGGRAPH'95 Proceedings, Edited by Robert Cook, Los Angeles, 1995, 173-182.
- [13] ZORIN, D., SCHRODER, P., AND SWELDENS, W. Interactive multiresolution mesh editing. SIGGRAPH'97 Proceedings, Edited by John M. Snyder, Los Angeles, 1997, 259-268.
- [14] KOBBELT, L., CAMPAGNA, S., VORSATZ, J., AND SEIDEL, H. P. Interactive multi-resolution modeling on arbitrary meshes. SIGGRAPH'98 Proceedings, Edited by Michael Cohen etc., Orlando Florida, 1998, 105-114.
- [15] MORETON, H., AND SEQUIN, C. Functional optimization for fair surface design. SIGGRAPH'92 Proceedings, Edited by , Orlando Florida, 1992, 167-176.
- [16] WELCH, W., AND WITKIN, A. Variational surface modeling. SIGGRAPH'92 Proceedings, Edited by , Orlando Florida, 1992, 157-166.
- [17] TAUBIN, G. A signal processing approach to fair surface design. SIGGRAPH'95 Proceedings, Edited by Robert Cook, Los Angeles, 1995, 351-358.
- [18] DESBRUN, M., MEYER, M., SCHRODER, P., AND BARR, A. H. Implicit fairing of irregular meshes using diffusion and curvature flow. SIGGRAPH 99 Proceedings, Edited by, Los Angeles, 1999, 316-324.
- [19] 刘新国, 鲍虎军, 彭群生, 多边形网格的平滑去噪声算法, 自然科学进展, No.7, 2000.
- [20] Liu Xinguo, Bao Hujun, Peng Qunsheng, Noise removal algorithm for polygonal meshes, Nature Science Progress, 2000, July.

Volume Preserved Multiresolution Mesh Modeling

Liu Xinguo¹ Bao Hujun¹ Heng Pheng-Ann² Peng Qunsheng¹

¹State Key Lab. of CAD&CG in Zhejiang University, Zhejiang Hangzhou 310027

²Dept. of Computer Science and Engineering, Chinese University of Hong Kong, HongKong

Abstract In this paper we present a novel multiresolution mesh modeling algorithm. The algorithm first produces a multiresolution representation for the polygonal mesh by introducing volume preserving constraints. By combining with a non-shrink smoothing scheme, the mesh can be edited effectively in a multiresolution fairing way. Unlike other algorithms, our algorithm automatically recovers details in edited regions using volume preserving constraints and optimization technique rather than expensive local frames. Experiments show that our algorithm is very stable and efficient, and can produce complex models.

Keywords: Geometric Modeling, Multiresolution Representation, Polygonal Mesh, Fairing