

文章编号 1004-924X(2019)04-0971-13

三角网格模型的特征保持混合折叠简化

曹增欢, 黄常标*, 郑 红

(华侨大学 福建省特种能场制造重点实验室, 厦门市数字化视觉测量重点实验室, 福建 厦门 361021)

摘要:在增材制造、逆向工程等领域,广泛存在包含大量甚至海量数据的三角网格模型。为便于存储并提高处理效率,经常需要进行网格模型简化。但在网格简化过程中存在特征保持、简化率和简化效率冲突的问题,为更好地平衡简化结果和简化效率,提出了特征保持的混合折叠算法。在基于曲度精确计算新顶点、最大距离高效计算折叠代价的基础上,对边界特征区域和非边界特征区域采用边折叠方式进行保特征简化,对非特征区域则采用三角形折叠方法进行高效简化,最后通过偏差和网格正则度对简化结果作出误差评价。算法实例表明:混合折叠算法的模型细节特征保持较好,简化前后变形较小且效率适中。

关键词:三角网格模型;网格简化;混合折叠;细节特征保持

中图分类号:TP391.7 **文献标识码:**A **doi:**10.3788/OPE.20192704.0971

Feature preserving mesh simplification based on hybrid collapse

CAO Zeng-huan, HUANG Chang-biao*, ZHENG hong

(Fujian Key Laboratory of Special Energy Manufacturing,
Xiamen Key Laboratory of Digital Vision Measurement,
Huaqiao University, Xiamen 361021, China)

* Corresponding author, E-mail: huangcb@hqu.edu.cn

Abstract: In additive manufacturing, reverse engineering, and other fields, many triangular mesh models were used to process big data. To facilitate storage and improve processing efficiency, simplifying the mesh model was necessary. However, conflicts existed among feature preservation, simplification rate, and simplification efficiency in mesh simplification. For improved balancing of results and efficiency of simplification, a method based on curvedness was proposed to compute new vertices more precisely. In addition, a method based on the max distance was proposed to compute collapse cost efficiently. The edge collapse was adopted for preserving the features of boundary and non-boundary feature regions, and the triangle collapse was applied to simplify the non-feature areas efficiently. The mesh regularity of the simplified model and the deviation between the original and simplified models were used as evaluation criteria for mesh simplification. Extensive experiments demonstrate that the mesh simplification algorithm with hybrid collapse can effectively preserve the features of a complicated mesh model with fewer errors and moderate efficiency.

收稿日期:2018-08-06; **修订日期:**2018-10-12.

基金项目:国家科技支撑计划资助项目(No. 2015BAF24B00);福建省科技重大专项资助项目(No. 2014HZ0004-3);福建省引导性项目资助(No. 2018H0020, No. 2017H0019, No. 2016H0020)

Key words: triangle mesh; mesh simplification; hybrid collapse; feature-preserving

1 引言

增材制造^[1]中所需制造的 STL (Stereolithography) 模型通常是由几万、几十万、甚至成百上千万个三角面片组成,在医学^[2]、逆向工程^[3]等领域存在包含大量甚至海量数据的复杂三角网格模型。为便于计算机存储和分析,通常需要进行简化处理,因此如何有效简化复杂三角网格模型也成为数字化制造领域内研究的热点问题之一。

三角网格模型简化的本质是减少三角面片和网格顶点的数目。目前常用的几何元素折叠法依据原始模型进行相应地简化,故能较好地保留原模型的拓扑信息,而被广泛研究。几何元素折叠法是指在与原模型有一定几何误差的前提下,折叠对原始模型影响较弱的几何元素,主要有边折叠算法^[2,4,8]和三角形折叠算法^[9,14]两种。其中,三角形折叠一次可以简化掉 4 个三角面片,简化效率较高,在曲率变化不大处折叠效果较好,但特征保持性较差;而边折叠算法对于细节特征保持性较好,但效率较低。也有将网格模型拟合为曲面模型的方法^[15],虽然能够极大地减少顶点数量,但其结果已不再是三角网格模型。

几何元素折叠方法需要解决折叠对象的确定和新顶点的选择这两个关键问题。现有算法通过计算网格三角形折叠代价确定折叠对象,折叠代价计算主要有二次误差测度法^[4,6,8,12] (Quadric Error Metric, QEM)、特征保持法^[10,14] 以及加权平均法^[2,11,13] 等方法。其中二次误差测度法原理简单,但是需要通过矩阵来求解折叠代价,在不通过加速手段进行简化处理的前提下效率较低;特征保持法对于细节特征保持较好,但对于复杂模型往往得不到很好的处理结果;加权平均法思路简单、计算较快,但对特征复杂模型无法取得较好的效果,且当简化程度增大时细节特征丢失严重,模型变形较大。

对于新顶点的选择,现有算法可以分为子集选择法^[2,4]和优化选择法^[5,8,15]。子集选择法即选择三角面片中的顶点或边上的点,该方法计算速

度较快,但简化模型与原模型相差较大;优化选择法是求取空间中的一点使得折叠代价最小,该方法简化效果虽然较好,但是计算速度较慢。

为更好地平衡简化结果和简化效率,本文提出一种特征保持的混合折叠网格简化算法。通过提出基于曲度^[16]的优化选择法来生成折叠后的新顶点,以减少简化误差;将传统 QEM 方法改进为最大距离法来确定折叠对象,可更加高效地确定是否折叠面片;简化过程中,先对边界三角形进行边折叠,对曲率变化较大的特征区域采用边折叠方式,对于非边界中曲率变化较小的非特征区域采用三角形折叠方式,以获得更好的简化效果。

2 算法流程

针对复杂三角网格模型简化的特征保持要求,本文算法以反映顶点局部几何信息的曲度作为权值计算折叠后的新顶点,以最大距离作为折叠代价计算方法控制面片简化的顺序;简化过程中,先对边界区域进行边迭代简化操作,然后通过二面角法识别模型的非边界特征区域,对识别的特征区域进行边折叠操作,对非特征区域进行三角形折叠操作完成模型的简化。算法流程如图 1 所示,其步骤如下:

Step 1. 折叠后新顶点计算。引入反映顶点局部几何信息的曲度值,进行新顶点权值的计算,以减小模型简化过程中发生的变形;

Step 2. 折叠代价的计算。通过改进的 QEM 方法(点面距离),计算各待折叠对象的折叠代价,通过控制边或面片的简化顺序来控制模型简化前后的变形程度,以保持重要细节特征;

Step 3. 边界特征简化。基于半边数据结构识别出模型的边界特征区域,在保持模型边界特征的前提下进行边折叠简化;

Step 4. 特征区域边折叠处理。对识别出的特征区域进行边折叠处理,在减少模型面片的同时保持重要细节特征;

Step 5. 非特征区域三角形折叠处理。对于非特征区域进行三角形折叠简化。

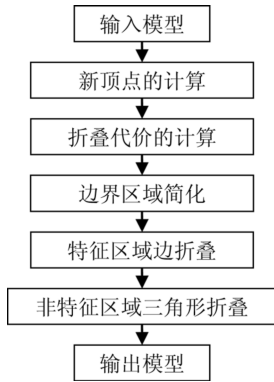


图 1 算法流程图

Fig. 1 Flow chart of proposed algorithm

3 新顶点的计算

为提高简化效率以及获得更小的简化误差,提出一种基于曲度的新点优化选择方法,并对不同简化对象采取不同的新顶点计算方法。

在曲面的微分几何信息中,高斯曲率和平均曲率是能够反映曲面信息的几何不变量,可表示点所在曲面的局部弯曲程度,但存在高斯曲率值对圆柱面、平均曲率对鞍形点不敏感的情况。而曲度^[16]则能较全面地反映模型表面的几何信息。曲度是描述曲面上某一点微分几何信息的重要特征,可反映三维曲面在局部的凹凸或弯曲程度。曲度计算式如下:

$$C_p = 4K_H^2 - 2K_G, \quad (1)$$

其中: K_H 为顶点平均曲率, K_G 为顶点高斯曲率。 K_H 和 K_G 通过估算法^[17]求得。

3.1 特征区域简化新顶点的计算

三角网格模型上存在法矢或曲率变化较大的特征区域和边界特征区域。这些区域含有丰富的细节特征,因此采用边折叠简化方法保留特征。边折叠新顶点 v_0 的计算公式如下:

$$v_0 = \frac{C_{p1}}{C_{p1} + C_{p2}} v_1 + \frac{C_{p2}}{C_{p1} + C_{p2}} v_2, \quad (2)$$

其中: v_1 和 v_2 为特征边的起点和终点, C_{p1} 和 C_{p2} 为起点和终点的曲度值。

3.2 非特征区域简化新顶点的计算

对非特征区域三角形则采取三角形折叠方式,折叠后新顶点 v_0 的计算公式如式(3)所示:

$$v_0 = \frac{C_{p1}}{C_{p1} + C_{p2} + C_{p3}} v_1 + \frac{C_{p2}}{C_{p1} + C_{p2} + C_{p3}} v_2 + \frac{C_{p3}}{C_{p1} + C_{p2} + C_{p3}} v_3, \quad (3)$$

其中: v_1, v_2 和 v_3 为待折叠三角形的顶点, C_{p1}, C_{p2} 和 C_{p3} 为顶点曲度值。

表 1 列出了 3 种新点生成方法的效果比较。其中, R 为简化率, N_T 为简化后面片数, d_{max} 为最大正偏差, d_{min} 为最大负偏差, d_{SD} 为标准偏差, d_{RMS} 为均方差。从表 1 可以看出,随着简化率的增大,曲度优选法较优于其他两种方法,简化前后模型变形较小,且算法效率适中。

表 1 不同新点计算方法的 Bunny 模型简化统计数据

Tab. 1 Statistical data of Bunny simplification by different methods of new vertex computation

方案	R/%	N_T	T/s	d_{max}/mm	d_{min}/mm	d_{SD}/mm	d_{RMS}/mm
三点选一法	60	24 658	20.470	0.851 1	-8.002 6	0.063 2	0.064 2
	90	6 726	28.018	1.760 7	-8.928 0	0.202 2	0.221 6
重心优选法	60	25 106	8.740	0.801 9	-9.316 2	0.081 1	0.084 3
	90	6 790	24.180	1.323 2	-8.881 9	0.284 3	0.332 5
曲度优选法	60	24 418	22.156	0.801 7	-0.837 1	0.059 6	0.062 4
	90	6 310	32.230	1.344 1	-1.252 1	0.249 1	0.286 5

4 折叠代价的计算

为控制简化精度,须通过折叠代价控制简化顺序。虽然 QEM 算法折叠代价原理简单、易于实现,但是需要求解矩阵,效率较低。为此,提出基于最大距离的折叠代价计算方法,即通过计算

点到折叠对象一阶邻域三角面片所在平面的最大距离,以此作为折叠对象的折叠代价,本文中的折叠对象为边或三角面片,并用符号 Ob 表示。

已知折叠后的新顶点 v_0 , T 为 v_0 所对应折叠对象 Ob 的一阶邻域三角面片 T_s 中的某个面片, v' 为 T 中任意一个顶点, u 为 T 的单位法向量。则点 v_0 到 T 所在平面的距离为向量 $R = (v_0 - v')$

在 u 方向的长度,其计算公式如式(4):

$$\| \mathbf{R} \| = | (u_0 - v') \cdot \mathbf{u} |. \quad (4)$$

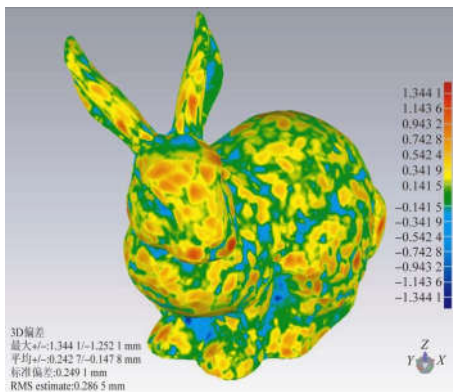
则折叠对象 Ob 的折叠代价如下:

$$\Delta(Ob) = \max_{T_s} (\| \mathbf{R} \|). \quad (5)$$

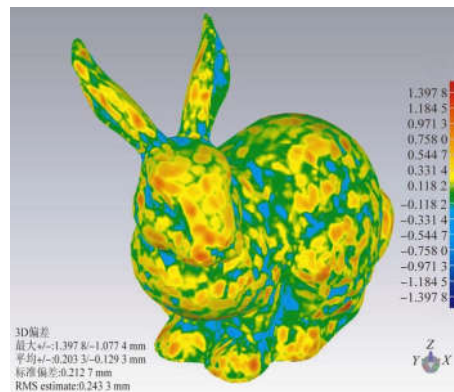
采用式(2)、式(3)计算新顶点后,将最大距离法与 QEM 法对 Bunny 模型、Hand 模型、Horse 模型简化 90%后的结果进行对比分析。图 2 为不同模型简化后的误差云图,云图数值为偏差值,标准

偏差和均方差值越小,说明简化前后模型变形较小。图 2 中红色区域代表正向变形,蓝色区域代表负向变形、绿色区域为未变形(彩图见期刊电子版)。

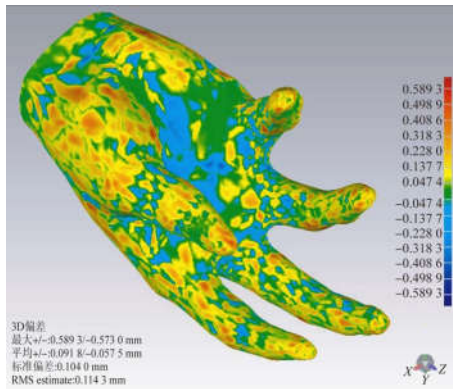
表 2 给出了将模型简化掉 90%后各模型所需的简化时间、偏差分析结果以及简化所得模型正则率情况(正则率计算详见 6.2 小节)。可见,在误差基本相同的情况下,最大距离法的计算效率要高不少。



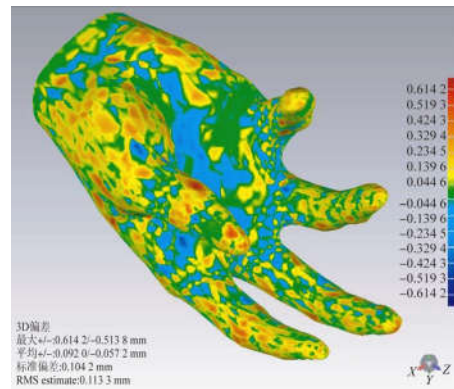
(a)最大距离法
(a) Method of the max-distance



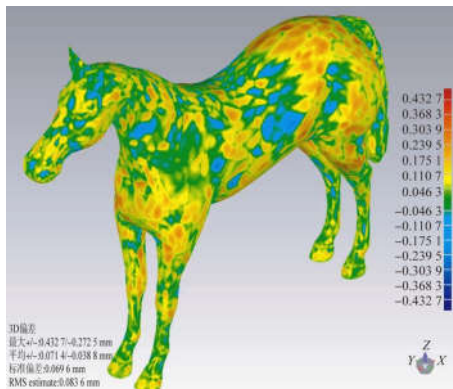
(b)QEM 方法
(b) Method of QEM



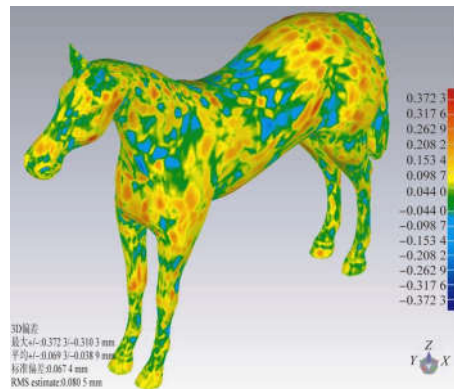
(c)最大距离法
(c) Method of the max-distance



(d)QEM 方法
(d) Method of QEM



(e)最大距离法
(e) Method of the max-distance



(f)QEM 方法
(f) Method of QEM

图 2 模型简化云图对比

Fig. 2 Comparison of simplification results for cloud picture

表 2 不同折叠代价的模型简化统计数据

Tab. 2 Statistical data of model simplification with different collapse costs

模型	方案	N_T	正则率/%	时间/s	d_{max}/mm	d_{min}/mm	d_{SD}/mm	d_{RMS}/mm
Bunny	最大距离法	6 310	83.98	35.55 8	1.344 1	-1.252 1	0.249 1	0.286 5
	QEM 法	6 516	85.49	43.385	1.397 8	-1.077 4	0.212 7	0.243 3
Hand	最大距离法	8 906	86.37	37.536	0.589 3	-0.573 0	0.091 8	-0.057 5
	QEM 法	8 388	87.83	50.763	0.614 2	-0.513 8	0.092 0	-0.057 2
Horse	最大距离法	14 630	60.88	64.430	0.432 7	-0.272 6	0.071 4	-0.038 8
	QEM 法	13 488	62.37	91.377	0.372 3	-0.310 3	0.069 3	-0.038 9

5 模型简化处理

对模型采用混合折叠网格简化算法,具体包括孔洞边界的边折叠、特征区域的边折叠以及非特征区域的三角形折叠,在保持边界和特征的前提下,提高简化效率。

5.1 边界特征区域简化

在原始网格中往往存在边界特征(如图 3),为保证孔洞边界特征不被过度简化,可先对边界区域进行识别,然后将边界区域进行边折叠处理,通过简化精度控制简化程度可保持原始模型中的特征,并提高算法效率。

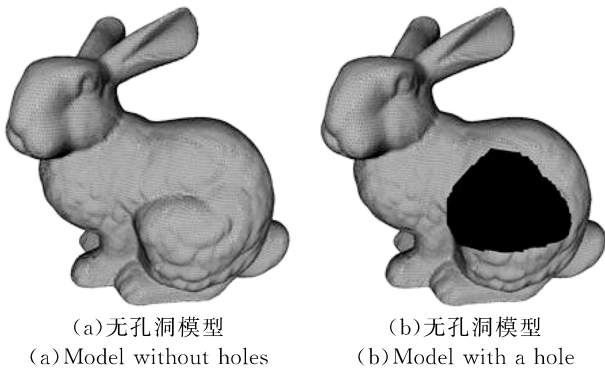


图 3 带孔洞特征的 Bunny 模型

Fig. 3 Bunny model with feature of hole

在对模型进行简化处理时,可能存在一个或多个边界区域,为使边界信息不被丢失,提出边界区域折叠算法,流程如下:

Step 1. 识别模型中的所有边界边,对查找出的各边界边一阶邻域面片进行标记;将各边界边按邻接顺序存储到边的 vector 容器 BordRing

中,并将各边的终点按顺序存储到顶点的 vector 容器 m_vecPnts 中;

Step 2. 识别边界边 E_i 中的尖锐特征边。识别尖锐特征边时遵循两个原则,一个是计算边的两特征角,另一个是通过式(5)计算边的折叠代价。如图 4 所示,由公式(6)计算边 E_1 的两特征角 θ_1 和 θ_2 ,若两特征角均小于阈值,转入 Step 3;若特征角和折叠代价都不满足给定的阈值则跳转到步骤 Step 4。

$$\theta = \arccos(v_i v_{i-1}, v_{i+1} v_i). \quad (6)$$

Step 3. 标记该边界边,取下一条边界边继续 Step 2;

Step 4. 由公式(2)计算折叠后的新顶点 v_i ,从边界边表中删除待折叠 E_i ,用新顶点 v_i ,更新与 E_i 起点和终点相关的边界边及 E_i 一阶邻域中的边、三角面片信息;

Step 5. 若边界边容器 BordRing 中无可简化边,则边界处理结束,否则跳转到 Step 2 继续简化。

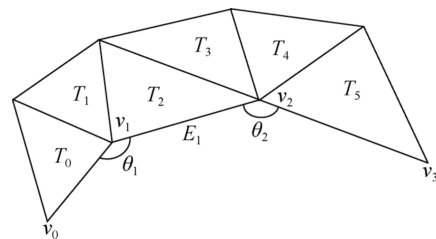


图 4 边界边特征角示意图

Fig. 4 Illustration of feature angle of boundary edges

5.2 内部区域简化

5.2.1 内部特征区域识别

原始三角网格上往往存在较多的细节特征,在简化过程中必须保留。根据基于边分块的

想,尖锐特征边处的两共边三角面片法矢夹角较大。当两个三角面片法矢夹角大于给定的阈值时,则可认为两个三角面片共享的边为尖锐边^[18],定义 E_{ij} 为面片 T_i 和面片 T_j 的邻接边, $\varphi(i, j)$ 为二面角值, \mathbf{n}_i 和 \mathbf{n}_j 为三角面片的单位法矢,则边 E_{ij} 的二面角(Dihedral Angle)值计算如下:

$$\varphi(E_{ij}) = \arccos(\mathbf{n}_i \cdot \mathbf{n}_j). \quad (7)$$

二面角法避开了复杂的曲率计算,如文献^[19],并且能够高效准确的识别特征边。二面角值大于给定阈值的边是文中所识别出的特征边。通过实验发现:阈值取 45° 时,特征识别效果较好(如图 5 所示)。

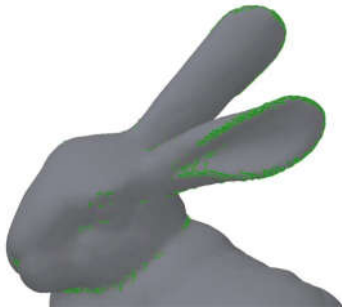


图 5 Bunny 模型头部特征边识别结果

Fig. 5 Illustration of recognition result of feature edges in the head of Bunny

5.2.2 内部特征区域简化

边折叠操作是遍历所有特征边,删除或者折叠与可折叠特征边相关联的顶点、边和面,从而得到较稀疏的特征网格。为防止特征区域被过度简化,可设定边折叠迭代次数,在经过设定次数迭代简化后将简化后的区域进行冻结,不做后续的三角形简化。边折叠简化流程如下:

Step 1. 识别边界处理后的三角网格模型中的特征边,将其添加到特征边容器中;

Step 2. 遍历特征边容器,取出边 E_i , 搜索其一阶邻域面片看是否有“已使用”或“已冻结”标志,若有则跳转到 Step 6, 否则继续;

Step 3. 按照公式(2)计算新顶点 v_0 , 遍历特征边的一阶邻域面片,按公式(5)计算所有特征边折叠代价;

Step 4. 将所有特征边按折叠代价大小排序,取出边表中折叠代价较小的边 E_i , 用新顶点 v_0 替代面片、边中相应的顶点,更新顶点链表,删除特征边 E_i ;

Step 5. 若遍历结束,则清空特征边表并结束

特征区域边折叠操作,否则,继续下一步骤;

Step 6. 从特征边容器中取下一条边,跳转到 Step 2。

5.2.3 内部非特征区域简化

三角形折叠操作是将满足一定条件的三角形,在按“折叠代价”排序后,将三角面片序号小于预设序列长度值的三角形作为待折叠三角形。用新顶点代替待折叠三角形,连接新顶点和面片“一阶邻域环”中的顶点并删除相应的面片和顶点即完成三角形折叠操作。

Step 1. 遍历所有三角面片,按照公式(3)求取折叠后的新顶点,并通过公式(5)求取所有未被标记的三角面片的折叠误差,将折叠误差和新顶点存储在三角形的数据结构中;

Step 2. 将所有未被标记三角面片按折叠代价进行升序排列,得到一个折叠误差序列;

Step 3. 以 N 表示原始模型中三角面片的数量, r 表示单次简化率,当折叠误差序列小于 $N(1-r)$ 时,可对序列中折叠误差最小的三角形 T_0 进行三角形折叠操作,如图 6 所示;

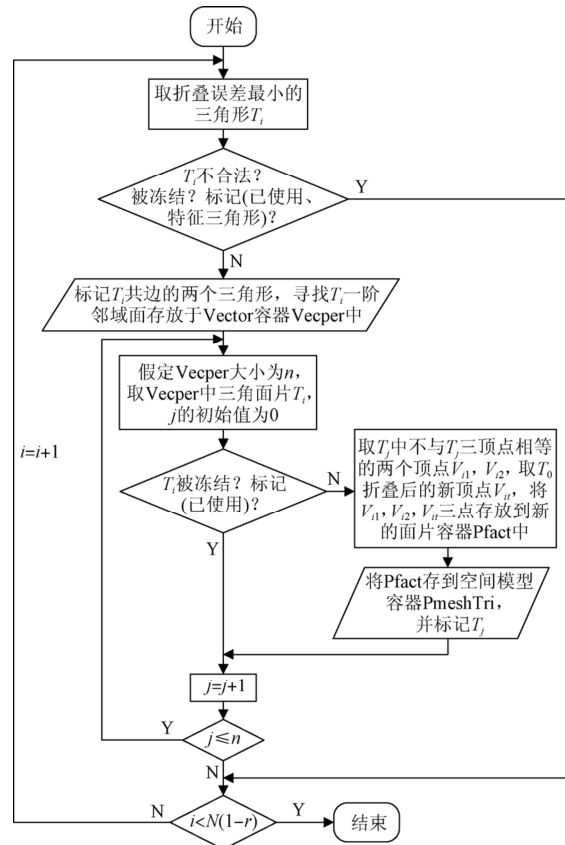


图 6 算法流程图

Fig. 6 Flow chart of algorithm

Step 4. 遍历原始模型所有三角面片,将未经标记的三角面片存入新的模型数据结构 PMeshTri 中。若遍历完成,则停止简化操作;

Step 5. 判断是否达到简化要求,若否,则以新得到的模型 PMeshTri 为新的原始模型继续 Step 1~Step 4 的简化操作;否则,输出简化后的网格模型并进行显示。

6 模型简化处理

6.1 折叠合法性检查

本文所处理的模型包括孔洞模型和有完整拓扑、流形的模型,对于具有非流形的模型(孔洞模型除外)不做处理。流形定义为:一条边有且仅有 2 个面片共用。如图 7 所示,对一个三角形 T_0 进行折叠可能出现以下情况:

(1) 三角形 T_0 及其一阶邻域内三角形可能会被重复多次折叠。

(2) 折叠后三角形边可能由两个以上的三角形共用。

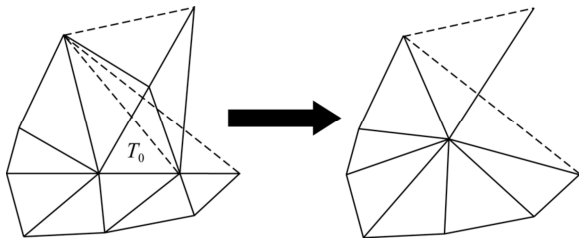


图 7 非流形三角形折叠

Fig. 7 Collapse of triangles with non-manifold edges

对于情况(1),检查检查 T_0 及其一阶邻域内三角形是否被标记,若是,则不对 T_0 进行折叠操作。对于情况(2),比较与此 T_0 相关联的三角面片的法向量在折叠前后的变化情况,如果法向量变化超过 45° ,则不对 T_0 进行折叠操作。

6.2 网格质量评价标准

对复杂三角网格模型进行简化操作首先要有一个较好的网格评判标准,而网格质量主要体现在三角面片单元的正则度上,正则度就是指三角形趋近于正三角形的程度。判定网格的质量,计算方法如式(8):

$$r=2(\cos \alpha+\cos \beta+\cos \gamma-1), \quad (8)$$

其中: α, β 和 γ 为三角形的 3 个内角, r 描述了三角形的质量, $0 < r \leq 1$, r 值越大,三角形的质量越好;三角形为正三角形时, $r=1$;当三角形为狭长三角形时, $r \approx 0$ 。

为度量简化前后三角网格的平均质量,本文使用公式(9)来判定狭长三角形:

$$r'=4 \sqrt{3} A / (l_1^2+l_2^2+l_3^2), \quad (9)$$

其中: A 表示三角形的面积; l_1, l_2, l_3 为三角形的边长;对于等边三角形, $r'=1$;对于狭长三角形, $r' \approx 0$ 。

本文三角网格正则率定义为:简化后三角形中小于 r' 三角形的个数与简化后三角形总数的比值。

7 实验验证与分析

实验验证在 windows 平台 VC++ 2013 环境下, Intel (R) Core (TM) i3-2240 3.40 GHz CPU, 4 GB RAM 的 PC 机结合 OpenGL 编程实现所提及的算法及相关对比算法,并结合普林斯顿形状基准 Benchmark (<http://segeval.cs.princeton.edu/>) 的多个模型进行实验验证。

7.1 边界区域简化实例分析

现有网格简化算法在对边界处理时可分为两类:一类是对边界直接冻结不做处理,另一类是直接对边界进行简化操作。为验证算法的简化效果,对包含 33 233 个顶点、66 302 个面片的 Bunny 孔洞模型进行实验,并编程实现了对边界直接冻结^[12]和直接对边界折叠^[11]。

为验证本文的边界处理算法简化效果,图 8 给出了云图分析结果(彩图见期刊电子版)。图 8 (b)云图显示,文献[12]总体效果较好;图 8 (c)中,标准偏差和平均偏差数值较大,因此网格变形较大不适合采用;图 8 (d)中边界周边都是绿色的,代表边界变形较小,并且整体网格变形较小。

分析可知:文献[12]算法未对模型边界进行简化处理;文献[11]算法得到的结果边界信息完全丢失了而且还出现了新的边界;本文算法的边界保持性较好,网格较为均匀。因此,本文所提算法具有较好的边界保持性。

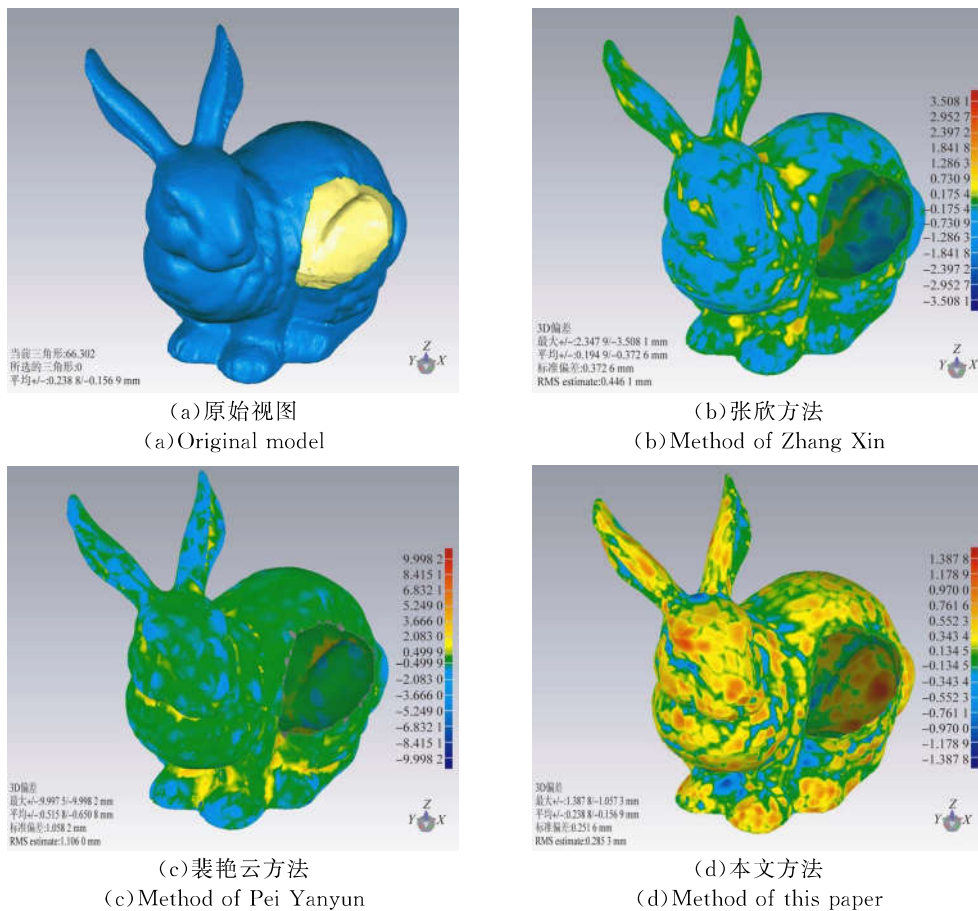


图 8 边界简化结果云图对比

Fig. 8 Comparison of cloud picture of simplification results

7.2 简化率分析比较

简化率是指模型被简化的面片数与原网格模型面片数的比值。该值越大,简化程度越高,越有利于后续对模型进行操作,但是随着简化程度增大,模型的特征丢失将愈加严重,因此采取合适的策略控制简化率是必要的。

边折叠算法和三角形折叠算法都是重要的网格简化方法,对于简化率的控制有着明显的效果。为了选择较好的简化策略,分别以边折叠算法^[8]、三角形折叠算法^[11]以及本文提出的混合折叠算法对包含 34 834 个顶点、69 664 个三角面片和丰富细节特征的 Bunny 模型进行简化实验,表 3 为 Bunny 模型实验结果数值,图 9 为简化结果云图,云图中的颜色是根据当前最大、最小误差给出的相对误差大小,不同图衡量值不一样(彩图见期刊电子版)。表 3 的简化率值中,文献[8]的 73%为

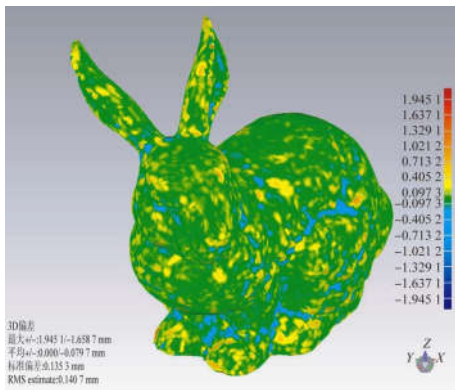
程序迭代简化后获得的最高值,其余值为设定值。

分析表 3 中的数据 and 图 9 的简化结果可得:边折叠算法的特征保持性较好但是简化效率较低,而三角形折叠算法简化效率相对较高但是简化前后模型变形较大,特征保持性较差,混合折叠算法相对来说是这两种算法中较优的简化,可以获得较好的简化结果。

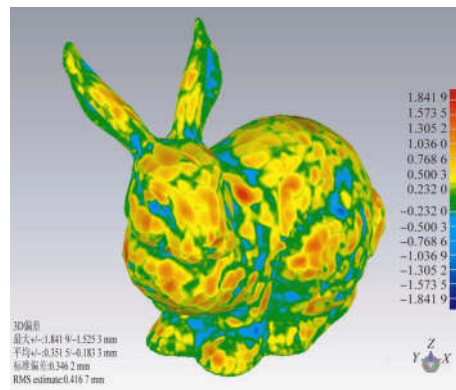
表 3 中文献[11-12]和本文算法都为三角形简化算法。其中文献[12]通过在二次误差测度(QEM)中加入特征保持因子计算三角形的折叠代价,新点生成通过解非线性方程组;文献[11]以三角形不平滑度与三角形面积开方之积作为三角形权值来控制三角形简化顺序,以各个顶点不平滑度在三角形三个顶点不平滑度之和中所占的比例作为权值,以三角形三个顶点的加权平均值作为新顶点的位置。

表 3 Bunny 模型简化统计数据
Tab.3 Statistical data of simplification of Bunny

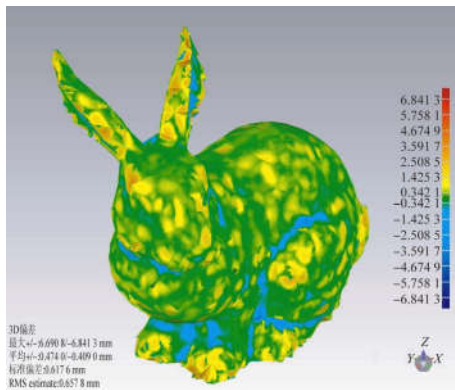
方案	R/%	N_T	正则率/%	时间/s	d_{max}/mm	d_{min}/mm	d_{SD}/mm	d_{RMS}/mm
文献[8]	60	27 004	87.50	34.790	0.734 1	-0.608 7	0.056 4	0.057 9
	73	18 872	71.57	46.360	1.945 1	-1.658 7	0.135 3	0.1407
文献[12]	60	23 608	89.52	55.504	0.568 8	-0.603 6	0.077 4	0.083 7
	90	6 900	80.25	77.165	1.841 9	-1.525 3	0.346 2	0.416 7
文献[11]	60	23 032	90.73	24.110	2.834 5	-1.916 3	0.157 8	0.162 6
	90	6 868	73.06	36.870	6.690 8	-6.841 3	0.617 6	0.657 8
本文算法	60	24 418	91.73	22.156	0.807 1	-0.837 1	0.059 6	0.062 4
	90	6 310	83.98	31.230	1.344 1	-1.252 1	0.249 1	0.286 5



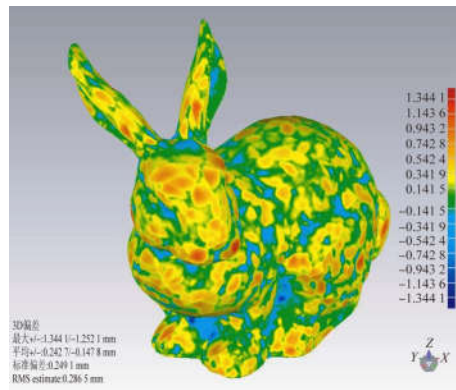
(a) 李现民方法
(a) Method of Li Xianmin



(b) 张欣方法
(b) Method of Zhang Xin



(c) 裴艳云方法
(c) Method of Pei Yanyun



(d) 本文方法
(d) Method of this paper

图 9 本文算法与不同的简化算法简化效果对比(简化 90%)

Fig. 9 Comparison of simplification of Standard Bunny by different methods (90% simplified)

从图 9 和表 3 可以看出:文献[12]和本文算法对于细节特征较明显的头部和脚部区域保持较好,但文献[12]算法效率最低且模型三角面片正则性较差。文献[11]算法虽然简化效率最高,但是从云图可以看出,模型变性非常大,尤其是在模型特征区域发生了较大变形。本文算法在细节特征保持上比文献[12]要好,且模型脊背较平坦的

区域保存完整,能有效防止平坦区域过度连续简化,且模型变形不大、效率适中。

由文献[8]算法、文献[11]算法、文献[12]算法以及本文算法的对比可知:采用单一的简化方法难以同时获得高简化率和小的简化误差,而本文的保特征混合折叠网格简化算法既能保证原始模型中的特征区域不被简化又能提高算法效率。

7.3 本文算法模型适应性测试分析

为验证本文算法的简化效果,对包含丰富细节特征 Bunny、Bunny 孔洞模型、Hand、Horse 模型进行了简化处理,图 10 为各模型处理后的网格效果。图 10(a)为包含丰富细节特征的 Bunny 模型的原始模型,包含 34 834 个顶点、69 664 个三角面片;图 10(b)和图 10(c)为简化率 60%、90% 的简化 Bunny 模型;图 10(d) Bunny 孔洞模型的原始模型,包含 33 233 个顶点、66 302 个三角面片;图 10(e)和图 10(f)为简化率 60%、90% 的 Bunny 孔洞模型。图 10(g)为 Hand 模型的原始模型,包含 45 279 个顶点、90 554 个三角面片;

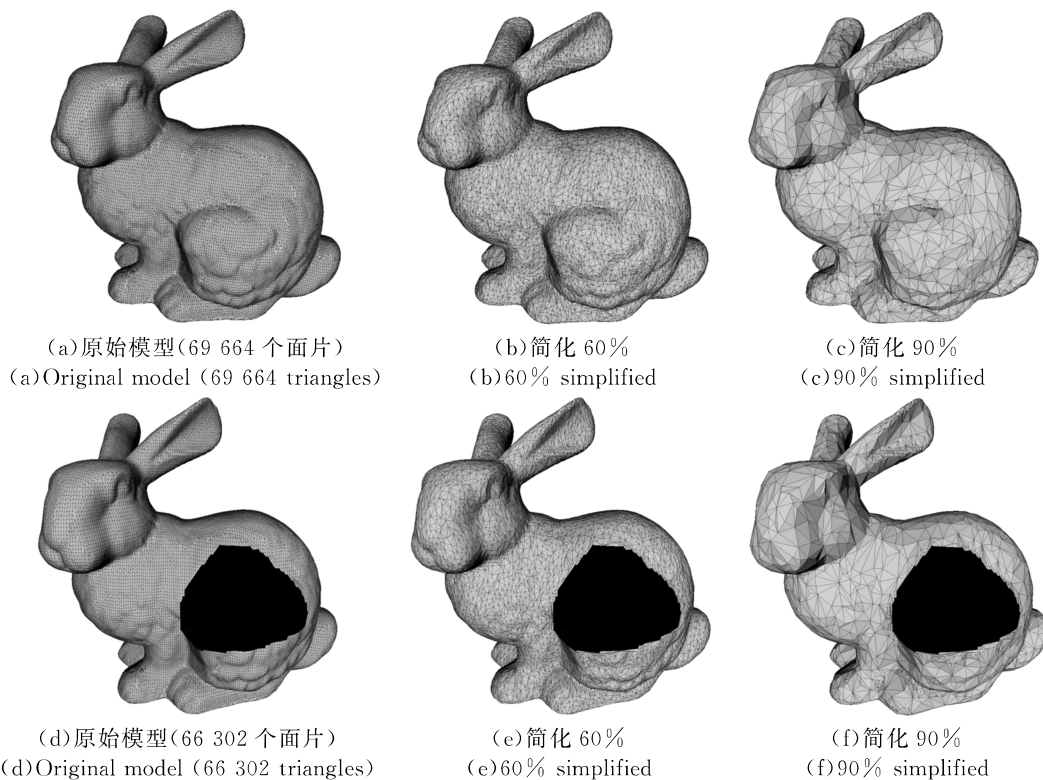
图 10(h)和图 10(i)为简化率 60%、90% 的简化 Hand 模型;图 10(j)Horse 模型的原始模型,包含 74 387 个顶点、148 762 个三角面片;图 10(k)和图 10(l)为简化率 60%、90% 的简化 Horse 模型。

从图 10(c)、图 10(f)、图 10(i)、图 10(l)中所显示的简化模型中可以看到,不管是简单的模型、带有孔洞特征的模型还是面片数较多的模型,本文的算法都适用,并且模型的细节特征都能保持的较完整,孔洞边界信息保持得较好。表 4 中给出的数据也能显示,该算法的简化速率较快,模型简化后网格质量较好。

表 4 模型简化统计数据

Tab. 4 Statistical data of simplification of models

方案	R/%	N_T	T/s	正则率/%	d_{\max}/mm	d_{\min}/mm	d_{SD}/mm	d_{RMS}/mm
Bunny	60	24 418	22.156	91.73	0.807 1	-0.837 1	0.0596	0.062 4
	90	6 310	32.230	83.98	1.344 1	-1.252 1	0.249 1	0.286 5
Bunny (孔洞)	60	23 468	17.217	91.90	0.870 5	-1.025 0	0.046 1	-0.035 2
	90	6 248	23.596	84.07	1.367 8	-1.057 3	0.238 8	-0.156 9
Hand	60	31 492	38.965	94.41	0.301 7	-0.164 9	0.0163	-0.010 9
	90	8 906	86.370	37.54	0.589 3	-0.573 0	0.091 8	-0.057 5
Horse	60	50 791	44.260	69.34	0.099 5	-0.104 4	0.014 6	-0.008 7
	90	14 630	64.430	60.88	0.432 7	-0.272 5	0.071 4	-0.038 8



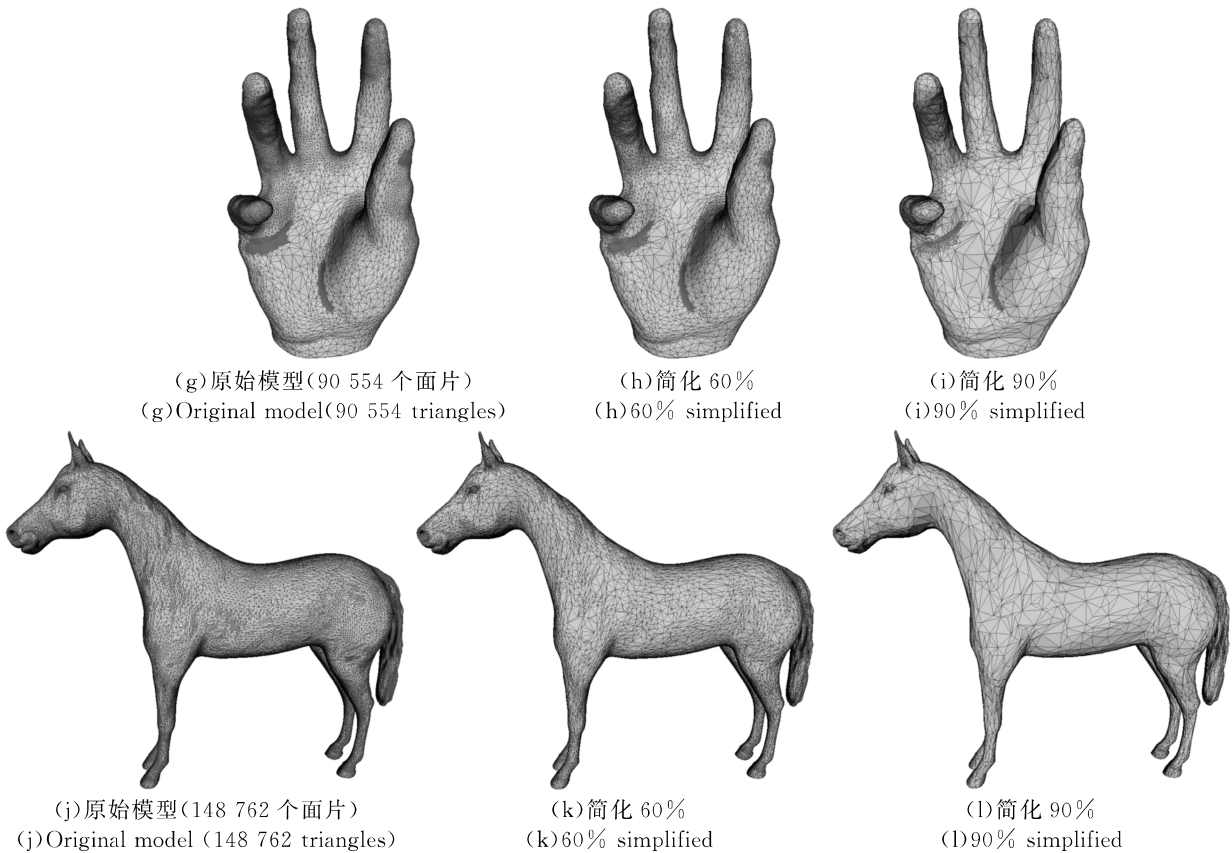


图 10 本文算法对各实验模型的简化效果

Fig. 10 Comparison of simplification results of different models by proposed method

7.4 简化后模型制造效果展示

为研究模型简化前后对数字化制造效率的影响,本文通过 VC++ 编程实现模型读取和 3D 打印中的分层算法^[20],并对简化前后 Bunny 模型在 3D 打印中的处理效率进行比较。表 5 中给出的是 Bunny 在处理前和被简化掉 90% 面片后执行读取和分层算法的时间效率对比。实验结果表明:被简化后的模型大大地提高了数字化制造的效率。此外,图 11 给出了简化前后的 Bunny 模型在相同参数设置条件下通过 FDM 打印机打印

出的实物效果。通过整体图和细节特征放大对比分析可以发现:简化前后模型的细节特征都保留得很好,简化对模型的外观并未有太大的影响。

表 5 Bunny 3D 打印成型统计数据

Tab. 5 Data of 3D printer of Bunny

项目	0%简化率	90%简化率	简化加速比/s
模型读取/s	6.373 0	1.147 9	5.225 1
分层切片/s	13.510 9	2.085 0	11.425 9



(a)原始模型制造结果(69 664 个面片)

(a)Fabrication results of original model (69 664 triangles)



(b)简化模型制造结果(90%简化)

(b)Fabrication results of simplified model(90% simplified)

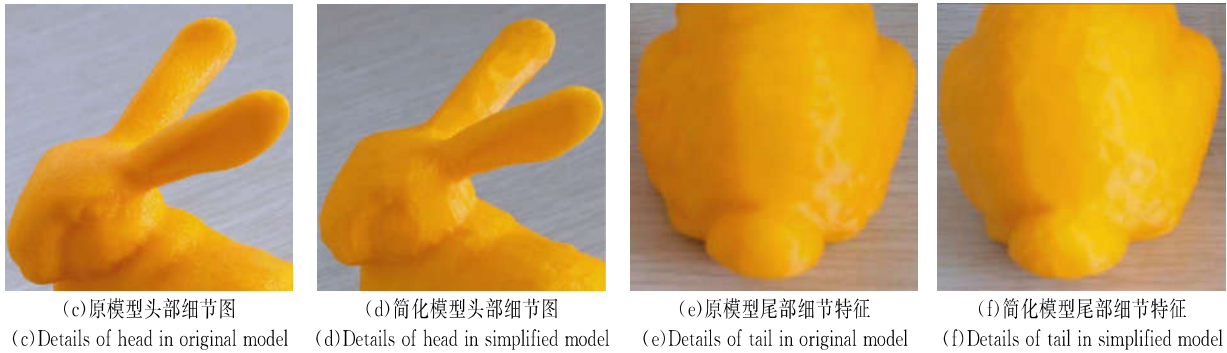


图 11 简化前后 Bunny 模型制造实物效果

Fig. 11 Comparison of fabrication results of Bunny

8 结 论

本文的网格简化算法基于曲度信息,在边界特征和细节特征提取的基础上对不同的对象采取不同的简化策略,实现了较高的特征保持性,简化前后模型变形较小,算法效率较适中。算法的主要特点体现在:提出了边界的改进边折叠、特征区域的边折叠和非特征区域的三角面片折叠的混合折叠算法,较好地解决了复杂三角网格模型简化

的特征保持和简化效率的平衡问题;通过将反映顶点局部几何信息的曲度加入到计算折叠后新网格顶点的权值计算中,增强了简化算法对不同模型的适应性,有效控制了简化前后模型的变形问题;提出最大距离法来改进传统的 QEM 算法,提高了简化算法的运行效率,大大节省了计算机的内存开销。

由于不同的场合下,对于特征信息需保留的程度不同,因此特征识别算法阈值以及简化率可以根据实际需求通过软件交互界面进行适当调整。

参考文献:

- [1] 耿国华, 石晨晨, 魏潇然, 等. 3D 打印中的模型分割与打包 [J]. 光学精密工程, 2016, 24(6): 1439-1447.
- [2] KENG G H, SHI CH CH, WEI X R, *et al.*. Model segmentation and packaging in 3D printing [J]. *Opt. Precision Eng.*, 2016, 24(6):1439-1447. (in Chinese)
- [3] DASSI F, ETTINGER B, PEROTTO S, *et al.*. A mesh simplification strategy for a spatial regression analysis over the cortical surface of the brain [J]. *Applied Numerical Mathematics*, 2015, 90:111-131.
- [4] 袁小翠, 吴禄镇, 陈华伟, 等. 特征保持点云数据精简 [J]. 光学精密工程, 2015, 23(9):2666-2676.
- [5] YUAN X C, WU L ZH, CHEN H W, *et al.*. Point cloud adaptive simplification of feature extraction [J]. *Opt. Precision Eng.*, 2015, 23(9):2666-2676. (in Chinese)
- [6] MICHAUD C, MELLADO N, PAULIN M. Mesh simplification with curvature error metric [C]. *Eurographics 2017, Lyon, France, Posters*, 2017.
- [7] LEE H, KYUNG M H. Parallel mesh simplification using embedded tree collapsing [J]. *Visual Computer*, 2016, 32(6-8):967-976.
- [8] CABIDDU D, ATTENE M. Large mesh simplification for distributed environments [J]. *Computers & Graphics*, 2015, 51:81-89.
- [9] LI Y B, ZHU Q. A new mesh simplification algorithm based on quadric error metrics [C]. *International Conference on Advanced Computer Theory and Engineering, Phuket, Thailand: ICACTE*, 2008: 528-532.
- [10] 李现民, 李桂清, 张小玲, 等. 基于子分规则的边折叠简化方法 [J]. 计算机辅助设计与图形学报, 2002, 14(1):8-13.
- [11] LI X M, LI G Q, ZHANG X L, *et al.*. Edge collapse simplification based on subdivision [J]. *Journal of Computer-Aided Design Computer Graphics*, 2002, 14(1):8-13. (in Chinese)
- [12] 段黎明, 杨尚朋, 张霞, 等. 基于遗传算法的三角网格折叠简化 [J]. 光学精密工程, 2018, 26(6): 1489-1496.
- [13] DUAN L M, YANG SH P, ZHANG X, *et al.*. Collapsing simplification of triangular mesh based on genetic algorithm [J]. *Opt. Precision Eng.*, 2018, 26(6):1489-1496. (in Chinese)
- [14] 段黎明, 邵辉, 李中明, 等. 高效率的三角网格模型保特征简化方法 [J]. 光学精密工程, 2017, 25(2):460-468.
- [15] DUAN L M, SHAO H, LI ZH M, *et al.*. Simpli-

- fication method for feature preserving of efficient triangular mesh model [J]. *Opt. Precision Eng.*, 2017, 25(2):460-468. (in Chinese)
- [11] 裴艳云, 陈飞翔. 一种基于不平滑度的网格简化算法 [J]. *计算机工程与应用*, 2013, 49(14):174-177.
PEI Y Y, CHEN F X. Mesh simplification algorithm based on non-smoothness [J]. *Computer Engineering and Applications*, 2013, 49(14):174-177. (in Chinese)
- [12] 张欣, 秦茂玲, 谢堂龙. 基于特征保持的三角形折叠网格简化算法 [J]. *计算机技术与发展*, 2012, 22(1):94-97.
ZHANG X, QIN M L, XIE L T. Mesh simplification algorithm of triangle collapse based on feature preserving [J]. *Computer Technology and Development*, 2012, 22(1):94-97. (in Chinese)
- [13] 佟玉斌, 王辉. 基于曲面拟合的三角形网格简化 [J]. *计算机工程*, 2012, 38(8):242-245.
TONG Y B, WANG H. Triangular mesh simplification based on surface fitting [J]. *Computer Engineering*, 2012, 38(8):242-245. (in Chinese)
- [14] WEI J, LOU Y. Feature preserving mesh simplification using feature sensitive metric [J]. *Computer Science and Technology*, 2010, 25(3):595-605.
- [15] 王爱霖, 刘弘, 张桂娟. 基于谷脊线特征的三维网格模型简化方法 [J]. *计算机辅助设计与图形学学报*, 2014, 26(5):788-793.
WANG A L, LIU H, ZHANG G J. A mesh simplification method based on ridge-valley lines [J]. *Journal of Computer-Aided Design Computer Graphics*, 2014, 26(5):788-793. (in Chinese)
- [16] 周继来, 周明全, 耿国华, 等. 基于曲度特征的三维模型检索算法 [J]. *计算机应用*, 2016, 36(7):1914-1917.
ZHOU J L, ZHOU M Q, GENG G H. 3D model retrieval algorithm based on curvedness feature [J]. *Journal of Computer Applications*, 2016, 36(7):1914-1917. (in Chinese)
- [17] WANG J, YU Z. Surface feature based mesh segmentation [J]. *Computers & Graphics*, 2011, 35(3):661-667.
- [18] ARHID K, ZAKANI F R, BOUKSIM M, et al.. An Efficient Hierarchical 3D Mesh Segmentation Using Negative Curvature and Dihedral Angle [J]. *International Journal of Intelligent Engineering & Systems*, 2017, 10(5):143-152.
- [19] 崔鑫, 闫秀天, 李世鹏. 保持特征的散乱点云数据去噪 [J]. *光学精密工程*, 2017, 25(12):3169-3178.
CUI X, YAN X T, LI S H P. Feature preserving scattered point cloud denoising [J]. *Opt. Precision Eng.*, 2017, 25(12):3169-3178. (in Chinese)
- [20] DING Y, DWIVEDI R, KOVACEVIC R. Process planning for 8-axis robotized laser-based direct metal deposition system: A case on building revolved part [J]. *Robotics and Computer-Integrated Manufacturing*, 2017, 44:67-76.

作者简介:



曹增欢(1993—),男,江西宜春人,硕士研究生,2015年于华东交通大学获得学士学位,主要从事数字化几何处理/增材制造方面的研究。E-mail: 784139048@qq.com



黄常标(1976—),男,湖南郴州人,副教授,硕士生导师,CCF会员,2005年,2008年于华侨大学分别获得硕士、博士学位,主要从事数字化制造、CAD/CAM/增材制造研究。E-mail: huangcb@hqu.edu.cn