DOI: 10.3901/JME.2013.01.148

基于数学形态学与拓扑规则的三角网格修补算法*

蒋恒恒 李奇敏 汤宝平

(重庆大学机械传动国家重点实验室 重庆 400030)

摘要: 针对散乱点云数据在三角剖分过程中产生的拓扑缺陷,提出一种基于数学形态学运算和拓扑规则的网格拓扑修补算法。通过交互的方式选择需要修改的区域,使用自适应分层栅格的缺陷识别技术提取有拓扑缺陷的网格的顶点,从而确定待修复区域的边界,然后利用数学形态学的开启运算和闭合运算去除该修复区域的拓扑缺陷,并利用基于柄体理论的拓扑运算法则对该区域进行局部拓扑修改,生成二维流形的三角网格。应用实例表明,由于不需要对整个点云数据重新进行三角剖分,简化数据处理的过程,该算法具有运算速度快、结果准确性好的优点,并能较好地消除网格中的拓扑缺陷,有效地提高三角网格的显示精度,最终得到具有几何一致性和网格单元拓扑一致性的三角网格模型。

关键词: 点云 数学形态学 开启 闭合 拓扑缺陷 柄体算子 星形算子

中图分类号: TP391

Triangular Mesh Repair Algorithm Based on Mathematical Morphology and Topology Rule

JIANG Hengheng LI Qimin TANG Baoping

(The State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400030)

Abstract: For the topological defects generated during unorganized point cloud triangulation, a new method is presented for mesh repair based on mathematical morphology and topology rule. The repaired areas are chosen with the interactive ways, and the mesh vertexes with potential topological errors are found out by means of a defect detecting technology based on adaptive hierarchical grid so as to determine the boundary need to be repaired. Opening and closing of mathematical morphology being used, the topology of the model can be modified. The local topology repair is able to be finished and a 2-manifold triangle mesh is generated quickly through topological mesh operators based on Handlebody theory. Experimental results show that the triangulation of the whole point cloud is no longer needed so that the data processing is simplified. The proposed algorithm is fast and accurate, and the topological defects can be eliminated satisfactorily and output precision of triangular meshes can be improved efficiently. The triangular mesh model with geometrical and topological consistency is able to be formed.

Key words: Point clouds Mathematical morphology Opening Closing Topological error Handlebody operator Stellar operator

0 前言

三角剖分问题一直是近年来网格剖分技术中的一个热点研究问题^[1]。迄今为止已有很多成熟的三角剖分算法,但由于三维采样设备的固有局限性,以及实体内部的几何性质^[2],大多数剖分算法生成的网格模型存在着拓扑缺陷。BISCHOFF 等^[3]利用

等值面重构与形态算子相结合的方法对三角网格的拓扑缺陷进行修补,该方法虽然简单、易于编程,但修补后网格的精确度有待提高。ZHOU 等^[4]提出一种基于骨架算子的拓扑修补算法,该算法能够去除低亏格模型中的小 handles,但在重构网格的过程中容易引起模型变形。LOMENIE 等^[5]引入 α -Shapes 理论,提出一种基于形态算子与 Delaunay 三角网相结合的三角剖分算法,该算法可较好地处理边界部分的网格,但由于理论部分不够完善,适用范围有限。

以上几种方法在网格修补后、重构网格模型时

* 高等学校博士学科点专项科研基金(20090191120007)和国家自然科学基金(50905190)资助项目。20111222 收到初稿,20121016 收到修改稿

均使用隐式曲面法, 在表示实体边界的拓扑集合时采用传统的 Euler-Poincare 理论。隐式曲面法适合于构造任意拓扑的复杂形体, 在分割点云的过程中, 易于跟踪有拓扑缺陷的网格, 且曲面光顺性较高, 但该方法在构造复杂形状曲面时需要大量的基本体素, 用户对此难以有效控制, 因此不能进行细节造型^[6]。欧拉算子是低层算子, 实现较为简单, 计算量也较少, 但在网格修补的过程中, 会产生一些不合法的中间模型, 最终导致结果模型中出现不需要的亏格^[2], 因此欧拉算子通常要封装更高层的算子^[7]。为了保证网格在修补过程中的拓扑合法性, 为运算法则提供准确的抽象层次, 本文采用较高层次的拓扑算子对网格进行编辑。具体思想如下: 利用自适应栅格分割点云, 在此基础上, 引入形态学算子, 消除网格中的拓扑缺陷, 再利用基于柄体理论与星形理论的拓扑规则, 最终生成二流形曲面。该算法将基于面的二流形创建方法与自适应拓扑修改算法相结合, 避免了传统算法中特定等值面的构建, 不仅能够提高计算效率, 而且用户可以根据实际需要选择要修改的区域, 从而提高了修改精度。

1 算法描述

曲面拓扑重建所得网格模型上的缺陷可分为两种: 点云采样信息不完备导致曲面拓扑重建算法失效而产生的缺陷, 如图 1 所示; 未能根据点云类型选择恰当的曲面拓扑重建算法而导致的缺陷, 如图 2 所示。而拓扑缺陷将影响最终模型生成理想的亏格以及连通分支个数^[8]。弥补第一种拓扑缺陷的方法之一是提高采样密度, 即增加采样点的数量^[3]。如图 1b 所示, 将门框模型的采样点个数由 135 895 个增加到 194 216 个, 可以消除部分拓扑缺陷。提高采样密度不仅需要较好的采样设备, 同时会降低算法的运行速度, 如图 1a 的计算时间为 10.44 s, 图 1b 为 14.54 s; 而且用这种方法消除拓扑缺陷具有一定局限性, 如图 1b 的门框把手处仍有未填平的空洞。第二种缺陷是伪缺陷, 因为样点间距通常小于工艺孔洞, 可在曲面拓扑重建过程中避免误连接情况的发生。如图 2 中的汽车零件模型, 将其划分点云的栅格尺寸减少为原来的 2/3, 可部分修补拓扑缺陷, 如图 2c 所示; 将栅格尺寸减少为原来的 1/2, 可完全修补拓扑缺陷, 如图 2d 所示。但有时用户并不熟悉拓扑方面的知识, 而且在拓扑重建阶段解决伪缺陷也会增加算法的运行时间。如图 2b

的计算时间为 0.59 s, 图 2c 为 2.59 s, 图 2d 为 4.23 s。

图 1 门框模型

图 2 汽车零件模型

数学形态学是一门建立在严格数学理论基础上的学科, 在图像处理中应用广泛^[9], 如今也越来越多地应用到计算机辅助设计及计算机辅助几何设计中。数学形态学用具有一定形态的结构元素去度量和提取图形图像中的对应形状, 从而简化图形图像数据, 保持其基本形状特征, 同时除去其中的缺陷^[10]。本文采用形态学开启和闭合算子修补网格中的拓扑缺陷, 并结合柄体理论与星形理论的拓扑规则重构网格, 达到较好的效果。算法的具体步骤如下。

- (1) 输入点云或三角网格模型, 将其转换为自适应分层栅格集合。
- (2) 用户选择需要修改的区域(可选)。
- (3) 利用自适应分层栅格识别技术分析有缺陷的区域。
- (4) 应用形态算子(开启算子和闭合算子)处理有拓扑缺陷的区域(填补空洞, 创建空洞, 连接间断部分)。

(5) 根据拓扑规则对该区域进行局部拓扑修改, 最终生成二维流形的三角网格曲面。

下面对其中的关键步骤进一步展开论述。

1.1 基于分层栅格的缺陷识别技术

一般情况下, 三角网格的缺陷存在于曲面的外部或内部边缘处、采样密度突变处或者有局部特征的地方^[11]。为了识别这些缺陷, 本文改进柯映林等^[12]提出的基于栅格的点云边界特征直接提取算法, 采用分层栅格识别技术提取任意栅格的内外边界, 找到需要修复的区域。为更好地找到要修补的区域, 本文采用与用户交互相结合的方式: 由用户选择待修复区域, 算法自动确定待修复区域的边界。分层栅格识别算法分为以下三步: ① 建立分层栅格的拓扑连接数学模型, 产生边界种子栅格; ② 种子栅格沿边界方向生长, 产生所有边界栅格; ③ 从边界区域中提取点云的边界点集合。

对于用户选定的待修复区域, 首先使用自适应三角剖分方法中的栅格分层算法^[7], 用较小的栅格尺寸细化该区域的栅格, 再按照实格和空格对选中区域的空间栅格进行二值化处理。

汽车零件模型的分层栅格如图 3 所示。

图 3 分层栅格示意图(汽车零件模型)

令

$$g_l(x, y, z) = \sum_{i=-1}^1 \sum_{j=-1}^1 \sum_{k=-1}^1 |f_l(x, y, z) - f_l(x+i, y+j, z+k)| \quad (1)$$

式中 (x, y, z) ——栅格的坐标

(i, j, k) ——栅格的拓扑方向

$f_l(x, y, z)$ ——第 l 层空间栅格的二值化函数

$g_l(x, y, z)$ ——第 l 层种子边界栅格的函数

如果 $f_l = 1$, 表示该栅格为实格, 包含数据点; 否则 $f_l = 0$, 该栅格为空格。如果 $g_l(x, y, z) \geq 3$, 并且当 $g_l(x, y, z) = 3$ 时满足约束条件 $f_l(x+i, y+j, z+k) = 1$, 该空间实格是种子边界栅

格。种子边界栅格分三种类型: 当 $g_l(x, y, z) \geq 3$ 时, 说明原始曲面的边界比较平坦, 如图 4a 所示; 当 $g_l(x, y, z) = 4$ 时, 表明原始曲面的边界弯曲程度比较剧烈, 如图 4b 所示; 当 $g_l(x, y, z) = 5$ 时, 表明原始曲面的边界存在尖锐特征, 如图 4c 所示。一般情况下, 在网格修补中只需要考虑弯曲程度比较剧烈或含有尖锐特征的边界, 本文只考虑 $g_l(x, y, z) \geq 4$ 时的情况。

图 4 种子边界栅格的三种类型

很明显, 种子栅格集合不包含所有边界栅格, 为了获得所有边界栅格, 可以利用种子栅格自身的拓扑关系使种子栅格沿特定的拓扑方向进行带约束的生长。令

$$T_l(x, y, z, i, j, k) = |f_l(x, y, z) - f_l(x+i, y+j, z+k)|$$

$$G_l(x, y, z) = \sum_{i=-1}^1 \sum_{j=-1}^1 \sum_{k=-1}^1 |f_l(x, y, z) - f_l(x+i, y+j, z+k)|$$

$$f_l(x, y, z) = 1$$

$$-1 \leq i \leq 1 \quad -1 \leq j \leq 1 \quad -1 \leq k \leq 1$$

$$(i, j, k) \neq (0, 0, 0) \quad (2)$$

式中, $T_l(x, y, z, i, j, k)$ 为第 l 层边界种子栅格的拓扑关系函数, $G_l(x, y, z)$ 为约束条件。

利用式(2)实现种子边界栅格的生长。如果 $T_l(x, y, z, i, j, k) = 1$, 则表示该实格沿 (i, j, k) 方向的拓扑关系为空; 如果函数值为 $T_l(x, y, z, i, j, k) = 0$, 则表示实格沿 (i, j, k) 方向的拓扑关系不为空。种子边界栅格的生长情况可按照 $G_l(x, y, z)$ 将其分为两类: 当 $G_l(x, y, z) = 5$ 或 $G_l(x, y, z) = 6$ 时, 表示该种子边界栅格受参考约束拓扑关系的约束, 可沿 (i, j, k) 的拓扑方向进行生长; 其他情况该种子栅格不生长。

当生长完成后, 从每个栅格中提取边界特征点作为种子点, 然后在种子点的邻域内搜索其他边界点并进行优化, 最终获得边界点集(图 5、6 为渲染后的三角网格)。

(a) 人工选择

(b) 采用分层栅格识别技术提取边界点

图 5 门框模型(135 895 个点)

(a) 人工选择

(b) 采用分层栅格识别技术提取的边界点

图 6 汽车零件模型(6 691 个点)

1.2 形态算子

本文采用形态算子修复网格中存在拓扑缺陷的区域。形态学运算主要包括腐蚀、膨胀、开启和闭合 4 种基本运算, 可用于抑制噪声、特征提取、边缘检测等方面^[13]。

可以使用形态学腐蚀运算扩张空洞及细小间断, 膨胀运算弥补空洞。设 V 表示栅格集合, v 表示栅格中的一个数据点, 为了书写方便, 记腐蚀算子 $\varepsilon(v) = \{v \in V, v \text{ 的 } 26 \text{ 邻域都在 } V \text{ 中}\}$, 膨胀算子 $\delta(v) = \{v \in V \text{ 或 } v \text{ 在 } V \text{ 的 } 26 \text{ 邻域中}\}$ 。腐蚀算子描述了从栅格中移除数据点的过程, 而膨胀算子则描述如何将数据点扩展到其所在栅格的邻域中。采用形态学腐蚀和膨胀运算修补网格的方法如下: 设 $S = \cup\{S_l\}$ 表示待修补区域的栅格集合 (S_l 为待修补区域的第 l 层栅格), B 为待修补区域的边界点集合。根据式(1)分析栅格 S_l 的边界拓扑关系与其参考约束拓扑关系: 如果 $g_l(x, y, z) = 5$, 表明该边界栅格中含有局部尖锐特征点, 可判断此处曲面上包含小 Handle, 应使用腐蚀算子删除 B 及 B 的一环邻域, 去除小 Handle, 如图 7a 所示; 如果 $g_l(x, y, z) = 4$, 表明该边界栅格中包含了原始曲面不连续边界边上的点, 可判断此处曲面上含有短的间断或小的缺口及孔洞, 应使用膨胀算子将其连接或填充, 即搜索该栅格的 26 邻域, 并将 B 扩展到其邻域中, 如图 7b 所示。

(a) 腐蚀算子

(b) 膨胀算子

图 7 用形态算子处理数据点

扩展点的坐标可用式(3)计算

$$\begin{aligned}
 v_0 &: (v_x, v_y - w_d, v_z) \\
 v_1 &: (v_x, v_y + w_d, v_z) \\
 v_2 &: (v_x - l_d, v_y + w_d, v_z) \\
 v_3 &: (v_x + l_d, v_y + w_d, v_z) \\
 v_4 &: (v_x, v_y, v_z + h_d) \\
 v_5 &: (v_x, v_y, v_z - h_d) \\
 v_6 &: (v_x, v_y - w_d, v_z + h_d) \\
 v_7 &: (v_x - l_d, v_y - w_d, v_z) \\
 v_8 &: (v_x, v_y - w_d, v_z - h_d) \\
 v_9 &: (v_x + l_d, v_y - w_d, v_z) \\
 v_{10} &: (v_x, v_y + w_d, v_z + h_d) \\
 v_{11} &: (v_x - l_d, v_y + w_d, v_z) \\
 v_{12} &: (v_x, v_y + w_d, v_z - h_d) \\
 v_{13} &: (v_x + l_d, v_y + w_d, v_z) \\
 v_{14} &: (v_x - l_d, v_y, v_z + h_d) \\
 v_{15} &: (v_x - l_d, v_y, v_z - h_d) \\
 v_{16} &: (v_x + l_d, v_y, v_z + h_d) \\
 v_{17} &: (v_x + l_d, v_y, v_z - h_d) \\
 v_{18} &: (v_x - l_d, v_y - w_d, v_z + h_d) \\
 v_{19} &: (v_x - l_d, v_y - w_d, v_z - h_d) \\
 v_{20} &: (v_x + l_d, v_y - w_d, v_z + h_d) \\
 v_{21} &: (v_x + l_d, v_y - w_d, v_z - h_d) \\
 v_{22} &: (v_x - l_d, v_y + w_d, v_z + h_d) \\
 v_{23} &: (v_x - l_d, v_y + w_d, v_z - h_d) \\
 v_{24} &: (v_x + l_d, v_y + w_d, v_z + h_d) \\
 v_{25} &: (v_x + l_d, v_y + w_d, v_z - h_d)
 \end{aligned} \quad (3)$$

式中 $v_0 \sim v_{25}$ —— 26 邻域中的扩展点
 (v_x, v_y, v_z) —— 边界点的坐标

l_d, w_d, h_d —— 栅格的长、宽、高

腐蚀和膨胀运算较为简单, 但容易引起模型变形, 如图 8a、8b 所示。因此, 本文使用开启和闭合算子实现拓扑修改。开启运算是先对网格进行腐蚀, 然后对腐蚀的结果做膨胀运算。闭合运算是开运算的对偶运算, 定义为先对网格做膨胀运算, 然后对膨胀的结果做腐蚀运算。开启运算一般用来平滑网

格边缘,分离细小间断的周围部分,消除尖细的突出物;而闭合运算也可使边缘网格得到平滑效果,但它一般用来连接细小间断的周围部分,或消除小空洞、填补断裂的边缘网格,最终得到闭合区域。图 8c、8d 显示了开运算和闭运算处理的效果图,可以看出开启运算能够有效地消除模型区域之间的细小连接以及区域边缘的毛刺,而闭合运算可以较好地填充区域之间的空洞。

图 8 基本形态算子

图 9 与图 10 显示了用形态学闭合和开启运算修补后的网格。由此可见,开启和闭合运算都可以除去网格模型中不需要的亏格,同时保证模型不会产生全局几何失真。使用开启和闭合算子对三角网格进行修补,可以较好地消除各种拓扑缺陷,从而得到更好的表达效果。

图 9 用闭合算子修补网格(门框模型, 135 895 个点)

图 10 用开启算子修补网格(汽车零件模型, 6 691 个点)

1.3 拓扑简化

大多数算法在采用形态算子修复存在拓扑缺陷的区域后,要将体素集转化为二维流形网格,

即重新对点云进行三角剖分。这种方法的缺点显而易见,本文引入一种基于柄体理论与星形理论的拓扑算子^[8],仅对三角网格进行局部拓扑修改,大大提高了计算效率。这种拓扑理论所使用的运算包括:构建或删除网格,改变网格的结构和分辨率。

1.3.1 柄体算子

柄体理论在欧拉-庞加莱理论的基础上进行了改进,对于 n 维流形引入了几个新的拓扑不变量,其核心是研究流形边界改变时所发生的拓扑变化。

定义 1: 设 K 是单纯复形,单纯形 $\sigma \in K$ 。 σ 的三种局部邻域可用以下元素表示。

(1) σ 的开星形邻域

$$\text{star}(\sigma, K) = \{\tau \in K : \sigma \text{ 是 } \tau \text{ 的一个面}\}$$

(2) σ 的星形邻域

$$\overline{\text{star}}(\sigma, K) = \{\tau \in K : \tau \text{ 是 } \text{star}(\sigma, K) \text{ 中某个元素的一个面}\}$$

(3) σ 的环形邻域

$$\text{link}(\sigma, K) = \{\tau \in K : \tau \text{ 与 } \sigma \text{ 彼此独立且 } \sigma * \tau \in K\}$$

定义 2: 设 D^γ 表示 γ 维圆盘, ∂P 表示集合 P 的边界。对于二维流形,设索引为 λ 的柄体为 H_λ ,它是一对拓扑空间 (A_λ, B_λ) 。这里 $B_\lambda \subset A_\lambda, A_\lambda = D^\lambda \times D^{2-\lambda}$, 且 $B_\lambda = \partial D^\lambda \times D^{2-\lambda}$ 。

定理 1(柄体分解定理): 对于每个可定向曲面 S ,都存在该曲面的一个有限序列 $\{S_\omega\}, \omega=0,1,\dots,N$ ($S_0 = \emptyset, S_N = S$),通过将柄体 $H_\lambda = (A_\lambda, B_\lambda)$ 附加在曲面 $S_{\omega-1}$ 的边界上得到曲面 S_ω 。这个序列称作曲面 S 的柄体分解。

柄体分解定理是柄体理论的基石。曲面在附加柄体的过程中,为了保持其方向性而引起的拓扑变化,可以利用欧拉公式计算连通分支和边界曲线的数量。图 11 显示了用构建柄体算子链接边界边的三种情况。分解柄体算子是其逆运算。

图 11 用构建柄体算子链接边界

1.3.2 星形算子

星形理论用于定义三角网格中的某些变化,但不改变网格曲面的拓扑。该理论包括双星形移动和

星形细分运算。

定义 3: 设 K 是一个 n 维单纯复形, σ 是 r 维单纯形($\sigma \in K$), τ 是 $(n-r)$ 单纯形($\tau \notin K$), 且有 $\text{link}(\sigma, K) = \partial\tau$, 则称运算 $K(\sigma, \tau)$ 为双星形移动, 它通过删除 $\sigma * \partial\tau$ 和插入 $\partial\sigma * \tau$ 来改变 K 。

定义 4: 设 K 是一个二维单纯复形, σ 是 r 单纯形($\sigma \in K$), v 是 σ 中的一个顶点, 定义运算 (σ, v) 为删除一个 $\overline{\text{star}}(\sigma, K)$, 并用 $v * \partial\sigma * \text{link}(\sigma, K)$ 表示删除后的结果, 运算 (σ, v) 称作星形细分, 其逆运算 $(\sigma, K)^{-1}$ 称为星形链接。

双星形移动为原子运算, 它只对单纯形的邻域做局部改变, 同时保持其组合结构的完整性; 星形细分可以改变网格的连通性和分辨率。图 12 显示了边的 1 阶星形细分。

图 12 星形细分

1.3.3 拓扑算子的应用

柄体算子和星形算子对组合流形的基本变换有很大帮助, 因为它为运算法则提供了准确的抽象层次, 可以保证在任何情况下计算过程总是合法的。此外, 使用柄体算子和星形算子分裂边和链接顶点的理论, 可以保证在网格分解过程中, 每一层网格只有局部变化而不改变整个网格, 因此采用这套拓扑算子能够更好地实现自适应分辨率下的三角网格修补。

构建柄体算子可创建新的网格或将两条边界边链接成一条内部边, 分解柄体算子则用来删除网格或将一条内部边分裂为两条边界边。这两种柄体算子包括以下两种原子运算: ① 构建网格、删除网格。 $f = \text{create}(v_0, v_1, v_2)$, 生成一个新的三角面片 f ; $\text{destroy}(f)$, 删除面 f 定义的连通分支; ② 改变网格拓扑。 $\text{glue}(e_0, e_1)$ 用于识别两条边界边 e_0, e_1 , 将其连成一条内部边; $\text{unglue}(e)$ 可将一条内部边分裂为两条边界边。

星形算子用来改变网格的分辨率和结构, 它所包括的原子运算: ① 修改网格的连通性, $\text{flip}(e)$ 通过交换公共边 e , 改变由两个相邻三角形形成的网格区域的结构; ② 修改网格的分辨率, $\text{split}(e)$ 可将边 e 和其入射面二等分; $\text{split}(f)$ 将面 f 三等分; $\text{weld}(v)$ 是 split 算子的逆算子。

在实际应用中, 可以根据需要定义几个推导出

的算子, 这些算子由一组柄体算子和星形算子构成, 将其封装起来, 就成为高级运算。如 $\text{attach}(e, v)$ 表示在边界增加一个新的三角形, 该三角形包括原来的边 e 及一个新顶点 v ; $\text{detach}(f)$ 表示 attach 算子的逆算子, 从网格中删除一个三角形 f (可以是内部三角形, 也可以是边界三角形); $\text{remove}(v)$ 表示删除任意一顶点 v , 从而降低网格的分辨率。

图 13 显示了该拓扑算子集合的具体应用, 图 13a 是车床零件的点云模型(共 27 913 个点), 图 13b、13c 是其自适应三角剖分结果, 图 13d、13e 是用形态算子修补的网格, 图 13f、13g 是用该拓扑算子封闭边界。试验数据见表 1。由此可见, 在网格修补过程中如果仅采用形态算子, 容易产生大量狭长三角形, 从而导致在三角参数曲面片插值时引起较大的误差, 降低模型的逼真度, 产生急剧的视觉变化, 最终影响模型的显示效果。引入柄体算子和星形算子后, 不仅能够限制狭长三角形的生成, 很好地保持模型的几何特征, 提高逼近精度, 而且计算效率也得到很大提高。

图 13 拓扑算子的应用(车床零件模型)

表 1 车床零件模型的试验数据

算子	运行时间 t/s	生成三角形数 $n/\text{个}$	CPU 使用率 $u(\%)$
仅用形态算子	2.01	165 582	51
添加拓扑算子	0.39	2 383	20

2 试验及结果分析

本文所使用的计算机环境: Intel P4, CPU 2.60 GHz, 内存 2 GB。算法已用 Visual Studio.NET 2003

及 OpenGL 编程实现。

图 14a 为含噪声的佛陀点云模型(共 49 990 个点), 图 14b 为三角剖分结果(含有不需要的亏格), 图 14c 为骨架算子^[7]的修补结果, 图 14d 为 Geomagic Studio 软件的修补结果, 图 14e 为本文算法的修补结果。图 15 为佛陀模型放大的局部网格, 图 15a 为修补前网格(含有不需要的亏格), 图 15b 为骨架算子^[7]的修补结果, 图 15c 为 Geomagic Studio 软件的修补结果, 图 15d 为本文算法的修补结果。图 16a 为含噪声的汽车油箱点云模型(共 174 458 个点), 图 16b 为三角剖分结果(有空洞), 图 16c 为骨架算子^[8]的修补结果, 图 16d 为 Geomagic Studio 软件的修补结果, 图 16e 为本文算法的修补结果。图 17 为放大的局部网格, 图 17a 为修补前网格(有空洞), 图 17b 为骨架算子^[8]的修补结果, 图 17c 为 Geomagic Studio 软件的修补结果, 图 17d 为本文算法的修补结果。试验数据见表 2、3。

图 14 佛陀模型的拓扑修补

图 15 局部网格放大(佛陀模型)

图 16 汽车油箱模型的拓扑修补

图 17 局部网格放大(汽车油箱模型)

表 2 佛陀模型的试验数据

算子	运行时间 t/s	生成三角形数 $n/\text{个}$	CPU 使用率 $u(\%)$
骨架算子	18.01	247 870	51
Geomagic Studio	2.21	—	50
本文算法	0.25	529	20

表 3 汽车油箱模型的试验数据

算子	运行时间 t/s	生成三角形数 $n/\text{个}$	CPU 使用率 $u(\%)$
骨架算子	24.59	1 026 492	51
Geomagic Studio	2.94	—	51
本文算法	0.34	374	20

由图 14、15 及表 2、3 可以看出, 采用本文算法对三角网格进行修补, 较好地消除了一些网格上的拓扑缺陷, 有效地提高了三角网格显示的精度。与传统方法和当前流行的应用软件相比, 本文方法不需要重构整个三角网格曲面, 简化了数据处理的过程, 从而降低了计算代价及内存使用率, 且有更

好的表达效果, 符合工程实际需求, 达到了预期的效果。

3 结论

(1) 通过对网格进行综合拓扑处理, 很好地去除了三角网格的拓扑缺陷, 显示了数学形态学和拓扑理论在图形学建模领域中的较好应用。

(2) 由于引入了基于分层栅格的缺陷识别技术, 曲面的拓扑可以由用户控制, 既保证了提取出的误差区域精度高, 又较好地解决了传统方法存在的不足, 较之传统方法有着更为广阔的应用前景。

参 考 文 献

- [1] NOORUDDIN F S, TURK G. Simplification and repair of polygonal models using volumetric techniques[J]. IEEE Transactions on Visualization and Computer Graphics, 2003, 9(2): 191-205.
- [2] HETROY F, REY S, CARLOS A, et al. Mesh repair with user-friendly topology control[J]. Computer-Aided Design, 2011, 43(1): 101-113.
- [3] BISCHOFF S, KOBELT L P. Isosurface reconstruction with topology control[C]// Proceedings of the 10th Pacific Conference on Computer Graphics and Applications (PG'02), October 9-11, 2002, Tsinghua University, Beijing. 2002: 246-255.
- [4] ZHOU Qianyi, JU Tao, HU Shimin. Topology repair of solid models using skeletons[J]. IEEE Transactions on Visualization and Computer Graphics, 2007, 13(4): 675-685.
- [5] LOMENIE N, STAMON G. Morphological mesh filtering and a-objects[J]. Pattern Recognition Letters, 2008, 29(10): 1571-1579.
- [6] YANG Zhouwang, SEO Y H, KIM T W. Adaptive triangular-mesh reconstruction by mean-curvature-based refinement from point clouds using a moving parabolic approximation[J]. Computer-Aided Design, 2010, 42(1): 2-17.
- [7] LEWINER T, LOPES H, MEDEIROS E, et al. Topological mesh operators[J]. Computer Aided Geometric Design, 2010, 27(1): 1-22.
- [8] ZHANG Lei, LIU Ligang, GOTSMAN C, et al. Mesh reconstruction by meshless denoising and parameterization [J]. Computers & Graphics, 2010, 34(3): 198-208.
- [9] 吕哲, 王福利, 常玉清, 等. 改进的形态学骨架提取算法[J]. 计算机工程, 2009, 35(19): 23-25.
LÜ Zhe, WANG FuLi, CHANG Yuqing, et al. Improved morphological skeleton extraction algorithm[J]. Computer Engineering, 2009, 35(19): 23-25.
- [10] 王金玲, 段会川, 刘弘. 基于轮廓线度量的形态学骨架剪枝方法[J]. 计算机工程与设计, 2009, 30(9): 2283-2285.
WANG Jinling, DUAN Huichuan, LIU Hong. Skeleton pruning approach for morphological skeletons based on contour metric[J]. Computer Engineering and Design, 2009, 30(9): 2283-2285.
- [11] 滕书娟. 一种保持视觉特征的 LOD 模型简化算法[J]. 计算机工程与应用, 2010, 46(33): 164-168.
TENG Shujuan. LOD model simplification method based on preserved vision feature[J]. Computer Engineering and Applications, 2010, 46(33): 164-168.
- [12] 柯映林, 范树迁. 基于点云的边界特征直接提取技术[J]. 机械工程学报, 2004, 40(9): 116-120.
KE Yinglin, FAN Shuqian. Research on direct extraction of boundary from point clouds[J]. Chinese Journal of Mechanical Engineering, 2004, 40(9): 116-120.
- [13] 胡艳, 张瑞林. 基于小波模极大值和形态学的图像边缘检测算法[J]. 计算机应用与软件, 2010, 27(9): 75-76.
HU Yan, ZHANG Ruilin. Edge detection approach for images based on wavelet modulus maximum and morphology[J]. Computer Applications and Software, 2010, 27(9): 75-76.

作者简介: 蒋恒恒, 女, 1973 年出生, 博士研究生。主要研究方向为计算机辅助设计和计算机动画。

E-mail: jhh@cqu.edu.cn

李奇敏(通信作者), 男, 1976 年出生, 博士, 副教授。主要研究方向为反求工程、曲线、曲面造型技术和小波分析。

E-mail: qim_li@163.com

汤宝平, 男, 1971 年出生, 博士, 教授, 博士研究生导师。主要研究方向为测试计量技术与虚拟仪器。

E-mail: bptang@cqu.edu.cn