

基于检测球控制的网格模型简化算法研究*

周儒荣, 唐杰, 张丽艳, 周来水

(南京航空航天大学 CAD/CAM 工程研究中心, 江苏 南京 210016)

E-mail: zlsme@nuaa.edu.cn

http://www.nuaa.edu.cn

摘要: 在逆向工程、计算机图形学等应用领域中,经常采用多边形网格模型(多为三角形网格)来描述几何形体,但网格中三角片数目往往非常庞大,为了保证对模型的后续操作能有效地进行,有必要在满足一定精度的条件下对其进行简化,提出了一种基于检测球控制简化精度的网格模型简化算法,该算法运行速度快,简化效果好。

关键词: 三角网格; 网格简化; 层次细节

中图法分类号: TP391 **文献标识码:** A

三角网格模型在许多领域得到了广泛的应用,如计算机视觉、计算机仿真、逆向工程、数控加工的刀具轨迹计算等。但是,三角网格模型的一大特征就是模型中三角片的数目庞大,例如,曲面重构技术往往要从海量的散乱点集重构具有海量三角片的三角网格模型,而这种三角片数目庞大的网格模型十分不利于显示、传输及其他后续操作。因而有必要在满足一定精度的条件下对其进行简化。

近几年来,国内外许多学者对三角网格模型的简化算法进行了研究。Hoppe^[1]等人提出了一种基于能量优化的网格模型简化算法,该算法通过建立一个能量函数来描述网格模型的状态,并定义了删除边、调换边、拆分边这3种单元操作,在网格简化过程中重复进行这3种单元操作,直到能量函数值不再降低为止,改变能量函数中的系数值可以获得不同简化效果的简化网格模型,该算法可获得较为优化的简化网格模型,但计算量大,运行速度较慢,而且网格简化参数的几何意义不明显。Schroeder^[2]等人提出的算法通过在网格局部区域中评估某一顶点到相关平均平面的距离来确定该顶点是否被删除,如果删除该点以及包含该点的三角片,则对形成的孔洞重新进行三角化,该算法能大量、快速地删除三角片,但由于在简化过程中误差不断积累,因而不能保证简化网格模型与初始网格模型很好地进行匹配。Garland^[3]在其提出的算法中判断某个顶点能否被删除的准则是,该点到所有与其相关的三角片的距离的均方根是否满足精度条件,该算法在用于多细节层次描述模型的生成时取得了较好的效果,但不易控制初始网格与简化网格之间的最大误差。陶志良^[4]也提出了一种基于能量评估的网格模型简化算法,但与Hoppe的算法一样,该算法也存在计算量大、运行速度较慢、网格简化参数的几何意义不明显等问题。

基于上述讨论,本文针对逆向工程中的曲面重构这一课题,提出了一种快速、有效且可直接控

* 收稿日期: 1999-11-02; 修改日期: 2000-06-26

基金项目: 国家自然科学基金资助项目(59905013); 国家863高科技发展计划资助项目(863-511-942-022); 航空基金资助项目(981H52095)。

作者简介: 周儒荣(1935—),男,福建长汀人,教授,博士生导师,主要研究领域为CAD/CAPP/CAM/PDM,逆向工程,快速原型制造; 唐杰(1971—),男,江苏南京人,博士,讲师,主要研究领域为逆向工程,计算机图形学; 张丽艳(1967—),女,辽宁宽甸人,副教授,主要研究领域为逆向工程,计算机图形学,计算机辅助几何设计; 周来水(1962—),男,河北保定人,博士,教授,博士生导师,主要研究领域为计算机辅助设计与制造。

制简化精度的网格简化算法,该算法基于删除边单元操作,利用检测球控制网格简化精度.算法自动化程度高,不产生积累误差,可以保证简化网格模型与初始网格模型很好地进行匹配.

本文第1节介绍网格的描述方法及一些基本概念,第2节是具体算法,第3节提出了一种检测球数目的控制算法,第4节为算法应用,最后是结论.

1 基本概念

为描述方便,先定义一些基本概念.

1.1 网格模型的数学描述

三角网格模型是由三维空间中的三角片通过边和顶点连接而成的分片线性曲面,其中每条边最多只能包含在两个三角片中.一般地,我们将网格模型 M 定义为一个二元组 (K, V) , 其中 $V = \{V_1, \dots, V_m\}$, $V_i \in R^3$, 表示网格模型 M 的顶点在三维空间中的位置; K 为表征网格拓扑结构的单纯复形. 一个单纯复形包含一组单形, 其中 $\{1\}, \dots, \{m\}$ 称为顶点, $\{i, j\}$ 等称为边, $\{i, j, k\}$ 称为面.

1.2 网格模型的误差评估

网格简化的目的就是要获得比初始网格模型简单,但要尽可能与原网格模型相似的简化网格模型.为了能够量化地评估以某个简化网格模型代替初始网格模型所产生误差的大小,有必要建立一种衡量两个网格模型之间差异的误差评估方法.

三维空间中一点 x 到网格模型 M 的距离 d_E 定义为

$$d_E(x, M) = \min_{y \in M} (d(x, y)), \quad (1)$$

其中 $d(x, y)$ 为点 x 和点 y 的欧氏距离.

网格模型 M_0 到 M 的单向 Hausdorff 距离 d_s 定义为

$$d_s(M_0, M) = \min_{x \in M_0} (d_E(x, M)). \quad (2)$$

网格模型 M_0 到 M 的对称 Hausdorff 距离 d_H 定义为

$$d_H(M_0, M) = \min(d_s(M_0, M), d_s(M, M_0)). \quad (3)$$

本文采用对称 Hausdorff 距离来评估两个网格模型之间的差异.若网格 M_0 到 M 的对称 Hausdorff 距离 d_H 小于 ϵ ,则以网格 M 代替 M_0 时产生的误差小于 ϵ .

2 算法描述

2.1 算法基本步骤

本文算法的基本步骤如下:

- (1) 读入初始网格模型 M_0 , 网格模型数据结构初始化, 读入用户指定的简化精度 ϵ_0 ;
- (2) 在 M_0 表面采样检测点集 E ;
- (3) 建立待删除边集, 根据近似曲率值, 对待删除的边进行排序;
- (4) 选择待删除边集中的第 1 条边;
- (5) 计算删除这条边引起的误差是否满足精度要求;
- (6) 如果误差不满足简化精度要求, 选择待删除边集中的下一条边, 执行步骤(5)
- (7) 如果误差满足简化精度要求, 则删除该边, 将该边邻域三角片的检测点传递给相应的三角片, 并重新对待删除的边排序, 执行步骤(4);

(8) 直到待删除边集为空为止.

2.2 检测球的选取

检测点用于控制简化后的网格模型与初始网格模型的匹配,因而要求检测点能记录足够多的网格细节信息.在平坦的网格面上仅需要少量的检测点,而在模型起伏变化较大的部位则需要较多的检测点来记录网格模型的局部特征.本文采用了一种基于特征检测点采样方法:给定任一初始网格 $M_0=(K,V)$,假定 M_0 无自交现象且能很好地表现出模型的各个细节.首先,顶点集 V 中的点可全部作为检测点.其次,在棱边和尖角等区域应采集更多的检测点.定义 B_θ 是这样一个边集: $\{i,j\}$ 为其中的边,若包含 $\{i,j\}$ 的两个三角片所形成的二面角小于阈值 θ ,则认为 $\{i,j\}$ 为特征边.取特征边上的检测点集为

$$E_\theta = \{v | v=1/2(v_i+v_j), (i,j) \in B_\theta\}, \tag{4}$$

最终点集 E 为

$$E = V \cup E_\theta. \tag{5}$$

当检测点采样好以后,即分配至其所属的三角片.这里,我们引入检测球的概念.定义以某个检测点为球心,用户指定的简化精度 ϵ_0 为半径的球为一个检测球.对于每个检测球,若与某个三角片相交,则将构成检测球的检测点分配至该三角片.

2.3 建立待删除边集

在网格较为平坦的区域,三角片被删除的可能性较大.选择一个好的删除顺序有助于获得具有良好的三角片形状的简化网格.

本文根据离散曲面片近似曲率的计算方法,提出一种选择删除边顺序的方法.

离散曲面中某一顶点的近似曲率公式为

$$k = \frac{2\pi - \sum_i \beta_i}{\frac{1}{3} \sum_i a_i}. \tag{6}$$

其中 β_i 为三角片在该顶点处的角度, a_i 为三角片的面积.

对于网格中某条边 $\{i,j\}$,我们定义其删除的优先级为

$$k_c = k_i + k_j. \tag{7}$$

在网格数据初始化以后,即对所有的待删除边按式(7)从小到大进行排序.这样就可以在网格简化时,首先删除网格中最为平坦区域的三角片,而对于较为复杂的区域可以尽量少删除.

2.4 判断删除某条边是否满足要求

- 判断删除边是否满足精度要求

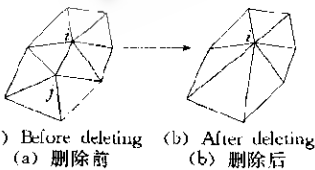


Fig.1 Edge deleting
图1 删除边

如图1所示,当删除一条边 $\{i,j\}$ 之后,三角网格模型仅在该条边的邻域范围内发生了变化,而三角网格模型的其他区域并未发生变化.记删除前网格模型为 M ,删除后为 M' .设这条边的邻域所包含的三角片集为 T_{ij} , T_{ij} 中所有三角片包含的检测球集为 E_{ij} .当 $\{i,j\}$ 被删除时,三角片集 T_{ij} 变为 T'_{ij} (如图1所示).此时,若 E_{ij} 中的每一个检测点所对应的检测球至少与 T'_{ij} 中的一个三角片相交,则该删除边操作满足精度要求.上述条件保证了 M 到 M' 的单向 Hausdorff 距离小于用户指定

的要求 ϵ_0 .

当某条边 $\{i, j\}$ 被删除时, 这条边的两个端点被合并为一个新顶点. 确定新顶点的位置有许多不同的方法. 为了提高算法速度, 在不影响算法效果的前提下, 本文采用子集法, 即以 v_i 或 v_j 作为合并后的新顶点. 这就可以保证删除边 $\{i, j\}$ 后的网格模型 M' 到删除前的网格模型 M 的单向 Hausdorff 距离也小于用户指定的要求, 从而以较小的计算量保证了 M 到 M' 的对称 Hausdorff 距离小于用户指定的要求.

• 判断删除边是否满足形状要求

在实际网格简化过程中, 若不加控制, 有可能出现狭长三角片, 即三角片的某一个内角接近 180° , 而这种狭长三角片对后续处理是不利的. 在本文的算法中, 为避免上述情况的发生, 如果生成的三角片的三个内角中有一个接近 180° 或有两个接近 90° , 则不进行删除边的单元操作.

• 判断删除边是否满足法矢连续性要求

如图 2 所示为平面三角片. 删除边 $\{1, 2\}$ 后的结果如图 2(b) 所示. 显然, 如图 2(b) 所示的三角网格是不正确的, 因为三角片 $\{1, 3, 4\}$ 与三角片 $\{3, 4, 5\}$ 相交了. 出现这种情况的原因是由于三角片 $\{2, 3, 4\}$ 的法矢在删除边 $\{1, 2\}$ 前后发生了突变, 此处为反向. 因此, 在简化过程中必须检查新生成的三角片的法矢与原三角片的法矢相比有无突变, 若有, 则不能进行该删除边的单元操作. 在本文中, 当新生成的三角片的法矢与原三角片的法矢之间的夹角超过 45° 时, 就认为该删除边单元操作不满足法矢连续性要求.

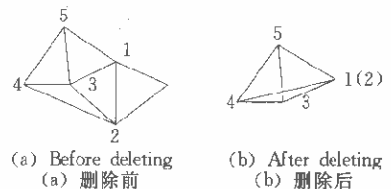


Fig. 2 Check of normal vector consistency
图2 法矢连续性的检查

2.5 检测点重新分配

为使删除边 $\{i, j\}$ 所引起的简化误差评估的是简化中间网格模型 M_i 与初始网格模型 M_0 之间的误差, 而不是与上一步简化网格模型 M_{i-1} 之间的误差, 必须保留被删除三角片所包含的检测点, 并重新分配边 $\{i, j\}$ 邻域三角片所包含的检测点.

设边 $\{i, j\}$ 的邻域所包含的三角片集为 T_{ij} , T_{ij} 中所有三角片包含的检测球集为 E_{ij} . 当 $\{i, j\}$ 被删除时, 三角片集 T_{ij} 变为 T'_{ij} (如图 1 所示). 对于 E_{ij} 中的任意一个检测点, 若它所对应的检测球与 T' 中的某个三角片相交, 则将该检测点分配至该三角片.

检测点的重新分配可以在网格简化过程中很好地保留初始网格模型的形状信息, 这就使得我们在每一步删除操作中所进行的误差检查是控制简化网格模型 M_{i+1} 与 M_0 之间的误差, 而不是仅仅控制 M_{i+1} 与上一步简化网格模型 M_i 之间的误差, 因而可以较好地保证简化网格模型与初始网格模型之间的匹配.

2.6 待删除边的重新排序

当某条边 $\{i, j\}$ 被删除之后, 网格模型在该局部区域的形状发生了变化, 因而要对该局部区域的边的删除顺序进行重新排序. 如图 1 所示, 在这条边被删除之后, 所有包含 T'_{ij} 中顶点的边的删除顺序都要进行重排.

3 检测球数目的控制

如前所述, 检测点是检测球的球心, 因此, 控制检测球的数日也就是要控制检测点的数目. 在初

始网格模型中,每个三角片一般包含 3 个检测点,即三角片的 3 个顶点(特征边周围的三角片包含 3 个以上检测点).随着简化过程的进行,每个三角片包含的检测点的数目不断增加.简化的程度越大,每个三角片包含的检测点的数目就越多,这在网格平坦处尤为明显.三角片的检测点数目过多增加了简化步骤的计算量,影响了网格模型简化的速度,因此必须控制三角片包含的检测点的数目.

如图 3 所示,删除边 $\{i, j\}$ 之后,顶点 $\{i\}$ 移至顶点 $\{j\}$,三角片 $\{i, m, n\}$ 将绕边 $\{m, n\}$ 转动.点 E 为一个检测点, E' 为 E 在边 $\{m, n\}$ 上的投影, α 为线段 EE' 与三角片 $\{i, m, n\}$ 的夹角, $-90^\circ < \alpha < 90^\circ$. 当点 E 在三角片上时, $\alpha = 0^\circ$. 可以看出,当三角片 $\{i, m, n\}$ 绕边 $\{m, n\}$ 转动时,最有可能不满足精度条件的检测点是 α 角最大和最小的点.因此,当三角片所包含的检测点的数目不超过 9 个时,保留所有的检测点.当某个三角片的检测点数目超过 9 个时,作以下处理:保留三角片的 3 个顶点所对应的 3 个检测点;对三角片的每一条边,保留 α 值最大和最小时所对应的检测点.

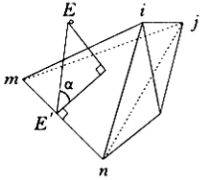


Fig. 3 Deformation of triangle
图3 三角片变形

这样,每个三角片所包含的检测点数就不会超过 9 个,既加快了网格的简化速度,又能较好地保证网格简化精度.

我们在奔腾 III, 450MHz CPU, 128MB 内存的微机上,对如图 4 所示的兔子模型进行了简化.如果不采用检测球控制技术,从 69 351 个三角片简化到 1 118 个三角片所用 CPU 时间为 226s,而采用检测球控制技术为 178s. 同样地,对于如图 5 所示的网格模型,如果不采用检测球控制技术,从 12 946 个三角片简化到 395 个三角片所用 CPU 时间为 42s,而采用检测球控制技术为 32s,速度提高了 30% 以上,且简化结果基本上一致.可见,采用检测球控制技术可以有效地提高网格简化的速度.

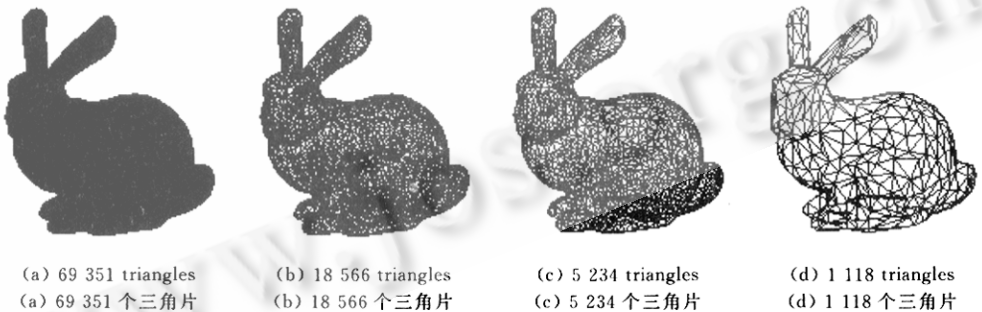


Fig. 4 Bunny model
图 4 Bunny 模型

4 算法的应用

P. Cigoni 等人^[5]给出了他们对几种三角网格模型简化算法的测试结果.他们选用了两个三角网格模型:

(1) Bunny(如图 4(a)所示),这是一个兔子模型,由 Stanford 大学提供.该模型有 34 834 个顶点,69 451 个三角片.模型的外接包围盒为 $15.6 \times 15.4 \times 12.1$.

(2) Fandisk(如图 5(a)所示),这是一个机械零件模型,由 Hoppe 等人^[1]提供.该模型有 6 475 个顶点,12 946 个三角片.模型的外接包围盒为 $4.8 \times 5.6 \times 2.7$.

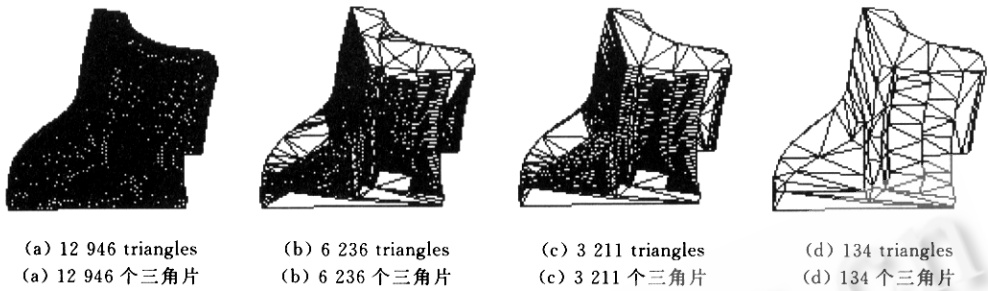


Fig. 5 Fandisk model
图 5 Fandisk 模型

Cigoni 比较了 5 种简化算法: Schroeder 算法^[2]、Cohen 算法^[6]、Ciampalini 算法^[7]、Hoppe 算法^[1]和 Garland 算法^[3]. 前 3 种算法所用的机器为 SGI Indigo2, R4400, 200MHz CPU, 16KB 数据缓存, 16KB 指令缓存, 1MB 二级缓存, 128MB 内存. 第 4 种算法受其代码的限制, 是在 Digital 3000/900, Alpha 275MHz CPU, 128MB 内存上运行的. 最后一种算法是由 Garland 在 SGI Indigo2, R4400 250MHz CPU, 128MB 内存机器上运行的^[6].

我们用 VC6.0 语言实现了本文提出的算法, 并在奔腾 III, 450MHz CPU, 128MB 内存的微机对上述两个网格模型进行了简化. 简化结果如图 4 和图 5 所示.

表 1 和表 2 为本文提出的算法与 P. Cigoni 考察的 5 种算法的运行速度的比较. 虽然本文的算法是在微机上测试的, 但基本上可以看出, 与 Cigoni 列出的几种算法相比, 本文的算法在速度上的优越性是比较明显的. 从图 5 中可以看出, 简化网格模型中的三角片沿曲率较小的方向排列, 这是一种较为优化的分布方式.

Table 1 Comparison between speed of the algorithm of this paper and the other five listed by Cigoni on model Bunny (s)
表 1 本文的算法与 Cigoni 列出的 5 种算法对 Bunny 模型的简化速度比较 (秒)

Algorithm ^② \ Deleting ratio ^① (%)	50	75	90
Schroeder	43.97	37.55	46.22
Cohen	932.17	942.93	944.66
Ciampalini	208.95	371.03	388.68
Hoppe	7.100	7.400	7.600
Garland	20.50	24.70	26.42
The algorithm of this paper ^③	95.05	54.46	29.82

①删除率, ②算法, ③本文的算法.

Table 2 Comparison between speed of the algorithm of this paper and the other five listed by Cigoni on model Fandisk (s)
表 2 本文的算法与 Cigoni 列出的 5 种算法对 Fandisk 模型的简化速度比较 (秒)

Algorithm ^② \ Deleting ratio ^① (%)	50	75	90
Schroeder	4.50	7.38	10.00
Cohen	203.49	228.14	185.22
Ciampalini	28.26	37.72	52.26
Hoppe	2.000	2.100	2.300
Garland	42.74	44.41	45.11
The algorithm of this paper ^③	16.26	8.84	5.59

①删除率, ②算法, ③本文的算法.

5 结 论

网格模型简化算法是逆向工程、计算机图形学等领域中的关键技术. 本文提出了一种三角网格模型的简化算法. 算法简单、明了, 速度快, 且能较好地控制网格的简化精度. 通过合理地选择表征网格细节的检测点集及建立删除边的优先队列, 使简化网格既保留了初始网格的细节特征, 又有较大的压缩比例. 直接以对称 Hausdorff 距离为精度控制变量, 保证了简化网格与初始网格之间的最大误差满足要求, 具有明显的几何意义和工程意义, 便于应用. 本文还提出了一种控制检测球数目的算法, 提高了算法运行速度. 文中给出一组应用实例, 说明了算法的有效性. 该算法可用于逆向工程中的网格简化, 也可用于多层次细节模型的生成.

References:

- [1] Hoppe, H., DeRose, T., Duchamp, T., *et al.* Mesh optimization. *Computer Graphics*, 1993, 27(special issue on SIGGRAPH'93):19~26.
- [2] Schroeder, W., Zarge, J., Lorensen, W. Decimation of triangle meshes. *Computer Graphics*, 1992, 26(special issue on SIGGRAPH'92):65~70.
- [3] Garland, M., Heckbert, P. S. Surface simplification using quadric error metrics. *Computer Graphics*, 1997, 31(special issue on SIGGRAPH'97):209~216.
- [4] Tao, Zhi-liang, Pan, Zhi-geng, Shi, Jiao-ying. Mesh simplification algorithm based on energy evaluation and its application. *Journal of Software*, 1997, 8(12):881~888 (in Chinese).
- [5] Cignoni, P., Montani, C., Scopigno, R. A comparison of mesh simplification algorithms. *Computer and Graphics*, 1998, 22(1):37~54.
- [6] Cohen, J., Varshney, A., Manocha, D., *et al.* Simplification envelopes. *Computer Graphics*, 1996, 30(special issue on SIGGRAPH'96):119~128.
- [7] Ciampalini, A., Cignoni, P., Montani, C., *et al.* Multiresolution decimation based on global error. *The Visual Computer*, 1997, 13(5):228~246.

附中文参考文献:

- [4] 陶志良, 潘志庚, 石教英. 基于能量评估的网格简化算法及其应用. *软件学报*, 1997, 12(8):881~888.

Study on Mesh Simplification Algorithm Based on Controlling by Examining Ball*

ZHOU Ru-rong, TANG Jie, ZHANG Li-yan, ZHOU Lai-shui

(Research Center of CAD/CAM Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

E-mail: zlsme@nuaa.edu.cn

http://www.nuaa.edu.cn

Abstract: Polyhedral meshes (especially triangle meshes), which are characterized by large amount of triangles, are often used for many applications such as reverse engineering, computer visualization, etc. In order to improve the efficiency of subsequent processing, a simplification process is required to reduce the number of vertices of the mesh under error controlling. An algorithm for mesh simplification with bounded error was present in this paper. And the algorithm has high efficiency and good simplifying results.

Key words: triangle mesh; mesh simplification; level of detail

* Received November 2, 1999; accepted June 26, 2000

Supported by the National Natural Science Foundation of China under Grant No. 59905013; the National High Technology Development 863 Program of China under Grant No. 863-511-942-022; the Aeronautical Science Foundation of China under Grant No. 98H52095