

基于多尺度邻域平移法向估计算法 ——点云的法向量估计

黄明琪, 张杰, 史路冰

辽宁师范大学, 辽宁 大连

收稿日期: 2022年8月23日; 录用日期: 2022年9月18日; 发布日期: 2022年9月27日

摘要

本文以邻域漂移的思想为基础, 提出一种简洁高效的点云法向估计算法, 该算法在保证效率的同时可以有效克服边界点的法向估计效果较差的问题。首先, 设计了一种多尺度候选邻域集的生成方法, 该方法以当前点的所有近邻点为中心, 通过多尺度的策略丰富了候选邻域集的内容, 进而提高了最优邻域的质量; 其次在最优邻域的评价上不仅采用协方差对候选邻域集的“平坦程度”进行刻画, 而且也考虑了候选邻域与当前点的距离, 从而筛选出最合理的最优邻域, 提高了法向的质量。实验结果表明, 该算法可以有效的克服噪声和非均匀采样等问题, 很好的恢复模型的尖锐特征, 更好的权衡质量和时间。

关键词

邻域平移, 最优邻域, 法向估计

A Normal Estimation Algorithm Based on Multi-Scale Neighborhood Translation —Point Cloud Normal Estimation

Mingqi Huang, Jie Zhang, Lubing Shi

Liaoning Normal University, Dalian Liaoning

Received: Aug. 23rd, 2022; accepted: Sep. 18th, 2022; published: Sep. 27th, 2022

Abstract

Based on the idea of neighborhood drift, this paper proposes a simple and efficient point cloud normal estimation algorithm, which can effectively overcome the problem of poor normal estimation effect of boundary points while ensuring efficiency. First, a method for generating a multi-scale

candidate neighborhood set is designed. This method takes all the neighbors of the current point as the center, and enriches the content of the candidate neighborhood set through multi-scale strategies, thereby improving the optimal neighborhood. Secondly, in the evaluation of the optimal neighborhood, not only the covariance is used to describe the “flatness” of the candidate neighborhood set, but also the distance between the candidate neighborhood and the current point is considered, so as to screen out the most reasonable optimal neighborhood domain, which improves the quality of the normal. The experimental results show that the algorithm can effectively overcome the problems of noise and non-uniform sampling, restore the sharp features of the model well, and better trade off quality and time.

Keywords

Neighborhood Shift, Optimal Neighborhood, Normal Estimation

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着 3D 扫描仪的普及，点云的获取越来越方便。点云的法向量作为点云的重要属性之一，在很多算法中具有重要作用，例如点云的重建、渲染、分割等。然而，准确快速的估计出点云的法向仍然是一项比较艰巨的任务。尤其是在点云的获取过程中，会受到噪声、遮挡等客观原因的影响，这也加大了点云法向估计的误差。

近年来，针对点云法向估计有很多算法，包括：基于 Voronoi/Delaunay 的方法[1] [2] [3] [4]；基于法向修正的方法[5] [6] [7] [8]；基于邻域拟合的方法[9] [10] [11] [12] [13]。其中基于 Voronoi/Delaunay 的方法[1] [2] [3] [4]主要通过三角剖分来构建点云之间的拓扑关系，再根据构建出来的拓扑关系重建法向，但是，它对噪声比较敏感。基于法向修正算法是一类基于平滑化的滤波方法，主要通过不同的方法对初始法向进行校正和离群滤波。基于邻域拟合的方法最早是 Hoppe 等人[9]提出的，也称主成分分析法(Principal Component Analysis)，简称 PCA。该算法将邻域内所有点用一平面来进行拟合，并将当前点的法向定义为所拟合平面的法向。当该点位于光滑区域，PCA 可以准确估计出该点的法向；当该点位于边界处、角点等位置，其邻域由多个光滑曲面组成，在拟合时受到位于其他光滑曲面的影响，估计出的法向过于光滑。该类算法是以当前点为中心构造候选邻域，具有一定的局限性。因此，现如今大多学者将这一条件放宽，提出以当前点的近邻点为中心构造邻域进行法向估计。

Cao 等人[14]受图像滤波的启发，提出邻域漂移的思想来进行法向估计。首先将尖锐特征处的点分为边缘点和角点，针对尖锐特征和角点分别设计了不同策略用于候选邻域的生成：对于边缘点，通过引入方向约束算法，从边缘点的邻域中来选择点构造候选邻域，且该候选邻域要包含当前点在内；对于角点，采用自适应的迭代方式来为其选出一个位于光滑平面的邻域。为了最终得到的邻域比较“平坦”，利用局部邻域的协方差分析作为候选邻域的评价准则。该方法可以很好的解决位于尖锐特征处点的拟合效果较差的问题，但是对于特征点选取候选邻域集的方式过于复杂，效率很低，可行性较差。Zhou 等人[15]将多尺度的思想用于候选邻域集的构造中，找到包含当前点在内的多个尺度的邻域构成候选邻域集，从而丰富了候选邻域集的选取范围，囊括了更多的可选条件，减小了误差。然后对每一候选邻域求解一最优的拟合平面，将邻域内点与拟合平面的距离作为评价准则找到一最优邻域用于法向估计。虽然该算法

的评价准则更加准确，但是在求解最优拟合平面时耗时较长。为了提高效率，Zhang 等人[16]设计一种更为快捷的候选邻域构造算法和最优邻域的评价准则。首先将当前点的所有近邻点对应的邻域作为候选邻域，再用协方差分析找到一个最优邻域用于法向估计。该方法不需要利用搜索法来判断当前点是否在邻域中，极大的提高了效率。然而，该算法对于候选邻域的构造采取单一的尺度，对于复杂形状的模型和非均匀采样处理的效果并不好；在最优邻域的确定上，没有考虑当前点与拟合平面的距离，筛选的过于粗糙，误差较大。

本文对候选邻域集的构造和最优邻域的筛选进行创新，提出了一种新算法。在候选邻域集的构造中，将多尺度思想引入其中，并设计了两种方案：第一种，以当前点的所有近邻点为中心，采用多个尺度构造候选邻域集；第二种，根据当前点与近邻点的距离远近的不同，对位于不同邻域尺度的点采用不同的尺度构造候选邻域集。在最优邻域的筛选上，不仅采用协方差对候选邻域集的“平坦程度”进行刻画，而且也考虑了候选邻域与当前点的距离。针对该距离的引入，提出了两种不同的最优邻域评价方法：第一种：首先根据当前点到候选邻域的距离远近程度先对候选邻域集进行一次筛选，再通过余下的候选邻域的协方差矩阵进行分析找到最优邻域；第二种：通过将协方差分析的结果和点到平面的距离相结合来找到一个最优邻域。本文对提出的算法进行不同组合测试，实验结果表明，所有的组合都明显优于以往算法，且时间消耗上都较低。但是从候选邻域集的两种方案来看，第一种构造候选邻域集的方案在效果上更好，因为该方法筛选更多的候选邻域；第二种方案在效果上略次于第一种方案。总之，本文提出的算法，可以准确的估计出点云的法向。

2. 本文算法

对于一组有噪声的点云数据 $P = \{p_1, p_2, \dots, p_N\}$ (N 为采样点个数, $p_i = \{x_i, y_i, z_i\}$)，本文提出基于多尺度的邻域平移法向估计算法，该算法可以高效的筛选出候选邻域和最优邻域。分为 3 步：

第一步：点云预处理。首先利用 PCA 算法估计出点云 P 中的所有点的初始法向，再通过协方差矩阵的特征值来对所有点距尖锐特征的远近程度进行刻画，将点分为平滑点和特征点。

第二步：筛选特征点的最优邻域。对于特征点 p_t ，首先构造其邻域，然后通过本文提出的两版构造多尺度候选邻域集的方法对邻域内所有点构造多尺度的候选邻域集，再通过给出的评价方法对其进行筛选得到一最优邻域。

第三步：确定最终法向。对于平滑点，初始法向即为最终法向；对于特征点，对第二步筛选出的最优邻域使用 PCA 算法得到最终法向。算法流程如图 1 所示。

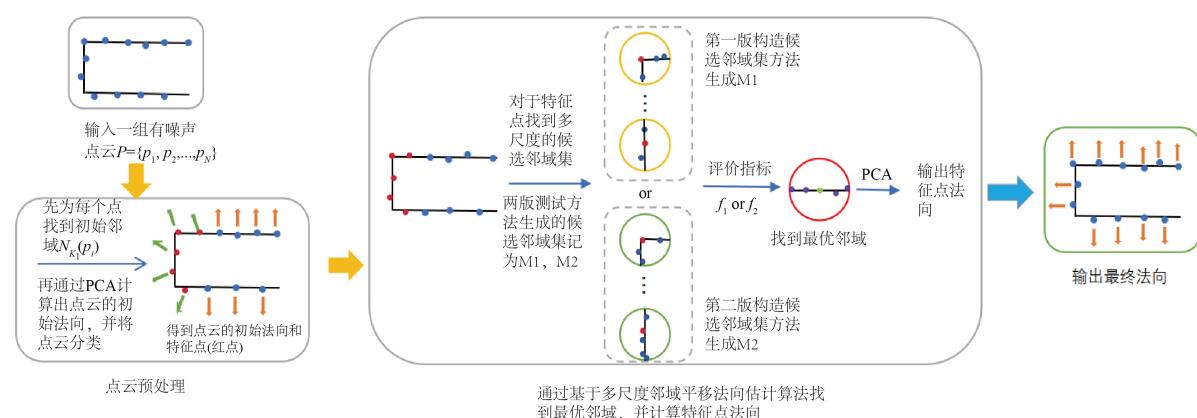


Figure 1. Arithmetic figure
图 1. 算法流程图

2.1. 点云预处理

对于点云 P 中任意一点 p_i , 找到初始邻域 $N_{K_1}(p_i)$, 即 $N_{K_1}(p_i) = \{p_i^j \mid p_i^j \text{是点 } p_i \text{ 的 } K_1 \text{ 个近邻点}\}$ 。本文采用与文献[16]同样的方法对每个点的初始法向和特征权重进行估计。首先, 对这些初始邻域构造对应的局部邻域协方差矩阵, 定义如下:

$$C_i = \frac{1}{K_1} \begin{bmatrix} p_i^1 - \bar{p} \\ p_i^2 - \bar{p} \\ \vdots \\ p_i^{K_1} - \bar{p} \end{bmatrix} \times \begin{bmatrix} p_i^1 - \bar{p} \\ p_i^2 - \bar{p} \\ \vdots \\ p_i^{K_1} - \bar{p} \end{bmatrix}^T$$

$$p_i^j \in N_{K_1}(p_i) \quad (1)$$

其中 $p_i^j \in N_{K_1}(p_i)$ 是点 p_i 的邻域点, \bar{p} 是局部邻域 $N_{K_1}(p_i)$ 的平均值, \times 代表矩阵相乘, 将协方差矩阵 C_i 的奇异值记为 μ_0, μ_1, μ_2 , 其中 $\mu_2 \geq \mu_1 \geq \mu_0 \geq 0$ 。

点 p_i 的初始法向定义为上述协方差矩阵中最小特征值 μ_0 对应的特征向量, 并且对于点 p_i 的特征权重 ω_i 定义如下:

$$\omega_i = \frac{\mu_0}{\mu_0 + \mu_1 + \mu_2} \quad (2)$$

点的特征权重 ω_i 刻画了该点位于尖锐特征处的概率。当 ω_i 较小时, 说明该点远离尖锐特征; 当 ω_i 较大时, 说明该点靠近尖锐特征。在本文中将 $\omega_i > \omega_s$ (ω_s 为给定的阈值) 时, 定义该点是特征点, 否则定义该点是平滑点。

本文使用文献[16]来对 ω_s 进行自动取值, 然后用 T_ω 代表特征权重 $\{\omega_i\}_{i=1}^N$ 的分布, 并通过最小化下面的函数来对该分布进行光滑处理:

$$\overline{T_\omega} = \arg \min_{\overline{T_\omega}} \left\| \overline{T_\omega} - T_\omega \right\|_F + \varepsilon \left\| A \overline{T_\omega} \right\|_1 \quad (3)$$

其中 ε 是权衡参数; A 为一阶差分矩阵; $\|\cdot\|_F$ 、 $\|\cdot\|_1$ 分别为 F 范数和 L_1 范数; ω_s 定义为 T_ω 的第一个峰值后的平缓点。

2.2. 筛选特征点的最优邻域

2.2.1. 构造多尺度的候选邻域集

对于特征点 p_t , 以该点的近邻点为中心构造邻域作为候选邻域。为了使得构造的候选邻域充分靠近当前点 p_t , 本文以 K_2 ($K_2 < K_1$) 为尺度构造 p_t 的邻域 $N_{K_2}(p_t)$, 即 $N_{K_2}(p_t) = \{p_t^j \mid p_t^j \text{是点 } p_t \text{ 的 } K_2 \text{ 个近邻点}\}$ 。文献[16]直接对邻域 $N_{K_2}(p_t)$ 中所有点构造单尺度的邻域作为候选邻域集。本文为了丰富候选邻域集, 采用两版方案构造多尺度的候选邻域集:

第一版: 以邻域 $N_{K_2}(p_t)$ 中的所有点为中心, $K_1; K_2; K_3$ ($K_1 > K_2 > K_3$) 为不同的三个尺度构造候选邻域 $N_{K_1}(p_t^j)$, $N_{K_2}(p_t^j)$, $N_{K_3}(p_t^j)$, 所有候选邻域构成候选邻域集 M_1 :

$$M_1 = \left\{ N_{K_1}(p_t^1), N_{K_1}(p_t^2), \dots, N_{K_1}(p_t^{K_2}), N_{K_2}(p_t^1), N_{K_2}(p_t^2), N_{K_2}(p_t^3), \dots, N_{K_2}(p_t^{K_2}), \right. \\ \left. N_{K_3}(p_t^1), N_{K_3}(p_t^2), \dots, N_{K_3}(p_t^{K_2}) \right\}$$

在实验中 $K_2 = K_1/2$, $K_3 = K_1/4$, 该版本的方法可以囊括多个候选邻域, 可选条件丰富, 对于效果有明显提升。但是在该方法下的候选邻域比较多, 在速度上有一定的局限性。因此, 本文又提出了第二

版构造候选邻域集的方法。

第二版：以与当前点的距离为基础，构造多尺度候选邻域集。如图 2 所示：对于与当前点 p_t 较远的点，一方面为了使所在邻域尽可能靠近当前点，另一方面由于该点远离尖锐特征，所在的平面较平坦。所以选择一个较大的尺度来构造邻域即可；对于与当前点 p_t 较近的点，靠近尖锐特征处，其所在邻域仍可能跨越尖锐特征，需要对邻域做仔细的分割，采用多个尺度构造候选邻域。

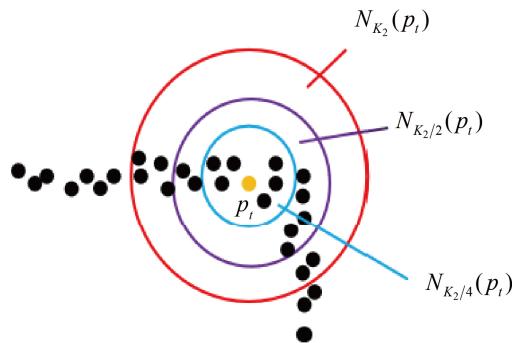


Figure 2. Effects at different neighborhood scales
图 2. 不同邻域尺度下的效果

具体来说，首先对邻域 $N_{K_2}(p_t)$ 进行分割，将其分成三部分，对于处于不同邻域尺度范围的点，处理的方式不同，详细如下：

a: 对于邻域尺度在 $0 \sim K_2/4$ 内的点(图 2 中蓝色圈内的点)，以 $K_1; K_2; K_3$ 为尺度构造多尺度候选邻域 $N_{K_1}(p_t^s), N_{K_2}(p_t^s), N_{K_3}(p_t^s)$ ；

b: 对于邻域尺度在 $K_2/4-K_2/2$ 内的点(图 2 中紫色圈除去蓝色圈的点)，以 K_1, K_2 为尺度构造多尺度候选邻域 $N_{K_1}(p_t^u), N_{K_2}(p_t^u)$ ；

c: 对于邻域尺度在 $K_2/2-K_2$ 内的点(图 2 中红色圈除去紫色圈的点)

所有的候选邻域构成候选邻域集 M_2 :

$$M_2 = \left\{ N_{K_1}(p_t^1), \dots, N_{K_1}(p_t^{K_2/4}), N_{K_2}(p_t^1), \dots, N_{K_2}(p_t^{K_2/4}), N_{K_3}(p_t^1), \dots, \right. \\ \left. N_{K_3}(p_t^{K_2/4}), \dots, N_{K_1}(p_t^{K_2/2}), \dots, N_{K_2}(p_t^{K_2/2}), \dots, N_{K_1}(p_t^{K_2}) \right\}$$

实验结果表明，由于第一版构造的候选邻域集相比第二版较多，可选条件丰富，因此效果更加优越；第二版的效果略次于第一版，但是时间上优于第一版。

2.2.2. 确定最优邻域

接下来从候选邻域集中选择最优邻域 N^* 用于法向估计。本文不但用协方差对邻域的“平坦程度”进行分析，而且也考虑了拟合平面与当前点的“远近程度”。在邻域的“平坦程度”上，用公式(1)对每个候选邻域 $N_{K_r}(p_t^h)$ ($r=1, 2, 3; h=1, \dots, K_2$) 构造对应的局部邻域协方差矩阵，然后再通过公式(2)得到对应的特征权重 $\omega(N_{K_r}(p_t^h))$ ($r=1, 2, 3; h=1, \dots, K_2$)。为了使所选的邻域比较平坦，需要在所有候选邻域特征权重中找到较小的值。除此之外，希望所选出的最优邻域的拟合平面要尽可能的靠近当前点，即对该邻域与当前点的远近程度进行刻画。首先对候选邻域 $N_{K_r}(p_t^h)$ ($r=1, 2, 3; h=1, \dots, K_2$) 的拟合平面构成的集合记为 $\theta(N_{K_r}(p_t^h))$ ($r=1, 2, 3; h=1, \dots, K_2$)，然后再通过如下公式计算出当前点与候选邻域的拟

合平面之间的距离:

$$D(N_{K_r}(p_t^h)) = \left| (p_t - p_t^h) \cdot \overline{V(\theta_{K_r}(p_t^h))} \right|, \quad r=1,2,3; \quad h=1,\dots,K_2 \quad (4)$$

其中 p_t 代表当前点, p_t^h 代表候选邻域的中心点, $\overline{V(\theta_{K_r}(p_t^h))}$ 代表候选邻域的拟合平面的法向, \cdot 代表内积, $||$ 代表绝对值。

为了将两种评价指标结合起来, 本文采用两种方法:

第一种(f_1): 给定一个阈值 λ , 根据得到的 $D(N_{K_r}(p_t^h))$, 先对候选邻域集进行第一波筛选。当 $D(N_{K_r}(p_t^h)) > \lambda$ 时, 说明这个候选邻域与当前点的距离较远, 将其舍掉。将余下的符合远近条件的候选邻域构成的集合记为 $N = \{N_{K_p}(p_t^q) | D(N_{K_p}(p_t^q)) \leq \lambda\}$, 并将所对应的特征权重构成的集合记为 $\omega = \{\omega(N_{K_p}(p_t^q)) | \text{其中 } N_{K_p}(p_t^q) \in N\}$ 。然后找到 ω 中的最小值 ω^* , 其对应邻域 N^* 即为最优邻域:

$$N^* = \arg \min_{N^*} \omega \quad (5)$$

第二种(f_2): 对于候选邻域 $N_{K_r}(p_t^h)$ ($r=1,2,3; \quad h=1,\dots,K_2$), 将它对应的特征权重 $\omega(N_{K_r}(p_t^h))$ ($r=1,2,3; \quad h=1,\dots,K_2$) 和到当前点的距离 $D(N_{K_r}(p_t^h))$ ($r=1,2,3; \quad h=1,\dots,K_2$) 做加权组合得到最终的评价准则 $f(N_{K_r}(p_t^h))$ 。具体定义如下:

$$f(N_{K_r}(p_t^h)) = D(N_{K_r}(p_t^h)) + \lambda * \omega(N_{K_r}(p_t^h)), \quad r=1,2,3; \quad h=1,\dots,K_2 \quad (6)$$

其中 λ 是一个权衡参数用于平衡 $D(N_{K_r}(p_t^h))$ 和 $\omega(N_{K_r}(p_t^h))$ 。接下来将所得的结果构成集合

$F = \{f(N_{K_r}(p_t^h)) | r=1,2,3; h=1,\dots,K_2\}$, 然后找到集合 F 中的最小值, 其对应的邻域为最优邻域 N^* :

$$N^* = \arg \min_{N^*} F \quad (7)$$

与文献[16]类似, 对于最优点, 如果依旧选择 K_1 为尺度构造邻域, 最终邻域仍然可能跨越边界处。为了保障最优邻域不跨越边界处, 将邻域的尺度缩小, 以 K_2 为尺度构造最优邻域 N^* 。

2.3. 最终法向的确定

对于平滑点, 其邻域为一个光滑的曲面, PCA 可以准确的对邻域拟合。因此, 初始法向即为最终法向; 对于特征点, 邻域可能由多个曲面构成, 直接 PCA 无法估计出准确的法向。因此, 对得到的最优邻域进行处理得到最终法向。

3. 实验结果及分析

为了评估算法的性能, 将本文算法与一些具有代表性的算法进行比较, 包括 HF、LRRfast、PatchShift、PCA、PCV、RNE、ShiftPCA, 其中 HF 使用 HF_cubes 模型进行比较。所有实验使用的电脑配置为 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40 GHz 1.38 GHz。本文在确定候选邻域集时采用两种不同的方案, 在最优邻域的评估时也采用两种不同的方案, 组合起来共有四种方案, 在文章中用方案 1、方案 2、方案 3、方案 4 代表。其中方案 1 代表用第二版候选邻域集和第一种评价指标组合的方法, 同时阈值 $\lambda = 0.01$; 方案 2 代表用第二版候选邻域集和第二种评价指标组合的方法, 同时阈值 $\lambda = 0.5$; 方案 3 代表用第一版候选邻域集和第一种评价指标组合的方法, 同时阈值 $\lambda = 0.05$; 方案 4 代表用第一版候选

邻域集和第二种评价指标组合的方法，同时阈值 $\lambda = 0.005$ 。

本文对大量模型进行测试：所有模型都添加了均值为 0 的噪声，标准差定义为点云之间平均距离的百分比，大小为 30%，40%，50%，60%。

3.1. 参数的确定

本文算法的参数主要如下：

$K_1; K_2; K_3$ ：构造多尺度邻域时选择的三种邻域尺度；

λ ：指标函数的阈值。

其中噪声的大小与 $K_1; K_2; K_3$ 的取值成正比。在本文的实验中设置 $K_1 = 100, K_2 = 50, K_3 = 25$ 。对于阈值 λ ，通过大量实验发现阈值大小对实验结果有很大影响，本文展示了其中 4 个结果较好的值：0.005、0.05、0.01、0.5。其他算法的邻域尺度大小均设置为 100，所有算法的其余相关参数均采用默认值。

3.2. 评价函数

本文使用带有阈值的均方根误差(RMS_{τ})来评价，定义如下：

$$RMS_{\tau} = \sqrt{\frac{1}{P} \sum_{p \in P} \left(f(\langle n_p, \hat{n}_p \rangle) \right)^2} \quad (8)$$

$$f(\langle n_p, \hat{n}_p \rangle) = \begin{cases} \langle n_p, \hat{n}_p \rangle & \text{当 } \langle n_p, \hat{n}_p \rangle < \tau \\ \pi/2 & \text{其他} \end{cases} \quad (9)$$

其中 $\langle n_p, \hat{n}_p \rangle$ 是法向 n_p 与 \hat{n}_p 之间的夹角， n_p 是点 p 的真实法向， \hat{n}_p 是点 p 的估计法向； N_p 是点 p 的真实法向集合， \hat{N}_p 是点 p 的估计法向集；当每个点输出一个法向时，集合 \hat{N}_p 为 1。本文取 $\tau = 10^\circ$ ，当 n_p 与 \hat{n}_p 的值大于 τ 时，认为该点估计的法向不准确(定义该点为坏点)。

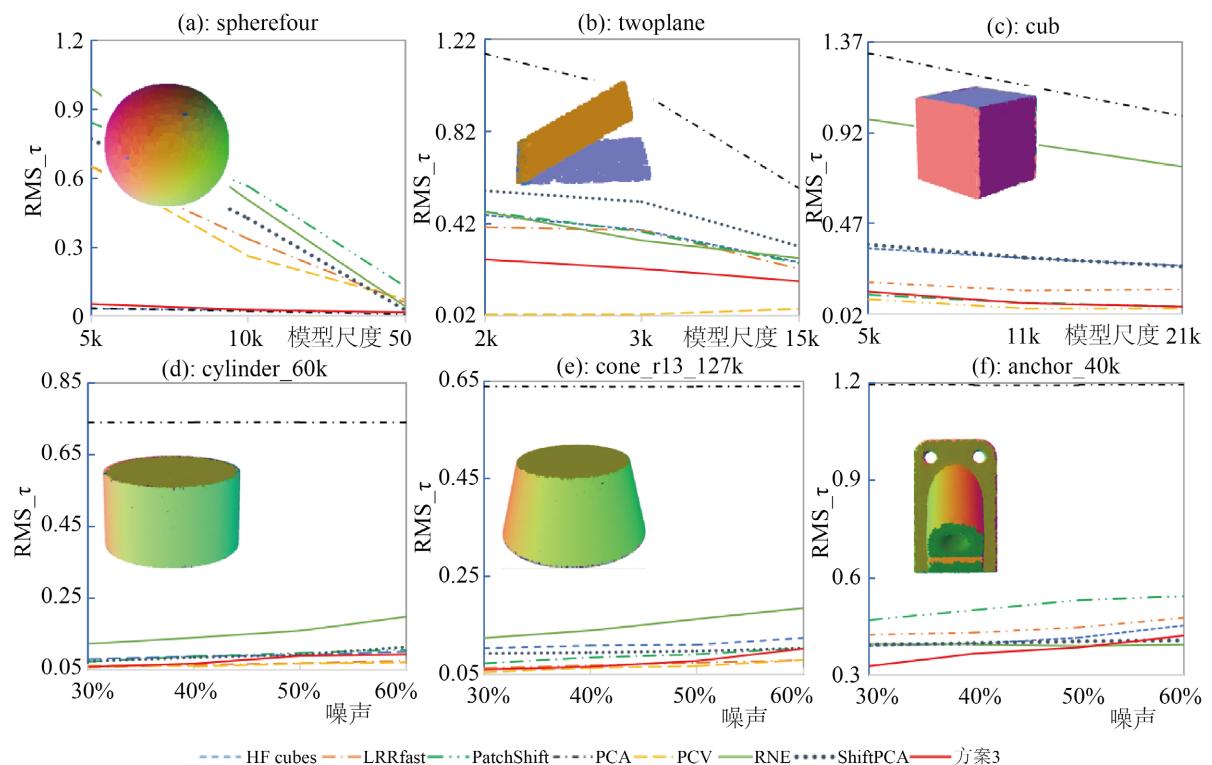
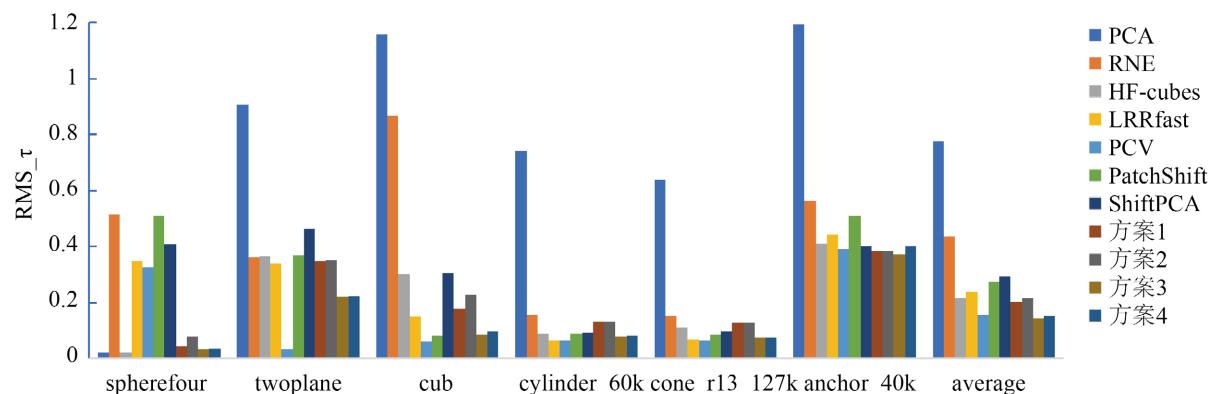
3.3. 结果分析

图 3 展示了不同算法在不同模型上 RMS_{τ} 的变化情况，由于本文的四种组合方案在效果上基本一致，差别不明显，为了更加清晰的显示本文算法与以前算法之间的差别，在图 3 中只展示了组合方案中较好的方案 3。图 4 展示了不同算法在每个模型以及所有模型上 RMS_{τ} 均值的比较。图 5 展示了不同模型坏点的可视化情况(红色点显示坏点)。图 6 展示了不同算法在各个模型上的法向估计结果的可视化。

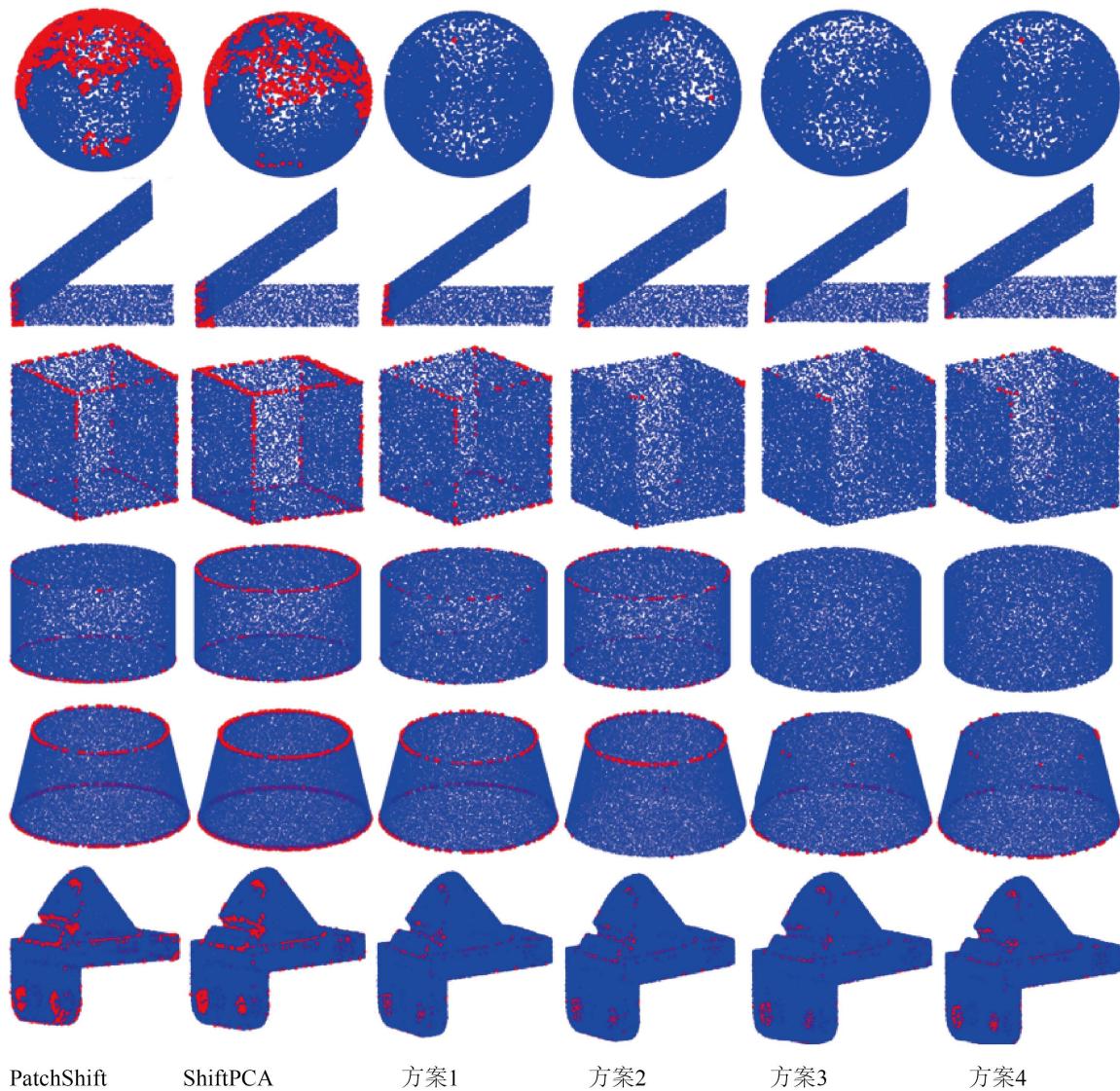
如图 3 和图 4 所示，即使在非均匀采样下，PCA 算法仍然可以准确的估计出光滑曲面的法向，如模型 spherefour 所示，但是对于尖锐特征模型(如图 3 中的(b)到(f)所示)，估计出的法向准确性较低，相比其他算法， RMS_{τ} 值最高。这是因为 PCA 选择整个邻域进行平面拟合，对于特征点，其邻域可能由多个光滑曲面构成，在拟合时易受其他光滑平面的影响，导致估计出的法向过于光滑，误差较大。

RNE、HF-cubes、LRRfast 和 PCV 在处理尖锐特征模型时，效果都比 PCA 优越，并且其中 PCV 的效果最佳，其 RMS_{τ} 值是最低的(如图 3 中的(b)到(f)所示)；在处理光滑模型时，HF-cubes 的效果与 PCA 持平，但 RNE、PCV 和 LRRfast 的效果与 PCA 相比较差(如模型 spherefour 所示)。

PatchShift 和 ShiftPCA 都是基于邻域漂移的算法。其中 PatchShift 对于简单的特征模型处理的较好，其 RMS_{τ} 值较低(如 cylinder 模型和 cone 模型)，但是在处理光滑模型和较复杂的特征模型时，效果较差(如 spherefour 模型和 anchor 模型)。ShiftPCA 对于特征模型处理效果比较好(如图 3 中的(d)到(f)所示)，但是在处理非均匀采样上效果较差，其 RMS_{τ} 值较高(如图 3 中的(a)到(c)所示)。总体上来看，PatchShift 和 ShiftPCA 的 RMS_{τ} 值高于本文算法，并且 PatchShift 的时间消耗也较高(如表 1 所示)。

**Figure 3.** RMS_{τ} on different models vary with noise or model scale**图 3.** 不同模型上 RMS_{τ} 值随噪声或模型尺度的变化情况**Figure 4.** Average RMS_{τ} across different models**图 4.** 不同模型上 RMS_{τ} 的均值**Table 1.** Sum and time comparison of RMS_{τ} , RMS of different algorithms**表 1.** 各类算法的 RMS_{τ} 、 RMS 值和时间比较

算法	HF_c	LRR fast	PatchShift	PCA	PCV	RNE	ShiftPCA	方案 1	方案 2	方案 3	方案 4
编码	C	M&C	M	M	M	C	M	M	M	M	M
RMS_{τ}	0.22	0.23	0.27	0.79	0.16	0.42	0.28	0.20	0.22	0.15	0.16
时间/s	17.05	185.97	249.7	11.8	95.2	38.9	6.69	12.45	16.05	16.74	16.99

**Figure 5.** Different arithmetic visualization**图 5.** 不同算法上的可视化比较

本文提出的四种组合方案在不同模型和噪声尺度上都得到了较好的结果。从定量分析上看，本文算法在不同模型和噪声尺度上的的 RMS_{τ} 值都比较低，并且其中方案 3 和方案 4 (即采用第一版构造候选邻域集的方法)的效果比方案 1 和方案 2 优越，但是时间消耗要比方案 1 和方案 2 略高。从可视化结果上看，本文算法对光滑模型的保持和尖锐特征的刻画都权衡的比较好(如图 5 和图 6 所示)，在保证运行速度的同时可以得到高质量的法向。

3.4. 时间比较

表 1 展示了不同算法的 RMS_{τ} 和时间的均值。从中能看出用 MATLAB (表中记为“M”) 实现的 ShiftPCA 的运行速度是最快的，PCA 的效率次之，但是这两类算法的 RMS_{τ} 值相对较大。在 MATLAB 下实现的 LRRfast、PCV、RNE 和 PatchShift (包括 C/C++混编的记为“M&C”) 运行速度相对较慢。本文提出的四种方案在时间和效果上都比较优越，在效果上，本文算法的 RMS_{τ} 值比较低，其中方案 1 和方

案 2 仅次于 PCV 并且与 HF_c 持平, 明显优于其他算法, 方案 3 和方案 4 的 RMS_{τ} 值可以达到最低; 在时间上, 本文算法仅次于 PCA 和 ShiftPCA, 显著优于其他算法。总之, 本文的算法可以很好的平衡算法的性能和效率, 可行性较强。

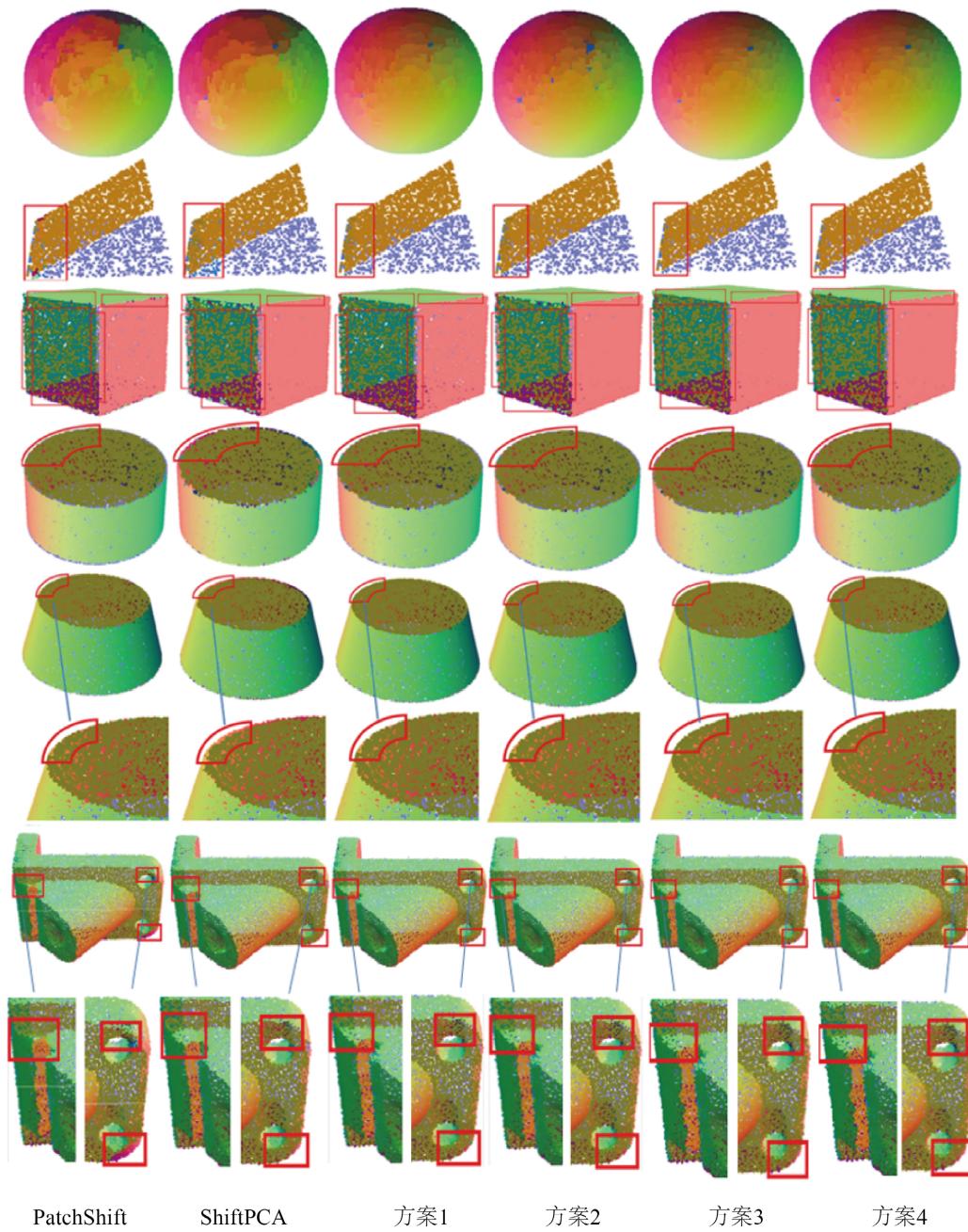


Figure 6. Estimation visual comparison of different algorithm
图 6. 不同算法的法向估计可视化结果

4. 结论

本文提出的基于多尺度邻域平移法向估计算法, 可以快速准确的估计出点云的法向。本文在候选邻

域集的构造和最优邻域的筛选上进行了创新，对候选邻域的构造提出多尺度的思想并且将当前点与候选邻域的距离以及对邻域的协方差分析作为最优邻域的筛选指标。从而极大地减小了误差，其误差值从之前的 0.28 降到了 0.15，同时也保证了运行的速度，可以准确的估计出点云法向，可行性较强。但本文算法没有考虑尖锐特征情况下的非均匀采样问题。如何在确保准确性和提高算法效率的基础上，考虑尖锐特征下的非均匀采样问题，这是未来的一项重要工作。

基金项目

国家自然科学基金(62076115, 61702245); 大连市青年科技之星(2020RQ053)。

参考文献

- [1] Bern, M.W. and Amenta, N. (1999) Surface Reconstruction by Voronoi Filtering. *Discrete and Computational Geometry*, **22**, 481-504. <https://doi.org/10.1007/PL00009475>
- [2] Dey, T.K. and Goswami, S. (2006) Provable Surface reconstruction from Noisy Samples. *Computational Geometry*, **35**, 124-141. <https://doi.org/10.1016/j.comgeo.2005.10.006>
- [3] Feng, H.Y. and Ouyang, D. (2005) On the Normal Vector Estimation for Point Cloud Data from Smooth Surfaces. *Computer-Aided Design*, **37**, 1071-1079. <https://doi.org/10.1016/j.cad.2004.11.005>
- [4] Alliez, P., Cohen-Steiner, D., Tong, Y. and Desbrun, M. (2007) Voronoi-Based Variational Reconstruction Reconstruction of Unoriented Point Sets. *Proceedings of the 5th Eurographics Symposium on Geometry Processing*, Barcelona, 4-6 July 2007, 39-48.
- [5] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D. and Silva, C.T. (2001) Point Set Surfaces. *Proceedings Visualization 2001. VIS '01*, San Diego, 21-26 October 2001, 21-29. <https://doi.org/10.1109/VISUAL.2001.964489>
- [6] Flix, C., Rui, U. and Rivera, M. (2007) Surface-Normal Estimation with Neighborhood Reorganization for 3D Reconstruction. *12th Iberoamerican Congress on Pattern Recognition (CIARP 2007)*, Valparaiso, 13-16 November 2007, 321-330. https://doi.org/10.1007/978-3-540-76725-1_34
- [7] Jones, T.R., Durand, F. and Zwicker, M. (2004) Normal Improvement for Point Rendering. *IEEE Computer Graphics and Applications*, **24**, 53-56. <https://doi.org/10.1109/MCG.2004.14>
- [8] Ztireli, A.C., Guennebaud, G. and Gross, M. (2009) Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression. *Computer Graphics Forum*, **28**, 493-501. <https://doi.org/10.1111/j.1467-8659.2009.01388.x>
- [9] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W. (1992) Surface Reconstruction from Unorganized Points. *ACM SIGGRAPH Computer Graphics*, **26**, 71-78. <https://doi.org/10.1145/142920.134011>
- [10] Gross, M. and Pfister, H. (2007) Point-Based Graphics. The Morgan Kaufmann Series in Computer Graphics, Morgan Kaufmann Publishers Inc., San Francisco.
- [11] Pauly, M., Keiser, R., Kobbelt, L. and Gross, M.H. (2003) Shape Modeling with Point-sampled Geometry. *ACM Transactions on Graphics*, **22**, 641-650. <https://doi.org/10.1145/882262.882319>
- [12] Mitra, N.J., Nguyen, A. and Guibas, L.J. (2004) Estimating Surface Normals in Noisy Point Cloud Data. *International Journal of Computational Geometry and Applications*, **14**, 261-276. <https://doi.org/10.1142/S0218195904001470>
- [13] Yoon, M., Lee, Y., Lee, S., Ivrissimtzis, I.P. and Seidel, H. (2007) Surface and Normal Ensembles for Surface Reconstruction. *Computer-Aided Design*, **39**, 408-420. <https://doi.org/10.1016/j.cad.2007.02.008>
- [14] Cao, J., Chen, H., Zhang, J., Li, Y., Liu, X. and Zou, C. (2017) Normal Estimation via Shifted Neighborhood for Point Cloud. *Journal of Computational and Applied Mathematics*, **329**, 57-67. <https://doi.org/10.1016/j.cam.2017.04.027>
- [15] Zhou, H., Chen, H., Feng, Y., Wang, Q., Qin, J., Xie, H., et al. (2020) Geometry and Learning Co-Supported Normal Estimation for Unstructured Point Cloud. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, 13-19 June 2020, 13235-13244. <https://doi.org/10.1109/CVPR42600.2020.01325>
- [16] 张杰, 刘健, 王茜微, 等. 基于邻域漂移的点云法向估计算法研究[J]. 辽宁师范大学学报(自然科学版), 2021, 44(1): 13-20.