

基于邻域重心约束的点云模型光顺算法*

杨振羽⁺, 郑文庭, 彭群生

(浙江大学 CAD&CG 国家重点实验室, 浙江 杭州 310027)

A Neighborhood Centroid Constrained Fairing Algorithm for Point Clouds

YANG Zhen-Yu⁺, ZHENG Wen-Ting, PENG Qun-Sheng

(State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027, China)

+ Corresponding author: Phn: +86-571-87951045, E-mail: aediles@hotmail.com, <http://www.cad.zju.edu.cn>

Received 2004-04-05; Accepted 2004-07-05

Yang ZY, Zheng WT, Peng QS. A neighborhood centroid constrained fairing algorithm for point clouds. *Journal of Software*, 2004,15(Suppl.):230~238.

Abstract: A fairing algorithm with centroid constraints of neighborhood is presented for point-sampled geometry. By optimizing a local function based on the centroid constraints of neighborhood, local smoothing is achieved without surface reconstruction. This method overcomes the problems of extreme shrinkage and extreme convergence arisen from Laplacian smoothing operator so that it achieves little distortion. The experimental results verify that it is stable, fast and easy-to-use.

Key words: fairing; centroid constraint; neighborhood; point cloud

摘要: 从统计的角度分析点云模型的光顺除噪,并提出了一个针对随机噪声的邻域重心约束光顺算法.此方法通过极小化邻域重心约束的局部邻域目标函数达到局部光顺的目的,不需要进行表面重构并保证迭代过程中点云模型主体形状不变,有效地克服了拉普拉斯算子所造成的过度收缩与聚集现象.实验表明此方法不仅计算稳定、速度快,而且简便易用.

关键词: 光顺;重心约束;邻域;点云

在计算机图形学、科学计算可视化以及几何造型领域内,随着实体模型复杂度的不断增加,单纯用人工交互的方法生成模型工作量巨大,而用代数描述生成的模型又难以表达真实的表面细节.三维扫描设备由于具有获取信息量大、可表现丰富细节等优点,已经成为具有复杂几何外形景物建模的有效工具.然而,由于三维扫描设备获取的点云数据不可避免地存在测量误差与噪声,难以被直接应用;对点模型进行大规模压缩也会降低模型精度并生成大量的噪声,这在一定程度上降低了其应用价值.针对这两个问题,研究者提出多种算法来处理点云数据,如 MLS(Moving Least Square,移动最小二乘)方法^[1]与基于 Scale-Space 理论^[2]的拉普拉斯方法^[3]等.

* Supported by the National Natural Science Foundation of China under Grant No.60103017 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant Nos.2002CB312101, 2002CB312102 (国家重点基础研究发展规划(973))

作者简介: 杨振羽(1977-),男,浙江诸暨人,硕士,主要研究领域为计算机图形学;郑文庭(1974-),男,博士,副研究员,主要研究领域为计算机图形学,虚拟现实;彭群生(1947-),男,教授,博士生导师,主要研究领域为计算机图形学,虚拟现实,红外仿真成像技术.

在此之前,已有研究者对多边形网格的几何处理做了不少研究工作^[4-11].Taubin 在 1995 年将 Laplace 算子扩展到三角形网格模型^[4],并在此基础上为网格曲面设计了各种多项式线性滤波器,解决了网格曲面的磨光问题.这种方法不仅应用于三角形网格模型,也可以直接应用于基于点云表示的实体.Vollmer 提出了一种新的多边形网格 Laplace 光顺算法^[6],此方法将磨光后的顶点推向先前的位置来避免变形与收缩的问题.刘新国针对三角网格模型提出了一个基于三角形重心约束的 Laplace 光顺算法^[12].该方法将磨光问题转化为一个基于三角面片重心约束的能量泛函的极小化问题,并通过局部求解泛函能量最小来解决大规模三角网格模型的约束迭代问题.

与多边形网格相比,无拓扑无邻接关系的点云模型的光顺算法较少,主要有以下两大类:MLS 方法和 Laplace 方法.前者以 Alexa 等人^[1]为代表,他们用 MLS 方法逼近点云模型表面得到局部逼近的多项式曲面,并以此定义光滑的二维流形表面来表示点云.MLS 曲面被用来进行加密采样以及简化采样,并在绘制时动态加密采样来得到光滑的表面显示效果.此 MLS 方法实际上是一个光顺重采样方法.后者以 Pauly 等^[13]为代表,他们以 Laplace 光顺算子为基础来光顺点云并设计了一个多分辨率点模型造型框架,这个框架通过体积约束来解决 Laplace 光顺算子的收缩问题.

本文提出了一种基于邻域重心约束的点云模型光顺算法.接下来首先介绍一些相关概念并从理论上分析了点云表示模型的 Laplace 光顺除噪,然后介绍基于邻域重心约束的光顺算法,最后给出实现结果并总结全文.

1 预备知识与理论依据

点云模型的光顺造型问题可以描述如下:给定一个由含有几何噪声的点云表示的模型表面,光顺造型的目标是生成一个表面光滑、与原始模型表面尽可能接近的新点云.由于点云无拓扑邻接关系,光顺造型不仅是一个除噪过程,也是一个模型表面的光顺重建过程.

1.1 基本定义

点云(point cloud):密集三维点集,或称为密集离散点集.点云均以集合 $P = \{p_i\}_{i \in \text{Index}(P)}$, $p_i \in R^3$ 表示,其中 $\text{Index}(P)$ 代表采样点的索引集(即下标集,通常以 $[1, n]$ 表示,其中 $n = |P|$ 为集合 P 内元素个数).

邻域结构:由于离散点云无拓扑无邻接关系,唯一可以利用的是一个松散的邻域结构.对于采样点 $p \in P$,定义其邻域 $\text{Nbhd}(p)$ 为一个索引集,即 $\text{Nbhd}(p) \subset \text{Index}(P)$.对于任何 $p_i, i \in \text{Nbhd}(p)$, p_i 满足邻域条件.特别地,每个采样点的邻域 $\text{Nbhd}(p)$ 只跟 p 相关,跟邻域内任何别的采样点的邻域 $\text{Nbhd}(p_i), i \in \text{Nbhd}(p)$ 无关.

k -最近邻域: $\text{Nbhd}(p)^k$ 定义为与采样点 p 最近的 k 个采样点的索引集合(如果 $p \in P$ 则包含其自身),这里采用的距离为欧氏距离.设 σ 为满足下列条件的置换算子: $\|p_{\sigma(1)} - p\| \geq 0$ 并且 $\|p_{\sigma(i)} - p\| \leq \|p_{\sigma(i+1)} - p\|, i \in [1, n-1]$, 则 $\text{Nbhd}(p)^k = \{\sigma(1), \sigma(2), \dots, \sigma(k)\}$. $\text{Nbhd}(p)^k$ 定义了一个中心位于 p 、半径为 $r_p^k = \|p_{\sigma(k)} - p\|$ 的球 s_p^k , 当且仅当 $i \in \text{Nbhd}(p)^k$ 时 p_i 位于 s_p^k 内.通常 k 取值为 8 到 20 之间, k 取值太小或者太大都不能很好地表示局部特性.

1.2 Laplace光顺算子

Laplace 光顺算子主要应用于三角网格的磨光,这个磨光过程可以看作是一个扩散过程^[5]:

$$\partial S / \partial t = \mathcal{L}(S) \quad (1)$$

通过在时间轴上的积分,噪声能量很快地扩散到它的邻域中,使得整个模型变得光滑起来.如果采用显式的 Euler 积分方法,有 $S^{n+1} = (1 + \Delta t \cdot \mathcal{L})S^n$. 于是可以得到一系列越来越光滑的模型表示(S^0 表示初始模型表示, S^n 表示第 n 次迭代后的模型表示).

1.3 空间分解

Laplace 算子是一个线性算子,如果此算子同时是一个与作用域无关的算子(与三维离散点的空间位置无关),我们定义其为 L^0 算子.对于三维空间的每一维, L^0 都是一个线性算子并且各维之间线性无关.对于一个邻域 $\text{Nbhd}(p)$, 可以将空间 R^3 分解为两个子空间的直和:

$$R^3 = N \oplus S^2 \quad (2)$$

其中 N 为邻域在 p 点处沿法向量方向的一维空间,而 S^2 为过 p 点处的二维切平面空间.因此对此邻域的 Laplace 迭代可以分解为两个相互独立的子空间上的迭代.

1.4 噪声分布

对于一个带噪声 X 的点云模型,其数学描述如下:

$$P_{noised} = P_{origin} + X \quad (3)$$

对于单个噪声点 $p = (p_N, p_{S^2})$,经过 L^0 算子作用其位置被邻域内其他点矫正到邻域 $Nbhd(p)$ 的重心附近,

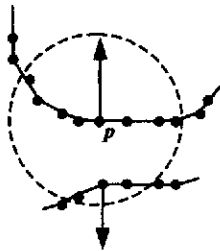


图 1 歧义的邻域

而噪声能量被分散到邻域内其他点.对于非均匀分布的点云以及掺杂了几何噪声的点云来说,邻域重心通常不与邻域结构的中心点 p 重合,这导致 p 偏离原来的位置并向点云密集处偏移,从而造成“马太效应”.

从统计的角度看,假设邻域 $Nbhd(p)$ 近似平面分布,而噪声呈 $X \sim N(0, \sigma^2)$ 分布,则邻域重心的噪声分布为 $\bar{X} \sim N(0, \sigma^2 / |Nbhd(p)|)$.邻域规模越大,邻域重心的噪声就越小.对于 L^0 算子,如果固定邻域重心,则邻域内所有点在 N 空间的分量将向重心处聚集,从而达到光顺的效果. p 在 N 空间内的投影点被不断迭代矫正到邻域重心在 N 空间内的投影处,因此迭代后的点云更接近于原始表面,而且邻域规模越大光顺效果越好.此外,约束邻域重心能有效地制止点云在 S^2 空间内的过度聚集.

1.5 与多边形网格光顺比较

相对于多边形模型,对点云做光顺处理无须考虑连接噪声(connectivity noise)与拓扑噪声(topology noise),而只需考虑几何噪声(geometry noise)即可.这使得光顺处理的工作量减少,却同时给光顺处理增加了很大的难度.与三角网格不同,缺乏拓扑关系与连接关系使得点云表面无法用代数方式精确表达.

对于离散的多边形网格曲面,设 $Adj(i)$ 代表与顶点 v_i 有邻接关系的顶点的集合,也称为顶点 v_i 的 1-环邻域.顶点 v_i 及其 1-环邻域张成一个局部 2 维流形表面.但对于离散点云表示的实体来说,采样点 p 及其邻域 $Nbhd(p)$ 不能张成一个精确的局部 2 维流形表面.这是因为无法确定 $Nbhd(p)$ 中的点是否属于同一表面(图 1),噪声存在的时候这个问题更加明显.

邻域太小缺乏足够的表面信息,邻域过大又容易导致歧义:当两个表面很接近时,过大的邻域会同时含有这两个表面的点(图 1),此时光顺算法会误将这两个表面合并成同一表面.另外,邻域过大导致特征不明显,特征很容易被当成噪声而被磨光.我们的经验是对于比较小的噪声,邻域大小取值为 8~20 时光顺效果比较好.

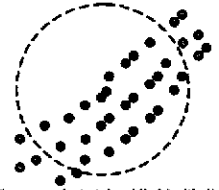


图 2 多层扫描的数据

1.6 应用于扫描数据

三维激光扫描获取的点云模型通常是多次扫描的合成结果.由于没有经过事先矫正,各扫描层之间往往有一定的差距(图 2).如果选取邻域足够大,同时包括了多个扫描层,则基于邻域约束的光顺算法会将这些离散层压缩到一个光滑的表面.因此只要选取适当的邻域,基于邻域约束的光顺算法可以在一定程度上解决拼接错位的问题.

2 邻域重心约束光顺算法

各种形式的离散 Laplace 算子的成功之处在于它们能够光顺大型的多边形网格.它们不要求解复杂的优化方程组,而只是对网格不断地施以 Laplace 算子进行磨光迭代,在时间和空间上仅具有线性复杂度.只要选取适当的邻域,Laplace 算子也能应用于点云的光顺,迭代的时间与空间复杂度也同样具有线性关系.为维持算法的稳定性,此邻域结构在迭代过程中一直保存.

2.1 邻域选取

本文所采用的邻域带有距离约束.给定 d_{max} 为最大允许距离约束, d 为初始距离约束值.对每一个点 p ,先求出以 p 为中心、 d 为半径包围球内的所有点 $Nbhd_d(p) = \{i | \|p_i - p\| < d\}$,然后再对 $Nbhd_d(p)$ 中的点作一次 k -最近邻域查找(图 3,虚线圈表示初始包围球,实线圈表示球 s_p^k). 我们有: $Nbhd(p) = Nbhd(p)^k \subseteq Nbhd_d(p)$. 如果 $|Nbhd_d(p)| < k$, 则此点 p 的邻域 $Nbhd(p)$ 为只含有 $|Nbhd_d(p)|$ 个点 $Nbhd(p)^{|Nbhd_d(p)|}$.

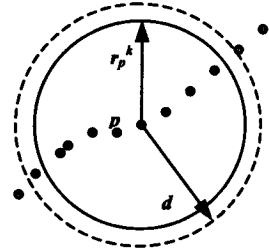


图 3 带距离约束的邻域

2.2 局部求解算法

我们采用^[12]中所用的局部求解算法来求解单个邻域 $Nbhd(p_i)$ 的带约束的泛函极小化问题:

$$E(Nbhd(p_i)) = \sum_{j \in Nbhd(p_i)} |L(p_j)| = \|AP - Q\|^2 = (P^T A^T - Q^T)(AP - Q) \tag{4}$$

其中 $A = \begin{bmatrix} 1 & \omega_{01} & \cdots & \omega_{0k} \\ \omega_{10} & 1 & \cdots & \omega_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_{k0} & \omega_{k1} & \cdots & 1 \end{bmatrix}, P = \begin{bmatrix} p_{i0} \\ p_{i1} \\ \vdots \\ p_{ik} \end{bmatrix}, Q = \begin{bmatrix} q_{i0} \\ q_{i1} \\ \vdots \\ q_{ik} \end{bmatrix}, k = |Nbhd(p_i)|, q_{im} = - \sum_{j \in R(i_m)} \omega_{mj} p_j,$

$$R(i_m) = Nbhd(p_m) - Nbhd(p_i) - \{i_m\}$$

重心约束条件为: $BP = C$, 其中 $B = [1 \ 1 \ \cdots \ 1], C = BP^0, P^0 = [p_{i0}^0 \ p_{i1}^0 \ \cdots \ p_{ik}^0]$

此时条件极值可以简化表示为:
$$\begin{cases} \min (P^T A^T - Q^T)(AP - Q) \\ BP = C \end{cases} \tag{5}$$

用 Lagrange 乘子法解得
$$P = A^{-1}Q + \lambda A^{-1}(BA^{-1})^T \tag{6}$$

其中
$$\lambda = \frac{C - BA^{-1}Q}{\|BA^{-1}\|^2} \tag{7}$$

矩阵 A 中权因子 ω_{ij} 选取如下:
$$\omega_{mj} = \begin{cases} \frac{-1}{|Nbhd(p_m) - \{i_m\}|} & j \in Nbhd(p_m) - \{i_m\} \\ 0 & \text{Others} \end{cases} \tag{8}$$

这样这个局部 Laplace 算子为 L^0 算子.一般情况下同一个点分属于多个邻域结构,每次局部迭代将产生新的不同位置,最终迭代生成的新位置取各次迭代结果的平均.整个过程可以用一个算子简单描述: $P^{n+1} = L_l(P^n)$.

2.3 邻域控制

当且仅当每一行的和为 0 时矩阵 A 奇异(见附录).此时对所有 m ,有 $Nbhd(p_m) \subseteq Nbhd(p_i)$,即此邻域群是自包含的.这说明此邻域群孤立于其他邻域结构.错误或者噪声过大均会导致上述情况发生,此时有两种不同的处理方法可供选择:一是将邻域内的点当成错误点全部剔除,另外一种办法是适当增大距离约束 d 为 d_c 并重构此邻域群,使得新生成的邻域群同附近其他邻域结构发生关系.

根据 3.1 的定义, d_{max} 为最大允许距离约束, d_c 为包含当前 $Nbhd(p)$ 的邻域 $Nbhd_{d_c}(p)$ 的距离约束,如果 $Nbhd(p)$ 定义的矩阵 A 奇异,则通过如下策略来控制邻域并剔除错误:

- (1) 如果 $d_c > d_{max}$, 将 $Nbhd(p)$ 中所有点判别为错误点并剔除;
- (2) 如果 $d_c \leq d_{max}$, 则对于 $Nbhd(p)$ 中每个点 p_i , 取 $d_{ci}' = \min(2d_{ci}, d_{max} + \varepsilon)$ 并建立新的邻域 $Nbhd(p_i) = Nbhd_{d_{ci}'}(p_i)$, 其中 $\varepsilon = 0.001d_{max}$. 如果新生成 $Nbhd(p)$ 的矩阵 A 非奇,则保留这些点;否则转 1.

在绝大多数情况下,只要邻域足够大,矩阵计算是稳定的.

2.4 局部光顺

对于众多造型场合,往往只需要局部光顺即可.这可以通过简单的初始点约束来实现:

$$P^{n+1} = (I - A)P^0 + AL_1(P^n) \quad (9)$$

其中 $A = \text{diag}\{\alpha_1, \alpha_2, \dots, \alpha_{|P|}\}$, $\alpha_i \in [0, 1] (i=1, 2, \dots, |P|)$. α_i 为初始点约束参数,由用户确定. α_i 取 1 时对点 p_i 作无约束磨光, α_i 取 0 时 p_i 不磨光.对于非封闭的点云模型,迭代会导致边界处点沿表面切线方向向内漂移,此时我们可以设这些边界点 α 值为 0 来解决这一问题,也可以选取适当的 α 值来达到某种折中.

2.5 算法实现

数据定义:

<i>Porigin</i> :	原始点云位置数组;	<i>Pold</i> :	迭代前点云的位置数组;
<i>Pnew</i> :	迭代后新点云位置数组;	<i>New</i> :	记录点云位置的临时数组;
<i>Cnew</i> :	记录点云新位置个数的临时数组;	<i>Alpha</i> :	原点约束权值数组;
<i>k</i> :	记录邻域大小的临时变量;		

单步迭代算法流程:

```

Pnew = 0;    Cnew = 0;    /* 初始化 */
for each point p in P {
    k = |Nbhd(p)|;        if (k <= 1)    continue;
    根据邻域结构 Nbhd(p)以及数组 Pold 初始化 A, P, Q;
    用邻域重心约束算法计算邻域内各点的新位置 New[i1], New[i2], ..., New[ik];
    for (j = 1; j <= k; j++) {
        Pnew[ij] += New[ij];    Cnew[ij] ++;
    }
}
for each point p in P
    if (! Cnew[i])    Pnew[i] = Pnew[i] / Cnew[i];
    else    Pnew[i] = Pold[i];
for each point p in P    Pnew[i] = (1 - Alpha[i]) * Porigin[i] + Alpha[i] * Pnew[i];

```

整个迭代过程描述如下:

```

for each point p in P
    if (p 需要磨光)    建立 k-邻域结构 Nbhd(p);
    else    建立只含 p 的邻域结构 Nbhd(p);
邻域控制并剔除错误点;
while (! 迭代中止)    执行单步迭代;

```

3 结果与分析

我们在 PIV2.4G、256M 内存的 PC 机上实现了这个算法,其中矩阵运算部分调用了 Intel 的数学库 Intel Math Kernel Library 6.1 来优化计算,同时用八叉树加速邻域结构的构造.当邻域平均点数为 8~20 时,平均每个邻域的构建时间为 70~150ms.点云模型 Screwdriver, Venus 的来源为 Cyberware 公司提供的模型数据,Boy 模型为自己用三维激光扫描仪扫描获取的数据.表 1 是部分模型的全局光顺实例(图 4、图 5),其中每个模型每个点都被加了一定的高斯噪声并用 Splat 技术绘制.邻域重心约束光顺算法收敛非常快,只需 2、3 步迭代就基本收敛,并且在迭代过程中保持了形体的稳定.如图 4 所示,screwdriver 顶部不仅被光顺,尖锐的形状也得到保持而没有收缩.

表 1 两个光顺实例

Model	Number of points	Average size of neighborhood	Construction time (s)	Single iteration time (s)
Screwdriver	27 152	16	1.05	0.31
		26	2.18	0.73
		36	2.25	1.53
Venus	134 345	16	29	6.40
		26	46.5	17.7

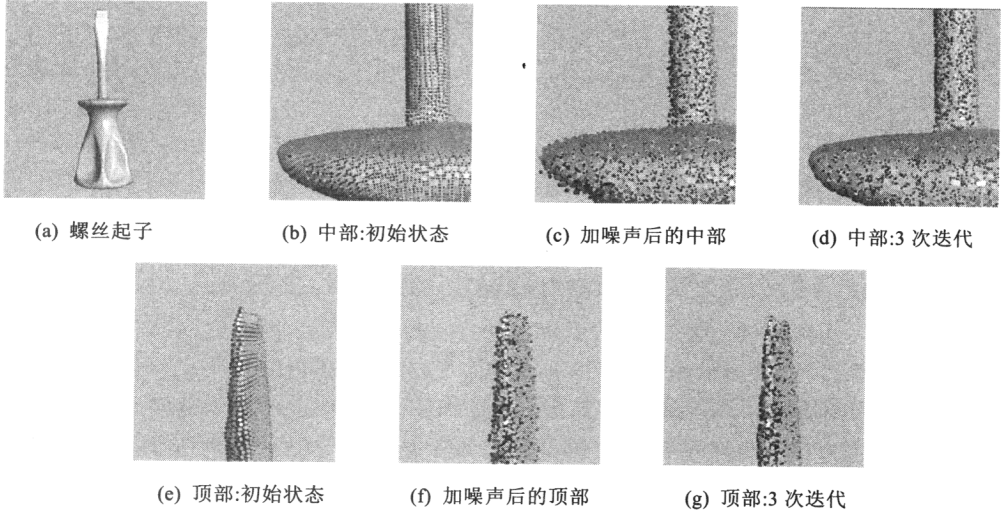


图 4 Screwdriver 的全局光顺

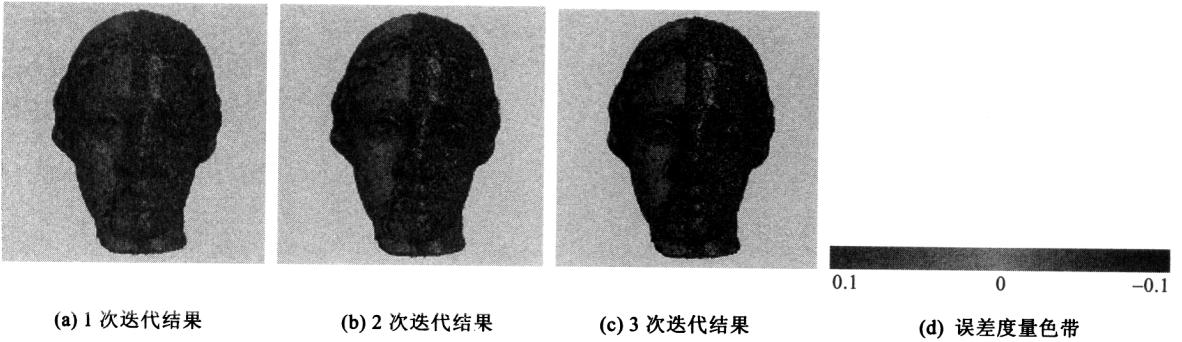


图 5 Venus 的局部光顺(左半部分)

3.1 数据模拟与误差评定

数据点的数值由精确值迭加噪声来模拟:

$$\hat{p}(x, y, z) = p(x, y, z) + E_{scale}(dx, dy, dz) \tag{10}$$

其中 $p(x, y, z)$ 为输入模型的位置,从文件读取; dx, dy, dz 为模拟高斯分布的随机数,满足 $-1 \leq dx, dy, dz \leq 1$; E_{scale} 为噪声幅度.计算精度由相对均方差误差 σ 以及相对最大误差 δ_{max} 来度量.设 T 为待光顺点集索引集,则:

$$\sigma = \sqrt{\frac{1}{|T|} \sum_{i \in T} \left(\frac{\|\hat{p}_i - p_i\|}{E_{scale}} \right)^2}, \delta_{max} = \max_{i \in T} \left\{ \frac{\|\hat{p}_i - p_i\|}{E_{scale}} \right\} \tag{11}$$

除此之外,我们还定义了每个磨光点在 N 空间与 S^2 空间内的相对均方差误差 $N\sigma, S^2\sigma$ 以及相对最大误差 $N\delta_{max}, S^2\delta_{max}$,计算方法类似于 σ 与 δ_{max} .法向空间 N 从文件读取得到,同原始输入点的法向方向一致.若不能读取法向量,则用协方差矩阵特征分析方法^[14]来估算.特别地,在法向空间 N 内的误差计算是有符号的,法向方向即为正向.

表 2 全局光顺误差统计表

Model	Average size of neighborhoods	Noised					
		$N\delta_{\max}$	$N\sigma$	$S^2\delta_{\max}$	$S^2\sigma$	δ_{\max}	σ
Screwdriver	16	0.872	0.141	1.030	0.152	1.030	0.155
	26						
Venus	8	0.949	0.141	1.027	0.151	1.030	0.154
	13						
	20						
After 1 iteration							
Screwdriver	16	0.707	0.052	1.185	0.170	1.186	0.169
	26	0.850	0.072	1.650	0.169	1.652	0.168
Venus	8	0.548	0.032	1.094	0.157	1.099	0.155
	13	0.551	0.032	1.081	0.161	1.082	0.160
	20	0.579	0.035	1.090	0.161	1.097	0.160
After 3 iterations							
Screwdriver	16	0.768	0.059	1.262	0.171	1.262	0.171
	26	0.882	0.084	1.709	0.171	1.712	0.171
Venus	8	0.584	0.025	1.106	0.157	1.107	0.157
	13	0.544	0.030	1.079	0.162	1.080	0.161
	20	0.613	0.038	1.096	0.162	1.106	0.161

3.2 计算结果

图 5 是 Venus 头像的局部光顺的例子,我们只对头像的左边作光顺.其中 $E_{scale} = r_{\max}$, 平均邻域大小为 26.图中红色表示相对误差为正,即点沿法向向外偏,蓝色点反之,其中相对误差的比例基数为 r_{\max} .可以清楚地看到,本文的光顺算法在法向方向上的磨光效果特别好,达到了光顺的效果.在原先比较平坦的地方点基本恢复到原来表面的位置,而在原先突起的地方向内偏移,在原先凹陷的地方向外偏,也就是说不仅点云被光顺,细节特征也被磨光.统计结果见表 2,此时 $E_{scale} = r_{\max}$.图 6 是一个扫描实例,平均邻域大小取值为 30,Boy 模型光顺后达到了理想的效果.

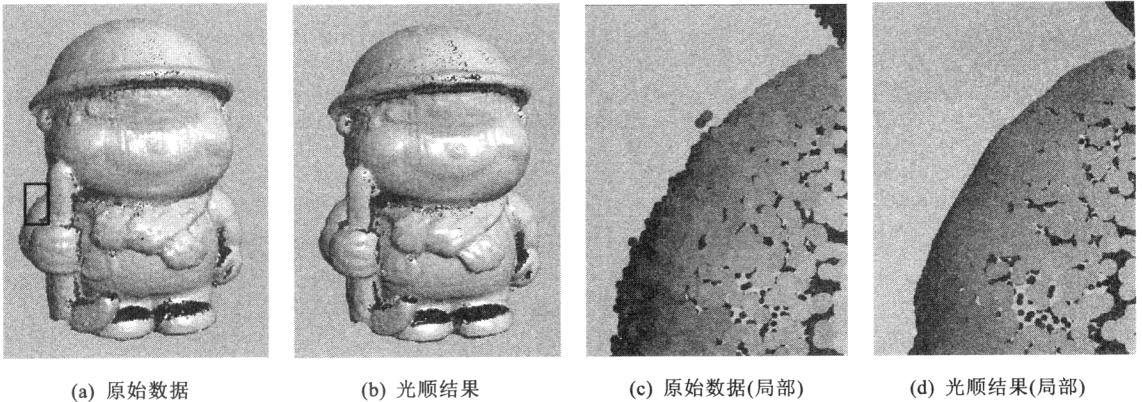


图 6 光顺扫描获取的点云

4 讨论

对于光滑表面,邻域越大则光顺效果越好.反之对于细节丰富的局部区域,邻域越大细节损失也越多,结果反而表现为误差增大.这是因为相对于较大的邻域尺寸,细微特征的起伏也被当成噪声而被磨光.图 5 正好反映了这一点:在光滑的脸部误差越来越小,而在嘴角、眼角以及头发等部位误差反而越来越大,这说明细节特征被光顺而丢失.

L^0 算子是一个各向同性的算子,法向方向的聚集表现为光顺,而水平方向上的聚集表现为采样点的漂移.漂移现象在网格模型中也是一个严重的问题,三角网格模型可以通过顶点间连接的能量最小化来达到顶点均匀分布的效果,而点模型可以通过均匀重采样来解决这个问题.

实际上本文的算法可以很容易推广为一个各向异性的光顺算法:对每个待光顺点 p 的 $Nbhd(p)$,采用^[15]的

方法作协方差分析并计算出 p 处局部表面的法向(Normal) N_p 以及曲面变化度(Surface Variation) $\sigma_n(p)$ 。如果 $\sigma_n(p)$ 小于某个预先给定的阈值 ε_p , 可以认为此局部表面近似于平面, 此时只须对该邻域作 N_p 方向的约束迭代; 如果 $\sigma_n(p) \geq \varepsilon_p$, 可以认为此局部表面处于细节丰富的区域, 对此邻域的约束迭代无方向性限制。这样就能实现各向异性的光顺并能极大地减轻水平漂移现象。

邻域越大对计算时间与空间存储的要求也越大, 因此存在一个时间、空间与光顺效果的平衡点。

5 结束语

本文对 Laplace 算子在点模型信号处理中的应用作了探讨, 并针对点云表示模型的光顺造型提出了独自的看法与算法: 1. 从统计的角度分析点云模型的光顺除噪; 2. 邻域结构为最基本的数据结构, 无需引入其他辅助工具, 也无需进行任何表面重建; 3. 提出了一个针对随机噪声的邻域重心约束的光顺算法, 保证迭代过程中主体形状不变; 4. 通过初始点约束将其推广为局部光顺算法。未来的工作主要集中在特征保持的约束光顺。

References:

- [1] Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva CT. Point set surfaces. In: Ertl T, Joy KI, Varshney A, eds. Proc. of IEEE Visualization 2001. Los Alamitos: IEEE Computer Society Press, 2001. 21~28.
- [2] Linderberg T. Scale-Space Theory in Computer Vision. Norwell: Kluwer Academic Publishers, 1994.
- [3] Pauly M. Point primitives for interactive modeling and processing of 3D geometry[Ph.D. Thesis]. Zürich: ETH Zürich, 2003.
- [4] Taubin G: A signal processing approach to fair surface design. In: Cook R, ed. Proc. of ACM SIGGRAPH'95. New York: ACM Press, 1995. 351~358.
- [5] Desbrun M, Meyer M, Schröder P, Barr AH. Implicit fairing of irregular meshes using diffusion and curvature flow. In: Rockwood A, ed. Proc. of ACM SIGGRAPH'99. New York: ACM Press, 1999. 317~324.
- [6] Vollmer J, Mencl R, Müller H. Improved laplacian smoothing of noisy surface meshes. In: Brunder P, Scopigno R, eds. Proc. of EuroGraphics'99. Milan: Eurographics Association and Blackwell Publishers, 1999,18(3).
- [7] Guskov I, Sweldens W, Schröder P. Multiresolution signal processing for meshes. In: Rockwood A, ed. Proc. of ACM SIGGRAPH'99. New York: ACM Press, 1999. 371~380.
- [8] Kobbelt LP. Discrete fairing and variational subdivision for freeform surface design. The Visual Computer, 2000,16(3/4): 142~150.
- [9] Peng J, Strela V, Zorin D. A Simple Algorithm for Surface Denoising. In: Ertl T, Joy KI, Varshney A, eds. Proc. of IEEE Visualization 2001, Los Alamitos, CA: IEEE Computer Society Press, 2001. 107-112.
- [10] Jones TR, Durand F, Desbrun M. Non-Iterative, feature-preserving mesh smoothing. In: Hart JC ed. Proc. of ACM SIGGRAPH 2003. New York: ACM Press, 2003. 943~949.
- [11] Fleishman S, Drori I, Cohen-Or D. Bilateral mesh denoising. In: Hart JC, ed. Proc. of ACM SIGGRAPH 2003. New York: ACM Press, 2003. 950~953.
- [12] Liu XG. Three dimensional geometry compression [Ph.D. Thesis]. Hangzhou: Zhejiang University, 2001 (in Chinese with English abstract).
- [13] Pauly M, Kobbelt LP, Gross M. Multiresolution modeling of point-sampled geometry. Technical Report, CS #379, Zürich: ETH Zürich, 2002.
- [14] Sagan H. Introduction to the Calculus of Variations. New York: Dover Publications, 1969.
- [15] Pauly M, Gross M, and Kobbelt LP. Efficient Simplification of Point-sampled Surfaces. In: Proc. of IEEE Visualization 2002. Los Alamitos: IEEE Computer Society Press, 2002. 163~170.

附中文参考文献:

- [12] 刘新国. 三维几何压缩[博士学位论文]. 杭州: 浙江大学, 2001.

附录:

定义 D1: 矩阵 $M = (m_{ij})_{n \times n}$ 称为可约的, 若存在置换阵 P , 使得 $PAP^T = \begin{pmatrix} M_{11} & M_{12} \\ 0 & M_{22} \end{pmatrix}$, 其中 M_{11}, M_{22} 为方阵.

定理 T1: 设 $M = (m_{ij})_{n \times n}$ 弱对角占优并且不可约, 则 M 非奇.

证明: 在式(4)中, 对称矩阵 A 的每一行都满足: $\sum_{j \neq i_m} |\omega_{i_m j}| = \sum_{j \in \text{Nbhd}(p_i) \cap (\text{Nbhd}(p_m) - \{i_m\})} |\omega_{i_m j}| \leq \sum_{j \in \text{Nbhd}(p_m) - \{i_m\}} |\omega_{i_m j}| = 1$. 如果

矩阵 A 的每一行上式等号成立, 则根据式(8)矩阵 A 每一行和均为 0, 此时 A 奇异. 反过来若上式对 A 的某一行等号不成立, 则 A 是弱对角占优矩阵, 同时 A 不可约. 由定理 T1, 此时矩阵 A 非奇. 另一方面, A 可约 \Leftrightarrow 存在 $\text{Nbhd}(p_i)$ 的非空真子集 J , 有: 对 $\forall m \in J, n \in \bar{J}, \omega_{mn} = 0$. 由于 $i \in \text{Nbhd}(p_i)$, 对 $\forall j \in \text{Nbhd}(p_i)$, 有 $\omega_{ij} = \omega_{ji} \neq 0$, 因此 A 不可约. 综上所述, A 奇异当且仅当 A 的每一行的和均为 0. □