

# 一种面向三维点集的快速表面重构算法

梁荣华<sup>1,2)</sup> 陈 纯<sup>1)</sup> 潘志庚<sup>3)</sup> 张 慧<sup>1)</sup>

<sup>1)</sup> (浙江大学计算机科学与工程系, 杭州 310027) <sup>2)</sup> (杭州电子工业学院计算机学院, 杭州 310037)

<sup>3)</sup> (浙江大学 CAD&CG 国家重点实验室, 杭州 310027)

**摘 要** 在对目前比较流行的空间三角化算法进行对比研究的基础上,对 Hugues Hoppe 提出的算法进行了改进,即借鉴 Marching Cubes 算法的基本思想,首先通过自动选取适当的参数,用包围盒方法将三维散乱点划分为数据区域;然后求取点的切平面及法向,同时采用广度优先算法遍历数据点来调整法向和快速地求取 Marching Cubes 的等势函数;最后用基于查表法的 Marching Cubes 来输出三角面片,即得到表面模型.实验结果表明,改进后的算法效率有较大的提高.新算法不仅适用于表面三维散乱点数据,也可以对体数据进行重构,具有一定的通用性.

**关键词** 三维重建 空间三角化 广度优先 Marching Cubes 等值面

**中图法分类号:** TP391.41 **文献标识码:** A **文章编号:** 1006-8961(2003)01-0063-05

## A Fast Model Reconstruction Algorithm for 3D Unorganized Points

LIANG Rong-hua<sup>1,2)</sup>, CHEN Chun<sup>1)</sup>, PAN Zhi-geng<sup>3)</sup>, ZHANG Hui<sup>1)</sup>

<sup>1)</sup> (Department of Computer Science and Engineering, Zhejiang University, Hangzhou 310027)

<sup>2)</sup> (School of Computer Science and Engineering, Hangzhou Institute of Electronics Engineering, Hangzhou 310037)

<sup>3)</sup> (State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310027)

**Abstract** Surface model reconstruction from 3D unorganized points (points of surface or volume) is of great importance in variable fields such as computer vision, images based modeling, 3D reconstruction based on images, scientific computing visualization, etc.. Many approaches have proposed to resolve the problem, such as 3D Delaunay triangulation, CDT, Qull Hull. etc. What makes the problem very difficult is that the reconstruction surface is convex and the efficiency of algorithms is not high. In this paper we present a fast model reconstruction algorithm for 3D unorganized points based on Hugues Hoppe algorithm. First, input points are divided into small logical "cubes" whose size can be decided automatically from the unorganized points according to Marching Cubes. Then the tangible planes and normal vectors at each point are calculated and all of the normal vectors are orientated to the outside of surface based on WFS(Wide First Searching). Finally, the function of iso-surface of scalar field for Marching Cubes algorithm can be obtained. In addition, the algorithm improves the efficiency of Marching Cubes by looking up tables. Finally the model can be obtained by the output of Marching Cubes composed of triangular meshes. Experimental results show the high efficiency of the algorithm. And the algorithm can be applied to not only the points of the surface but also the volume data (such as 3D scanning data, MRI data).

**Keywords** 3D reconstruction, 3D triangulation, Wide first searching(WFS), Marching Cubes, Iso surface

## 0 引 言

在三维模型重构中,一个基本的问题是,如果已经得到了物理表面的散乱三维数据点(没有任何其

他信息,如法向、标量场等),如何以这些数据点为基础来快速构造物体的三维表面模型.

最典型的重构方法是 Delaunay 空间三角化方法,虽然在二维平面区域问题中, Delaunay 三角化方法取得了相当大的成功,但在 3D 网格自动生成

中,却遇到了巨大困难,因为在三维网格生成中,不仅最大-最小角判据的对角线交换规则不再成立,而且基于外接圆判据的Delaunay三角化一般也不再能保证生成的网格质量<sup>[1~3]</sup>。虽然通过限制Delaunay三角化<sup>[4,5]</sup>,但缺点是边界重构不能保证质量。杨钦等人通过分析CDT中存在的问题,提出并论证了边界边、边界面片,在三角化中的存在条件,还讨论了存在性条件在实际工程中的应用范围,从而充实了CDT的研究基础,并为三维Delaunay三角化算法的设计提供了理论依据<sup>[6]</sup>。网上已有相关方面的软件开发包<sup>[7]</sup>,在<http://www.geom.umn.edu/software/qhull>处可以下载开发包QuickHull。QuickHull虽能够利用输入的三维点来重构表面三角形网格,但获取的表面是凸的,而对于有凹的表面则需要编写新算法。

崔汉国等人提出了一种在三维任意区域中,对点集进行三角剖分的算法<sup>[8]</sup>。该算法首先对输入的点进行排序,并将前 $i$ 个点进行标准的Delaunay三角剖分;然后通过新增顶点来局部改变“四面体”对,从而形成表面的三角剖分,但算法假定有通用的孔洞,如果输入的点动态增加或减少,则不能对生成的剖分三角形进行修改。王育等人提出了基于二维Delaunay的空间三角剖分技术<sup>[9]</sup>。该方法首先对每个数据点的局部拓扑重建,然后通过自动矫正局部数据点的非法连接关系,以增量扩张的方式来把局部三角网拼接成一张标准的整体二维流形网格,但算法不适用于等值面及体数据。周焰等人基于三维重建物体截面轮廓线的控制点,先对每两条相邻轮廓线的控制点进行配对,然后在这样两条轮廓线之间进行三角剖分<sup>[10]</sup>。但这种方法只能应用于体数据。

Hugues Hoppe对三维重构算法作了详细的描述<sup>[11,12]</sup>。该算法能够从表面数据及体数据(如三维断层扫描的数据和核磁共振方法获得的三维数据)来恢复表面,但对于边界及尖锐特征不能很好地生成,同时对于海量数据,算法运行效率不高,还需要手工调整参数。周儒荣等人首先根据测点的近邻来估算法矢,然后对法矢进行调整,以使法矢都指向曲面外侧,最后输出三角网格模型<sup>[13]</sup>。该算法不仅在计算切平面、空间划分方面提高了效率,并改善了曲面边界及尖锐棱边区域的重建效果,还解决了法矢方向传播中可能出现的局部“孤岛”问题,但算法效率还有待进一步提高。

本文的研究目标是面向人脸的三维重建,即在

三维重建时,首先通过相机定标、对应的特征点提取以及运动分析与三维点生成来得到人脸的三维散乱点,再根据这些点来重构脸部的表面模型。本文算法在Hugues Hoppe工作的基础上进行,借鉴Marching Cubes算法<sup>[14,15]</sup>的思想,以便能够自动完成参数的调整,使划分数据的区域比较优化,另外,还采用广度优先算法对数据点进行遍历来调整法向量,从而较好地获得了Marching Cubes的等势函数,同时在速度上进行了改进。本文算法不仅可重构有尖锐特征的表面(凸的,或凹的表面),还可以对体数据进行重构。

## 1 算法描述

### 1.1 定义及假设

假定有三维点集 $X = \{x_1, x_2, \dots, x_n\}$ ,其中, $x_i \in \mathbb{R}^3, i = 1, 2, \dots, n$ ;并设所求的表面是 $U$ 。参照文献<sup>[11]</sup>、<sup>[13]</sup>,给出以下的定义及假设。

**定义1** 设 $U$ 表面上的点 $y_i = x_i + e_i$ ,其中, $|e_i| \leq \delta$ ,称 $X$ 是 $\delta$ -噪声。

**定义2** 若 $X$ 是 $0$ -噪声,则以 $U$ 表面上的点 $p$ 为球心, $r$ 为半径的球中至少包含一点,称 $X$ 是 $r$ -深度。

**定义3** 点 $x$ 的 $k$ -近邻是指 $X$ 点集中与点 $x$ 的直线距离最近的 $k$ 个点,记为 $B(x)$ 。

**定义4** 点 $p$ 到面 $U$ 的距离函数为 $D_n(p) = sD(p, U)$ ,其中, $D(p, U)$ 表示点 $p$ 到面 $U$ 的距离,是一个正数, $s$ 的值为 $+1$ 或 $-1$ ,这依赖于点 $p$ 与面 $U$ 的位置关系,当 $p$ 在面 $U$ 的外法向方向时, $s$ 为 $+1$ ,否则为 $-1$ 。

**假设** 以下假设三维点集 $X$ 是 $\delta$ -噪声, $r$ -深度,这样的假设是有意义的,若 $X$ 是 $0$ -噪声, $r$ -深度,则根据定义,表面上不可能有半径大于 $r$ 的空洞,如果给定 $X$ 是 $\delta$ -噪声, $r$ -深度,那么显然点 $p$ 满足 $D(p, U) > r + \delta$ ,且点 $p$ 不可能在面 $U$ 上。

根据前面的假定,如果能够求出距离函数 $D_n$ ,则就可以计算表面 $U$ ,因为只要计算得到点 $p$ 到表面 $U$ 的距离函数值等于 $0$ 的点,即可求出集合 $Z(D_n) = \{p, D_n(p) = 0\}$ 。最后通过Marching Cubes算法,以 $D_n$ 作为场函数,用Marching Cubes算法输出三角形面片即可。

### 1.2 总体算法步骤

算法可以用以下步骤描述:

(1) 根据  $X$  估算出  $r + \delta$ , 并算出邻集  $B(X)$  的  $k$  值.

(2) 计算  $B(X)$  中每个点的切平面, 然后求每个切平面的法向.

(3) 法向调整, 使每个切平面的法向指向曲面的外侧.

(4) 计算距离函数  $D_u$ .

(5) 通过 Marching Cubes 输出表面  $U$  上的三角形面片.

下面重点介绍算法步骤(1)、(3)及(5), 其他的详细内容可以参见文献[11]、[13].

### 1.3 $r + \delta$ 及 $k$ -邻集的计算

从给定的数据中, 初步估算  $r + \delta$

$$r + \delta = \sqrt{\frac{(x_{\max} - x_{\min})(y_{\max} - y_{\min})(z_{\max} - z_{\min})}{t}}$$

其中,  $t$  是三维点集  $X$  中点的个数,  $(x_{\max}, x_{\min}, y_{\max}, y_{\min}, z_{\max}, z_{\min})$  分别表示数据集中坐标  $X, Y, Z$  轴方向最大坐标值与最小坐标值.

对于每个三维数据点使用的数据结构如下:

```
struct node{
    int index;           // 编号
    float xyz[3];       // 坐标
    struct node * B;    // 邻集
    float fnormal[3];   // 法向量
    BOOL isvisit;      // 点是否访问过
};
```

借鉴 Marching Cubes 算法中, 关于逻辑立方体(也称为包围盒)的定义, 可将包围盒保存起来, 用于对散乱点的数据分区. 这样分区的好处是易于找到每个点的邻集. 若将每个包围盒的大小取为  $(r + \delta) \times (r + \delta) \times (r + \delta)$ , 在坐标  $X, Y, Z$  轴方向分别取  $l, m, n$  个包围盒, 则总的包围盒个数为  $l \times m \times n$ .

确定  $k$  邻集的算法步骤如下:

(1) 选取点  $x$ , 确定所在的包围盒.

(2) 在包围盒周围 6 个方向选取相邻的包围盒, 找出包围盒中所有的数据点, 加到邻集中.

(3) 重复  $l \times m \times n$  次步骤(1)和(2), 即可以找到所有的邻集.

### 1.4 基于广度优先的法向调整

计算点  $p$  到切平面  $T_p(x_i)$  的距离函数为

$$d(p, T_p(x_i)) = \mathbf{n}_i \cdot (\mathbf{p} - \mathbf{o}_i) \quad (1)$$

其中,  $\mathbf{o}_i$  是  $T_p(x_i)$  的中心点,  $\mathbf{n}_i$  是  $T_p(x_i)$  的法向. 在计算切平面法向时, 需先求出以下半正定矩阵

$$C = \sum_{p \in B(x_i)} (\mathbf{p} - \mathbf{o}_i)(\mathbf{p} - \mathbf{o}_i)^T \quad (2)$$

其对应的最小值单位特征向量  $\mathbf{V}_i$ , 取  $\mathbf{n}_i$  为  $\mathbf{V}_i$  或  $-\mathbf{V}_i$  即可.

根据定义 4 及式(1), 计算距离时要对切平面  $T_p(x_i)$  的法向进行调整, 使其指向表面外侧.

一种法向调整方法是通过 Riemannian 图的 MST(最小生成树)来遍历整个结点进行, 本文采用一个更高效的算法, 即通过广度优先遍历整个结点来对结点进行调整. 算法如下:

(1) 在所有的结点中, 选取  $Z$  轴坐标最大值的结点  $x_i$ , 作为根结点, 再根据其邻集算出切平面及法向  $\mathbf{n}_i$  (见 1.3 节), 并标识该结点已经访问过.

(2) 找出  $B(x_i)$  中没有访问的点  $x_p$ , 同样算出切平面及法向  $\mathbf{n}_p$ . 如果  $\mathbf{n}_i \cdot \mathbf{n}_p < 0$ , 则调整  $\mathbf{n}_p$  为反向.

(3) 如果  $B(x_i)$  中所有的结点已经访问过, 就进行下一层的访问  $B(B(x_i))$ , 并转向步骤 2; 如果都访问过, 则转向步骤 4.

(4) 如果还有其他结点没有访问过, 则在未访问过的结点中, 选取其中  $Z$  轴坐标值最大的结点, 转向步骤 1, 否则结束.

这样对于边界结点, 如根及叶子结点, 可以保证切平面的光滑, 用同样的方法也可以解决局部孤岛问题.

### 1.5 三角面片的输出

Marching Cubes 算法是用于三维数据场等值面生成的经典算法, 也是体素单元内等值面抽取技术的代表. 该算法的基本思想是: 首先逐个处理数据场中所有的立方体, 再判断立方体的 8 个顶点是否跨越待求等值面, 而对于跨越等值面的立方体, 则采用插值法计算出等值面与立方体边的交点; 最后根据立方体每一顶点与等值面的相对位置, 将等值面与立方体边的交点按一定方向连接生成的等值面, 作为等值面在该立方体内的一个逼近表示.

由于 8 个顶点在等值面之上及之下有  $2^8 = 256$  种可能, 改进的算法是对 8 个立方体顶点及 12 条边进行编号, 即使用一个边表来记录对等值面与边的交集, 并用三角形表按顺序记录对本立方体输出的三角形有贡献的边, 这两个常量表示如下:

(1) 边表, 记录等值面与 12 条边的位置:

```
Bit12 edgeTable[256];
```

(2) 三角形表, 记录输出的三角形网格的边:

```
int triTable[256][16];
```

查表法的详细步骤如下:

(1) 在  $l \times m \times n$  个立方体中, 选取一个未处理过的立方体, 根据式(1)计算每个顶点的距离值  $f$ , 其中,  $f$  的值为 1, 0, undefined, 分别表示该立方体在曲面上方、曲面下方、没有定义。

(2) 如果  $f$  不等于 undefined, 则转向第 3 步, 否则转向第 1 步。

(3) 根据 8 个顶点的  $f$  值得到 0 与 1 数字串, 设与该串对应的十进制值为  $x$ , 则通过查找  $edgeTable[x]$ , 即得到曲面与本立方体相交的边。

(4) 计算  $edgeTable[x]$  中的  $y$  值, 然后查找  $triTable[y]$ , 再通过线性插值来计算曲面与立方体

交点。

(5) 输出本立方体的三角面片。

## 2 实验结果分析

本文算法已用 Visual C++ 6.0 在 PⅢ 700 的 PC 机上实现, 输入的数据来源于从 Internet 下载的模型的表面三维点集, 本文算法与文献[11]算法比较的输出结果见图 1、图 2。由图 1、图 2 可见, 本文算法重构的表面与实际模型接近, 也消除了“孤岛”问题, 而文献[11]的算法仍存在“孤岛”问题(图 1(d)、图 2(d))。

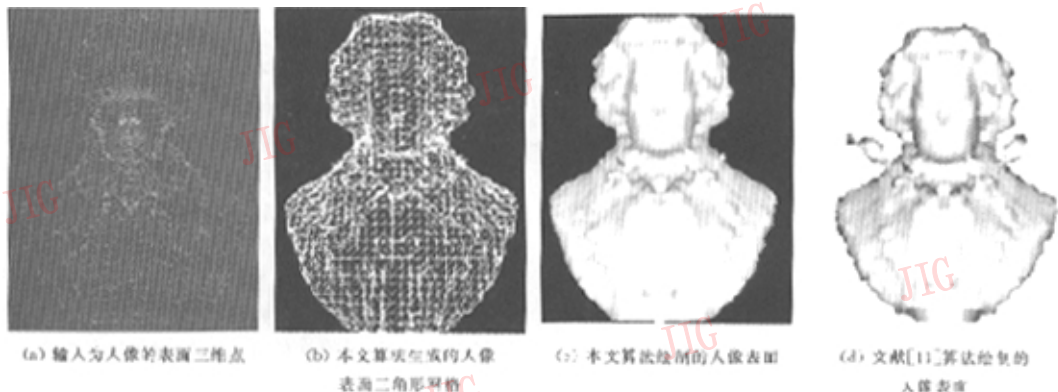


图 1 人像表面三维重构不同算法的实验比较

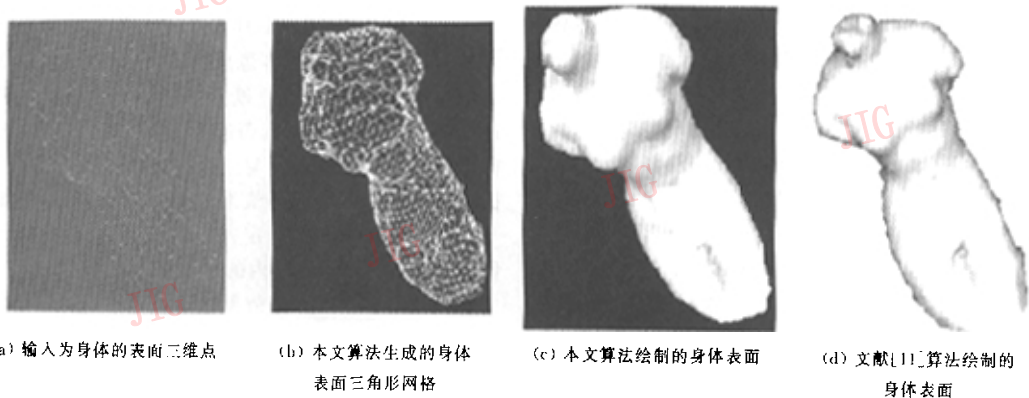


图 2 身体表面三维重构不同算法的实验比较

根据 1.3 及 1.4 节提出的算法与文献[11]算法的复杂度比较如表 1 所示。

对 Marching Cubes 算法的改进, 由于在查表上

进行了优化, 从而加快了运算速度(如表 2 所示)。本文提出的算法还可以自动估算  $r + \delta$ , 而在文献[11]中则需要人为指定。

表1 不同算法时间复杂度比较

比较内容	文献[11]算法的时间复杂度	本文算法的时间复杂度
查找 $B(X)$	$O(n^2)$	$O(n \lceil \log k \rceil)$
法向调整	$O(n^2)$	$O(n \log n)$

表2 运行实例

实例	点数	生成三角形数	单位:s		
			文献[11]算法所用时间	本文算法所用时间	传统MC 本文MC 时间 运算时间
人像	2 655	6 133	8.2	5.1	3.0 1.4
人身	711	1 602	5.1	3.1	1.6 0.9

注：表中的MC是Marching Cubes的简写。

### 3 结论

本文提出了一个快速、稳定的散乱点表面模型重构算法,用该算法进行表面三维重构时,首先根据输入的三维散乱点来自动计算包围盒的高度,同时求取邻域和计算切平面的法向;然后采用广度优先的方法来调整法向,以使其指向表面的外侧;最后,通过改进的Marching Cubes算法来输出表面的三角面片.本文算法已经成功应用于“面向人脸的三维重建”系统中,实验结果表明,本文算法不仅效率较高,而且可视化效果也较好.另外,它还能应用于三维断层扫描及MRI获取的体数据.

### 参考文献

- 1 Joe B. Delaunay triangular meshes in convex polygons [J]. SIAM J. Sci. Stat. Comput., 1986,7(2):514~539.
- 2 Joe B. Delaunay versus max-min solid angle triangulations for three-dimensional mesh generation[J]. Int. J. Numer. Methods Eng., 1991,31:987~997.
- 3 Joe B. Three-dimensional triangulations from local transformations [J]. SIAM J. Sci. Stat. Comput., 1989,19(4):718~741.
- 4 Shewchuk J R. A condition guaranteeing the existence of higher dimensional constrained Delaunay triangulations [A]. In: Proceedings of the Fourteenth Annual Symposium on Computational Geometry[C], Minneapolis, Minnesota, USA, 1998:76~85.
- 5 Shewchuk J R. Sweep algorithms for constructing higher dimensional constrained Delaunay triangulations [A]. In: Proceedings of the Sixteenth Annual Symposium on Computational Geometry[C], Hong Kong, 2000:350~359.
- 6 杨秋,徐永安,陈其明等. 三维约束Delaunay三角化的研究[J]. 计算机辅助设计与图形学学报, 2000,12(8):590~594.
- 7 Barber C B, Dobkin D P, Huhdanpaa H T. The quickhull

algorithm for convex hulls [J]. ACM Transactions on Mathematical Software, 1996,22(4):469~483.

- 8 崔汉国,胡瑞安,金端峰等. 三维任意区域中点集的三角剖分算法[J]. 计算机辅助设计与图形学学报, 1995,7(2):103~108.
- 9 王青,王旭涛,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.
- 10 周焰,李德华,陈振羽等. 三维物体表面三角划分的快速算法[J]. 中国图象图形学报, 2000,5A(9):764~768.
- 11 Hoppe H, DeRose T, Duchamp T, McDonald J et al. Surface reconstruction from unorganized points[J]. Computer Graphics (SIGGRAPH'92 Proceedings), 1992,26(2):71~78.
- 12 Hoppe H. Surface reconstruction from unorganized points [D]. University of Washington, USA, 1994.
- 13 周儒荣,张丽艳,苏旭等. 海量散乱点的曲面重建算法研究[J]. 软件学报, 2001,12(2):249~255.
- 14 Lorensen W E, Cline H E. Marching cubes: a high resolution 3D surface construction algorithm [J]. Computer Graphics, 1987,21(4):163~169.
- 15 Lorensen W E, Cline H E. Two algorithms for the 3D reconstruction of tomograms [J]. Medical Physics, 1998,15(3):225~233.



梁荣华 1974年生,浙江大学计算机科学与工程系博士研究生,杭州电子工业学院教师.研究方向为计算机视觉、计算机图形学、人工智能等.



陈纯 1955年生,1990年获浙江大学计算机应用专业博士学位,现为浙江大学计算机系主任,教授,博士生导师.研究方向为计算机图形图像处理、计算机视觉、人工智能、CAD/CAM、CSCW等.



潘志庚 1965年生,1993年获浙江大学计算机应用专业博士学位,现为浙江大学CAD&CG国家重点实验室研究员,博士生导师.研究方向为虚拟现实、多媒体计算、分布式图形、汉字信息处理、电子商务等.



张慧 1977年生,浙江大学计算机科学与工程系硕士研究生.研究方向为计算机视觉、计算机图形学.