

点云隐式曲面快速重建算法研究

王连哲^{1**}, 韩俊刚^{1*}, 卢升¹, 唐海鹏², 齐全³, 冯宁宇⁴, 汤少杰⁵¹西安邮电大学计算机学院, 陕西 西安 710121;²南密西西比大学计算机科学与工程学院, 密西西比 哈迪斯堡 39406;³石河子大学信息科学与技术学院, 新疆 石河子 832000;⁴宁夏医科大学总医院耳鼻咽喉头颈外科, 宁夏 银川 750004;⁵西安邮电大学自动化学院, 陕西 西安 710121

摘要 提出一种点云数据隐式曲面高效重建算法。该算法首先基于传统径向基函数隐式曲面重建算法对点云数据进行低解析度、低精度快速插值, 然后采用三线性插值对点云数据进行高解析度、低精度插值, 最后根据欧氏距离确定点云零水平集附近需要处理的区域, 处理过程中只对区域内点云数据进行滤波降噪。与传统方法相比, 本文算法既可以保证曲面重建精度, 又可以缩短计算时间。在头部点云数据的曲面重建过程中, 本文算法能够实现与传统算法相近的精度, 同时使插值运算时间减少 63.21%。

关键词 机器视觉; 点云重建; 隐式曲面重建; 三线性插值

中图分类号 TP391.9

文献标志码 A

doi: 10.3788/LOP202158.0415003

Fast Reconstruction Algorithm of Point Cloud Implicit Surface

Wang Lianzhe^{1**}, Han Jungang^{1*}, Lu Sheng¹, Tang Haipeng², Qi Quan³,
Feng Ningyu⁴, Tang Shaojie⁵^{1*} School of Computer, Xi'an University of Posts and Telecommunications, Xi'an, Shaanxi, 710121, China;² School of Computing Sciences and Computer Engineering, University of Southern Mississippi, Hattiesburg, Mississippi, 39406, China;³ College of Information Science and Technology, Shihezi University, Shihezi, Xinjiang, 832000, China;⁴ Otolaryngological Wards, General Hospital of Ningxia Medical University, Yinchuan, Ningxia, 750004, China;⁵ School of Automation, Xi'an University of Posts and Telecommunications, Xi'an, Shaanxi, 710121, China

Abstract This study proposes an efficient algorithm for an implicit reconstruction of point cloud data. First, the algorithm quickly interpolates the point cloud data with low resolution and low precision on the basis of the traditional radial basis function implicit surface reconstruction algorithm and then interpolates the point cloud data with high resolution and low precision using trilinear interpolation. Finally, the proposed algorithm determines the area near the zero level set of the point cloud data according to the Euclidean distance, and only for the points within the area, the point cloud data are denoised by statistical filtering algorithm. Compared with the traditional method, this algorithm can not only ensure the accuracy of surface reconstruction but also reduce the calculation time. This algorithm achieves the similar accuracy to that of the traditional method and a reduction in the operation time by 63.21% in the surface reconstruction experiment of the head point cloud data.

收稿日期: 2020-06-08; 修回日期: 2020-07-25; 录用日期: 2020-08-11

基金项目: 陕西省教育厅科学研究计划项目(15JK1673)、陕西省自然科学基金项目(2016JM8034, 2020SF377)、宁夏回族自治区科技厅 2020 年自治区重点研发计划项目(2020ZDYF0150)、西安邮电大学创新创业项目(2018SC-03)

* E-mail: hjg@xupt.edu.cn; ** E-mail: lianzhe.wang@foxmail.com

Key words machine vision; point cloud reconstruction; implicit surface reconstruction; trilinear interpolation
OCIS codes 150.6910; 150.1135

1 引言

近年来,随着三维(3D)扫描仪器性价比的不断提高,获取点云数据越来越便捷,而基于三角面片表示的物体 3D 模型复杂性不断增加,处理难度不断增大,点云数据在计算机图形学中变得日益重要。如何精确描述点云数据,进而提取其特征,辅助实际应用,成为当前热门且具有挑战性的课题^[1]。

在点云数据曲面重建过程中,3D 扫描仪器在扫描目标物体时不仅会产生海量点云数据,还会因扫描过程中的种种不可控因素产生各种噪声,进而导致扫描得到的点云数据出现孔洞、变形、缺失等问题。因此,通过点云数据重建目标物体的 3D 模型:一方面,需要选择合适的数学模型,以提高点云数据的描述精度,修补点云数据的缺陷;另一方面,要求算法尽量简单,运算复杂度有效降低,以从稀疏点云中构建出高精度 3D 模型^[1]。目前解决此类问题的主流方法是基于径向基函数(RBF)的隐式曲面点云重建算法。

Franke^[2]于 1982 年采用 RBF 对散乱点云进行插值拟合,此方法应用于小型稀疏点云数据拟合时稳定且精确,但是 RBF 全局支撑性质导致此方法应用于大规模点云时难以精确拟合,且效率低下。Savchenko 等^[3]在 Franke 的方法上加入了一次多项式约束,以提升拟合精度,但效果不佳且计算量增大。2001 年,Carr 等^[4]提出先计算出表面法向量,再根据点云数据的法向量约束进行曲面重建的方法,但该方法计算量大且其法向方向不易计算。Blinn^[5]提出了 Blinn 圆球模型,采用具有等势场值的点集来定义曲面。圆球模型被定义为 $b \exp(-ar^2)$,其中, b 为高度, a 为标准偏差, r 为空间中点到等势场的距离,这种模型有很多种,其等势场大小可用关于 r 的函数 $\Phi(r)$ 来定义。对于 Blinn 模型中的 $\Phi(r) = \exp(-ar^2)$ 也常被称为高斯 RBF,具有全局支撑性质。2004 年,Li 等^[6]提出了基于椭球约束^[7-8]的 RBF 隐式曲面重建算法,适用于重建小型稀疏闭合点云数据。2016 年,Cuomo 等^[9]基于 RBF 隐式曲面拟合算法提出获取问题参数和结果的解决方案以及避免伪影的见解。

三线性插值算法是线性插值的扩展,常应用于图像像素补充和三维模型解析度提升中^[10]。本文

综合了 RBF 隐式曲面重建理论和三线性插值理论,利用带有椭球约束的 RBF 算法对点云进行快速插值,并利用三线性插值避免了传统 RBF 算法计算过程中线性方程组不断增大、求解时间长的问题,可以在提高点云曲面精度,保证重建曲面细节特征不丢失的同时,减少传统 RBF 方法的运算时间。

2 带有椭球约束的 RBF 隐式曲面重建

点云表面重建的常用方法有两种——显式重建和隐式重建。显式重建通常参数化描述^[11]表面的位置,可表示为映射 $S: \mathbf{R}^2 \rightarrow \mathbf{R}^3$,其中, \mathbf{R} 为实数集;隐式重建通常用标量函数的 0 等值面来描述,可表示为映射 $F: \mathbf{R}^3 \rightarrow \mathbf{R}$ 。采用隐式曲面对点云数据进行全局描述,适用于解决碰撞检测、变形、布尔运算等问题^[1]。本文综合 RBF 理论和椭球约束理论,采用 Blinn 提出的高斯 RBF 对点云数据进行曲面重建。

2.1 RBF 隐式曲面拟合

通常基于 RBF 的原始隐式曲面的表达式为

$$F(\mathbf{X}) = \sum_{i=1}^n \lambda_i \Phi_i(\|\mathbf{X} - \mathbf{p}_i\|) + P(\mathbf{X}), \quad (1)$$

式中: $\mathbf{X} = (x, y, z)$ 为任意空间点位置坐标; λ_i 为基于 RBF 的 Φ_i 的待定权值; $\|\cdot\|$ 表示欧氏距离; \mathbf{p}_i 为原始点云中任意点位置坐标, $\mathbf{p}_i = (x_i, y_i, z_i)$, $i = 1, 2, \dots, n$; $P(\mathbf{X}) = c_1x + c_2y + c_3z + c_4$ 为 \mathbf{X} 的待定一次多项式, c_1, c_2, c_3, c_4 为多项式系数,这里要求未知系数满足正交条件,即

$$\sum_{i=1}^n \lambda_i = \sum_{i=1}^n \lambda_i x_i = \sum_{i=1}^n \lambda_i y_i = \sum_{i=1}^n \lambda_i z_i = 0. \quad (2)$$

故(1)式可改写成

$$\begin{bmatrix} \Phi_{11} & \cdots & \Phi_{1n} & x_1 & y_1 & z_1 & 1 \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Phi_{n1} & \cdots & \Phi_{nn} & x_n & y_n & z_n & 1 \\ x_1 & \cdots & x_n & 0 & 0 & 0 & 0 \\ y_1 & \cdots & y_n & 0 & 0 & 0 & 0 \\ z_1 & \cdots & z_n & 0 & 0 & 0 & 0 \\ 1 & \cdots & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (3)$$

简记为

$$\begin{bmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda} \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \quad (4)$$

$$A_{ij} = \Phi_{ij} = \Phi(\|p_i - p_j\|), \quad (5)$$

$$P = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 \end{bmatrix}. \quad (6)$$

通过解此线性方程组,可得 RBF 权值 $\lambda_i (i=1, 2, \dots, n)$ 和一次多项式系数 $c=(c_1, c_2, c_3, c_4)$, 把求得的权值 λ_i 和一次多项式系数 c 代入(3)式,就可得到隐式曲面插值函数 $F(\mathbf{X})$ 。

2.2 带有椭球约束的 RBF 隐式曲面拟合

令 $\mathbf{X}=(x, y, z)$ 代表任意空间点位置坐标,常见的中心二次曲面或抛物面方程可表示为

$$P(\mathbf{X}) = ax^2 + by^2 + cz^2 + 2fyz + 2gxz + 2hxy + 2px + 2qy + 2rz + d = 0, \quad (7)$$

令

$$I = a + b + c, \quad (8)$$

$$J = ab + bc + ac - f^2 - g^2 - h^2, \quad (9)$$

$$K = \begin{vmatrix} a & h & g \\ h & b & f \\ g & f & c \end{vmatrix}, \quad (10)$$

其中 I, K, J 为二次曲面的不变量,当且仅当 $J > 0$, 且 $IK > 0$ 时(7)式为椭球方程。

当 $4J - I^2 > 0$ 时能够确定(7)式为椭球方程,这从几何意义方面易于论证。首先将(7)式严格地转换为标准形式,即

$$Ax^2 + By^2 + Cz^2 + Px + Qy + Rz + D = 0, \quad (11)$$

$$u^3 - Iu^2 + Ju - K = 0, \quad (12)$$

令

$$\rho = \frac{4J - I^2}{A^2 + B^2 + C^2}, \quad (13)$$

则 ρ 在旋转和平移下是不变的。可以证明,当且仅当 $A=B=C, |\rho| \leq 1$ 时,(7)式为球面方程。进一步地,当一个根趋于无穷大或两个根趋于零时, ρ 的值趋于 -1 ,此时由(7)式定义的椭圆形是扁平的。因此可证明 ρ 的值可用于测量椭球的椭圆度, ρ 值越大,二次曲面越接近球形,相反, ρ 值越小,椭圆形越平坦或越长且越薄。

现在假设(1)式中的多项式是二次的,并具有如下形式:

$$P(\mathbf{X}) = d + 2px + 2qy + 2rz + ax^2 + by^2 + cz^2 + 2fyz + 2gxz + 2hxy,$$

即以抛物面(椭球)表达式作为 RBF 拟合函数的约束多项式。在该情况下(4)式中的 c 为

$$c = (d, p, q, r, a, b, c, f, g, h)^T, \quad (14)$$

令 $\{p_i(x_i, y_i, z_i)\}_{i=1}^n$ 表示待拟合的点云,则(4)式中

$$P = (p_1, p_2, p_3, \dots, p_n)^T, \quad (15)$$

其中

$$p_i^T = (1, 2x_i, 2y_i, 2z_i, x_i^2, y_i^2, z_i^2, 2y_i z_i, 2x_i z_i, 2x_i y_i), \quad (16)$$

此时(4)式可写为

$$\begin{bmatrix} \Phi_{11} & \cdots & \Phi_{1n} & 1 & 2x_1 & 2y_1 & 2z_1 & x_1^2 & y_1^2 & z_1^2 & 2y_1 z_1 & 2x_1 z_1 & 2x_1 y_1 \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Phi_{n1} & \cdots & \Phi_{nn} & 1 & 2x_n & 2y_n & 2z_n & x_n^2 & y_n^2 & z_n^2 & 2y_n z_n & 2x_n z_n & 2x_n y_n \\ 1 & \cdots & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2x_1 & \cdots & 2x_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2y_1 & \cdots & 2y_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2z_1 & \cdots & 2z_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_1^2 & \cdots & x_n^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ y_1^2 & \cdots & y_n^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2y_1 z_1 & \cdots & 2y_n z_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2x_1 z_1 & \cdots & 2x_n z_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2x_1 y_1 & \cdots & 2x_n y_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ d \\ p \\ q \\ r \\ a \\ b \\ c \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (17)$$

此时隐式曲面拟合问题就转化为在 $4J - I^2 > 0$ 的约束条件下(4)式的求解问题。

3 点云隐式曲面快速重建

RBF 隐式曲面重建算法本质上是对点云数据

作插值运算。当需要根据点云数据拟合出高解析度、高精度的曲面时,需要为每个采样点增加一个或多个偏移支撑点,这样就增大了线性方程组的规模,但会导致系统计算开销和存储开销增大。为了解决这个问题,本文综合带有椭球约束的 RBF 隐式曲面拟合理论^[6]与三线性插值理论,在实现快速插值提升点云曲面解析度的基础上,根据点云间欧氏距离更新插值所得曲面,保证点云数据细节特征精确。

该算法不需要计算点云数据法向量,提高了效率和精度,适用于解决稀疏点云曲面的重建问题。算法流程如图 1 所示,首先基于带有椭球约束的 RBF 隐式曲面算法对点云进行低分辨率、低解析度插值重建曲面,再使用三线性插值算法进行快速插值,最后根据欧氏距离计算点云曲面的表面区域范围,在点云曲面的表面区域范围内对点云数据进行优化,以确保重建曲面的精度。

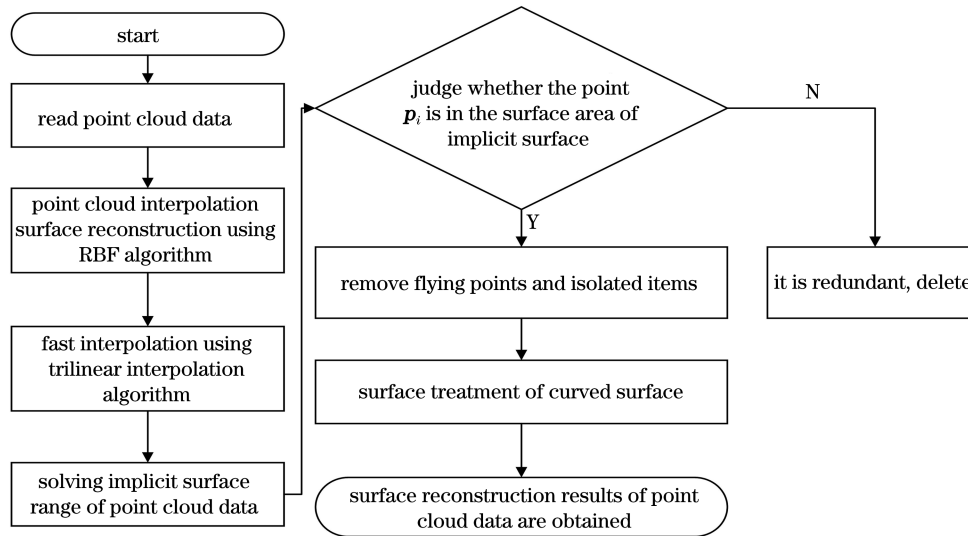


图 1 点云隐式曲面快速重建算法流程图

Fig. 1 Flow chart of fast reconstruction algorithm for point cloud implicit surface

3.1 RBF 隐式曲面快速插值重建算法

在对点云数据使用 RBF 插值建立隐函数曲面的过程中,手动增大 RBF 插值时选取采样点的间隔步长,可以达到减少插值时间的目的。手动增大采样步长的优势在于既缩短了插值时间,又避免了点云数据下采样造成的点云特征丢失,保留了点云数据之间的关联性,对后续处理至关重要。而对于减少 RBF 插值点数带来的空间解析度损失,可以引入三线性插值算法快速对点云隐式曲面进行插值,补充细节,提高点云曲面的解析度。

三线性插值是在 3D 空间上进行多元插值的一种方法。结合点云数据,3D 空间中待插值的点数据需依赖该点四周的点数据进行求解,过程与 3D 空间格点上的函数数据插值过程一致。

对于空间中的 C 点,它的坐标值线性近似于局部轴向矩形棱镜内的中间点 $C(x, y, z)$ 处的函数值。求解具体过程可描述为:在有步距为 1 的周期性立方网格上,如图 2 所示,取 x_d, y_d, z_d 为待计算距离小于 x, y, z 的最大整数的差值,即

$$\begin{cases} x_d = (x - x_0)/(x_1 - x_0) \\ y_d = (y - y_0)/(y_1 - y_0) \\ z_d = (z - z_0)/(z_1 - z_0) \end{cases}, \quad (18)$$

式中: x_d, y_d, z_d 均为单位化之后的值,取值范围在区间 $[0, 1]$ 上; x_0 为 x 下方的晶格点; x_1 为 x 上方的晶格点; y_0 为 y 下方的晶格点; y_1 为 y 上方的晶格点; z_0 为 z 下方的晶格点; z_1 为 z 上方的晶格点。

首先沿 x 轴插值,可以得到

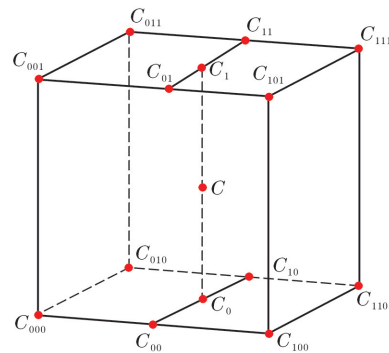


图 2 空间中 C 点插值过程示意

Fig. 2 Interpolation process of point C in space

$$\begin{cases} C_{00} = C_{000}(1 - x_d) + C_{100}x_d \\ C_{01} = C_{001}(1 - x_d) + C_{101}x_d \\ C_{10} = C_{010}(1 - x_d) + C_{110}x_d \\ C_{11} = C_{011}(1 - x_d) + C_{111}x_d \end{cases}, \quad (19)$$

然后,沿着 y 轴插值,可以得到

$$\begin{cases} C_0 = C_{00}(1 - y_d) + C_{10}y_d \\ C_1 = C_{01}(1 - y_d) + C_{11}y_d \end{cases}, \quad (20)$$

最后沿着 z 轴插值,可以得到

$$C = C_0(1 - z_d) + C_1z_d. \quad (21)$$

此时 C 即为点云数据中依据四周点数据插值计算所得的点。在点云曲面重建过程中引入三线性插值算法既可以快速补充点云细节,提升解析度,又可以避免插入大量噪声点和孤立项,为后续曲面表面更新工作节省了大量时间。

3.2 隐式曲面表面优化

在对点云数据隐式曲面快速插值,提升点云解析度之后,为了保证重建所得曲面的精度,需要对插值后的曲面进一步作去噪处理。本课题组在文献[12]中设计了计算不同点云间尺度因子的算法。

给定两个尺度各异的点云,其中点云 $P = \{p_i(x_i, y_i, z_i)\}_{i=1}^{N_p}$ 表示插值结果点云, $Q = \{q_j(x_j, y_j, z_j)\}_{j=1}^{N_q}$ 为原始点云, N_p 和 N_q 分别表示点云 P 与 Q 中的点数且 N_p 和 N_q 可任意取值。定义旋转矩阵 $R = R_a \times R_b \times R_c$, 表达式为

$$\begin{cases} R_a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_a & -\sin \alpha_a \\ 0 & \sin \alpha_a & \cos \alpha_a \end{bmatrix} \\ R_b = \begin{bmatrix} \cos \alpha_b & 0 & \sin \alpha_b \\ 0 & 1 & 0 \\ -\sin \alpha_b & 0 & \cos \alpha_b \end{bmatrix} \\ R_c = \begin{bmatrix} \cos \alpha_c & -\sin \alpha_c & 0 \\ \sin \alpha_c & \cos \alpha_c & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{cases}. \quad (22)$$

1) 分别计算点集 P 和 Q 的质心并记作 μ_p 和 μ_q , 表达式为

$$\begin{cases} \mu_p = \frac{1}{N_p} \sum_{i=1}^{N_p} p_i \\ \mu_q = \frac{1}{N_q} \sum_{i=1}^{N_q} q_i \end{cases}; \quad (23)$$

2) 计算点集 P 和 Q 中点到质心的距离,取其均值

$$\begin{cases} \text{dist}(\mu_p, P) = \sum_{i=1}^{N_p} \|\mu_p - p_i\|^2 \\ \text{dist}(\mu_q, Q) = \sum_{i=1}^{N_q} \|\mu_q - q_i\|^2 \end{cases}; \quad (24)$$

3) 对尺度区间进行粗略估算,尺度因子可表示为

$$s = \text{dist}(\mu_p, P) / \text{dist}(\mu_q, Q). \quad (25)$$

将插值后的点云与原始点云作归一化处理,由文献[12]可知此项操作产生的误差很小 ($10^{-30} \sim 10^{-4}$)。

计算原始点云数据与插值所得点云数据之间的欧氏距离,可以据此设置点云表面的区域范围。对表面区域范围内的点进行去噪和去除孤立项的操作,对点云表面区域范围之外的点直接删除,节省运算时间。令 $P = \{p_i(x_i, y_i, z_i)\}_{i=1}^n$ 表示插值点云, $Q = \{q_j(x_j, y_j, z_j)\}_{j=1}^m$ 表示原始点云, L 表示两点间欧氏距离,则插值点云与原始点云的欧氏距离为

$$L = \|p_i - q_j\|, i = 1, 2, \dots, n; j = 1, 2, \dots, m. \quad (26)$$

选择距原始点云零水平集一定距离的区域,作为点云表面区域,去除区域外的冗余部分,可得插值点云的表面待处理区域,在此区域内进行降噪处理,可缩短降噪处理的运算时间。将经过三线性插值处理后的散乱点云数据表示为具有厚度的点区域,如图 3 所示。

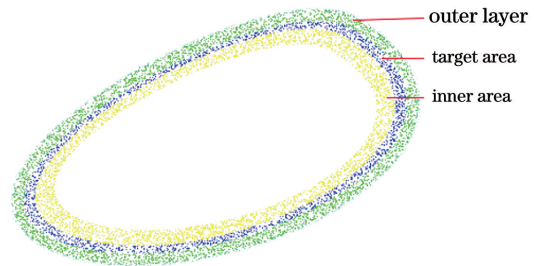


图 3 点云表面区域示意

Fig. 3 Surface area of point cloud

点云降噪是数据预处理中的一项基本操作,并且噪声滤除的效果直接影响点云曲面重建的质量。无论是基于 RBF 对点云数据插值,还是采用三线性插值算法对点云数据插值,都会不可避免地产生噪声点,故必须借助点云降噪算法进行去噪。统计滤波与半径滤波是点云去噪处理的基本方法^[13],李仁忠等^[14]提出了一种基于方法库的点云去噪与精简算法,该算法可有效去除点云模型的不同尺度噪声,其他去噪方法^[15]还有小波去噪、

基于偏微分方程(PDE)的非线性全变分方法和双边滤波等。

由于本文算法是针对小型稀疏点云来实现的,根据几何理论,通过计算有限邻域平均距离,可以较容易地检测出距离较近的点。本研究选择适用于小型稀疏点云数据的统计滤波降噪算法^[13]进行降噪。在点云曲面表面范围内采用统计滤波进行降噪处理,首先读取插值后待处理点云,为统计滤波器设定邻域搜索点数 K 。然后计算每个查询点到 K 个邻域点的平均距离,再求得离散点的筛选范围,进行筛选去噪,并保存去噪后点云和噪声点。最后通过可视化模块观察去噪效果,若去噪效果较差,则调整参数,重新采取统计滤波操作。采用该算法滤除离散点后,所有点的邻域平均距离将保持在设定大小范围内波动。算法流程如图 4 所示。

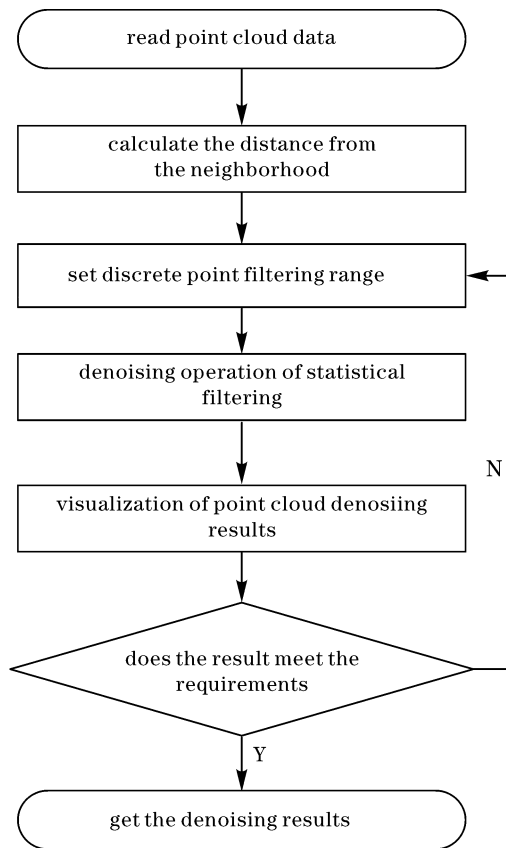


图 4 统计滤波去噪流程图

Fig. 4 Statistical filtering and denoising flow chart

本文先对点云数据进行快速插值,再对插值结果作曲面表面去噪优化,保证重建曲面精度。实验结果证明,在对同一点云数据实现同等解析度和精度的表面重建任务中,相对于只依赖 RBF 的插值算法,本文所提算法可以大幅度缩短运算时间。

4 实验及分析

为验证本文算法的实际效果,以脸部点云为原始数据,与文献[6,10]做了对比实验。实验硬件环境为 Intel i7-8550U, 16 G 内存,软件环境为 Windows10 64 位操作系统、MATLAB2016a、C++ 和 OpenCV。

脸部点云数据共有 2113 个点,空间中点云散点图如图 5(a)、(b)所示,头部点云数据共有 1628 个点,空间中散点图如图 5(c)、(d)所示,斯坦福 Bunny 点云均匀采样处理后共取 1889 个点,空间中散点图见图 5(e)、(f)所示。

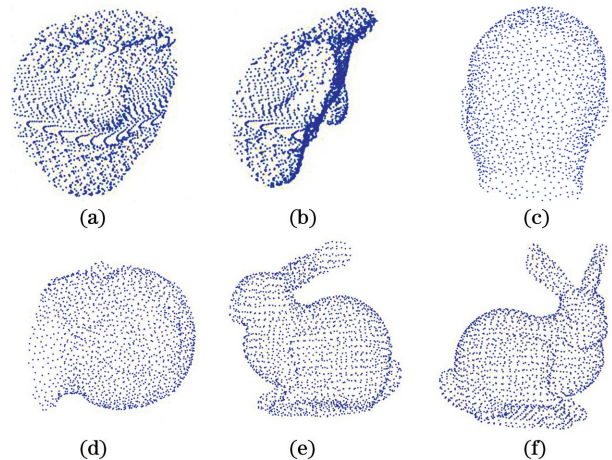


图 5 脸部点云与头部点云展示。(a)脸部点云正面视角;(b)脸部点云右侧视角;(c)头部点云正面视角;(d)头部点云侧面视角;(e)Bunny 点云左侧视角;(f)Bunny 点云右侧视角

Fig. 5 Facial point cloud and head point cloud are shown. (a) Facial point cloud positive perspective; (b) right perspective of facial point cloud; (c) head point cloud frontal view; (d) side view of head point cloud; (e) left view of Bunny point cloud; (f) right view of Bunny point cloud

首先从点云数据插值方面进行分析,在使用不同的采样点间隔步长时,脸部点云与头部点云插值计算的时间如表 1 所示,从表中插值时间的变化可知,减小 RBF 插值的点数可以显著缩短点云数据表面重建运算的时间。

从点云表面重建结果分析:对比图 6(a)、(b)可以看出,本文算法在缩短运算时间的同时,亦可保证重建结果准确;对比图 6(c)、(d)可以看出,本文算法去除了传统 RBF 表面重建算法拟合头部点云时颈部产生的噪点。

表 1 实验选取点云数据插值运算时间

Table 1 Interpolation operation time of point cloud data selected in experiment

Point cloud	Sampling interval	Number of RBF interpolation points	Number of trilinear interpolation points	Total number of points	RBF interpolation time /s	Time spent on trilinear interpolation /s	Total interpolation time /s
Face	1	42260	0	44373	50.1784	0	50.1780
	2	21120	39352	62585	6.7663	0.0709	6.8372
	4	10560	47493	60166	2.2115	0.0739	2.2854
Head	1	3536	0	5164	14.6615	0	14.6615
	2	1768	3877	7273	6.2975	0.0172	6.3147
	4	884	4479	6991	5.3746	0.0183	5.3929
Bunny	1	4120	0	5991	111.2606	0	111.2606
	2	2051	4499	8439	15.2753	0.1451	15.5508
	4	1026	5196	8111	5.0470	0.2410	5.4304

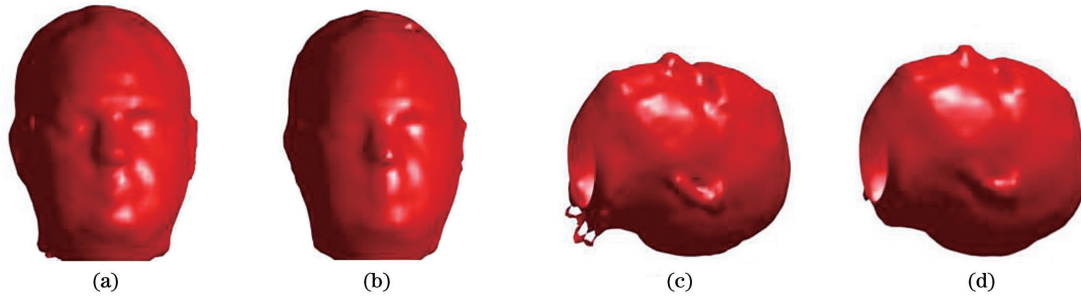


图 6 头部点云重建结果对比。(a)(c)文献[6]方法重建结果；(b)(d)本文算法间隔步长取 6 时重建结果
Fig. 6 Head point cloud reconstruction results contrast. (a)(c) Reconstruction result of method in Ref. [6]; (b)(d) reconstruction result when interval step size of algorithm in this paper is set to 6

图 7(b)中浅色部分为本文算法插值结果,深色点为原始点云,对比图 7(a)可见本文算法在保证插

值精度的同时,去除了头部点云颈部的噪点[图 7(b)中左下角深色点]。

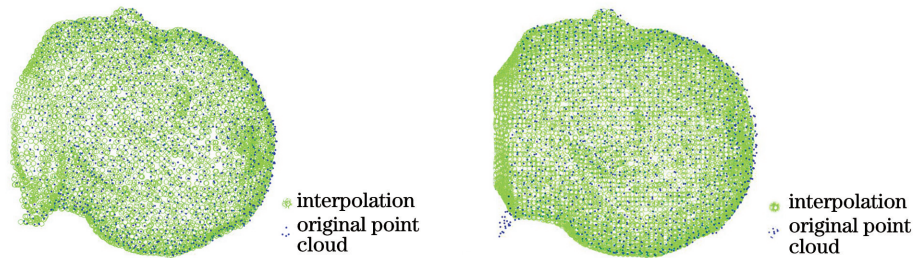


图 7 头部点云插值结果散点图对比。(a) 原始 RBF 算法插值结果；(b) 本文算法插值结果
Fig. 7 Comparison of head point cloud interpolation results in scatter plots. (a) Interpolation result of original RBF algorithm; (b) interpolation result of algorithm in this paper

文献[6]中的方法对所选脸部点云的插值结果如图 8(a)所示,对点云曲面的重建结果如图 8(b)所示,由于带有椭圆约束的 RBF 隐式曲面拟合算法的缺陷,半封闭点云插值重建曲面会出现部分冗余块,这从图 8(a)、(b)中脸部点云表面周围有冗余块可以看出。图 9(a)、(b)分别为采用本文算法,采样步

长取值为 8 时,对脸部点云数据进行插值处理的结果和对脸部点云数据曲面重建的结果。从局部放大图 9(c)可以看到,人脸部点云嘴唇及上颚处特征清晰,脸部点云的细部特征得以保留,由此证明本文算法在缩短曲面重建运算时间的同时,可以保证特征效果良好。图 9(d)为本文算法对斯坦福 Bunny 点

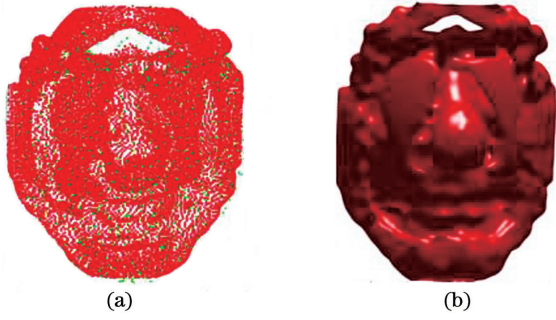


图 8 采用文献[6]方法的脸部点云插值和重建结果。(a)脸部点云数据插值结果;(b)脸部点云曲面重建结果
Fig. 8 Facial point cloud interpolation and reconstruction results were obtained by using the method in Ref. [6]. (a) Interpolation result of facial point cloud data; (b) reconstruction result of facial point cloud surface

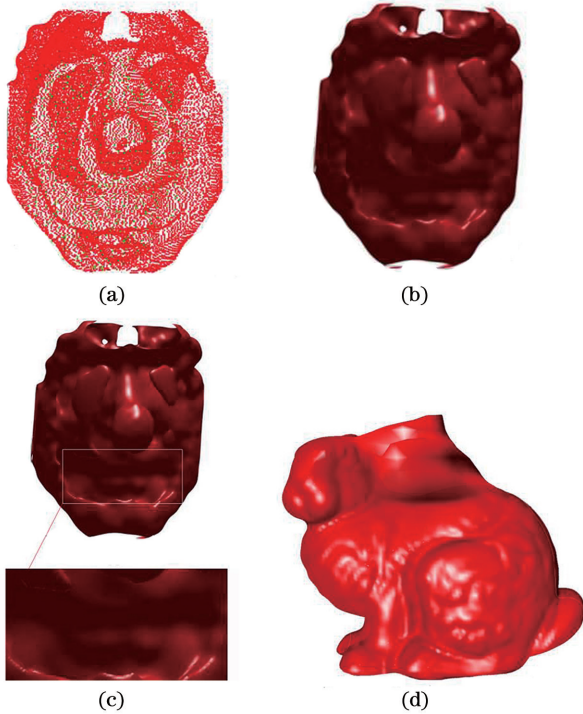


图 9 本文方法的插值和重建结果。(a)本文算法插值结果;(b)本文算法曲面重建结果;(c)实验选取的人脸点云重建结果局部放大;(d)斯坦福 Bunny 点云表面重建结果

Fig. 9 Interpolation and reconstruction results of the proposed algorithm. (a) Interpolation result of the proposed algorithm; (b) surface reconstruction result of algorithm in this paper; (c) local magnification of facial point cloud surface reconstruction result; (d) surface reconstruction result of Stanford Bunny point cloud

云曲面的重建结果,由于实验算法选取椭球约束,兔

子耳朵处产生了冗余块,但身体细节特征得到了较好展现。

图 10(a)、(b)分别为头部点云与脸部点云在选取不同采样间隔时的表面重建运算时间与对比原始 RBF 表面重建偏差变化曲线,图中横轴为不同的采样间隔,值为 1 时为原始 RBF 表面重建算法。

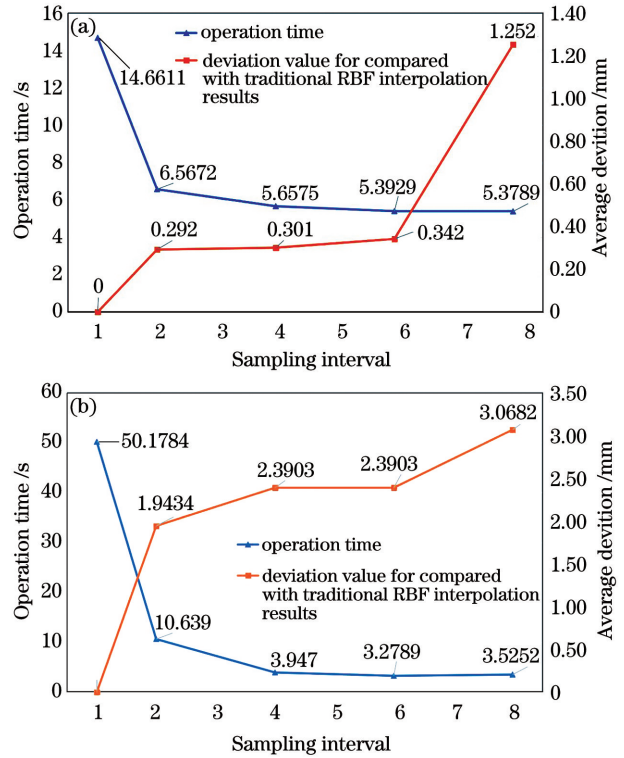


图 10 选取不同采样步长时点云数据曲面重建时间与偏差变化曲线。(a)头部点云的曲线;(b)脸部点云的曲线

Fig. 10 Curves of reconstruction time and deviation of point cloud data surface at different sampling step sizes. (a) Curve of point cloud on head; (b) curve of point cloud on face

从点云曲面重建精度方面分析,采用本文算法,设采样点间隔值为 6,对头部点云进行表面重建,运算时间为 5.3929 s,对比原始 RBF 算法 14.6611 s,运算时间缩短了 63.22%。基于逆向工程技术专业软件 ImageWare 中的“点云偏差计算”功能^[16],与原始 RBF 算法重建结果作对比,绝对平均偏差仅为 0.342 mm。对脸部点云数据重建时,运算时间为 3.9994 s,对比原始 RBF 算法 50.1784 s,运算时间缩短了 92.03%。如图 11 所示,当采样间隔继续增加时,点云插值结果偏差会迅速增大,故本研究取采样间隔为 6 进行实验评估。

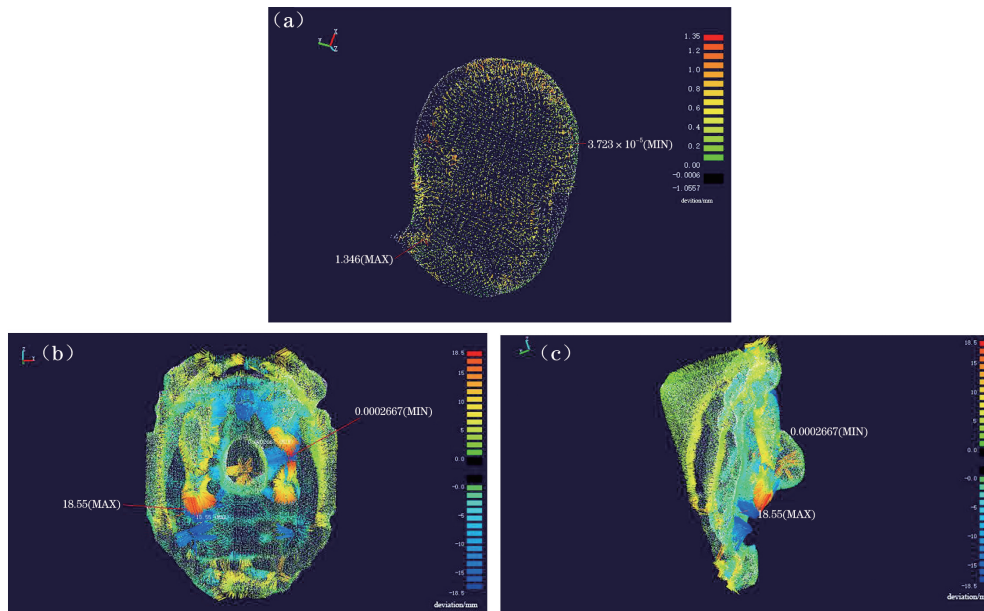


图 11 基于 ImageWare 对点云插值结果的偏差评估图。(a)头部点云;(b)正面脸部点云;(c)侧面脸部点云

Fig. 11 Deviation evaluation diagram of point cloud interpolation results in ImageWare. (a) Head point cloud; (b) positive view of facial point cloud; (c) side view of facial point cloud

5 结 论

基于 RBF 隐式曲面理论改进了点云数据隐式曲面重建算法,在 RBF 算法对点云数据插值重建曲面的过程中引入三线性插值理论,弥补了 RBF 插值过程中线性方程组不断增大、计算时间长的缺陷,同时根据欧氏距离选取点云曲面表面附近区域为待处理区域,在区域内采用统计滤波算法对点云数据进行去噪,保证了隐式曲面的精度。在实验过程中,将本文算法重建的点云数据与原始数据根据尺度因子作归一化处理,导入逆向工程专业软件 ImageWare 中,与传统 RBF 隐式曲面重建算法的插值结果进行点云偏差计算,结果表明,本文算法在不损失点云曲面重建精度的同时,能大幅度缩短原始 RBF 算法的运算时间。本文算法基于带有椭球约束的 RBF 实现,适合于解决小型稀疏闭合点云的曲面重建问题。

参 考 文 献

- [1] Xu L M, Wu G. Review of implicit surface reconstruction from point cloud dataset[J]. Computer Science, 2017, 44(S2): 19-23, 28.
徐利敏, 吴刚. 点云数据集的隐式曲面重构研究进展[J]. 计算机科学, 2017, 44(S2): 19-23, 28.
- [2] Franke R. Scattered data interpolation: tests of some methods[J]. Mathematics of Computation, 1982, 38(157): 181-200.
- [3] Savchenko V V, Pasko A A, Okunev O G, et al. Function representation of solids reconstructed from scattered surface points and contours[J]. Computer Graphics Forum, 1995, 14(4): 181-188.
- [4] Carr J C, Beatson R K, Cherrie J B, et al. Reconstruction and representation of 3D objects with radial basis functions [C] // SIGGRAPH' 01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. New York: Association for Computing Machinery, 2001: 67-76.
- [5] Blinn J F. A generalization of algebraic surface drawing[J]. ACM Transactions on Graphics, 1982, 1(3): 235-256.
- [6] Li Q, Wills D, Phillips R, et al. Implicit fitting using radial basis functions with ellipsoid constraint [J]. Computer Graphics Forum, 2004, 23(1): 55-69.
- [7] Klein P P. On the ellipsoid and plane intersection equation[J]. Applied Mathematics, 2012, 3(11): 1634-1640.
- [8] Beck A, Sabach S. An improved ellipsoid method for solving convex differentiable optimization problems [J]. Operations Research Letters, 2012, 40(6): 541-545.
- [9] Cuomo S, Galletti A, Giunta G, et al. Reconstruction of implicit curves and surfaces via RBF interpolation [J]. Applied Numerical Mathematics, 2017, 116: 157-171.
- [10] Kim S M, Park J C, Lee K H. Depth-image based full 3D modeling using trilinear interpolation and

- distance transform [C] // Depth-image based full 3D modeling using trilinear interpolation and distance transform. New York: Association for Computing Machinery, 2005: 259-260.
- [11] Pang Z Y, Zhou Z F, Wang L D, et al. Improved three-dimensional reconstruction algorithm for point cloud data [J]. *Laser & Optoelectronics Progress*, 2020, 57(2): 021102.
庞正雅, 周志峰, 王立端, 等. 改进的点云数据三维重建算法[J]. *激光与光电子学进展*, 2020, 57(2): 021102.
- [12] Lu S, Han J G, Wang L Z, et al. Two stage variable scale 3D point cloud registration algorithm [J]. *Laser & Optoelectronics Progress*, 2020, 57(20): 201503.
卢升, 韩俊刚, 王连哲, 等. 两阶段变尺度三维点云配准算法研究[J]. *激光与光电子学进展*, 2020, 57(20): 201503.
- [13] Yang P, Liu D E, Liu J Y, et al. Mine ground point cloud extraction algorithm based on statistical filtering and density clustering [J]. *Laser & Optoelectronics Progress*, 2020, 57(2): 021107.
杨鹏, 刘德儿, 刘靖钰, 等. 结合统计滤波与密度聚类的矿山地面点云提取算法[J]. *激光与光电子学进展*, 2020, 57(2): 021107.
- [14] Li R Z, Yang M, Ran Y, et al. Point cloud denoising and simplification algorithm based on method library [J]. *Laser & Optoelectronics Progress*, 2018, 55(1): 011008.
李仁忠, 杨曼, 冉媛, 等. 基于方法库的点云去噪与精简算法[J]. *激光与光电子学进展*, 2018, 55(1): 011008.
- [15] Huang W M, Li Y W, Wen P Z, et al. Algorithm for 3D point cloud denoising [C] // 2009 Third International Conference on Genetic and Evolutionary Computing, October 14-17, 2009, Guilin, China. New York: IEEE Press, 2009: 574-577.
- [16] Mo H J, Zhou J J. Study of surface reconstruction based on Imageware [J]. *Modern Manufacturing Engineering*, 2014(9): 21-25.
莫海军, 周佳军. 基于 Imageware 的曲面重构研究[J]. *现代制造工程*, 2014(9): 21-25.