

一种基于投影的散乱数据表面增量重建算法

田庆国 葛宝臻 郁道银 吕且妮 卢波

(天津大学精密仪器与光电子工程学院, 光电信息技术科学教育部重点实验室, 天津 300072)

摘要 针对 3 维散乱数据场提出了一种表面重建算法。根据空间曲面的局平特性和平面三角化的基本原则, 在参考点的切平面上对邻域点按角度排序, 应用可见性准则删除不可见点后, 相邻邻域点和参考点形成三角网格。将平面上的网格关系对应到空间, 以增量方式重建反映散乱数据场拓扑关系的空间曲面。设定角度阈值优化网格, 判断空间曲面的边界和孔洞。对多个数据场进行重建并对结果进行分析。对多个数据场进行重建并对结果进行分析表明, 算法具有原理简单, 重建速度快, 重建效果好的特点。

关键词 散乱数据场 表面重建 曲面局平特性 三角网格 可见性准则

中图法分类号: TP391 **文献标识码:** A **文章编号:** 1006-8961(2006)10-1409-05

An Incremental Projection-based Surface Reconstruction Algorithm from Scattered Data

TIAN Qing-guo, GE Bao-zhen, YU Dao-yin, LÜ Qie-ni, LU Bo

(College of Precision Instruments and Opto-electronics Engineering, Tianjin University, Key Laboratory of Opto-electronics Information and Technical Science, Ministry of Education, Tianjin 300072)

Abstract To 3D scattered data, a novel algorithm for surface reconstruction is presented. Based on local flatness of surface and fundamental rule of planar triangulation, this method sorted neighbor points around reference point, R on tangent plane at R. After deleting invisible points by visibility criterion, any two adjacent points formed triangle with R. At last the planar mesh relationship returned to the space and object surface, which correctly reflects point cloud topology, is reconstructed progressively. By predefining angle thresholds, this algorithm can optimize triangles and test surface boundary and holes automatically. Several scattered data reconstructions are completed using VC++ , and the results show that this algorithm is easy to implement, and can reconstruct more authentic 3D surface with a faster speed.

Keywords scattered data, surface reconstruction, local flatness of surface, triangle mesh, visibility criterion

1 引言

物体表面模型重建是计算几何, 计算机可视化以及逆向工程领域中重要的研究内容。散乱数据场(scattered data)指点与点之间毫无内在联系的数据场, 是工程应用和算法研究经常用到的数据类型。针对这种数据类型已经提出了多种表面重建的方

法。Attali^[1]首先将包围 3 维数据场的最小立方体细分为分离的小单元, 然后在各小单元中寻找等值面; 而 Amenta^[2]直接用含有曲面片的小单元代替物体表面。这类方法难以重建复杂拓扑的曲面形状。Hoope^[3]等人提出基于距离函数的表面重建算法, 首先构造点到物体表面的(有向)距离场, 然后计算距离场的零等值面, 即为重建曲面。不过该算法涉及到复杂的法向一致性检查和等值面抽取, 重建非

基金项目: 国家自然科学基金项目(60277009); 天津市科技攻关培育项目(05YFGPX05000)

收稿日期: 2006-02-20; **改回日期:** 2006-05-31

第一作者简介: 田庆国(1973 ~), 男, 天津大学精密仪器与光电子工程学院光学工程专业在读博士研究生。主要研究方向为计算机图形学、3 维可视化。E-mail: tianqg_zhwx@163.com

常耗时。在 Green^[4] 实现 2 维空间 Voronoi 图的计算和 Delaunay 三角化,特别是 Bowyer^[5] 和 Watson^[6] 把结果推广到任意维之后,出现了大量利用 Delaunay 三角剖分实现 3 维表面增量重建的算法^[7,8]。这些算法分别应用 2 维或 3 维的 Delaunay 三角剖分在邻域内重构三角网格,渐进地完成曲面重建。Delaunay 三角化符合最大内角最小化原则,而且能够重建复杂曲面,但是 Voronoi 图的计算和 Delaunay 三角化都是非常费时的。本文提出了一种新的基于投影的快速表面重建算法。

2 基于投影的网格重建

2.1 算法思想及相关概念

平面三角网格化的基本准则是任何两条线段除了在顶点相交外,不能出现交叉现象。如果两点的连线和已知的线段 l 交叉,其中的一点称为另一点的不可见点,线段 l 称为阻挡边。虽然这一准则不是空间不可见的充分条件,但是当 3 维点云采样密度足够大且局部均匀时,可以用平面不可见准则判断相应空间点的位置关系。

为了便于表述,下面给出点和边的概念。

如果正在搜索点 r 的邻域点,则称点 r 为参考点。没有任何附属三角形的点称为自由点。已有部分附属三角形的点,没有被选作参考点的为边缘点,已被选作参考点的为边界点。如果某点的所有附属三角形均被找出,称该点为完成点。

连接点 r 和邻域点 p 的边分为两类。若有两个三角形和该边相连,则称该边为内部边,同时称点 p 为点 r 的内部顶点。若只有一个三角形和该边相连,称该边为边界边,同时称点 p 为点 r 的边界顶点。点 r 的所有内部顶点和边界顶点统称为点 r 的附属顶点。

2.2 局部三角化算法流程

(1) 对原始点云数组预处理。按 x 坐标从小到大排序分层;所有点均标记为自由点;将 x 坐标最小的点放入参考点数组 R 中,并令 R 中元素个数 $j=1$ 。

(2) 从 RA 中取出首元素 r ,并使 $j=j+1$ 。在点云数组中搜索 r 的满足要求的邻域点集 C_r 。

(3) 构造 r 的投影平面 P_r ,将 C_r 向 P_r 投影,投影点集称为 C_p 。

(4) 对点集 C_p 进行可见性判别。删除不满足要求的点之后的点集称为 C'_p 。

(5) 连接点 r 和点集 C'_p 中的点,构成平面三角形,将连接关系返回到 3 维空间。

(6) 修改 C'_p 中点的属性,将 C'_p 中的边缘点加入到 R ,同时增加 j 。

(7) 如果 $j \neq 0$;返回第 2 步;否则算法结束。

2.3 算法流程中的主要步骤

2.3.1 邻域搜索

快速搜索参考点的邻域点集非常重要。不同于已有的搜索算法^[9,10],本文提出了一种基于分层搜索的方法,如图 1 所示。从点 r 所在层向两侧分别搜索邻域点,直到待选点所在层与点 r 所在层之间的距离大于邻域半径 D ,结束搜索,其中 $D=nd$ 。在初始阶段, d 是距 r 最近的点到 r 的距离, n 为邻域半径系数是常数,可以取 $n=3$ 。在重建中 d 是点 r 的附属顶点到 r 的最远距离,此时可取 $n=1 \sim 1.5$ 。从邻域半径的选择看,点 b 可能是点 a 的邻域点,反之却未必。但是这种不对称性并不会影响重建网格的拓扑正确性。

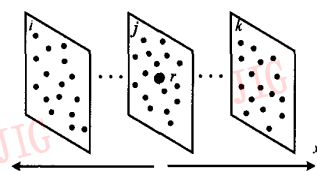


图 1 分层搜索参考点 r 的邻域点

Fig. 1 Finding adjacent points layer after layer

为了保证搜索到的邻域点不会过多,可以同时设置邻域点数阈值 k 。本方法不要求一定选择参考点的 k 个邻域点,而是保证邻域点数不会多于 k 个。在该阶段,邻域点集不包含参考点的边界顶点。

2.3.2 C_p 中点的可见性判别

参考点 r 的投影平面 P_r 定义为通过点 r ,法向量为 V 的平面。通过如下方法可以确定 V :定义一个 3×3 的对称矩阵

$$C = \sum_{i=1}^m (p_i - r)^T (p_i - r) \quad (1)$$

其中, p_i 是参考点 r 的第 i 个邻域点, m 是邻域点数。容易证明,矩阵 C 的最小特征值对应的单位特征向量是法向量 V 的很好近似。在 P_r 上建立局部坐标系,以参考点 r 为原点, r 和 p_1 的投影点的连线作为 x 轴,法向量 V 作为 z 轴。将点集 C_p 逆时针排序,如图 2(a) 所示。此时点集 C_p 还不包括边界顶点 M 和 N 。在应用可见性准则删除不可见点之前,对 C_p

应用角度准则进行删减是必要的。因为如果 $\angle p_i r p_{i+1}$ 非常小,将生成细长的三角形。为了优化三角形的形状,设置最小角度阈值 $\alpha_{\min}$,然后对点集 C_p 进行分组。如 $\angle p_3 r p_4 < \alpha_{\min}$,则 p_3 和 p_4 分在一组, $\angle p_5 r p_6 < \alpha_{\min}$,则 p_5 和 p_6 分在一组。其他的点各自为一组。应用角度准则删减邻域点的步骤如下:

(1) 如果某组中点数为 1,转第 3 步,否则执行第 2 步;

(2) 只保留该组中到参考点距离最近的点;

(3) 比较该点和参考点的边界顶点与参考点形成的夹角。如果小于 $\alpha_{\min}$,删除该点,否则保留该点;

(4) 所有的组均处理完,算法结束;否则转第 1 步处理下一组。

通过角度准则处理后参考点 r 的邻域点集如图 2(b) 所示,邻域点 3 和 6 已被删除,同时在这一步中将 r 的边界顶点 M 和 N 加入到 C_p 中。

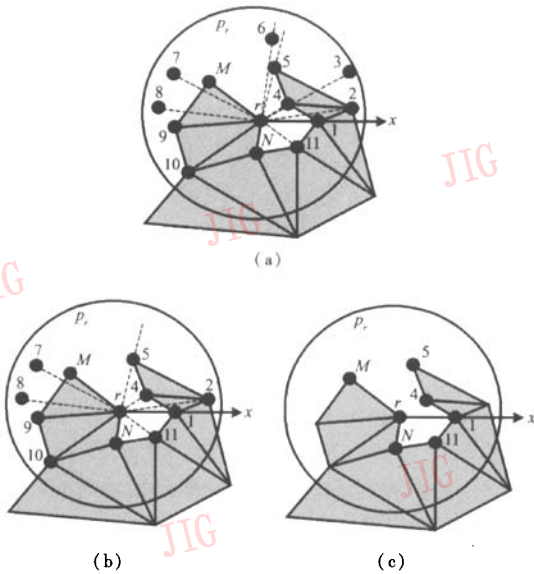


图 2 在 P_r 上对邻域点集 C_p 进行处理
Fig. 2 Deal with C_p on projection plane P_r

Gopi^[10]指出,只有边界边才能成为阻挡边,因此通过可见性准则删除不可见点时可以先找出点 r 和 C_p 中各点的附属边界边。因为点 r 的内部顶点不再和参考点形成三角形,因此首先删除 C_p 中点 r 的内部顶点,如图 2(b) 中的点 9 和 10。然后判断剩余的邻域点与参考点的连线是否和某条边界边交叉,如果交叉则删除该点,否则保留该点。图 2(c)

是通过可见性准则删除邻域点之后的结果,点 2, 7, 8, 9, 10 已被删除。该步处理之后参考点 r 的邻域点集 C_p 变成了 C_p^r 。

2.3.3 局部三角网格化及修改邻域点属性

一般地,点集 C_p^r 中相邻两点可以和参考点 r 构成三角形,但是还有一种情况需要特别考虑,如图 3 所示。点 r 是参考点, p, t 是 r 的边界顶点,而 q, s 是 r 的内部顶点,通过上面的处理, q, s 已被删除,但是点 p, t 明显不能和 r 构成三角形。可以看出这种情况只在 p, t 均是参考点的边界顶点时发生,因此如果 C_p^r 中的相邻点均是参考点的边界顶点时要判断它们之间是否存在参考点的内部顶点。如果有,则不能形成三角形,否则可以。为了优化新生成的三角形并且检测物体的边界,可以设置最大角度阈值 $\alpha_{\max}$ 。只有相邻点和参考点形成的夹角小于 $\alpha_{\max}$ 时,才能形成三角形。

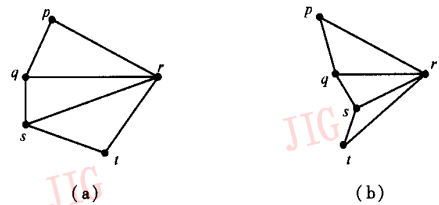


图 3 边界顶点不能和参考点构成三角形的两种情况

Fig. 3 Two examples that boundary vertices cannot form triangle with r

在局部三角网格化过程中需要对新三角形的法向量进行调整。设三角形 T_1 和 T_2 是有公共顶点的两个三角形, T_1 的法向量 N_1 已经指向体外, T_2 的法向量 N_2 和 N_1 若满足:

(1) $N_1 \cdot N_2 > 0$, N_2 指向体外,不需要调整;

(2) $N_1 \cdot N_2 < 0$, N_2 指向体内,需要将 N_2 的 3 个分量分别反向。

对于封闭曲面, x 坐标最小的点处三角形法向量的 x 分量小于 0。因此增量重建过程从 x 坐标最小的点开始,在进行局部网格化的同时,调整三角形的法向量。

局部三角化完成之后,需要修改参考点 r 和 C_p^r 中点的属性。将 C_p^r 中的自由点均改为边缘点,并加入到 R 尾部。如果 C_p^r 中的某点已经是边缘点或边界点,判断其是否已经成为了完成点。对于 r ,如果和其相连的三角形已全部找出,将其修改为完成点,否则其为边界点。

3 算法实例和分析

应用 VC ++ 和 OpenGL 在 PC 机上实现该算法, 并对不同的 3 维散乱数据场进行重建实验。首先对图 3 的情况进行实验验证, 结果如图 4 所示。图 4(a)是某点云数据的一部分, 图 4(b)是没有考虑参考点的两个边界顶点能否和参考点构成三角形的结果。从图中可以看出出现了交叉的线段; 而图 4(c)对两个边界顶点和参考点能否构成三角形进行判断, 交叉现象消失了。

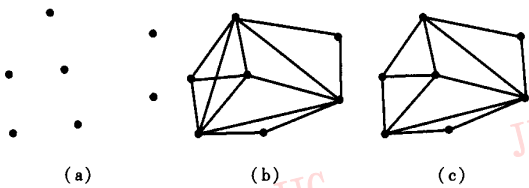


图 4 两种局部三角网格化的结果

Fig. 4 Two different results of local triangulation

图 5(a)对单位球的表面进行采样得到相应的 3 维点云数据, 然后重建得到相应的三角网格表示, 在重建中, 邻域半径系数选为 $n = 1.1$ 。图 6(a) 是人的脸谱的 3 维点云数据, 图 6(b)是重建出的光照模式下的表面效果图。图 7(a)是马的点云图, 图 7(b)是表面模型。图 6(b)和图 7(b)均是在邻域半径系数 $n = 1.1$ 时的重建结果。

表 1 给出了不同的 n 对重建结果的影响。可以看出不同的邻域半径系数 n 对生成的三角面片数影响不大, 但是对重建时间的影响很大。随着 n 的增大, 重建时间显著增加, 特别是大点云数据。对一般的散乱数据场, $n = 1.1$ 能得到满意的重建效果。

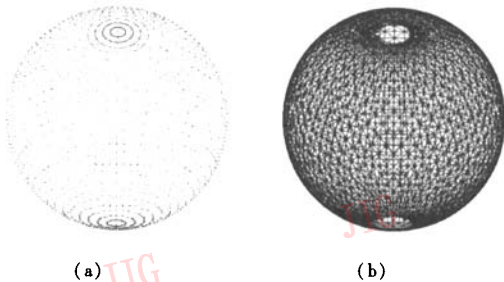


图 5 球的点云图和三角网格化($n = 1.1$)

Fig. 5 Ball's point cloud and triangle mesh($n = 1.1$)

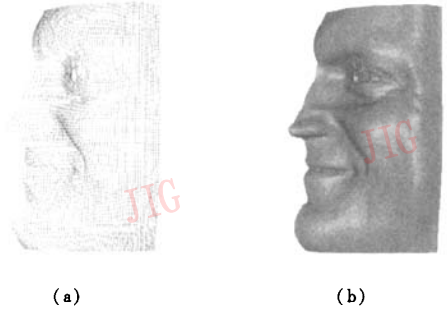


图 6 Domino 的点云图和表面重建($n = 1.1$)

Fig. 6 Domino's point cloud and reconstruction result($n = 1.1$)

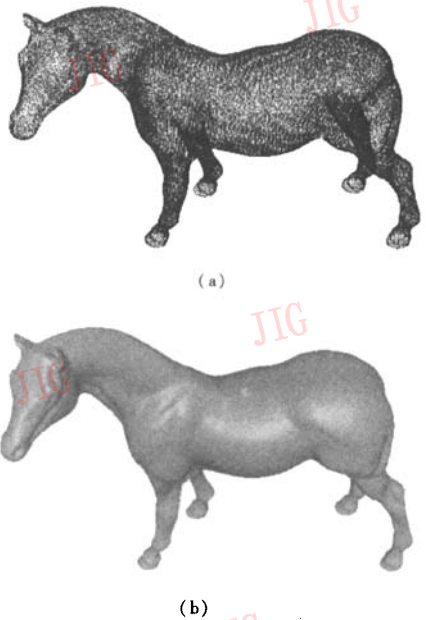


图 7 马的点云图和表面重建结果($n = 1.1$)

Fig. 7 Horse's point cloud and reconstruction result($n = 1.1$)

表 1 邻域半径系数 n 对重建结果的影响

Tab. 1 Effect of different adjacent radius coefficient n to reconstruction results

例子	点数	n	生成的三角面片数	运行时间 (s)
Ball	3 121	1.1	5 176	0.531
		1.3	5 709	0.672
		1.5	5 708	0.893
Domino	11 885	1.1	23 292	3.310
		1.3	23 293	4.134
		1.5	23 297	4.775
Horse	48 485	1.1	96 694	25.922
		1.3	96 816	29.937
		1.5	96 811	32.765

4 结 论

本文提出的算法在只有物体表面散乱数据场而没有任何附加信息的情况下,能够从一点出发,渐进地完成邻域点的搜索、投影、删减、网格化和法矢量的自动调整,最后重建出反映数据场正确拓扑结构的网格曲面。该方法具有算法简单,易于实现,节省内存,重建速度快的优点。从重建实例看,算法效果良好。

参考文献 (References)

- 1 Dominique Attali. R-regular shape reconstruction from unorganized points[A]. In: Proceedings of the Thirteenth Annual Symposium on Computational Geometry[C], Nice, France, 1997: 248 ~ 273.
- 2 Amenta N, Bern M, Kamvyselis M. A new Voronoi-based surface reconstruction algorithm[A]. In: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques[C], Orlando, FL, USA, 1998: 415 ~ 421.
- 3 Hoopé H, Deroose T, Duchamp T, *et al.* Surface reconstruction from unorganized points [A]. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques[C], Chicago, USA, 1992: 71 ~ 78.
- 4 Green P J, Sibson R. Computing Dirichlet tessellations in the plane [J]. The Computer Journal, 1978, 21(2): 168 ~ 173.
- 5 Bowyer A. Computing Dirichlet tessellations [J]. The Computer Journal, 1981, 24(2): 162 ~ 166.
- 6 Watson D F. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes [J]. The Computer Journal, 1981, 24(2): 167 ~ 172.
- 7 WANG Qing, WANG Rong-qing, BAO Hu-jun. A fast progressive surface reconstruction algorithm for unorganized points[J]. Journal of Software, 2000, 11(9): 1221 ~ 1227. [王青,王融清,鲍虎军等. 散乱数据点的增量快速曲面重建算法[J]. 软件学报, 2000, 11(9): 1221 ~ 1227.]
- 8 TAN Jian-rong, LI Li-xin. An algorithm for topology reconstruction from unorganized points based on local flatness of surface[J]. Journal of Software, 2002, 13(11): 2121 ~ 2126. [谭建荣,李立新. 基于曲面局平特性的散乱数据拓扑重建算法[J]. 软件学报, 2002, 13(11): 2121 ~ 2126.]
- 9 Xiong Bang-shu, He Ming-yi, Yu Hua-jing. Algorithm for finding k-nearest neighbors of scattered points[J]. Journal of Computer-aided Design & Computer Graphics, 2004, 16(7): 909 ~ 912. [熊邦书, 何明一, 俞华璟. 3 维散乱数据的 k 个最近邻域快速搜索算法[J]. 计算机辅助设计与图形学学报, 2004, 16(7): 909 ~ 912.]
- 10 Gopi M, Shankar Krishnan. A fast and efficient projection-based approach for surface reconstruction[A]. In: Proceedings of the XV Brazilian Symposium on Computer Graphics and Image Processing [C], Washington DC, EUA, 2002: 179 ~ 186.