

基于曲率改进的 K4PCS 点云配准算法

黄 鹤¹ 赵夫群¹

HUANG He ZHAO Fuqun

摘 要

点云配准就是根据源点云和目标点云的匹配点对求解最优刚性变换矩阵,从而实现点云对齐的过程,目前已在同步定位与测绘、三维重建、逆向工程等领域得到广泛应用。针对现有算法存在的配准精度低、耗时长等问题,提出一种基于曲率改进的 K4PCS 点云配准算法。粗配准阶段首先计算点云的法向角度和马氏距离方差两个特征来构造特征描述符,筛选特征点集,然后通过曲率改进 K4PCS 算法进行粗配准;精配准阶段使用 TrICP 算法来完成精确配准。实验采用 Bunny、Dragon 和 Buddha 点云模型,结果表明所提出的算法比已有的两种配准算法在平均精度上提高了约 25%,平均耗时降低了约 22%,是一种有效的点云配准算法。

关键词

点云配准;特征点;马氏距离;曲率;四点共线集

doi: 10.3969/j.issn.1672-9528.2024.10.001

0 引言

近年来,激光雷达技术发展迅速,点云被认为是更广泛的场景中 3D 可视化最合适的数据,能够提供被测物体表面的空间坐标,从而解决 3D 场景重建、虚拟现实与增强现实等任务^[1-2]。然而,在实际应用中,由于受环境遮挡、物体自身的破损程度、采集设备的分辨率参数等因素的影响,无法一次获得完整的点云。因此,需要对不同姿态下进行多次连续扫描得到的点云进行配准。点云配准的实质是求解一个旋转矩阵和平移向量,并利用该刚体变换将两片或多片点云转换为同一坐标系。目前,点云配准已广泛应用于城市规划、地貌测量等领域^[3-4]。

点云配准算法中最经典的是迭代最近点(iterative closest point, ICP)算法^[5],该算法通过不断迭代计算最优变换,直到满足收敛条件。但是由于其存在初始位姿要求高、计算效率低等问题,许多学者提出了多种改进 ICP 算法。如 Chetverikov 等人^[6]提出的裁剪迭代最近点(the trimmed iterative closest point, TrICP)算法,对对应点之间的欧氏距离排序,只使用一定比例的点对计算变换矩阵;Bouaziz 等人^[7]提出的稀疏迭代最近点(sparse iterative closest point, SICP)算法,通过参数的稀疏性有效控制离群点;Wu 等人^[8]采用最佳 BBF 算法进行数据降维,加快了 ICP 迭代速度;Aiger 等人^[9]提出四

点共线集(four-points congruent sets, 4PCS)算法,依据仿射不变性原理,在目标点云中寻找与源点云共面四点近似的全等四点子集,不需要依赖任何特征,但对点云重叠比例要求较高;在此基础上,Da 等人^[10]提出动态四点共线集(dynamic four-points congruent sets, D4PCS),每次迭代中动态控制点云的重叠程度的基础宽度,因此能够对不同重叠程度的曲面进行配准;Theiler 等人^[11]提出一种基于关键点的关键点四点共线集(keypoint-based four-points congruent sets, K4PCS),将提取的关键点作为 4PCS 算法的输入进行配准,从而更好地适应激光扫描的特点。

针对现有算法存在的配准精度低、耗时长等问题,提出一种基于曲率改进的 K4PCS 点云配准算法。首先,利用法向角度和马氏距离方差构造特征描述符来提取特征点,然后使用曲率改进的 K4PCS 算法对提取的特征点集进行粗配准,最后通过 TrICP 算法完成精确配准。该算法提取特征点集有助于提高后续配准速度,通过利用曲率改进的 K4PCS 进一步提高配准精度,而 TrICP 算法由于只对部分点对进行计算,因此加快了配准速度。

1 基于曲率改进的 K4PCS 点云粗配准

1.1 特征点提取

法向量、距离方差等几何特征反映了点云最基本的几何属性,对于表达点云的局部特征至关重要。此外,这些特征具有旋转平移不变性,能够有效描述,并且易于计算。因此,结合点云邻域的法向角度和马氏距离方差两个特征来构建一种新的特征描述符。

1. 西安财经大学信息学院 陕西西安 710100

[基金项目] 国家自然科学基金项目(62271393)

(1) 法向角度

法向量是点云中关键的几何特征，对于对应的两片分割点云，点的法向量与其邻域点的法向量夹角一般是相似的，因此可以利用这一特性来寻找分割点云的对应关系。

首先对于点云 P 中任一点 p_i ，搜索其 k 个邻域点，然后建立协方差矩阵 C ，定义为：

$$\begin{cases} C = \frac{1}{k} \sum_{j=1}^k (p_{ij} - \bar{p}_i)(p_{ij} - \bar{p}_i)^T \\ \bar{p}_i = \frac{1}{k} \sum_{j=1}^k p_{ij} \end{cases} \quad (1)$$

式中： $\bar{p}_i$ 为点 p_i 的邻域点 p_{ij} 的质心。

根据协方差矩阵，计算出特征值 λ_1 、 λ_2 、 λ_3 ($\lambda_1 \leq \lambda_2 \leq \lambda_3$) 及其对应的特征向量 V_1 、 V_2 、 V_3 ，则最小特征值 λ_1 对应的特征向量 V_1 即为点 p_i 的法向量 n_i 。

求出每个点的法向量后，点 p_i 与其邻域点 p_{ij} 法向量夹角 $\theta_{n_i, n_{ij}}$ 的余弦值可表示为：

$$\omega_i = \cos \theta_{n_i, n_{ij}} = \frac{n_i \cdot n_{ij}}{|n_i| \cdot |n_{ij}|} \quad (2)$$

式中： $||$ 表示为向量的模，夹角 $\theta_{n_i, n_{ij}}$ 的取值范围是 $[0^\circ, 180^\circ]$ 。

(2) 马氏距离方差

对于对应的特征点，其邻域内各邻近点的分布情况应较为相似，在此可以使用距离方差对其进行描述。马氏距离考虑了数据之间的相关性和不同尺度，通过对特征向量进行白化处理，将原始数据转换成不相关且具有相同方差的新的特征向量，在实际应用中具有较好的检测、聚类 and 识别等性能，因此能够在高维空间中更加准确地描述数据点之间的距离。

使用马氏距离作为距离参数，对于点 p_i ，设其半径为 r 的邻域点集 p_{ij} 所在球体投影到二维平面 L ，该平面垂直于点 p_i 的法向量 n_i ，如图 1 所示。

首先计算点云中各邻近点与二维切平面 L 之间的马氏距离 d_m ($m=1, 2, \dots, k$)，可定义为^[12]：

$$d_m(p_{ij} - p_{ij}') = \sqrt{(p_{ij} - p_{ij}')^T C^{-1} (p_{ij} - p_{ij}')} \quad (3)$$

式中： p_{ij}' 为点 p_{ij} 在切平面 L 上的投影点。

然后将该距离 d_m 的方差定义为：

$$\sigma(p_i) = \frac{1}{k} \sum_{m=1}^k (d_m - \bar{d}_i)^2 \quad (4)$$

式中： $\bar{d}_i$ 为各邻近点与切平面之间的马氏距离的平均值。

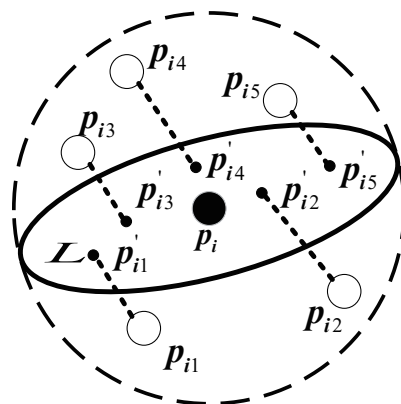
法向量和马氏距离方差都具有旋转平移不变性，因此将这两个特征融合为 $H(p_i)$ ，定义为：

$$H(p_i) = \frac{\alpha \sigma_i}{\omega_i} \quad (5)$$

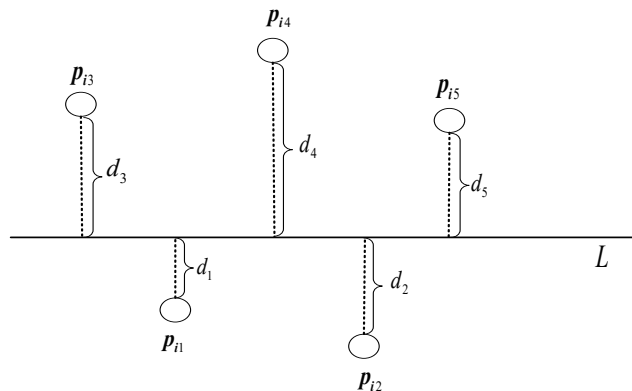
式中： i 为特征点的数量， α 为平衡参数。

由式 (5) 可知，马氏距离方差越大， p_i 是特征点的概率越高，因而 σ_i 与 $H(p_i)$ 成正比；法向夹角越小，其余弦值越大， p_i 是特征点的概率越低，因此法向夹角余弦值与 $H(p_i)$ 成反比。

因此，可以根据建立的特征参数来获取特征点对。当 $H(p_i)$ 大于阈值 ε 时，可将该点设为特征点；当 $H(p_i)$ 小于阈值 ε 时，该点为非特征点，应将其删去。由此可得到特征点集 $X = \{x_i\}_{i=1}^{N_x}$ 和 $Y = \{y_i\}_{i=1}^{N_y}$ ，其中 N_x 和 N_y 为特征点的数量。



(a) 切平面和投影点



(b) 邻近点与切面的马氏距离

图 1 马氏距离

1.2 基于曲率改进的 K4PCS 点云粗配准

1.2.1 曲率计算

曲率可以表征点云局部曲面的形状、凹凸性以及变化趋势。可以利用点云的局部曲率信息筛选特征点对，有效地减少配准的时间消耗，并减小配准失败的概率。

这里采用协方差分析法求取采样点的曲率值，根据协方差矩阵求解出特征值 λ_1 、 λ_2 、 λ_3 ($\lambda_1 \leq \lambda_2 \leq \lambda_3$)，则点 p_i 处的曲率表示为：

$$S(p_i) = \frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} \quad (6)$$

1.2.2 改进的 K4PCS

作为一种常见的点云配准方法, 4PCS 算法的思想是基于变换的仿射不变性, 通过在两点云的重叠区域内, 在一定距离内找出其对应的四点集, 并根据这些点对之间的关系求解变换矩阵。K4PCS 算法利用 3D 关键点作为 4PCS 算法的输入, 通过较少的下采样和关键点的提取更好地保留点云的显著特征。

对于给定源点云 X 和目标点云 Y , 首先在点集 X 中选择一个近似共面四点对 $M(a, b, c, d)$, 其中线 ac 与线 bd 相交于点 e , 如图 2 所示。定义交点比率 ε_1 和 ε_2 为:

$$\varepsilon_1 = \frac{\|a - e\|}{\|a - c\|} \quad (7)$$

$$\varepsilon_2 = \frac{\|d - e\|}{\|a - b\|} \quad (8)$$

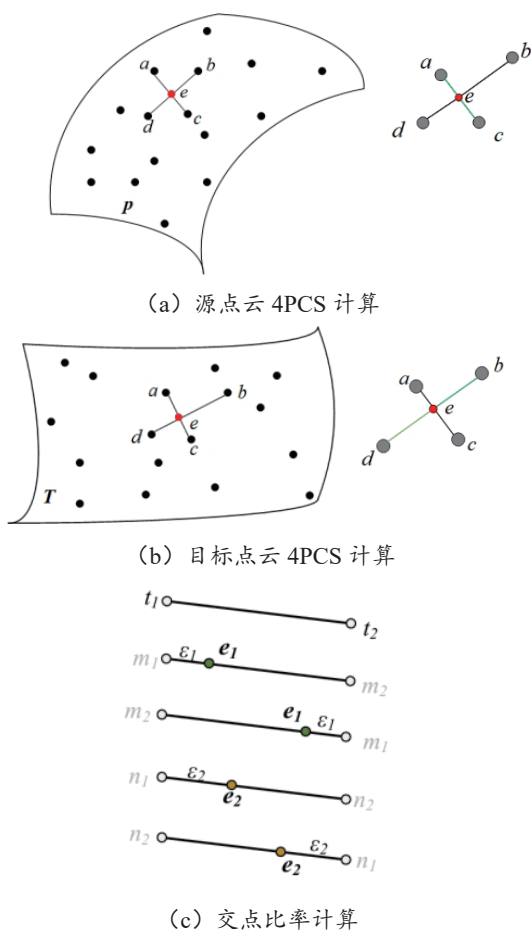


图 2 4PCS 原理

然而, 在刚体变换中, ε_1 和 ε_2 是不变的, 从目标点云 Y 中提取四个点集 $N(m_1, m_2, n_1, n_2)$, 经过仿射变换后, 可以与点集 M 一致或者近似同余, 即:

$$\frac{\|m_1 - e_1\|}{\|m_1 - e_2\|} \approx \varepsilon_1 \quad (9)$$

$$\frac{\|n_1 - e_2\|}{\|n_1 - n_2\|} \approx \varepsilon_1 \quad (10)$$

对于每对点 $\{t_1, t_2\} \in Y$, 根据 ε_1 和 ε_2 计算两个交叉点:

$$e_1 = x_1 + \varepsilon_1 \cdot (x_2 - x_1) \quad (11)$$

$$e_2 = x_2 + \varepsilon_2 \cdot (x_2 - x_1) \quad (12)$$

有效点对 $N'\{m_1, m_2\}$ 以及 $N''\{n_1, n_2\}$ 的交点比 N 需要与 M 的交点比近似, 则:

$$\|e_1(N') - e_2(N'')\| < \zeta \quad (13)$$

$$\|e_2(N') - e_1(N'')\| < \zeta \quad (14)$$

式中: ζ 为距离阈值, e_1 和 e_2 表示可以分离的最大距离。

设 $M(a, b, c, d)$ 为点云 X 中的四个共面点, $S(C_a, C_b, C_c, C_d)$ 是其对应的曲率, $d_1 = \|a - c\|$, $d_2 = \|b - d\|$ 。设 $C(x, y)$ 表示点 x 和 y 处的曲率平均值, 对于满足 $\|t_1 - t_2\| \approx d_1$ 或 $\|t_1 - t_2\| \approx d_2$ 的所有点对 $\{t_1, t_2\} \in Y$, 选择一个任意点 $t_i \in Y$, 计算其对应的曲率, 并提取所有满足 $C(t_i - t_j) \approx C(a, c)$ 或 $C(t_i - t_j) \approx C(b, d)$ 的点 $t_i \in Y$ 。

曲率改进的 K4PCS 算法的具体步骤如下。

(1) 首先计算特征参数 $H(p_i)$ 来提取源点云和目标点云中的特征点, 然后利用协方差分析法对特征点计算曲率值。

(2) 在点集 X 中选择一个近似共面四点对 M , 并根据式 (7) 和式 (8) 计算交点比率 ε_1 和 ε_2 。

(3) 根据式 (11) 和式 (12) 计算交叉点, 并根据两个交点比率, 在目标点云 Y 中提取满足式 (13) 和式 (14) 与 M 的交点比近似的四点集。

(4) 在 Y 中搜索与 M 近似相等的四点集, 并筛选满足 $C(t_i - t_j) \approx C(a, c)$ 或 $C(t_i - t_j) \approx C(b, d)$ 的点对。

(5) 通过最小二乘法计算变换矩阵 B , 不断迭代以上过程, 最后依据 LCP 准则获得最优变换矩阵 B 。

2 基于 TrICP 的点云精配准

为了进一步提高配准精度, 下面采用 TrICP 算法进行精配准, 配准步骤如下。

(1) 在目标点云中, 计算点云 Y 的最近点, 构成点对, 计算这些点对的欧氏距离 μ_i 。

(2) 按升序对欧氏距离 μ_i 进行排序, 选择顶部的 $\lfloor 0.7n \rfloor$ 个点, 从第一个点开始计算均方根误差 (RMSE), 具体公式为:

$$\text{RMSE} = \frac{\sum_{i=1}^{\lfloor 0.7n \rfloor} \mu_i}{\lfloor 0.7n \rfloor} \quad (15)$$

式中: n 为点云 Y 中点的数量。

(3) 如果 RMSE 小于设定值或数值迭代次数达到上限, 算法终止。否则, 继续下一步。

(4) 利用步骤(2)中选取的数组中 $[0.7n]$ 点对,采用最小二乘法求解旋转矩阵 R 和平移向量 T 。

(5) 将旋转平移矩阵应用在点云 Y 上,得到新的对应点云 Y' ,具体公式为:

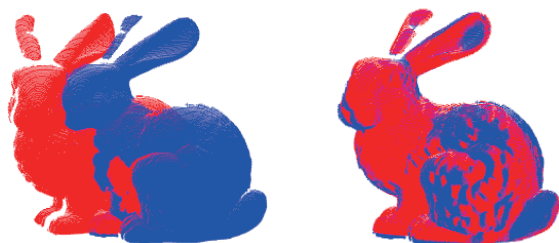
$$Y' = Y \times R + T \quad (16)$$

(6) $Y = Y'$, 返回步骤(1)。

3 实验结果与分析

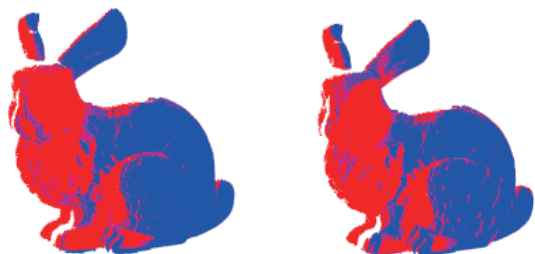
实验选取斯坦福公共点云数据模型中的Bunny、Dragon和Buddha验证以上算法的性能,并在VS2022环境下结合PCL库进行实现,硬件环境为Intel i3-8130CPU, 8 GB内存。

对于Bunny、Dragon和Buddha三个点云数据,分别采用ICP算法、4PCS+ICP算法和本文算法进行配准,配准结果如图3、图4、图5和表1所示。



(a) 待配准的 Bunny

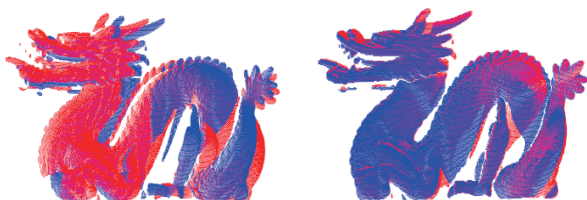
(b) ICP



(c) 4PCS+ICP

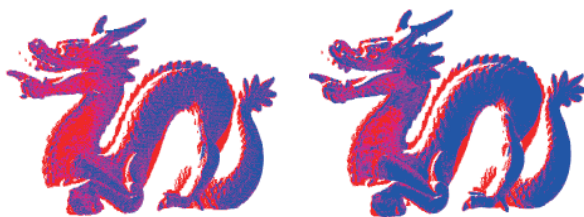
(d) 本文算法

图3 Bunny点云配准结果



(a) 待配准的 Dragon

(b) ICP



(c) 4PCS+ICP

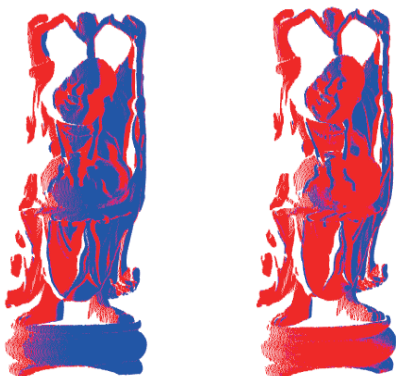
(d) 本文算法

图4 Dragon点云配准结果



(a) 待配准的 Buddha

(b) ICP



(c) 4PCS+ICP

(d) 本文算法

图5 Buddha点云配准结果

表1 公共点云配准结果

点云数据	点云数量	配准算法	配准误差/mm	耗时/s
Bunny	40 256, 40 097	ICP 算法	0.103 6	7.54
		4PCS+ICP 算法	0.095 1	6.88
		本文算法	0.074 0	5.83
Dragon	126 546, 132 000	ICP 算法	0.068 6	10.23
		4PCS+ICP 算法	0.051 1	8.67
		本文算法	0.042 4	7.27
Buddha	244 860, 244 200	ICP 算法	0.096 3	13.39
		4PCS+ICP 算法	0.087 9	11.93
		本文算法	0.071 9	9.45

根据图3、图4、图5和表1所示的实验结果,可以发现三种算法均能够有效处理公共点云数据模型,整体来看,配准效果最好的是本文算法,其次是4PCS+ICP算法,最后是ICP算法。与ICP算法相比,本文算法的平均配准精度提高了约31%,平均耗时降低了约27%;与4PCS+ICP算法相比,本文算法的平均配准精度提高了约19%,平均耗

时降低了约 17%。

这是由于 ICP 算法要求点云数据需要有良好的初始位姿,因此初始位姿较好的 Dragon 使用该算法的配准误差最低,并且 ICP 算法需要不断迭代来寻找对应点对,故 ICP 算法在三种方法中耗时最长;4PCS+ICP 算法采用 4PCS 对点云数据进行粗配准,使点云数据具有良好的初始位姿,但是配准精度和时间均有待提高;而本文算法是基于曲率改进的 K4PCS 点云配准算法,通过特征参数提取点云的特征点,减少了点云数据的规模,并利用曲率改进的 K4PCS 算法提高了配准精度,最后通过 TrICP 进一步缩短了配准时间,因此说明基于曲率改进的 K4PCS 点云配准算法是一种有效的点云配准算法。

4 总结

点云配准作为三维数据处理的关键步骤,其准确性直接影响到后续模型重建和实体描述的质量。因此,开发一种既精确又高效的点云配准算法至关重要。本文针对现有算法存在的配准精度不高、耗时长等问题,提出一种基于曲率改进的 K4PCS 点云配准算法,首先利用法向角度和马氏距离方差构造的特征描述符筛选特征点集,然后使用基于曲率改进的 K4PCS 算法进行粗配准,最后利用 TrICP 算法完成精确配准。实验采用斯坦福大学提供的 Bunny、Dragon 和 Buddha 模型,并与 ICP 算法和 4PCS+ICP 算法进行对比分析,结果表明该基于曲率改进的 K4PCS 点云配准算法在配准精度和耗时上均有明显提升,因此可以说本文算法是一种有效的点云配准算法。

参考文献:

- [1]YU H, HOU J, QIN Z, et al. Riga: rotation-invariant and globally-aware descriptors for point cloud registration[J]. IEEE transactions on pattern analysis and machine intelligence, 2024,46(5):3796-3812.
- [2]XU N, QIN R, SONG S. Point cloud registration for LiDAR and photogrammetric data: a critical synthesis and performance analysis on classic and deep learning algorithms[J]. ISPRS open journal of photogrammetry and remote sensing, 2023,8:100032.
- [3]WANG W, ZHANG Y, GE G, et al. A new approach toward corner detection for use in point cloud registration[J]. Remote sensing, 2023,15(13):33-55.
- [4]MONJI-AZAD S, HESSER J, LÖW N. A review of non-rigid transformations and learning-based 3D point cloud registration methods[J]. ISPRS journal of photogrammetry and remote sensing, 2023,196:58-72.
- [5]BESL P J, MC KAY N D. A method for registration of 3-D shapes[J]. IEEE transactions on pattern analysis & machine intelligence, 1992,14(2):239-256.
- [6]CHETVERIKOV D, SVIRKO D, STEPANOV D, et al. The trimmed iterative closest point algorithm[C]//Proceedings of the 2002 International Conference on Pattern Recognition. Piscataway:IEEE,2002:545-548.
- [7]BOUAZIZ S, TAGLIASACCHI A, PAULY M. Sparse iterative closest point[J]. Computer graphics forum, 2013,32:113-123.
- [8]WU P, LI W, YAN M, et al. 3D scene reconstruction based on improved ICP algorithm[J]. Microprocessors and microsystems, 2020,75:103064.
- [9]AIGER D, MITRA N J, COHEN-OR D. 4-points congruent sets for robust pairwise surface registration[J]. ACM transactions on graphics, 2008,27:1-10.
- [10]DA SILVA J P, BORGES D L, DE BARROS VIDAL F. A dynamic approach for approximate pairwise alignment based on 4-points congruence sets of 3D points[C]//Proceedings of the 2011 18th IEEE International Conference on Image Processing. Piscataway:IEEE,2011:889-892.
- [11]THEILER P W, WEGNER J D, SCHINDLER K. Keypoint-based 4-points congruent sets-automated marker-less registration of laser scans[J]. ISPRS journal of photogrammetry and remote sensing, 2014,96:149-163.
- [12]WALCZAK J, POREDA T, WOJCIECHOWSKI A. Effective planar cluster detection in point clouds using Histogram-Driven Kd-Like partition and shifted mahalanobis distance based regression[J]. Remote sensing, 2019,11(21):2465-2488.

【作者简介】

黄鹤(2000—),女,河南洛阳人,硕士研究生,研究方向:三维点云数据处理。

赵夫群(1982—),通信作者(email: fuqunzhao@126.com),女,山东临沂人,博士,教授,硕士生导师,研究方向:图形图像处理和三维重建。

(收稿日期:2024-07-04)