

基于 L_0 优化的网格曲面特征线提取算法

杨贤康, 潘茂东, 童伟华

(中国科学技术大学 数学科学学院, 合肥 230026)

摘 要: 基于网格曲面特征线的稀疏分布, 提出一种优化的特征线提取算法。对于给定的网格, 在每个面上计算一个值或向量作为输入。对输入的度量建立 L_0 优化模型, 使其在网格边上的跃变尽可能少且优化前后的变化较小。给出基于变量分裂技术与罚函数方法的交替方向优化算法, 并引入一种迭代的策略提升解的稀疏性, 以取得更高质量的特征线。实验结果表明, 该算法能有效提取网格曲面的特征线, 与 Crest lines 算法、变分算法等相比, 提高了特征线提取的质量和带噪数据的鲁棒性。

关键词: 网格曲面; 特征线提取; L_0 优化; 变量分裂; 交替方向优化算法

中文引用格式: 杨贤康, 潘茂东, 童伟华. 基于 L_0 优化的网格曲面特征线提取算法[J]. 计算机工程, 2019, 45(7): 251-257, 263.

英文引用格式: YANG Xiankang, PAN Maodong, TONG Weihua. Feature lines extraction algorithm on meshes based on L_0 optimization[J]. Computer Engineering, 2019, 45(7): 251-257, 263.

Feature Lines Extraction Algorithm on Meshes Based on L_0 Optimization

YANG Xiankang, PAN Maodong, TONG Weihua

(School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, China)

[Abstract] Based on the sparse distribution of feature lines on meshes, the paper proposes an optimized algorithm for feature lines extraction. For a given mesh, compute a scalar or a vector for each triangle as the input. Then, an optimized model is established for the measurement of the input so that the jumps on the edge of the meshes are as few as possible and the changes before and after optimization are minimized. An alternating direction optimization algorithm based on variable splitting technique and penalty function approach. In addition, a heuristic strategy is introduced to improve the sparsity of the solution, thus achieving higher quality of feature lines. Experimental results demonstrate that this method can effectively extract feature lines. Compared with some state-of-the-art methods such as Crest lines algorithm, variation algorithm and others, the proposed algorithm can improve the quality of feature lines extraction and the robustness for noisy data.

[Key words] meshes; feature lines extraction; L_0 optimization; variable splitting; alternating direction optimization algorithm
DOI: 10.19678/j.issn.1000-3428.0051288

0 概述

特征线是计算机图形学、计算机辅助设计及几何建模等领域广泛使用的高层次信息之一, 主要用于曲面设计和编辑, 网格去噪、参数化和变形等应用中。特征线虽易被人眼识别, 但其自动化提取却是一项具有挑战性的任务。特征是一个宽泛的概念, 抽象来说就是被观测物体的某种可度量的启发式属性。在网格处理中常用的特征线包括: 尖锐特征线, 脊线与谷线, 轮廓线等。这些线可通过数学语言来

刻画, 目前已出现一些特征线检测方法。尖锐特征线是指曲面法向上发生剧烈变化的曲线。文献[1]通过优化拟合误差及尖锐边的个数、位置来构造含有尖锐特征线的细分曲面控制网, 但该方法耗时较长。文献[2]将类似的技术用于重建点云的尖锐特征, 该方法虽易于实现, 但存在阈值参数选取困难、抗噪性差等问题。脊线与谷线是指曲面上沿主曲率方向取到极值的曲线。文献[3]提出先用光滑的隐式曲面来拟合网格模型, 然后利用网格顶点在隐式曲面的投影点来估计相应的曲率张量和曲率导数,

基金项目: 国家自然科学基金(11571338, 61877056); 浙江大学 CAD & CG 国家重点实验室开放课题(A1819)。

作者简介: 杨贤康(1993—), 男, 硕士研究生, 主研方向为计算机图形学; 潘茂东, 博士; 童伟华(通信作者), 副教授、博士。

收稿日期: 2018-04-23 **修回日期:** 2018-05-23 **E-mail:** yangxk@mail.ustc.edu.cn

最后通过追踪算法提取脊线与谷线。基于曲率极值的几何本质,文献[4]提出快速、有效的脊线与谷线提取算法,然而,这类算法的结果不可避免地依赖于高阶微分几何量的估计质量。轮廓线是指法向量与视线向量内积为零的曲线。文献[5]引入启发式轮廓线,即在曲面上沿视线方向的径向曲率为零,且在切平面上沿视线投影方向的导数大于零的点集。文献[6]采用主成分分析法和局部二次曲面拟合法对网格模型进行微分几何信息估算,并结合双阈值检测方法提出网格曲面特征线提取算法。文献[7]将图像处理中的 Mumford-Shah 模型推广到网格曲面上,由于其利用跟踪的方法提取特征线,对含噪声的网格特征线提取存在一定的困难。

稀疏表示与优化理论^[8]在信号处理、图像处理、机器学习和统计学等领域被广泛使用。近年来,有研究者尝试将该方法应用于数字几何处理中的若干问题^[9-10],例如保持特征去噪^[11-12]、网格分割^[13-14]、曲面重建^[15-17]等,取得了较好的效果。受此启发,在特征线具有稀疏分布的情况下,本文使用 L_0 优化方法提取网格曲面的特征线,使用 L_0 范数来抑制噪声的影响,以提高提取质量。

1 预备知识

1.1 三角网格表示

设 M 是给定的网格曲面,可由三元组 (V, E, T) 表示,其中, $V = \{v_1, v_2, \dots, v_{|V|}\}$ 为顶点集、 $E = \{e_1, e_2, \dots, e_{|E|}\}$ 为边集、 $T = \{t_1, t_2, \dots, t_{|T|}\}$ 为面集,顶点、边和面的个数分别记为 $|V|$ 、 $|E|$ 、 $|T|$ 。网格的每个面可以是三角形、四边形或其他多边形。为保证简单性与一般性,本文以三角网格为例。网格曲面的几何信息主要通过顶点的位置来定,拓扑信息由顶点、边和面之间的连接关系构成。若顶点 v 为边 e 的端点,则记为 $v < e$ 。类似地,若边 e 为三角形 t 的一条边,则记为 $e < t$ 。

在网格表示中,由于每条内边被 2 个三角形所共享,为方便离散梯度算子的描述,需要引入边 e 相对于其邻接三角形 t 的方向性概念。不失一般性,假定 M 是可定向的,其每个面 t 按逆时针方向给出外法向。对 M 中每条边 e 随机选定一个方向, $e < t$,若边 e 的定向和三角形 t 的定向一致,则记为 $\text{sgn}(e, t) = 1$; 否则 $\text{sgn}(e, t) = -1$ 。

1.2 离散梯度算子

设 $f(p)$ 是定义于网格 M 上的函数,在 M 的每个面 t 上取常数值,即:

$$f(p)|_{t_i} = f_i \in \mathbb{R}, \forall p \in t_i, i = 1, 2, \dots, |T|$$

则称 $f(p)$ 是网格 M 上的分片常函数。显然,函数 $f(p)$ 的梯度在每个面 t 的内部恒为零,而在面 t 的边 e 上可能取非零值,记为:

$$\nabla f(p)|_e = \begin{cases} \sum_{e < t} f_i \text{sgn}(e, t), & e \notin \partial M \\ 0, & e \in \partial M \end{cases}$$

其中, ∂M 表示网格 M 的边界。

不失一般性, M 上的分片常函数 $f(p)$ 可由向量 $f = (f_1, f_2, \dots, f_{|T|})^T \in \mathbb{R}^{|T|}$ 表示,此时离散梯度算子 $\nabla f = (\nabla f(p)|_{e_1}, \nabla f(p)|_{e_2}, \dots, \nabla f(p)|_{e_{|E|}})^T$ 可写成矩阵形式,即 $\nabla f \triangleq Gf, G \in \mathbb{R}^{|E| \times |T|}$,称 G 为离散梯度算子 ∇ 的矩阵表示。

1.3 单位外法向量空间

为叙述方便,除特别说明外,下文均以输入度量为单位外法向量进行说明。具体地,对于给定网格 M ,在每个面 t 上计算或读入一个单位外法向量 $n_t^{\text{in}} = (n_{t1}^{\text{in}}, n_{t2}^{\text{in}}, n_{t3}^{\text{in}})^T$ 。若记 $N^{\text{in}} = (n_1^{\text{in}}, n_2^{\text{in}}, \dots, n_{|T|}^{\text{in}})^T \in \mathbb{R}^{|T| \times 3}$,则 N^{in} 的每一列可视为 M 上的分片常函数。对于一般维数的输入度量,可做类似的处理。

在优化单位外法向量时,需要保持向量的单位性,引入 M 上的单位外法向量空间:

$$U_N = \{N: \|n_t\|_2 = 1, \forall t \in T\}$$

其中, $N = (n_1, n_2, \dots, n_{|T|})^T \in \mathbb{R}^{|T| \times 3}$, U_N 的特征函数记为:

$$\chi(N) = \begin{cases} 0, & N \in U_N \\ +\infty, & N \notin U_N \end{cases}$$

1.4 L_0 范数

在稀疏表示与优化理论中, L_0 范数常被用于描述向量的稀疏度。对于 n 维线性空间中的列向量 $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$, L_0 范数定义如下:

$$\|x\|_0 = \lim_{p \rightarrow 0} \|x\|_p = \lim_{p \rightarrow 0} \sum_{k=1}^n |x_k|^p = \#\{k: x_k \neq 0\}$$

其中,符号 $\#$ 表示取集合元素的个数,即 $\|x\|_0$ 等于向量 x 中非零分量的个数。特别地,当 $n = 1$ 时,若 $x_1 \neq 0$,有 $\|x\|_0 = 1$; 否则, $\|x\|_0 = 0$ 。

在本文中,对于 m 维行向量 $y = (y_1, y_2, \dots, y_m) \in \mathbb{R}^{1 \times m}$,有:

$$\|y\|_0 = \|(|y_1| + |y_2| + \dots + |y_m|)\|_0$$

即将行向量 y 视为一个整体,用于刻画非零向量。对于矩阵 $A = (a_{ij}) \in \mathbb{R}^{n \times m}$,将其写成 $A = (a_1, a_2, \dots, a_n)^T$ 的形式,则有:

$$\|A\|_0 = \sum_{i=1}^n \|a_i^T\|_0$$

其中, $a_i = (a_{i1}, a_{i2}, \dots, a_{im})^T, i = 1, 2, \dots, n$ 。

2 基于 L_0 优化的特征线提取算法

2.1 L_0 优化模型

对于给定的网格 M ,输入度量 $N^{\text{in}} = (n_1^{\text{in}}, n_2^{\text{in}}, \dots, n_{|T|}^{\text{in}})^T$,假设算法输出的度量为 $N = (n_1, n_2, \dots, n_{|T|})^T$, N 的梯度可表示为:

$$\nabla N = GN \triangleq (u_1, u_2, \dots, u_{|E|})^T \in \mathbb{R}^{|E| \times 3}$$

其中, $\mathbf{u}_i = (u_{i1}, u_{i2}, u_{i3})^T$, $i = 1, 2, \dots, |E|$, 稀疏度表示为 $\|\nabla N\|_0 = \sum_{i=1}^{|E|} \|\mathbf{u}_i^T\|_0$, 用于刻画输出度量在网格边上发生跃变的个数。稀疏度值越小表示特征的稀疏性越高。本文使用分片常数空间, 输入度量在每个面 t 上取常值或常向量。假定待提取的特征线位于网格的边上, 在特征线具有稀疏分布的情况下建立如下 L_0 优化模型:

$$\begin{aligned} \min_{N \in U_N} \|\nabla N\|_0 \\ \text{s. t. } \|N - N^{\text{in}}\|_F^2 < \varepsilon \end{aligned} \quad (1)$$

其中, $\|\cdot\|_F$ 表示矩阵的 Frobenius 范数, ε 为误差阈值。从稀疏表示与优化的角度来看, 该模型在允许的阈值范围内对输入度量进行扰动或去噪, 使得输出度量的梯度具有尽可能低的稀疏度。因此, 该模型不仅能检测出特征线, 还可以有效地抑制噪声的不良影响。容易看出, 通过引入参数 α 及 U_N 的特征函数 $\chi(N)$, 将式(1)改写成如下形式:

$$\begin{aligned} \min_N \|\nabla N\|_0 + \alpha \|N - N^{\text{in}}\|_F^2 + \chi(N) = \\ \min_{\{\mathbf{u}_i\}_{i=1}^{|E|}} \sum_{e=1}^{|E|} \|\mathbf{u}_e^T\|_0 + \alpha \sum_{t=1}^{|T|} \|\mathbf{n}_t - \mathbf{n}_t^{\text{in}}\|_2^2 + \chi(N) \end{aligned} \quad (2)$$

其中, $\alpha > 0$ 用于调节各项之间的权重, 值得注意的是, $\{\mathbf{u}_e\}$ 仅为中间变量, 可利用 $\{\mathbf{n}_t\}$ 表示。

2.2 模型求解

由于式(1)中含有非凸、非连续的 L_0 范数, 因此求解该模型是 NP 困难问题, 必须使用近似算法。首先, 通过变量分裂技术在每条边上引入辅助变量 $\mathbf{h}_e = (h_{e1}, h_{e2}, h_{e3})^T$, 将式(1)写成如下等价形式:

$$\begin{aligned} \min_{\{\mathbf{n}_t, \mathbf{h}_e\}} \sum_{e=1}^{|E|} \|\mathbf{h}_e^T\|_0 + \alpha \sum_{t=1}^{|T|} \|\mathbf{n}_t - \mathbf{n}_t^{\text{in}}\|_2^2 + \chi(N) \\ \text{s. t. } \mathbf{h}_e = \mathbf{u}_e, \quad \forall e \in E \end{aligned} \quad (3)$$

然后, 通过二次罚函数方法将其转化成无约束优化问题, 即引入参数 β 将式(1)进一步近似为如下形式:

$$\begin{aligned} \min_{\{\mathbf{n}_t, \mathbf{h}_e\}} \sum_{e=1}^{|E|} \|\mathbf{h}_e^T\|_0 + \alpha \sum_{t=1}^{|T|} \|\mathbf{n}_t - \mathbf{n}_t^{\text{in}}\|_2^2 + \\ \beta \sum_{e=1}^{|E|} \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 + \chi(N) \end{aligned} \quad (4)$$

其中, β 为罚因子。当 β 不断增大时, 式(4)的解逐步逼近式(3)的解。

最后, 利用交替方向优化方法来求解式(4), 即不断交替固定变量 $\{\mathbf{n}_t\}$ 与 $\{\mathbf{h}_e\}$, 逐步迭代求解直到满足预先设定的停止准则为止, 具体见算法1。

算法1 L_0 优化模型求解算法

输入 $N^{\text{in}}, \alpha, \beta_0, \beta_{\max}, \rho, \varepsilon$

输出 N

步骤1 $k \leftarrow 0, \beta \leftarrow \beta_0, N_0 \leftarrow N^{\text{in}}$ 。

步骤2 重复以下步骤直到 $\frac{\|N_k - N_{k-1}\|_F^2}{|F|} < M$

或者 $\beta > \beta_{\max}$:

1) 固定 N_k , 求解 H -子问题(5)得到 H_k 。

2) 固定 H_k , 求解 N -子问题(8)得 N_{k+1} 。

3) $\beta \leftarrow \rho\beta, k \leftarrow k+1$ 。

步骤3 返回 $N \leftarrow N_k$ 。

在算法1中, N^{in} 是输入网格的初始法向, α 是优化模型中的权值, β_0, β_{\max} 分别是罚因子的初始值以及最大值, ρ 是增长因子, ε 是终止误差, N 是优化后的网格面法向。 H -子问题与 N -子问题的具体求解方法如下:

1) H -子问题。当变量 $N = \{\mathbf{n}_t\}$ 固定时, 式(4)可以简化为:

$$\min_{\{\mathbf{h}_e\}} \sum_{e=1}^{|E|} \|\mathbf{h}_e^T\|_0 + \beta \sum_{e=1}^{|E|} \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 \quad (5)$$

上述优化问题可以进一步分解成若干个独立的子问题, 具体如下:

$$\sum_{e=1}^{|E|} \min_{\mathbf{h}_e} \|\mathbf{h}_e^T\|_0 + \beta \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 \quad (6)$$

对于这些子问题, 有如下命题:

命题1 式(6)的子问题具有如下显式解:

$$\mathbf{h}_e = \begin{cases} (0, 0, 0)^T, & \|\mathbf{u}_e\|_2^2 \leq 1/\beta \\ \mathbf{u}_e, & \text{其他} \end{cases} \quad (7)$$

证明 记 $E(\mathbf{h}_e) = \|\mathbf{h}_e^T\|_0 + \beta \|\mathbf{u}_e - \mathbf{h}_e\|_2^2$, 下面分2种情形进行讨论:

情形1 当 $\|\mathbf{u}_e\|_2^2 \leq 1/\beta$ 时, 若 $\mathbf{h}_e \neq (0, 0, 0)^T$, 则 $E(\mathbf{h}_e) = 1 + \beta \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 \geq \beta \|\mathbf{u}_e\|_2^2 + \beta \|\mathbf{u}_e - \mathbf{h}_e\|_2^2$; 若 $\mathbf{h}_e = (0, 0, 0)^T$, 则 $E(\mathbf{h}_e) = \beta \|\mathbf{u}_e\|_2^2$ 。容易看出, 当 $\mathbf{h}_e = (0, 0, 0)^T$ 时, 函数 $E(\mathbf{h}_e)$ 取最小值。

情形2 当 $\|\mathbf{u}_e\|_2^2 > 1/\beta$ 时, 有 $\mathbf{u}_e \neq (0, 0, 0)^T$ 。若 $\mathbf{h}_e \neq (0, 0, 0)^T$, 则 $E(\mathbf{h}_e) = 1 + \beta \|\mathbf{u}_e - \mathbf{h}_e\|_2^2$ 。在此情形下, 当 $\mathbf{h}_e = \mathbf{u}_e$ 时, $E(\mathbf{h}_e)$ 取最小值1; 若 $\mathbf{h}_e = (0, 0, 0)^T$, 则 $E(\mathbf{h}_e) = \beta \|\mathbf{u}_e\|_2^2 > 1$ 。容易看出, 当 $\mathbf{h}_e = \mathbf{u}_e$ 时, 函数 $E(\mathbf{h}_e)$ 取最小值。

综上所述, 证毕。

2) N -子问题。当变量 $H = \{\mathbf{h}_e\}$ 固定时, 式(4)可以简化为:

$$\min_{\{\mathbf{n}_t\}} \alpha \sum_{t=1}^{|T|} \|\mathbf{n}_t - \mathbf{n}_t^{\text{in}}\|_2^2 + \beta \sum_{e=1}^{|E|} \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 + \chi(N) \quad (8)$$

其中, $\chi(N)$ 为非线性项, 本文先将其忽略, 即将式(4)简化为式(9)的二次优化问题。

$$\min_{\{\mathbf{n}_t\}} \alpha \sum_{t=1}^{|T|} \|\mathbf{n}_t - \mathbf{n}_t^{\text{in}}\|_2^2 + \beta \sum_{e=1}^{|E|} \|\mathbf{u}_e - \mathbf{h}_e\|_2^2 \quad (9)$$

然后将所得的 $\{\mathbf{n}_t\}$ 单位化来进行处理。若记

$H = (\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{|E|})^T \in \mathbb{R}^{|E| \times 3}$, $N_i = N(:, i) \in \mathbb{R}^{|T|}$, $N_i^{\text{in}} = N^{\text{in}}(:, i) \in \mathbb{R}^{|T|}$, $H_i = H(:, i) \in \mathbb{R}^{|E|}$, $i = 1, 2, 3$, 则式(9)可写为:

$$\min_{\{n_i\}} \sum_{i=1}^3 \{ \alpha [N_i^T N_i - 2 N_i^T N_i^{\text{in}} + (N_i^{\text{in}})^T N_i^{\text{in}}] + \beta (N_i^T G^T G N_i - 2 N_i^T G^T H_i + H_i^T H_i) \} \quad (10)$$

其中, $G \in \mathbb{R}^{|E| \times |T|}$ 为离散梯度算子 ∇ 的矩阵表示。因此, 式(9)的解满足如下线性方程组:

$$(\alpha I + \beta G^T G) N_i = \alpha N_i^{\text{in}} + \beta G^T H_i, \quad i=1, 2, 3 \quad (11)$$

其中, I 表示 $|T|$ 阶单位矩阵。可以发现, 上述线性方程组的系数矩阵是稀疏、对称且正定的, 但每次迭代随 β 变化而改变, 因此采用双共轭梯度算法^[18]来求解。

2.3 稀疏性提升算法

在式(2)中, 参数 α 可用于调节特征线的稀疏度与逼近误差之间的权重。当特征线的稀疏度较高时, 许多伪特征边会被提取出来。此时, 可通过减小 α 来降低特征线的稀疏度, 消除多余的伪特征边。但是, 该方法在实践中并不能得到理想的结果, 其主要原因是减小 α 相当于增大了式(1)中的 ε , 使得输出度量的可扰动范围扩大, 即优化模型可行解的搜索区域变大, 从而使 L_0 优化模型的求解变得更为困难。因此, 本文提出一种迭代的贪婪策略来逐步降

低 L_0 优化问题的解的稀疏度, 具体地, 在进行第 $l+1$ 次迭代时, 求解如下改进的 L_0 优化模型:

$$\min_N \|\nabla N\|_0 + \alpha \|N - N^l\|_F^2 + \chi(N), \quad l=1, 2, \dots \quad (12)$$

其中, N^l 是第 l 次迭代所得的结果, 详见算法 2。

算法 2 稀疏性提升算法

输入 $N^{\text{in}}, \alpha, \beta_0, \beta_{\max}, \rho > 1, \varepsilon$, 迭代次数 L

输出 最终的特征线 Γ

步骤 1 $l \leftarrow 0, N^0 \leftarrow N^{\text{in}}$ 。

步骤 2 重复以下步骤, 直到 $E^l > \|\nabla N^{l-1}\|_0$ 或者 $l \geq L$ 。

1) 基于 $N^l, \alpha, \beta_0, \beta_{\max}, \rho, \varepsilon$, 调用算法 1 求解式(12), 得到解 N^{l+1} 。

2) $E^{l+1} \leftarrow \|\nabla N^{l+1}\|_0 + \alpha \|N^{l+1} - N^l\|_F^2, l \leftarrow l+1$ 。

3) 采用第 2.4 节的方法提取网格曲面的特征线 Γ 。

容易看出, 利用算法 2 计算出的式(12)的解序列 $\{N^l\}$ 满足 $\|\nabla N^{l+1}\|_0 < \|\nabla N^l\|_0$, 即解的稀疏度是递减的。通过选取合适的迭代次数 L , 可有效提高特征线的提取质量, 提取结果如图 1 所示。

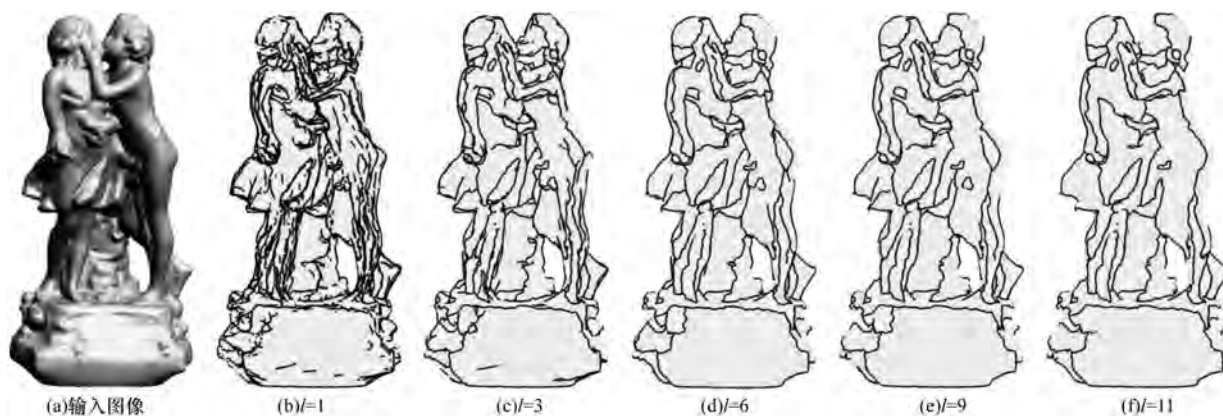


图 1 迭代次数不断增加时的特征线提取效果

2.4 特征线提取

根据式(1), 本文算法待提取的特征线出现在优化后度量值 N 发生跃变的位置, 即网格 M 满足如下条件的边集:

$$\Gamma = \{e | \|u_e^T\|_0 \neq 0, e \in E\}$$

其中, $u_e^T = \nabla N(e, :)$, $e = 1, 2, \dots, |E|$ 。然而, 由于本文采用近似算法求解式(1), 且受到数值计算误差的影响, 不能直接通过比较 $\|u_e^T\|_0$ 与 0 来判别特征边。因此, 在算法中, 使用如下标准来提取特征边:

$$\Gamma = \{e | \|u_e^T\|_2 > \delta, e \in E\}$$

其中, δ 为常数。本文所有实验结果均在 $\delta = 0.2$ 情形下取得。

3 实验结果与分析

本文使用 C++ 语言和 Matlab 实现算法研究, 下文实验均在一台配置 Intel Core i5 3.1 GHz CPU 及 8 GB RAM 的计算机上进行。

3.1 参数选取

在算法的执行过程中, 有一些参数需要用户设置, 主要包括 $\alpha, \beta_0, \beta_{\max}, \rho, \varepsilon, L$ 等。在本文实验中, 部分参数取默认值, 例如 $\beta_0 = 0.1, \rho = 1.4, \varepsilon = 1.0e-7$ 。其余参数依据具体情况进行设置, 通常都在一定的范围内选择, 例如, $\alpha \in [2, 40], \beta_{\max} \in [5\ 000 \beta_0, 20\ 000 \beta_0], L \in [1, 12]$, 具体数值见表 1。

表 1 实验参数设置与运行时间

网格模型	$ V $	$ T $	α	β_{\max}	L	运行时间/s
Sculpture	87 494	174 996	5.0	2 000	11	1 335
Fusee	62 088	124 244	10.0	1 000	10	498
Lucy	131 455	262 906	10.0	1 000	10	1 445
Trim-star	23 723	47 446	2.6	1 000	5	251
Dodecahedron	30 722	61 440	2.0	1 000	1	70
Camel	9 770	19 156	3.3	1 000	12	64
Santa	75 781	151 558	40.0	500	8	570
Horse	19 851	39 698	12.5	800	1	40
Dinosaur	126 614	253 224	10.0	1 000	10	1 321
Dancing Children	724 678	1 449 380	15.0	1 000	10	6 832

3.2 实验结果

为验证本文算法的有效性,对不同类型的网格模型进行测试,包括干净的网格曲面、带噪声的网格曲面、带边界的网格曲面以及带颜色属性的网格曲面。此外,将本文算法与 RTSC-1.5 算

法^[5]、Crest lines 算法^[4]和变分算法^[7]进行对比。为确保公平性,实验中尽可能按照论文作者提供的建议选取相应参数,以获得最佳实验结果。不同网格模型的提取结果如图 2 ~ 图 7 所示。

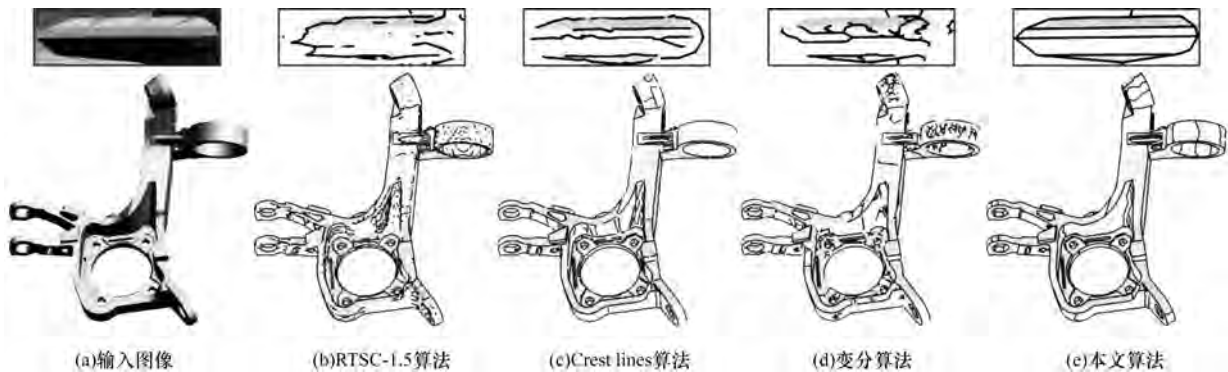


图 2 4 种算法对干净的 Fusee 网格模型提取特征线结果比较



图 3 4 种算法对干净的 Lucy 网格模型提取特征线结果比较

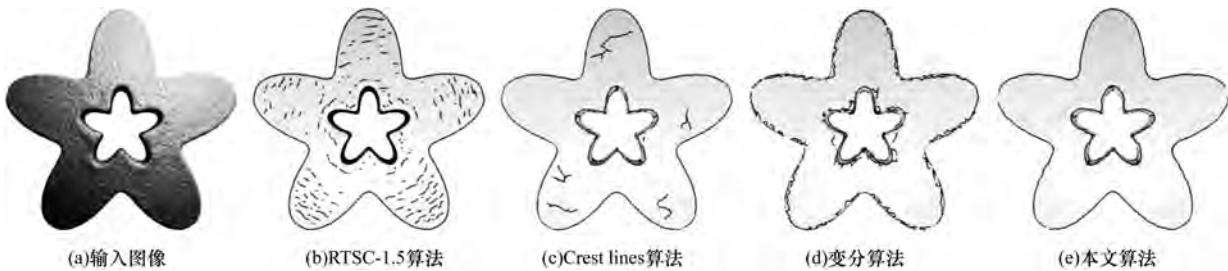


图 4 4 种算法对带 $\sigma = 0.05 \bar{\epsilon}$ 高斯噪声的 Trim-star 网格模型提取特征线结果比较

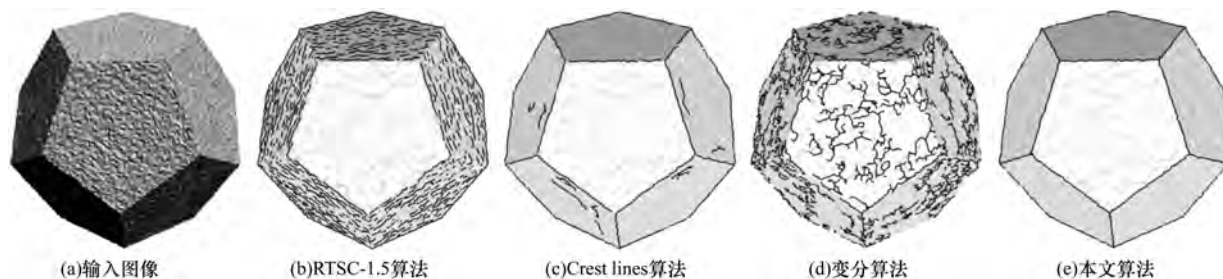


图5 4种算法对带 $\sigma=0.17\bar{e}$ 高斯噪声的Dodecahedron网格模型提取特征线结果比较

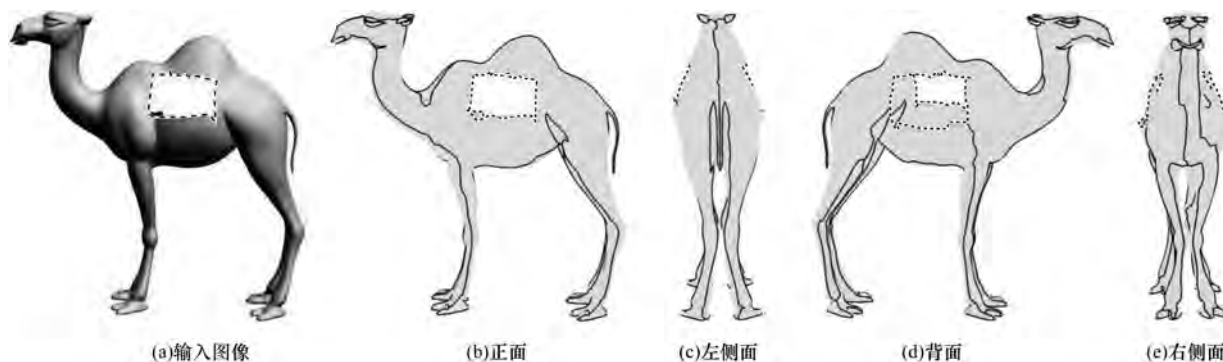


图6 本文算法对带边界的Camel网格模型特征线提取结果



图7 本文算法对带RGB颜色属性的Santa网格模型与Horse网格模型特征线提取结果

具体分析如下:

1) 干净的网格曲面

图2、图3分别是对于Fusse网格模型和Lucy网格模型采用4种不同算法提取特征线的结果。从局部放大图中容易看出,本文算法提取的特征线质量明显优于其他算法。与其他3种算法不同,本文算法无须利用跟踪的方法提取特征线,因此可避免特征线的后处理过程。

2) 带噪声的网格曲面

为了测试噪声对本文算法结果的影响,按如下步骤人工合成带噪声的网格模型:

步骤1 随机产生向量集 $\{d_v = (d_{v1}, d_{v2}, d_{v3})^T | v \in V\}$,其中,随机变量 d_{v1} 、 d_{v2} 、 d_{v3} 相互独立且均服从高斯分布 $N(\mu, \sigma^2) = (\sqrt{2\pi}\sigma)^{-1} \exp^{-(x-\mu)^2/2\sigma^2}$ 。本文实验 μ 恒取0。

步骤2 更新顶点位置 $\tilde{p}_v = p_v + d_v, \forall v \in V$,其

中, p_v 为顶点 v 的位置坐标。

步骤3 按更新后的顶点位置重新计算网格的输入度量 N^{in} 。

图4为对带 $\sigma=0.05\bar{e}$ 的高斯噪声的Trim-star网格模型进行特征线提取的结果,其中, \bar{e} 为网格 M 的平均边长。容易看出,在处理带噪声的数据时,本文算法的准确性明显高于其他3种算法。图5对带 $\sigma=0.17\bar{e}$ 高斯噪声的Dodecahedron网格模型的提取结果,其进一步显示了本文算法对噪声的鲁棒性。

3) 带边界的网格曲面

图6为本文算法对带边界的网格模型进行特征线提取的结果,其中虚线表示网格的边界。从图6可以看出,本文算法可有效地处理带边界或带洞的网格模型,且网格缺失的部分并未对本文算法的结果造成明显的影响。

4) 带颜色属性的网格曲面

为验证本文算法的通用性,对带 RGB 颜色属性的网格模型进行测试。在实验中输入的度量改为网格每个面上定义的 RGB 颜色信息,如图 7(a) 和图 7(c) 所示。通过本文的 L_0 优化算法,优化后的图像及相应的特征线见图 7(b) 和图 7(d)。可以看出,本文算法对于一般的输入度量也能取得较满意的结果。

3.3 算法运行时间

本文算法主要由 2 个嵌套的循环构成,其中外

循环为稀疏性提升的迭代过程,内循环为求解 L_0 模型的交替方向优化算法。容易看出,求解一系列大型稀疏线性方程组,即式(11),是本文算法主要的性能瓶颈,占用了大部分运行时间。为了测试算法的性能,对大型的网格模型进行对比实验,结果如图 8 所示。表 1 展示了本文算法对于不同模型所需的计算时间,可以看出,本文算法的运行速度还不够理想,其部分原因是本文算法使用 C++ 语言调用 Matlab 来求解稀疏线性方程组,未来可对如何加速进行研究。



图 8 本文算法对大型网格模型 Dinosaur 和 Dancing children 的特征线提取结果

4 结束语

由于特征线具有稀疏分布,本文提出一个基于 L_0 优化的特征线提取算法。对于给定的输入度量,通过求解 L_0 优化问题得到优化后的度量,使之在网格边上发生跃变的总数极小化,并使优化后的值与优化前的值之间的差异尽可能小。给出基于变量分裂技术与罚函数方法的交替方向优化算法,并引入一种迭代策略来提升 L_0 优化问题解的稀疏性,以取得更高质量的特征线。实验结果表明,该算法能有效提取网格曲面的特征线,与变分算法、Crest lines 算法、RTSC-1.5 算法相比,该算法提取质量更高,对带噪声的数据鲁棒性更好。然而,本文主要使用参数 α 来调节特征边的稀疏度与逼近误差之间的权重,未考虑特征线尺度的问题,同时本文算法的运行速度不够理想,因此,引入层次的优化模型并研究收敛速度更快的 L_0 优化求解算法将是下一步的研究方向。

参考文献

- [1] HOPPE H, DEROSE T, DUCHAMP T, et al. Piecewise smooth surface reconstruction [C]//Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques. New York, USA: ACM Press, 1994:295-302.
- [2] OHTAKE Y, BELYAEV A, ALEXA M, et al. Multi-level partition of unity implicits [C]//Proceedings of

- SIGGRAPH' 03. New York, USA: ACM Press, 2003: 463-470.
- [3] OHTAKE Y, BELYAEV A, SEIDEL H P. Ridge-valley lines on meshes via implicit surface fitting [C]//Proceedings of SIGGRAPH' 04. New York, USA: ACM Press, 2004:609-612.
- [4] YOSHIZAWA S, BELYAEV A, YOKOTA H, et al. Fast, robust, and faithful methods for detecting crest lines on meshes [J]. Computer Aided Geometric Design, 2008, 18(8):545-560.
- [5] DECARLO D, FINKELSTEIN A, RUSINKIEWICZ S, et al. Suggestive contours for conveying shape [C]//Proceedings of SIGGRAPH' 03. New York, USA: ACM Press, 2003:848-855.
- [6] 史皓良, 吴禄慎, 余喆琦. 散乱点云数据特征信息提取算法[J]. 计算机工程, 2017, 43(8):279-283.
- [7] TONG Weihua, TAI Xuecheng. A variation approach for detecting feature lines on meshes [J]. Journal of Computational Mathematics, 2016, 34(1):87-112.
- [8] ELAD M. Sparse and redundant representations: from theory to applications in signal and image processing[M]. Berlin, Germany: Springer, 2010.
- [9] XU Li, LU Cewu, XU Yi, et al. Image smoothing via L_0 gradient minimization[J]. ACM Transactions on Graphics, 2011, 30(6):1-12.
- [10] HE Lei, SCHAEFER S. Mesh denoising via L_0 minimization[J]. ACM Transactions on Graphics, 2013, 32(4):1-50.
- [11] WANG Ruimin, YANG Zhouwang, LIU Ligang, et al. Decoupling noise and features via weighted L_1 -analysis compressed sensing [J]. ACM Transactions on Graphics, 2014, 33(2):1-12.

(下转第 263 页)

(上接第 257 页)

- [12] ZHANG Huayan, WU Chunlin, ZHANG Juyong, et al. Variational mesh denoising using total variation and piecewise constant function space[J]. IEEE Transactions on Visualization and Computer Graphics, 2015, 21(7): 873-886.
- [13] ZHANG Juyong, ZHENG Jianmin, WU Chunlin, et al. Variational mesh decomposition[J]. ACM Transactions on Graphics, 2012, 31(3): 1-15.
- [14] LIU Xiuping, ZHANG Jie, LIU Risheng, et al. Low-rank 3D mesh segmentation and labeling with structure guiding[J]. Computers and Graphics, 2015, 46: 99-109.
- [15] AVRON H, SHARF A, GREIF C, et al. L1-sparse reconstruction of sharp point set surfaces[J]. ACM Transactions on Graphics, 2010, 29(5): 1-12.
- [16] XIONG Shiyao, ZHANG Juyong, ZHENG Jianmin, et al. Robust surface reconstruction via dictionary learning[J]. ACM Transactions on Graphics, 2014, 33(6): 1-3.
- [17] PAN Maodong, TONG Weihua, CHEN Fala. Compact implicit surface reconstruction via low-rank tensor approximation[J]. Computer-Aided Design, 2016, 78: 158-167.
- [18] GOLUB G H, LOAN C F V. Matrix Computation[M]. 4th ed. Washington D. C., USA: Johns Hopkins University Press, 2013.

编辑 樊丽娜