

11-20-2019

Adaptively Resampling 3D Mesh Models Based on Editable Features

Jiajia Dai

1. Institute of EduInfo Science & Engineering, Nanjing Normal University, Nanjing 210097, China; ;

Lipeng Fan

1. Institute of EduInfo Science & Engineering, Nanjing Normal University, Nanjing 210097, China; ;2. School of Economics and Management, Nanjing University of Science & Technology, Nanjing 210094, China;;

Mingyong Pang

1. Institute of EduInfo Science & Engineering, Nanjing Normal University, Nanjing 210097, China; ;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Adaptively Resampling 3D Mesh Models Based on Editable Features

Abstract

Abstract: We propose a hybrid algorithm for adaptively resampling 3D triangulations by user-defined editable features. *The method parameterizes a 3D mesh model into 2D parameter plane, and the geometric properties of the original model is calculated and represented on a planar domain. According to a constructed geometric image of the original model and user-defined editing information, the method creates a global density function for the resampled model. The sampling density function is employed to control distribution of samples in the 2D parameter domain. The method uses centroidal Voronoi tessellation technique to further optimize local distribution of the sampled points. The created samples in 2D domain are mapped to 3D space and the resulted model is obtained with adaptive sampling property.* Experiments show that the algorithm can deal with various mesh models efficiently and robustly. The distribution of vertices of resulted model is adaptive and can be controlled by user-defined features.

Keywords

adaptive sampling, Ricci flow, model editing, mesh parameterization

Recommended Citation

Dai Jiajia, Fan Lipeng, Pang Mingyong. Adaptively Resampling 3D Mesh Models Based on Editable Features[J]. Journal of System Simulation, 2019, 31(5): 853-860.

特征驱动的三维网格模型自适应重采样算法

戴佳佳¹, 范丽鹏^{1,2}, 庞明勇¹

(1. 南京师范大学教育信息工程研究所, 江苏 南京 210097; 2. 南京理工大学经济管理学院, 江苏 南京 210094)

摘要: 提出一种由可编辑特征驱动的三维网格模型自适应重采样算法, 该算法运用一组特征曲线控制重采样密度。将网格模型参数化到平面域, 用灰度几何图像表示原模型的局部几何信息; 由几何图像的灰度及用户编辑信息定义三维模型表面采样的密度控制函数; 该函数控制采样点在参数域中的疏密分布, 并采用重心Voronoi方法优化采样点的局部分布; 将采样结果映射到三维空间生成重采样模型。算法能有效地处理不同模型, 得到的重采样点分布具有局部特征自适应性, 用户以交互方式控制采样分布。

关键词: 自适应重采样; Ricci流; 模型编辑; 网格参数化

中图分类号: TP391

文献标识码: A

文章编号: 1004-731X(2019)05-0853-08

DOI: 10.16182/j.issn1004731x.joss.17-0169

Adaptively Resampling 3D Mesh Models Based on Editable Features

Dai Jiajia¹, Fan Lipeng^{1,2}, Pang Mingyong¹

(1. Institute of EduInfo Science & Engineering, Nanjing Normal University, Nanjing 210097, China;

2. School of Economics and Management, Nanjing University of Science & Technology, Nanjing 210094, China)

Abstract: We propose a hybrid algorithm for adaptively resampling 3D triangulations by user-defined editable features. The method parameterizes a 3D mesh model into 2D parameter plane, and the geometric properties of the original model is calculated and represented on a planar domain. According to a constructed geometric image of the original model and user-defined editing information, the method creates a global density function for the resampled model. The sampling density function is employed to control distribution of samples in the 2D parameter domain. The method uses centroidal Voronoi tessellation technique to further optimize local distribution of the sampled points. The created samples in 2D domain are mapped to 3D space and the resulted model is obtained with adaptive sampling property. Experiments show that the algorithm can deal with various mesh models efficiently and robustly. The distribution of vertices of resulted model is adaptive and can be controlled by user-defined features.

Keywords: adaptive sampling; Ricci flow; model editing; mesh parameterization

引言

三维网格重采样技术是优化模型数据表示、提

高网格模型质量的重要途径^[1], 也是计算机图形学、计算机视觉等领域的热点问题^[2-3]。目前, 三维网格模型重采样算法多由输入模型自动生成采样点集^[4], 用户难以对采样结果进行交互式控制。随着技术的发展, 人机交互问题越来越受到人们的关注^[5-6]。在一些实际应用中, 需要对重采样过程进行一定的人工干预, 使重采样操作能够在模型表



收稿日期: 2017-04-20 修回日期: 2017-07-04;
基金项目: 国家自然科学基金重点项目(41631175),
江苏省社会科学基金(15TQB005), 江苏省现代教育
技术研究课题(2014-R33356);
作者简介: 戴佳佳(1993-), 女, 安徽芜湖, 博士
生, 研究方向为数字几何处理。

<http://www.china-simulation.com>

• 853 •

面局部区域生成指定密度的采样结果。因此研究可编辑的三维网格重采样技术有着重要的意义。

自 Dippé 等^[7]将随机采样技术引入到图形学中以来, 采样理论得到了深入研究。现有的网格模型重采样算法可分为两类: 三维表面直接采样^[8-9]和基于参数化的采样方法^[10-11]。第一类方法通常依据不同策略直接在模型表面上进行采样, 如 Cline 等^[8]将经典的飞镖算法应用到三维模型表面, 对三维模型直接进行采样; Xu 等^[12]应用 CCVT (Capacity Constrained Voronoi Tessellation)^[9]技术对曲面网格进行采样, 其采样结果具有潜在的人为痕迹 (Artifacts); Chen 等^[13]将 CCVT 与 CVT (Centroidal Voronoi Tessellation)^[14]相结合, 通过 CapCVT (Capacity-constrained CVT) 方法对曲面网格进行采样, 显著地减少了规则性人为痕迹现象; Yan 等^[15]基于空区域方法来分析曲面表面最大 Poisson-disk 采样 (Maximal Poisson-disk Sampling), 进而对模型进行采样处理; Guo 等^[16]通过基于层次细分的方法来改进该算法, 提高了采样的质量和效率。通常, 该方法需要考虑测地线问题, 计算量较大。第二类方法先将三维模型参数化到平面区域, 然后在平面区域中利用二维采样技术得到二维采样点, 最后将二维采样点重新映射到三维空间中生成重采样结果, 如 Alliez 等^[11]先利用误差扩散方法生成初始采样点, 然后利用 CVT 方法优化采样点的位置, 进而重构三维网格; Peyré 等^[17]利用简化的三维网格模型来构建测地三维网格, 再根据距离信息参数化测地三维网格, 进而重构优化的三维模型; Quinn 等^[18]先通过参数化方法将人工分割的三维网格映射为平面, 再进行采样和重建三维模型。基于参数化的重采样方法中的参数化过程会带来一定的采样变形, 一般也不考虑用户交互编辑。

本文基于第二类方法, 提出一种可编辑特征驱动的三维网格模型自适应重采样算法。相比于现有方法, 本文算法的优点在于: 1) 可通过设置不同的密度函数来控制采样点的分布, 得到自适应的三维网格模型; 2) 可根据用户的输入, 对三维网格

进行局部编辑, 并能用编辑操作控制采样结果。

1 算法梗概

如图 1 所示, 本文算法的输入是三维网格模型 (称为“原模型”); 输出是重构的自适应三维网格模型。具体步骤如下: 1) 运用平面参数化技术将三维网格模型展平到二维平面域中, 得到原模型的平面参数化网格; 2) 根据平面网格二维坐标与原模型三维坐标之间的对应关系以及原模型中的几何信息, 构造相应的灰度图像; 3) 利用灰度信息以及用户编辑信息确定密度函数; 4) 在二维平面中, 根据密度函数构造加权 CVT, 生成自适应的采样点集; 5) 利用重心插值技术将自适应的二维采样点集映射到三维空间, 并重构三维网格模型, 从而实现三维网格的自适应重采样过程。

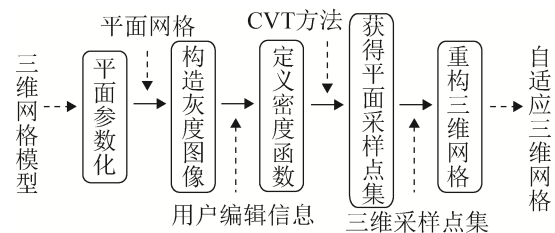


图 1 算法流程图

Fig. 1 Pipeline of our algorithm

为了优化本文算法的性能, 我们在上述步骤中引入了两个特别的操作过程: 1) 由于参数化过程所带来的变形影响着重采样的结果, 为了减少参数化过程中的变形对重采样结果的影响, 本文引入面积扭曲度量, 即在定义采样密度函数时, 将模型参数化前后的面积度量融入其中; 2) 在利用 CVT 进行重采样的过程中, 由于需要迭代多次才能使采样过程达到收敛状态。本文采用基于增/删点的加速迭代策略来减少迭代次数, 加速算法的计算过程。

2 平面参数化

早期的平面参数化往往需要先对模型进行分片处理, 再将各个模型片段逐一参数化到平面区域上。其不足在于相邻分片的参数化边界处难以保持

连续性^[19]。为解决此问题, 学者们提出了全局参数化方法^[20]。本文采用文献[21]中的方法实现曲面网格的平面参数化, 该方法可对具有不同拓扑结构的曲面使用统一的方法进行全局共形参数化。该方法基于 CP(Circle Packing)理论^[22], 将几何度量定义在模型的各项点处, 并利用离散 Ricci 流计算顶点的黎曼度量, 使其所诱导的高斯曲率与定义的目标曲率相一致。根据文献[22], 与传统的 CP 度量相比, 逆距离 CP 度量能够提高 Ricci 流的通用性和灵活性。本文采用逆距离 CP 度量作为 CP 度量。

本文平面参数化主要包括 3 个部分: 度量初始化、基于 Ricci 流的参数化和平面嵌入。

2.1 度量初始化

设 $M = (V, E, F)$ 为输入的原网格模型。其中 $V = \{v_i\}$, $E = \{e_{ij}\}$, $F = \{f_{ijk}\}$ 分别为 M 的顶点集、边集和面片集。这里, v_i 为 M 的第 i 个顶点, e_{ij} 是连接 v_i 和 v_j 的边, f_{ijk} 为由顶点 v_i , v_j , v_k 围成的面。进一步记 α_i^{jk} 为顶点 v_i 处的面 f_{ijk} 的内角, l_{ij} 为边 e_{ij} 的边长。另以顶点 v_i 为圆心, γ_i 为半径在网格表面做测地圆; φ_{ij} 表示边 e_{ij} 在参数化过程中对应的保角不变量; K_i 为顶点 v_i 处的高斯曲率, \bar{K}_i 为顶点 v_i 处的目标高斯曲率。度量初始化的具体步骤如下:

对所有的面 $f_{ijk} \in M$, 计算顶点 v_i , v_j , v_k 在面 f_{ijk} 中对应的半径 $\gamma_i^{jk} = (l_{ij} + l_{ki} - l_{jk})/2$; 计算每个顶点 $v_i \in M$ 对应的半径 $\gamma_i = \min_{jk} \gamma_i^{jk}$; 计算每条边 $e_{ij} \in M$ 对应的保角不变量 $\varphi_{ij} = (l_{ij}^2 - \gamma_i^2 - \gamma_j^2)/2\gamma_i\gamma_j$; 设定目标曲率处处为 0, 即 $\bar{K} \equiv 0$ 。

2.2 基于 Ricci 流的参数化

记含有 CP 度量的网格为 (M, Γ, Φ) , $\Gamma = \{\gamma_i\}$ 为顶点上的半径集合, $\Phi = \{\varphi_{ij}\}$ 为边上的保角不变量集合。Ricci 流参数化为保角映射, 定义该映射为: $(M, \Gamma, \Phi) \rightarrow (M, \bar{\Gamma}, \bar{\Phi})$, 其中 $\bar{\Gamma}$ 为诱导后的顶点上的半径集合。利用离散的 Ricci 流方程能得到与目标曲率相对应的目标 CP 度量。下面给出离散

Ricci 流方程:

$$\frac{d\gamma_i}{dt} = (\bar{K}_i - K_i)\gamma_i \quad (1)$$

式中: γ_i 和 K_i 分别为顶点 v_i 处测地圆的半径和预设的目标高斯曲率。令 $u_i = \ln \gamma_i$, 离散 Ricci 流方程被简化为:

$$\frac{du_i}{dt} = (\bar{K}_i - K_i) \quad (2)$$

该方程是能量函数的负梯度流。且能量函数的 Hessian 矩阵为正定矩阵, 所以方程存在全局最小解。通过牛顿迭代法计算该方程, 具体步骤如下:

1) 首先计算各项点的共形因子 $u_i = \ln \gamma_i$; 计算当前网格的黎曼度量 $l_{ij} = \sqrt{\gamma_i^2 + \gamma_j^2 + 2\gamma_i\gamma_j\varphi_{ij}}$; 然后利用余弦定理计算所有三角形的内角 $\alpha_k^{ij} = \arccos((l_{jk}^2 + l_{ki}^2 - l_{ij}^2)/2l_{jk}l_{ki})$;

2) 由

$$K(v) = \begin{cases} 2\pi - \sum_{f_{ijk} \in F} \alpha_i^{jk}, v \notin \partial M \\ \pi - \sum_{f_{ijk} \in F} \alpha_i^{jk}, v \in \partial M \end{cases} \quad (3)$$

计算各项点的离散高斯曲率 K_i , 其中 α_i^{jk} 为顶点 v 处面 f_{ijk} 内的角, ∂M 为网格的边界; 然后计算式(2)的 Hessian 矩阵 H ; 根据 $H\delta u = \bar{K} - K$ 计算 δu ; 更新共形因子 $u \leftarrow u + \delta u$ 及顶点的半径集合;

3) 计算 $\max_{v_i \in M} |\bar{K}_i - K_i|$, 并判断其值是否小于设定误差值 ε , 若小于设定误差值则结束迭代; 否则, 转步骤 1)。

2.3 平面嵌入

由参数化结果可以得到目标 CP 度量, 利用此度量将模型嵌入到平面区域, 具体过程如下:

首先利用余弦定理计算模型中所有的边长 l_{ij} : $l_{ij}^2 = \gamma_i^2 + \gamma_j^2 + 2\gamma_i\gamma_j\varphi_{ij}$; 再随机选择一个面作为嵌入平面的第一个面 f_{012} , 计算其 3 个内角 θ_0 , θ_1 , θ_2 ; 接着将面 f_{012} 嵌入到平面中, 计算其顶点 v_0 , v_1 , v_2 的二维平面坐标: $\tau(v_0) = (0, 0)$, $\tau(v_1) = (l_{01}, 0)$, $\tau(v_2) = (l_{20} \cos \theta_0, l_{20} \sin \theta_0)$; 然后将所有与面 f_{012} 相邻的面都压入一个堆栈; 然后根据从相邻面的堆栈弹出面 f_{ijk} 判断其 3 个顶点 v_i , v_j , v_k 是否都已嵌

入到平面中,若全部嵌入则弹出下一个面;若有顶点 v_k 未嵌入,则此时顶点 v_i 与 v_j 已嵌入到平面中。则以 v_i 与 v_j 为圆心,以 l_{ki} 和 l_{jk} 为半径的两个圆相交于两点,满足 $(\tau(v_j) - \tau(v_i)) \times (\tau(v_k) - \tau(v_i)) > 0$ 的交点就是 $\tau(v_k)$,把所有不在堆栈中和 f_{ijk} 相邻的面压入堆栈;最后重复上一步操作之至模型中所有面都被压入堆栈。

3 密度函数构造

3.1 构造灰度几何图像

根据原模型的局部曲率以及参数化过程中的面积扭曲度量来构造灰度几何图像。首先采用文献[23]中的方法计算原模型中每个顶点的曲率,然后根据参数化后的平面网格计算灰度图像中每个像素在原模型中对应的曲率,最后对平面网格外像素的曲率均设为 0,并将所有像素点的曲率记为 K_I 。

对面积扭曲的度量:先计算参数化后平面网格与原模型的总面积比值 $R_M = A_M / A'_M$,其中 A_M 和 A'_M 分别为平面网格和原模型的总面积;再分别计算顶点 v_i 1-环邻域的三角面在平面网格及原模型上的面积之和 A_i 与 A'_i 的比值 $R_{v_i} = A_i / A'_i$;进而确定顶点 v_i 处的面积扭曲度量 $d_{v_i} = R_{v_i} / R_M$;最后根据像素点与平面网格的位置关系及平面网格中每个顶点的面积扭曲度量,计算各像素点的面积扭曲度量,并将所有像素点面积扭曲值记为 A_I 。

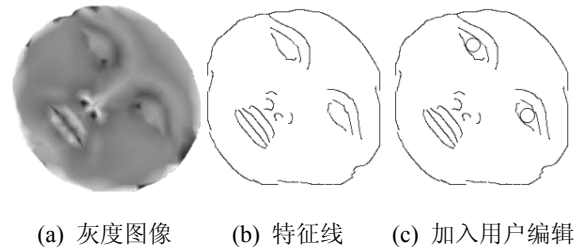
灰度图像 I 的像素灰度由像素点的曲率和面积扭曲值的线性组合 $I = k_1 K_I + k_2 A_I$ 得到,其中 k_1 , k_2 分别为控制系数。

3.2 定义密度函数

在对三维网格模型进行重采样的过程中,通过构造密度函数来调节采样点的疏密程度,进而保持原模型的细微特征以及用户编辑区域的形状。

上节中所构造的灰度图像蕴含着三维网格模型的曲率及面积扭曲度量信息,见图 2(a)。本文方法首先采用 Canny 算子^[24]来提取该图像的特征线,见图 2(b),并在特征线中添加用户编辑区域的形状信

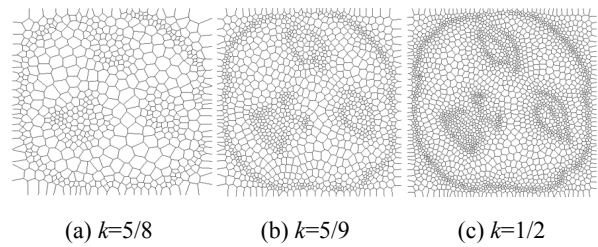
息,见图 2(c);然后根据采样点到特征线以及用户编辑区域的最近距离来定义密度函数,即 $\rho = 1/d^k$,其中 $d(d > 0)$ 为采样点到特征线的最短距离, $k(k > 0)$ 为控制参数。通过调整 k 值,可控制点集疏密的过渡情况, k 值越小,采样点的疏密过渡越平缓,如图 3 所示。



(a) 灰度图像 (b) 特征线 (c) 加入用户编辑

图 2 特征线及用户编辑

Fig. 2 Feature lines (middle) extracted from a gray image (left) and its edited version (right)



(a) $k=5/8$ (b) $k=5/9$ (c) $k=1/2$

图 3 不同 k 值的效果图

Fig. 3 Resampled 2D meshes with different k values

4 自适应重采样

4.1 定义加权 CVT

在欧式二维空间中,给定平面区域 Ω 内的点集 $\{x_i\}_{i=1}^n$,则每个点 x_i 对应的 Voronoi 区域 D_i 为^[25]:

$$D_i = \{x \in \Omega \mid d(x, x_i) \leq d(x, x_j),$$

$$\forall j \neq i, j \in [1, n]\} \quad (4)$$

式中: $d(*, *)$ 表示两点之间的欧式距离。由公式(4)可知, $v_i \cap v_j = \emptyset (i \neq j)$,则 $\{v_i\}_{i=1}^n$ 为平面区域 Ω 的一种 Voronoi 剖分结构。CVT 是 Voronoi 图的一种特殊形式,其能量函数如下:

$$E(x) = \sum_{i=1}^n \int_{D_i} \rho(x) \|x - x_i\|^2 dx \quad (5)$$

当 $E(x)$ 值最小时, x_i 为 D_i 区域的重心,即:

$$x_i = \int_{D_i} x \rho(x) dx / \int_{D_i} \rho(x) dx \quad (6)$$

式中: $\rho(x) > 0$ 为 3.2 节中的密度函数, 所得采样点集 $\{x_i\}_{i=1}^n$ 的疏密程度符合该密度函数的分布规律。

4.2 构建加权 CVT

在密度函数的基础上, 通过构建加权 CVT 来实现采样点集的自适应分布。CVT 构建步骤为:

1) 给定平面区域内的点集 $P = \{x_i\}_{i=1}^n$, 构建 P 的 Voronoi 图 $P_V = \{v_i\}_{i=1}^n$: 本文采用分治法来构建点集 P 的 Voronoi 图^[26];

2) 求得 P_V 中各区域的重心点, 得到点集 P' : 设 $v_i^k (k=1, 2, \dots, m)$ 为区域 D_i 的顶点, $\rho_i^k (k=1, 2, \dots, m)$ 为区域 D_i 各顶点的密度值, 则区域 D_i 的质心点 x'_i 为:

$$x'_i = \frac{1}{m} \sum_{k=1}^m (v_i^k \cdot \rho_i^k) \quad (7)$$

3) 将顶点 x_i 移至 x'_i 的位置;

4) 根据点集 P 计算能量函数 $E(x)$ 。若 $E(x)$ 小于一定阈值, 则得到符合密度函数的点集 P , 反之则重复执行步骤 1)~3)。

在上述 CVT 构建过程中, 使用的迭代方法为 Lloyd 迭代法^[27], $E(x)$ 在每一次的迭代过程中变化较小。为了减少迭代次数, 我们在步骤 3) 中采用本文 4.3 节中的迭代加速方法。

4.3 加速迭代策略

CVT 构建过程中, 迭代可使采样点的疏密程度能够近似地符合密度函数的分布规律, 即在距离特征线较近的地方采样密集, 反之采样稀疏。为了加速迭代的收敛速度, 本文在迭代过程中根据设定的条件, 在特征线附近增加采样点或减少采样点, 来加速迭代的收敛。增/删点的具体步骤为:

1) 计算点集的 Delaunay 三角网格;

2) 根据密度函数计算 Delaunay 三角网格中各三角形顶点的密度, 取三角形 3 个顶点密度的平均值作为该三角形的密度, 并将三角形的密度与面积之积作为该三角形的分辨率;

3) 对于 Delaunay 三角网格中的任一三角形, 若该三角形被标记, 则不进行处理。在未被标记的三角形中, 若其分辨率大于设定阈值, 则添加该三角形的重心点作为新采样点, 并标记其相邻三角形; 若其分辨率小于另一设定阈值, 则删除该三角形的顶点, 添加该三角形重心点作为采样点, 并标记其相邻三角形。

阈值大小与三角网格的分辨率 M (即所有三角形分辨率的平均值) 相关。大量实验表明, 当设定值为 $0.875 M$ 和 $0.125 M$ 时, 效果较好。

4.4 重构三维网格

本文通过将 4.2 节中的二维自适应采样点集映射到三维空间来重构三维模型, 从而实现三维网格的自适应重采样。

首先根据 Voronoi 图与 Delaunay 三角网格的对偶关系, 将二维采样点集三角网格化, 得到新的平面网格; 其次为了提高网格质量, 本文采用“边交换”策略, 对新平面网格顶点之间的连通度量进行优化; 然后根据平面参数化一一映射的特性, 将优化后的平面网格覆盖在参数化的平面网格上, 并通过判定每个采样点在参数化平面网格中三角形的位置, 确定其在原网格模型中的原始三角形; 最后运用重心插值技术, 根据原始三角形的三维坐标得到新采样点的三维坐标, 从而得到重构三维模型。

5 实验与结果

作者在 PC 机上运用 VC6.0 和 OpenGL 实现了三维网格模型的自适应重采样过程, 图 4 为整个算法的实现过程, 其中第一行从左至右分别为原网格模型、平面参数网格和灰度图像; 第二行从右至左分别为根据灰度图像提取的特征线集合、自适应的平面重采样结果和自适应的三维重采样结果; 第三行从右至左分别是加入用户编辑信息的特征线集合、用户编辑后的自适应平面重采样结果和用户编辑后的自适应三维重采样模型。

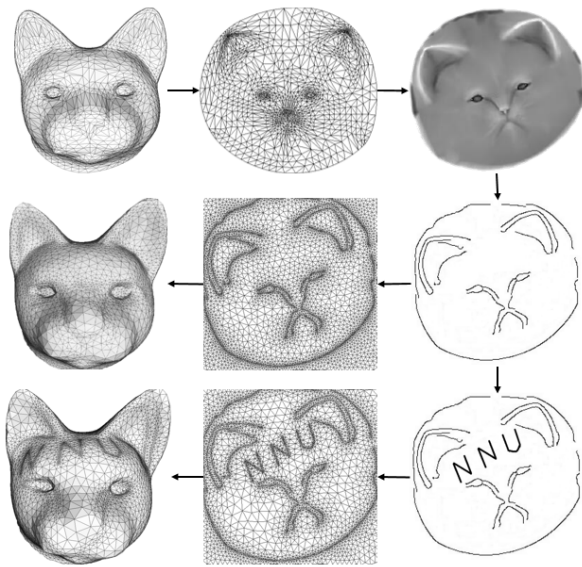


图 4 算法实现过程

Fig. 4 Implemented workflow of our algorithm

本文算法根据参数化后的平面网格和原三维模型中的信息绘制灰度图像,并根据灰度图像提取特征线集合,用户在特征线集合中添加编辑信息,进而构造采样的密度函数,运用加权 CVT 方法生成二维自适应采样点集,并将采样点集映射到三维空间中重构三维网格模型。如图 5 所示,对不同模型附加了用户编辑信息后的重采样结果,其中:第一列为带有用户编辑信息的特征线集合;第二列为由二维自适应采样点集生成的三角网格;第三列为重采样三维网格模型。本文算法使用加权 CVT 方法生成二维采样点集,首先设定采样点集中点的数目,然后从 100 个点开始采用加速迭代策略,依次增加点的数目直到超过定点数目为止,并使用 Lloyd 方法迭代直至点集的能量函数值低于阈值。另将最终点的数目设为 Lloyd 方法中迭代的点的总数,以此得出 Lloyd 迭代的次数。实验表明,相比于 Lloyd 方法,我们的方法迭代次数更少,且效果更好:具体如图 6 和表 1 所示。图 6 为同一模型同一密度函数,分别选取 2 097, 3 449 和 4 357 三种规模的点集,经过加权 CVT 后,获得的 Voronoi 图结果。其中(a)分别为 Lloyd 方法对三种规模的点集进行迭代后的 Voronoi 图,(b)为本文的方法对三种规模的点集迭代后得到的 Voronoi 图结果。表 1

是对同一密度函数,分别选取 636, 1 071, 1 879, 2 300 和 2 897 五种规模的点集,采集点集分布趋于稳定时需要迭代的次数。由表中数据可知,本文加速迭代策略相较于 Lloyd 方法,能够有效减少迭代计算的次数。对于不同密度函数重采样的结果如图 7 所示,第一列为参数化平面网格和原始三维网格,第二、三列分别为密度函数中 $k=5/8$, $k=5/9$ 的平面自适应采样点集的三角网格及重采样的三维网格模型。

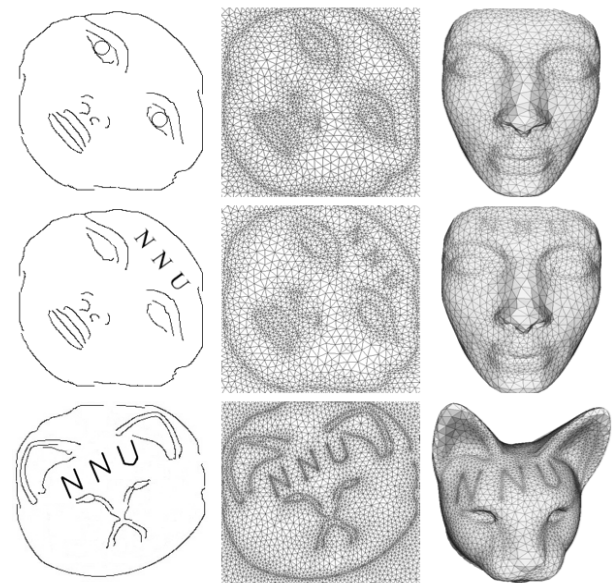
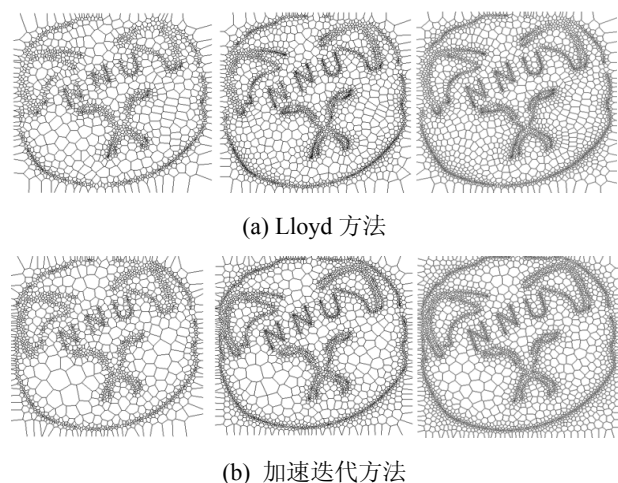


图 5 用户编辑的重采样结果

Fig. 5 Reconstructing edited 3D models



(a) Lloyd 方法

(b) 加速迭代方法

图 6 本文方法与 Lloyd 方法产生的 Voronoi 图的对比
Fig. 6 Comparing our method and Lloyd's method

表 1 不同规模点集的迭代次数

Tab. 1 A statistic of iteration number on point sets with different sizes

方法	点集数				
	636	1 071	1 879	2 300	2 897
Lloyd	161	605	741	825	814
ours	147	188	237	298	326

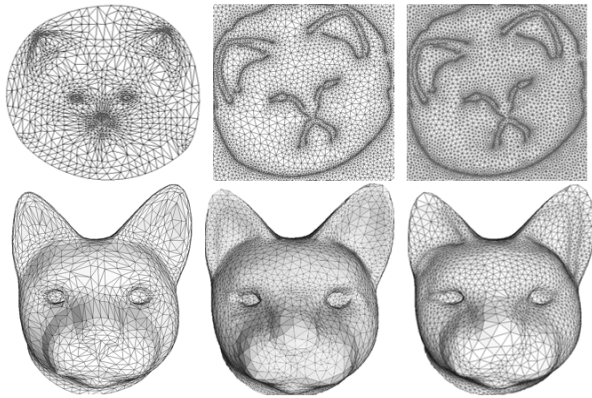


图 7 不同密度函数的重采样结果

Fig. 7 Sampling results of different density functions

6 结论

本文提出一种三维网格模型自适应重采样算法, 算法应用平面参数化方法和模型局部几何特征分析等构造采样密度控制函数, 并利用模型与参数域的对用关系实现三维重采样模型的重建。实验表明, 本文算法能够根据密度函数的变化, 产生不同疏密程度的采样点集, 得到的采样点集能够保持原三维模型的局部特征, 用户可通过编辑手段交互地控制重采样模型的局部采样密度与局部形状变化。

参考文献:

- [1] 赵向军, 路梅, 叶澄清. 全规整重网格化三维模型的压缩 [J]. 计算机辅助设计与图形学学报, 2007, 19(4): 436-441.
Zhao Xiangjun, Lu Mei, Ye Chengqing. Mesh compression by uniform resampling [J]. Journal of Computer-Aided Design and Computer Graphics, 2007, 19(4): 436-441.
- [2] Ebeida M, Davidson A, Patney A, et al. Efficient maximal Poisson-disk sampling [J]. ACM Transactions on Graphics (S0730-0301), 2011, 30(4): 49.
- [3] Wu F, Dong W, Kong Y, et al. Feature-aware natural texture synthesis [J]. The Visual Computer (S0178-2789),

2016, 32(1): 43-55.

- [4] Wachtel F, Pilleboue A, Coeurjolly D, et al. Fast tile-based adaptive sampling with user-specified Fourier spectra [J]. ACM Transactions on Graphics (S0730-0301), 2014, 33(4): 56:1-56:11.
- [5] Taylor J, Bordeaux L, Cashman T, et al. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences [J]. ACM Transactions on Graphics (S0730-0301), 2016, 35(4): 143.
- [6] Ostromoukhov V. Sampling with polyominoes [J]. ACM Transactions on Graphics (S0730-0301), 2007, 26(3): 78.
- [7] Dippé M, Wold E. Antialiasing through stochastic sampling [C]//The 12th annual conference on Computer graphics and interactive techniques, New York, ACM, 1985: 69-78.
- [8] Cline D, Jeschke S, White K, et al. Dart throwing on surfaces [C]//The Twentieth Eurographics Conference on Rendering, Girona, Spain, 2009: 1217-1226.
- [9] Balzer M, Schlömer T, Deussen O, et al. Capacity-constrained point distributions: a variant of Lloyd's method [J]. ACM Transactions on Graphics (S0730-0301), 2009, 28(6): 86.
- [10] Yan D, Guo J, Jia X, et al. Blue-noise remeshing with farthest point optimization [J]. Computer Graphics Forum (S1467-8659), 2014, 33(5): 167-176.
- [11] Alliez P, De Verdière É C, Devillers O, et al. Centroidal Voronoi diagrams for isotropic surface remeshing [J]. Graphical Models (S1524-0703), 2005, 67(3): 204-231.
- [12] Xu Y, Hu R, Gotsman C, et al. Blue noise sampling of surfaces [J]. Computers & Graphics (S0097-8493), 2012, 36(4): 232-240.
- [13] Chen Z, Yuan Z, Choi Y, et al. Variational blue noise sampling [J]. IEEE Transactions on Visualization & Computer Graphics (S1077-2626), 2012, 18(10): 1784-1796.
- [14] Yan D, Lévy B, Liu Y, et al. Isotropic remeshing with fast and exact computation of restricted Voronoi diagram [J]. Computer Graphics Forum (S1467-8659), 2009, 28(5): 1445-1454.
- [15] Yan D, Wonka P. Gap processing for adaptive maximal Poisson-disk sampling [J]. ACM Transactions on Graphics (S0730-0301), 2013, 32(5): 148.
- [16] Guo J, Yan D, Jia X, et al. Efficient maximal Poisson-disk sampling and remeshing on surfaces [J]. Computers & Graphics (S0097-8493), 2015, 46(3): 72-79.

- [17] Peyré G, Cohen L. Geodesic remeshing using front propagation [J]. *International Journal of Computer Vision* (S0920-5691), 2006, 69(1): 145-156.
- [18] Quinn J, Langbein F, Lai Y, et al. Generalized anisotropic stratified surface sampling [J]. *IEEE Transactions on Visualization & Computer Graphics* (S1077-2626), 2013, 19(7): 1143-1157.
- [19] 郭风华, 张彩明, 焦文江. 网格参数化研究进展[J]. *软件学报*, 2016, 27(1): 112-135.
Guo Fenghua, Zhang Caiming, Jiao Wenjiang. Research progress on mesh parameterization [J]. *Journal of Software*, 2016, 27(1): 112-135.
- [20] Khodakovsky A, Litke N, Schroder P. Globally smooth parameterizations with low distortion [J]. *ACM Transactions on Graphics* (S0730-0301), 2003, 22(3): 350-357.
- [21] Zeng W, Samaras D, Gu X, et al. Ricci flow for 3D shape analysis [J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (S0162-8828), 2010, 32(4): 662-677.
- [22] Guo R. Local rigidity of inversive distance circle packing [J]. *Transactions of the American Mathematical Society* (S0002-9947), 2009, 363(9): 4757-4776.
- [23] Meyer M, Desbrun M, Schröder P, et al. Discrete differential-geometry operators for triangulated 2-manifolds [M]. *Visualization & Mathematics III*. Springer-Verlag, Heidelberg, 2003: 35-57.
- [24] Canny J. A computational approach to edge detection [J]. *IEEE Transaction on Pattern Analysis & Machine Intelligence* (S0162-8828), 1986, 8(6): 679-698.
- [25] Gunzburger M. Advances in studies and applications of centroidal Voronoi tessellations [J]. *Numerical Mathematics Theory Methods & Applications* (S1004-8979), 2010, 3(2): 119-142.
- [26] Wang J, Cui C, Rui Y, et al. A parallel algorithm for constructing Voronoi diagrams based on point-set adaptive grouping [J]. *Journal of Graphics* (S2095-302X), 2014, 26(2): 434-446.
- [27] Hateley J, Wei H, Chen L. Fast methods for computing centroidal Voronoi tessellations [J]. *Journal of Scientific Computing* (S1573-7691), 2015, 63(1): 185-212.