基于三维点云模型的特征线提取算法*

刘 倩[1], 耿国华[1], 周明全[1,2], 赵璐璐[1], 李姬俊男[1]
(1. 西北大学 信息科学与技术学院 计算机系, 西安 710127; 2. 北京师范大学 信息科学与技术学院计算机系, 北京 100875)

摘 要: 针对以往算法存在无法区分尖锐和非尖锐特征点、提取的特征点与视角有关、特征点未连线等问题,提出一种基于高斯映射和曲率值分析的三维点云模型尖锐特征线提取算法。该算法先进行点云数据点的离散高斯映射,并将映射点集聚类;然后使用自适应迭代过程得到两个或多个面的相交线上曲率值和法向量发生突变的尖锐特征点,这些点与视角无关;最后,用改进的特征折线生长算法,将特征点连接,得到光顺特征线。实验证明,该算法具有良好的自适应性、抗噪性和准确性,是一种有效的三维模型特征线提取算法。

关键词: 高斯映射; 曲率计算; 点聚类; 自适应迭代; 折线生长
中图分类号: TP391.41 **文献标志码:** A **文章编号:** 1001-3695(2013)03-0933-05
doi:10.3969/j.issn.1001-3695.2013.03.076

Algorithm for feature line extraction based on 3D point cloud models

LIU Qian[1], GENG Guo-hua[1], ZHOU Ming-quan[1,2], ZHAO Lu-lu[1], LI Ji-jun-nan [1]
(1. *Dept. of Computer, School of Information Science & Technology, Northwest University, Xi'an* 710127, *China*; 2. *Dept. of Computer, School of Information Science & Technology, Beijing Normal University, Beijing* 100875, *China*)

Abstract: This paper proposed a sharp feature line extraction algorithm of 3D point cloud models based on Gaussian map and curvature value analysis, which aimed to solve the problems that previous algorithms existed, including could not distinguish sharp and non-sharp feature points, the extracted feature points were relative to perspective, or feature points weren't connected. First, this algorithm conducted discrete Gaussian map for point cloud data, and clustered these mapping point sets. Then it used an adaptive iterative procedure to get sharp feature points, these points mainty located on the intersection line of two or more point cloud surfaces, where curvature value or normal vector suddenly changed and they were independent of perspective. Finally, it used an improved feature polyline propagation algorithm, connected the feature points, and got smoothing feature lines. Experiments show that the algorithm has good adaptability, noise immunity and accuracy, it is an effective feature extraction algorithm for 3D models.

Key words: Gaussian map; curvature computation; points clustering; adaptive iterative; feature polyline propagation

0 引言

三维模型的特征线提取技术是近年来计算机可视化、数字几何处理、CAD 逆向工程等领域研究的热点课题。因为模型特征,尤其是边界特征和若干表面相交线特征,已被广泛用于曲面重建、数据分割、模型简化、曲线匹配和拼接等几何处理领域。

对于三维点云模型,其特征曲线提取过程就是对模型上的数据点进行分析计算,找出其中的特征点,并连接构成光滑特征曲线。针对数据点分布不均匀、点间缺乏拓扑连接信息的点云数据,许多学者都提出了相关的特征线提取算法:

a)基于投影的方法。Daniels 等人[1]在每个点的邻域内使用鲁棒移动最小二乘法(RMLS)进行曲面拟合,并计算当前点在曲面上的投影残差值,将投影残差值较大的点定义为潜在特征点。但由于算法是基于 RMLS 的, 时间代价较高。Pang 等人[2]根据局部表面拟合多项式计算出点云模型上每个点的主曲率和主方向,用绝对值较大的主曲率标志出谷脊潜在特征点,通过将得到的特征点投影到邻域点构成的主轴上,平滑潜在特征点,但该算法对稀疏点云模型有很好的计算效果。

b)基于曲率值、法向量等几何特征的方法。Mrigot 等人[3]通过对点云 Voronoi 单元的卷积协方差矩阵进行计算,得到点云曲面上某局部范围内的主曲率、主方向以及尖锐边的夹角,进而进行特征点的判断和提取,但计算复杂度较高。王丽辉等人[4]通过计算点云模型中某点与其邻域内其他点间距离的平均值、该点的曲率、法向值、该点法向和邻域点法向的夹角以及点云数据点的密度来进行特征检测,但该方法需要对相关参数设置和调整。

c)基于主成分分析的方法。Demarsin 等人[5]先用 PCA 方

收稿日期: 2012-07-26; **修回日期:** 2012-09-05 **基金项目:** 国家自然科学基金资助项目(61172170);国家教育部博士点基金项目(200806970014);陕西省自然科学基金资助项目(2011JQ8001,2010JQ8011);虚拟现实应用教育部工程研究中心开放基金资助项目(MEOB-NUEVRA200903)

作者简介: 刘倩(1984-),女,甘肃舟曲人,主要研究方向为三维可视化技术、图形图像处理(1141823100@qq.com);耿国华(1955-),教授,博导,主要研究方向为图形图像处理、可视化技术、人工智能;周明全(1953-),教授,博导,主要研究方向为计算机可视化技术、生物特征识别、中文信息处理;赵璐璐(1989-),陕西西安人,硕士研究生,主要研究方向为图形图像处理、可视化技术;李姬俊男(1987-),博士研究生,主要研究方向为可视化技术、三维模型处理.

法计算点云模型各点的法向量,在局部邻域内对法向量变化大的点用区域生长算法,得到数据的初始聚类,然后对不同聚类的边缘点云数据进行分割,并构造最小生成树得到封闭特征线,但该算法仅对均匀分布的数据有效。Pauly 等人[6]通过改变局部邻域的半径, 多尺度地处理含有噪声的模型,但该方法只适合于处理稀疏点云模型,对微细特征的提取无能为力。

d)基于视角的特征线提取方法。Decarlo 等人[7]使用基于视线向量的主观轮廓线的概念,提取三维模型形状的细节特征,但该算法只能找到可见的凸而非凹区域内的主观轮廓线。

本文使用基于高斯映射和曲率值判断的点云模型尖锐特征提取方法,可以弥补之前方法较难区分尖锐和非尖锐特征、提取的特征点数量大、噪声多、只提取与视角相关的特征等缺陷,且本文特征点提取算法具备以下两个特点:a)完备性,算法可检测出模型的所有尖锐特征点,即两个或者多个面的相交线上曲率值和法向量发生突变的特征点;b)准确性,指定了特征点所在的具体位置,不选择靠近但不属于尖锐特征边上的点。具体过程:a)对点云数据中的每个点进行离散高斯映射,并将映射点集进行基于距离的凝聚型层次聚类;b)根据聚类结果和曲率值分析进行特征点判断,并用自适应的迭代过程对算法的参数进行选择,得到最优的特征点集;c)使用改进的基于 PCA 分析的特征折线生长算法,将特征点连接,得到光顺的特征线。

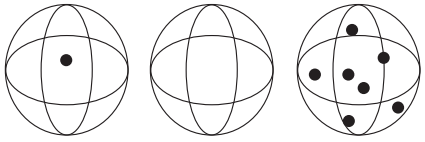
1 高斯映射理论基础

高斯映射已被广泛应用于曲面识别[8]、点云拟合[9]、点云分割[10]等领域。高斯映射的定义为,对一个分段光滑的空间三维曲面 Σ,计算其上所有点的单位法矢量,并将法矢的起点平移到单位球 S^2 的球心上,则法矢量的终点落在单位球面上,形成投影点的集合称为该曲面 Σ 的高斯图,曲面上的点向球面上映射的过程,称为高斯映射 $G_p: \Sigma \to S^2$,该单位球称为高斯球。对于一个具体的 3D 实物模型,其外表面常由多个曲面构成,各曲面高斯图的并集构成该模型整体的高斯图。在高斯球上,通常存在着三种不同分布的映射点:

a)点型聚类的零维分布点(平面上的点)。对平面上的点,其法矢量是一个常数,且其高斯曲率、最大、最小主曲率均为 0,通过高斯映射,平面上的点在高斯球上重合于一点,属于零维分布,如图 1(a)所示。

b)线型聚类的一维分布点。通常圆锥面、圆柱面、一般拉伸面上的点都称为抛物点。对任何一个抛物点 p_i,其法矢量 n_{p_i} 和轴线 l 总是成固定夹角 α,即 $\cos\alpha = n_{pi} \cdot l = c$,所有曲面上的抛物点通过高斯映射,会在高斯球上形成一条二次圆弧曲线,即高斯映射像呈线形聚类、一维分布,如图 1(b)所示。

c)面形聚类的二维分布点。通常球面、圆环面、一般直纹面上的点,通过高斯映射在高斯球上均匀分布,形成面形二维分布的点,如图 1(c)所示。



(a)点型聚类 (b)线型聚类 (c)面型聚类

图1 高斯映射点的分布

2 基于高斯映射的特征提取

对输入的 3D 离散点云模型数据,求出该模型的尖锐特征点子集,可以通过以下步骤来完成:

a)数据组织,计算寻找点集中每个点的邻接点,使用 k-d 树结构搜索点云数据,并用 Riemann 图存储当前点的 k-近邻。

b)对点云模型每个点及其邻域内的点,使用高斯映射的基本原理,计算其在高斯球上的离散映射点集。

c)特征检测,根据 b)的计算结果,使用测地距离准则来对高斯映射点集聚类,通过判断最终保留簇的个数、曲率值的分析,来判断当前点是否为尖锐特征点。

d)通过自适应的迭代过程,不断修改结果簇之间的最小距离 δ 和邻域大小 k 这两个参数,直到满足迭代的终止条件且找到最优化表示的特征点集。

2.1 数据组织

对 3D 模型点云 $P = \{p_i = (x_i, y_i, z_i) \in R^3 \mid i = 1, 2, \cdots, N\}$,寻找其中每个采样点 p_i 的局部 K-近邻邻域 $N_{p_i} = \{(x_j, y_j, z_j) \in R^3 \mid j = 1, 2, \cdots, k\}$,本文使用基于邻接表结构的无向 Riemann 图存储当前点的 K-近邻。若已知点云中所有点的索引为:index = 1, 2, …, N,对每个点用 k-d 树法搜索该点的邻域数据空间,找出到当前点 p_i 的欧氏距离 $\text{dist} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \mid (x_j, y_j, z_j) \in N_{p_i}$ 最近的 k 个点,并按距离值 dist 由小到大的顺序进行排序,依次填入如图 2 所示的邻接表(Riemann 图存储)结构中。

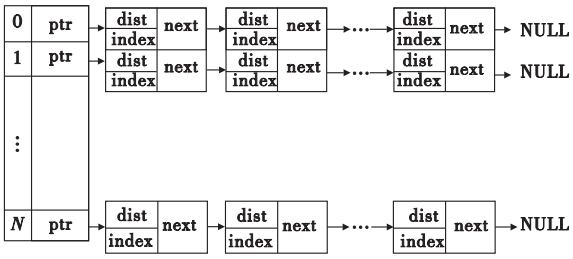


图2 Riemann 图存储结构

2.2 离散高斯映射

使用高斯映射的基本原理,在 3D 点云模型中取当前采样点 p_i,然后以 p_i 作为三角形的一个顶点与其邻域内的所有其他点构成 $k(k-1)/2$ 个三角形,如图 3 所示。设这些三角形构成集合 $T = \{\Delta_{mn} = (p_i, p_m, p_n) \in R^3 \mid 1 \leqslant m, n \leqslant k\}$,其中每个三角形 Δ_{mn} 的法向量为:$N_{mn} = p_i p_m \times p_i p_n$ 且 $N_{mn} = -N'_{nm}$。这些法向量与以 p_i 为球心的单位高斯球 S^2 的交点为高斯映射点,这样就省去了将法向量的起点平移到单位球原点进行高斯映射的步骤,则点 p_i 的 k 邻域内点的离散高斯映射定义为

$$G_p: T \to S^2$$

$$\Delta_{mn} \to x_{mn} = p_i + \frac{N_{mn}}{\| N_{mn} \|} \tag{1}$$

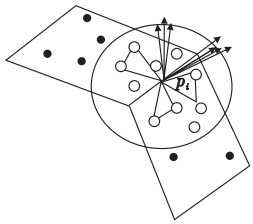


图3 局部邻接点的法向量计算

x_{mn} 为 p_i 点在高斯球上的映射点,同时,对上述三角形集 T,设其中所有三角形构成了连通区域 M 且面积为 Φ,将该区域使用高斯映射后得到高斯图 $G(M)$(其面积为 Φ^*),则曲面在点 p_i 邻近处的弯曲程度可以用 Φ^*/Φ 的比值来表示。曲面在已知点处的弯曲程度,可以用该比值当原曲面小区域的面积 Φ 收缩到点 p_i 时的极限来衡量[11],即 p_i 处曲面的高斯曲率的绝对值为

$$|k_G(p_i)| = \lim_{\Phi \to p_i}(\Phi^*/\Phi) = d(\Phi^*)/d(\Phi) \tag{2}$$

所以高斯映射在一定程度上反映了曲面上点的曲率值的大小,通过高斯映射能检测出曲面上弯曲程度较大的区域,即当原点云中某点邻域表面面积大且弯曲程度也较大大,相应的高斯球面映射区域的面积较大,映射点分布较广,可见曲率值的计算与邻域大小(面积)有关。若邻域太大,则违反所有邻接点处于同一连通域的条件,且远离采样点的其他点会对特征判断产生影响;而邻域太小,将会对噪声不敏感,影响特征点的判断;同时邻域 k 值的选取还取决于点云数据模型的密度、分布均匀性,当点云数据密集时可取小一些,为近似对某点 p_i 邻域均匀采样,将到 p_i 点的距离远小于邻域内点其他点到 p_i 的平均距离的点排除,使得它们不参与高斯映射计算;最后,参考 Pauly 等人[12]设置的检测非有效邻域的准则,通过对多个实验点云模型计算,确定邻域大小 k 的取值一般取为 8 ~ 32。此外还需注意的是,当曲面弯曲的厉害时,其法向量变化也大,因而相应的高斯映射图的面积也较大。所以通过高斯映射也能检测出法向发生变化的曲面点,具体取决于映射点在球面上距离值的大小,可以设定一个距离阈值用于对映射点进行聚类,这点将在 2.3 节中介绍。

2.3　特征检测

对上述 p_i 点的邻域三角形集 T 的高斯映射聚类行为的曲率值变化情况进行分析,可检测出点云数据的尖锐特征点。

2.3.1　映射点聚类

对点云模型上的每个采样点 p_i 的 K-近邻邻域进行高斯映射后,需要进行映射点的聚类,此过程中常出现反向簇,即集合 T 中每个三角形法向量的朝向不确定,每个高斯球上的映射点 x_{mn},在球的另一侧对应存在着对称的映射点 x_{nm},如图 4 所示,要将 3D 模型上每个点的高斯映射点聚类成簇,需选择一个合适的距离度量。本文先连接以点 p_i 为球心高斯球上所有对称的映射点,然后以两条对称点对连线的夹角作为距离度量,可证明高斯映射中任意两条连线间的夹角和高斯球上两点之间的测地距离是相等的。两条这样的连线间的最小夹角,应该满足以下条件:

$$d(x_{mn},x_{nk}) = \min(d_g(x_{mn},x_{kl}),d_g(x_{mn},x_{lk}))$$
$$d_g(x_{mn},x_{kl}) = \arccos(\langle n_{mn},n_{kl}\rangle) \tag{3}$$

其中:$d_g(x_{mn},x_{kl})$ 是映射点 x_{mn} 和 x_{kl} 在高斯球上的测地距离。映射点在高斯球上的聚类行为称为高斯映射聚类,许多聚类算法都要预先指定聚类簇的个数如 K-均值聚类等。本文中映射点最终聚类后的个数无法预先指定,所以使用自底向上的凝聚型层次聚类算法[13]。该算法在初始的最底层中以每个映射点作为独立的簇,以后每一层中计算所有簇两两间的距离,合并距离最近的两个簇,通过各层向上递推,逐步将小簇合并为大簇,直到满足预先设定的终止条件。

具体而言,当前点的 N 个映射点构成独立簇集 Data = $\{C_1,C_2,\cdots,C_N\}$,若已知 C_i 和 C_j 为待比较的两个簇,$|C|$ 为每个簇中元素的个数,d 为由上述式(3)定义的高斯映射后相反簇连线的夹角,则以每个簇中映射点间的平均距离 d_n 作为簇合并准则:

$$d_n(C_i,C_j) = \frac{1}{|C_i|\cdot|C_j|}\sum_{x\in C_i}\sum_{y\in C_j} d(x,y) \tag{4}$$

要将 data 中的 N 个对象进行聚类,其距离矩阵大小为 $N\times N$,则聚类算法描述如下:

a)将每个簇看做一个独立的对象,计算它们两两之间的距离 $d_n(C_i,C_j)$,得到初始化距离矩阵 $D_{N\times N} = [d_n(C_i,C_j)]$,$1\leqslant i,j\leqslant N$。

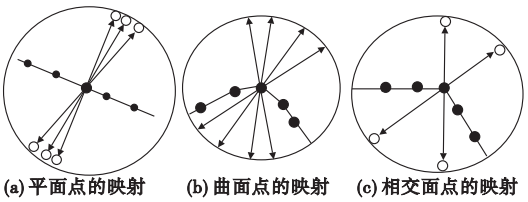
b)在距离矩阵 D 中,根据所有两两簇对间的距离,找到距离最近(最相似)的两个簇 C_r 和 C_s,$d_n(C_r,C_s) = \min\{d_n(C_i,C_j),1\leqslant i,j\leqslant N\}$。

c)将簇 C_r 和 C_s 合并 $C_r = C_r\cup C_s$,更新距离矩阵 D,删除 D 中簇 C_s 所在行和列的数据,并重新计算 C_r 簇所在行和列的距离值。

d)重复 b)c),直到剩余簇中任意两个簇之间的距离大于给定的阈值,即当矩阵 D 中各个元素的值都大于给定的阈值 $\delta\in(0,\frac{\pi}{2})$ 时聚类结束,δ 在该范围内取值主要考虑在高斯球一周 2π 的范围内最少将高斯映射点聚类成四个簇,以便进行下一步特征点的判断和识别。

2.3.2　特征点判断

通过以上聚类算法,对当前采样点 p_i 及其邻域内的高斯映射点:若以 p_i 为球心的高斯球上仅保留了一个簇,如图 4(a)则认为 p_i 点不是尖锐特征,因为潜在区域是个平面;若保留了 2 ~ 4 个簇,由于圆锥面、圆柱面、抛物面上的点通过高斯映射,在高斯球上最多聚类成四个簇,则该点 p_i 是特征点,如图 4(b);两个或多个面相交(最多四个),其相交线上点的高斯映射点,将在高斯球上呈现两个或多个簇对,如图 4(c);若高斯球上没有保留簇或者有多于四个簇,则认为该点不是特征点。



(a) 平面点的映射　(b) 曲面点的映射　(c) 相交面点的映射

图4　特征检测过程中聚类结果

这里,阈值 δ 表示所有法向映射后结果簇间的最小距离。对于两个曲面相交的情况,若 δ 值较小,则簇对应的法向量间的夹角较小(若两个相交曲面的内夹角较大,则交线处两曲面切平面的法向量间的夹角较小,高斯球上映射簇间的距离较小),通过高斯映射,可检测出两个面相交形成的钝角边上的特征点,同时锐角边邻域内可能会产生许多过估点。如果 δ 值较大,则表示簇间的距离较大(两个相邻面交线处切平面的法向量之间的夹角较大,而两个面之间的夹角可能较小),这样可以很好地区分处于直角或锐角边上的特征,但可能会忽略钝角边上的特征点[14];δ 太大,最终将得到包含所有高斯映射点的一个簇,当前采样点被认为是平面上的点而非特征点。

由以上分析可知,通过高斯映射可以检测出法向发生变化的曲面交线上的特征点。为了不遗漏由于磨损而平滑的曲面相交区域内特征点的检测,在特征点判断时还应加入曲率值的判断,即对当前点 p_i,若通过高斯映射没有被判定为特征点,但若该点 k 邻域内特征点数占邻域大小 k 的 10%,则需进一步判

断。由 2.2 节知,虽然通过高斯映射能算出曲面上某点的曲率值,但由于当前采样点及其 k 邻域点构成的连通区域面积较难计算,且曲率值正负性无法确定,所以本文将使用 Milroy 等人[15]提出的曲率估计方法,对该 p_i 点 k 邻域进行二次曲面拟合 $S(u,v)=au^2+buv+cv^2$,从而计算出曲面在该点处的最大和最小主曲率 k_1、k_2,相应的主方向 m_1、m_2 以及高斯曲率 k_G,公式如下:

$$m_1=\begin{cases}(c-a+\sqrt{(a-c)^2+b^2},-b) & a<c\\(b,c-a-\sqrt{(a-c)^2+b^2}) & a\geqslant c\end{cases}$$

$$m_2=\begin{cases}(b,c-a+\sqrt{(a-c)^2+b^2} & a<c\\(c-a-\sqrt{(a-c)^2+b^2},-b) & a\geqslant c\end{cases} \tag{5}$$

$$k_1=a+c-\sqrt{(a-c)^2+b^2},k_2=a+c+\sqrt{(a-c)^2+b^2}$$

$$k_G=k_1\cdot k_2$$

通常,曲面在某个已知点邻域内沿着不同方向上的弯曲程度是不同的。本文先在 p_i 的 k 邻域内,沿最大主曲率方向 m_1,找到左右两个邻近点 y_1、y_2,若最大主曲率值 $k_1(p_i)>k(y_1)$ 且 $k_1(p_i)>k(y_2)$,并且沿着最小主曲率 m_2 方向上,p_i 的最小主曲率值比其左右两个邻近点都小,则认为它是沿着曲面边界的曲率极值点[13];接着求 p_i 与邻域内其他特征点的高斯曲率变化均值 $\mu=\frac{1}{k}\sum_{q\in N(p_i)}(k_G(p_i)-k_G(q))$,若该值小于给定的阈值,则最终将其判定为候选特征点,并将这些点放入预先建立好的候选特征点列表中。

为了使得算法在完备性(找到所有特征)和准确性(指定特征的具体位置)之间找到平衡,可以使用迭代的方式来实现自适应的参数选择,从而找出模型上的特征点。

2.4 自适应的迭代过程

自适应迭代过程的实现,需要对邻域大小 k 和映射簇间的最小距离 δ 这两个参数进行设置,参数 k 范围的确定见 2.2 节所述,参数 δ 范围的确定见 2.3 节。在特征检测算法中,初始时将 δ 设置为 $(0,2/\pi)$ 内相对较小的值 0.1,这是为了得到所有相交面形成的钝角边上的特征点,初始时以产生过估计的锐角边特征点集为代价,实验结果如图 6 所示。邻域 k 取较大的值 20,使用高斯映射产生一个含较多特征点的数据集,然后通过局部地增加参数值 δ,来减少过估点的数目,同时减小邻域的大小,以丢弃被过估的、不会随 δ 值的增大而消失的锐角边特征点。本文算法使用以下迭代步骤,实现自适应迭代的特征提取:

a)对候选特征点列表中的每个特征点,设置初始化参数 $\delta=0.1$,$k=20$,并从列表中的第一个点出发,继续以下步骤。

b)计算该点邻域内候选特征点的个数,若该数目大于邻域大小 k 的 30%(通过反复实验,该参数一般取为 $0.1k\sim0.3k$,将会使最终特征线的显示清晰准确),则重新调整参数,将该点邻域大小 k 减 1,阈值 δ 增加 0.1,再对当前点邻域内的候选特征点使用高斯映射聚类和曲率值相结合的方法判断是否为特征点。若不是,则将该点从候选点集中删除;否则保留该点为特征点。

c)对当前候选点邻域内的每个特征点,重复 b)再次迭代地判断每个邻域候选点是否为特征点,直到给定邻域内的所有特征点检测完毕。

d)从候选点列表中选择下一个特征点重复 b)~c),直到邻域值 k 达到最小值 $k=8$,或邻域大小达到最大值 $\delta=1.2$(δ 取大于 1 小于 $2/\pi$ 的任何终止值均可,具体取决于所用的模型)。

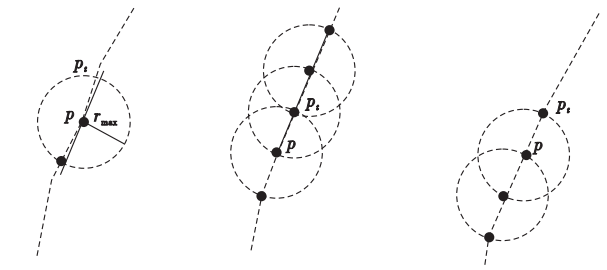
3 特征折线的生长

若将通过迭代优化的高斯映射算法得到的所有特征点集记为 F,则 F 中的点杂乱排列、不包含任何拓扑信息,因此还需要连线和平滑过程,形成连续光顺的特征线。常见的折线生长算法[1,16]是随机地从特征点集中选一个种子点作为生长点;然后将当前生长点及所有邻域点都投影到由协方差矩阵确定的主轴向量上,取投影最远的两个远端点作为新的生长点,这样得到的特征线可能会偏离原特征点集,且特征线可能不光顺,因此本文使用改进的折线生长方法,定义了新的特征折线的生长方向,并用 Bezier 曲线拟合各生长点,得到光顺的特征线。该算法需要由用户输入一个半径值 $r_{\max}$ 来作为每次折线生长的步长,并以此控制特征折线的精度。

首先,构造一个优先队列 Q,将特征点集 F 中的点按照高斯映射聚类簇的个数(2~4)由小到大的顺序放入 Q 中,这样就可以依次从 Q 中取出远离角点的模型顶点 p,作为特征折线的初始生长点。然后,以 p 为中心,以 $r_{\max}$ 为半径构造生长圆,取该圆内 $\mathrm{NBHD}(p)=\{p_j|p_j\in F,\|p_j-p\|\leqslant r_{\max}\}$ 范围内的点进行 PCA 分析,即计算出协方差矩阵最大特征值对应的特征向量 v_0。接着,将 $\mathrm{NBHD}(p)$ 内的每个点 p_j 投影到向量 v_0 上得到 p_j',并计算距离 $d_j=\|p_j-p'_j\|$,计算生长向量:

$$V_g=\sum_{p_j\in\mathrm{NBHD}(p)}\mathrm{e}^{-\left(\frac{\|p_j'-p\|d_j}{r_{\max}^2}\right)}\cdot\frac{p_j-p}{\|p_j-p\|} \tag{6}$$

初始生长时,离生长圆心 p 较远且偏离投影主轴 v_0 较远的点,对生长方向确定的贡献率较小,且邻域圆与生长向量的交点为 $p_t=p+r_{\max}\cdot v_g$,如图 5(a)所示;同时将 p 和 p_t 放入折线点集合 F_Z 中,下次以 p_t 为特征折线的新生长点,如图 5(b)所示;重复执行以上方法,进行迭代正向生长,当在正生长方向上找不到邻域点时,停止该方向的生长,并从队列 Q 中将 $\mathrm{NBHD}(p)$ 内的点删除,然后选择与点 p 生长方向的反方向 $-v_g$ 重新生长,如图 5(c)直至队列 Q 为空。这样会得到若干个折线点集合 F_Z,以每个特征折线点集合 F_Z 中的点为控制点,使用 k 次 Bezier 曲线 $F(t)=\sum_{i=0}^{k}b_iB_i^k(t)$,$b_i\in R^{k+1}$ 拟合这些点可以得到光顺的特征线。最后使用文献[1]中的方法进行折线间缺口的填补、角点的生成、多条折线的连接,最终得到完整光滑的特征线。



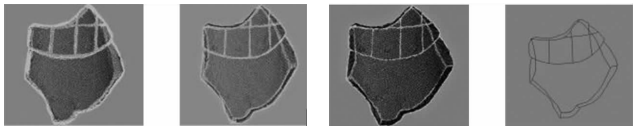
(a)构造邻域圆,初始生长 (b)迭代正向生长 (c)反向生长

图5 特征折线的生长

4 实验结果及其分析

在 VS2010 环境下采用 C++语言和 OpenGL 实现了本文的算法,通过对多个模型进行不同实验,可以发现本文方法能

较全、较准确地提取出模型上两个面相交处的尖锐特征。图 6 为对兵马俑铠甲碎片使用不同的参数进行实验的结果。其中:图 6(a)是当选择邻域大小 $k=20$、映射簇间的最小距离 $\delta=0.1$ 时所产生的许多过估计的特征点;(b)是参数改变之后的结果,可见过估计的点数目相对减少;(c)是通过迭代自适应的方式得到的特征点;(d)是通过折线生长算法得到的最终特征线。



(a) k=20,δ=0.1　(b) k=14,δ=0.7　(c) 自适应迭代　(d) 最终的特征线

图6　不同参数值对特征选择对比图

为了检测该算法对噪声的鲁棒性,对点云模型数据进行归一化,并构造一个球形包围盒,计算出它的半径 R。然后设定 ε 为噪声度,使模型上的每个点 p_i 沿着随机向量 $\rho(p_i)$ 扰动一到两次合成噪声,向量 $\rho(p_i)$ 的方向任意、长度小于 ε 与 R 的积。这样模型上有些点可能会移动到包围盒之外,使模型中点的个数相应地减少,得到的特征线会出现缺口。本文实验中分别取噪声度 ε 为 0.2%、0.9%。图 7 为加入不同噪声后对兵马俑碎片 2 提取特征线的结果。其中:图 7(a)为原始点云数据;(b)为不加噪声时从原点云数据中提取的特征线;(c)为加入较小的噪声 0.2% 时,提取的特征线,可以发现该特征线与原特征线相比,出现较小的毛刺,且线之间出现断裂,但特征线还是相对较光滑的;(d)为加入较大噪声 0.9% 时,得到的较粗糙特征线。从总体上来看,加入噪声后噪声特征点和原模型特征点的位置相差并不是很大,因此,本文算法对噪声干扰并不是很敏感。

(a) 原始点云数据　(b) 不加噪声的特征线　(c) 加入0.2%噪声的特征线　(d) 加入0.9%噪声的特征线

图7　带噪声的碎片2模型特征线提取对比图

将本文方法与 Daniels 等人[1]的 RMLS 特征提取方法进行比较。图 8(a)为原始兵马俑半身点云模型,(b)为本文方法的实验结果,(c)为 Daniels 方法的结果。通过比较,可见本文方法能完全提取俑碎片的所有尖锐特征曲线,具有完备性和准确性的优点;而 Daniels 的方法提出的特征曲线可能会丢失模型原有的部分特征,且 Daniels 由于使用了 RMLS 方法,计算时间较长。

(a) 原始点云模型　(b) 本文算法结果　(c) Daniels方法结果

图8　带噪声的俑半身模型特征线提取对比图

表 1　图 7 中的模型提取特征点及运行时间的数据统计

碎片 2	噪声率/%	原始点数	特征点数	运行时间/s
Daniels 方法	0	35050	4950	89.8
	0.2	34662	4630	80.6
本文方法	0	35050	5210	62.6
	0.2	34893	4886	58.2

表 1 为对于图 7 的碎片 2 加入噪声并使用两种不同算法,在提取特征点个数、运行时间上的比较。与 Daniels 的方法相比,本文方法提取的特征点数量较多,能准确地反映模型的基本特征,且所需的运行时间较少,算法效率较高。

5　结束语

使用迭代自适应高斯映射和曲率判断过程来提取模型尖锐特征点,可以避免因提取过多特征点而带来之后特征去噪的复杂步骤,算法简单易于实现,运行效率较高,且通过改进的折线生长算法,可以较准确地拟合原特征点集。实验证明,该方法具有良好的自适应性、抗噪性和准确性,但是本文方法只适用于封闭模型的尖锐特征边提取,无法提取模型细节特征。如何分段截取所得到的模型特征线,并用恰当的曲线匹配方式来实现三维碎片的拼接是今后需要进一步研究的问题。

参考文献:

[1] DANIELS J, HA L K, OCHOTTA T, *et al*. Robust smooth feature extraction from point clouds[C]//Proc of IEEE International Conference on Shape Modeling and Applications. Piscataway, NJ: IEEE Press, 2007: 123-136.

[2] PANG X F, PANG M Y, SONG Z. Extracting feature curves on point sets[J]. International Journal of Information Engineering and Electronic Business, 2011, 3(3): 1-7.

[3] MÉRIGOT Q, OVSJANIKOV M, GUIBAS L J. Robust Voronoi-based curvature and feature estimation from point clouds[J]. IEEE Trans on Visualization and Computer Graphics, 2011, 17(6): 743-756.

[4] 王丽辉,袁保宗. 三维散乱点云模型特征点检测[J]. 信号处理, 2011, 27(6): 932-938.

[5] DEMARSIN K, VANDERSTRAETEN D, VOLODINED T, *et al*. Detection of closed sharp edges in point clouds using normal and graph theory[J]. Computer Aided Design, 2007, 39(4): 276-283.

[6] LEE I K. Curve reconstruction from unorganized points[J]. Computer Aided Geometric Design, 2000, 17(2): 161-177.

[7] DeCARLO D, FINKELSTEIN A, RUSINKIEWICZ S, *et al*. Suggestive contours for conveying shape[C]//Proc of SIGGRAPH. New York: ACM Press, 2003: 848-855.

[8] 柯映林,李岸. 基于主方向高斯映射的旋转面特征提取[J]. 浙江大学学报:工学版, 2006, 40(6): 942-947.

[9] 李岸,管爱枝. 基于高斯映射的柱面与锥面点云拟合[J]. 组合机床与自动化加工技术, 2007(8): 28-32.

[10] 刘宇. 基于微分信息的散乱点云拼合和分割[D]. 湖北:华中科技大学, 2008.

[11] HORN B K P. Extended Gaussian images[J]. Proceedings of the IEEE, 1984, 72(12): 1671-1686.

[12] PAULY M, KEISER R, GROSS M. Multi-scale feature extraction on point-sampled surfaces[J]. Computer Graphics Forum, 2003, 22(3): 281-289.

[13] HASTIE T, TIBSHIRANI R, FRIEDMAN J. The elements of statistical learning: data mining, inference, and prediction[M]. 2nd ed. Boston: Springer, 2009.

[14] WEBER C, HAHMANN S, HAGEN H. Sharp feature detection in point clouds[C]//Proc of the Shape Modeling Inernational Conference. Washington DC: IEEE Computer Society, 2010: 175-186.

[15] MILROY M J, BRADLEY C, VICKERS G W. Segmentation of a wrap-around model using an active contour[J]. Computer Aided Design, 1997, 29(4): 299-230.

[16] 马骊溟,徐毅,李泽湘. 基于高斯曲率极值点的散乱点云数据特征点提取[J]. 系统仿真学报, 2008, 20(9): 2341-2344.