文章编号： 1001-3806(2020)06-0678-06

基于天牛须改进粒子群算法的点云配准方法

陈斯祺，张海洋*，赵长明，张子龙，王文鑫，张　明
（北京理工大学 光电学院 光电成像技术与系统教育部重点实验室，北京 100081）

摘要： 为了提高激光点云配准精度与配准速度，采用了基于天牛须算法改进的粒子群算法，以点云分布熵为寻优目标，寻找最优空间变换矩阵的点云粗配准，为点云精配准提供良好的初始条件。结果表明，点云分布熵较传统的均值平方差评价方式有更快的计算速度，基于天牛须算法改进的粒子群算法具有全局搜索能力强、计算速度快等特点，与传统点云粗配准方法相比，该方法配准速度提升了近 25%；在点云数据量大的条件下，表现出较快的配准速度。这一方法对如何提高激光点云配准速度具有参考意义。

关键词： 激光技术；激光雷达点云配准；点云分布熵；粒子群算法；天牛须算法
中图分类号： TN249　　**文献标志码：** A　　**doi**：10.7510/jgjs.issn.1001-3806.2020.06.005

Point cloud registration method based on particle swarm optimization algorithm improved by beetle antennae algorithm

CHEN Siqi, *ZHANG Haiyang*, *ZHAO Changming*, *ZHANG Zilong*, *WANG Wenxin*, *ZHANG Ming*
(Opto-electronics Imaging Technology and System Key Laboratory of Education Ministry, School of Optics & Photonics, Beijing Institute of Technology, Beijing 100081, China)

Abstract: In order to improve the accuracy and speed of LiDAR point cloud registration, a point cloud coarse entropy that used particle swarm algorithm based on the improved beetle algorithm was adopted. The method useed the point cloud distribution entropy as the optimization target to find the optimal spatial transformation matrix, and provided a good initial condition for the precise registration of the point cloud. Through theoretical analysis and simulation verification, it was proved that the point cloud entropy is an evaluation method which has faster calculation speed than the traditional mean squared difference, and the particle swarm optimization algorithm improved by beetle antennae algorithm has the characteristics of strong global search ability and fast calculation speed. Compared with the traditional point cloud coarse registration method, the registration method has improved the registration speed by nearly 25%. Under the condition of large amount of point cloud data, it still shows a faster registration speed. This method has a reference significance for how to improve the registration speed of laser point cloud.

Key words: laser technique; LiDAR point cloud registration; point cloud distribution entropy; particle swarm optimization algorithm; beetle antennae algorithm

引　言

激光雷达近年来在 3-D 场景感知及重建方面得到了广泛的应用，例如地形勘探、遗迹保护、城市建模、智能驾驶以及电力巡线等[1-6]，由激光雷达扫描获取的 3-D 点云数据经过点云滤波、特征点匹配、点云配准等步骤后获得，用于还原 3-D 场景[7-9]。点云配准是 3 维场景重建中至关重要的环节，配准方法可分为：手动配准、依赖于仪器的配准以及自动配准[10]。由于场景重建所需要的数据量较大，点云数量较多，手动配准与依赖于仪器的配准效率太低，一般采用自动配准方法[11]。自动配准方法所使用的算法不同，如何优化配准时间与精度成为了点云配准的主要研究方向。

最经典的点云配准方法是由 BESL 和 McKAY 提出的最近迭代点（iterative closest point，ICP）算法[12]，它能精度较高地实现点云配准，但计算量大，对点云数据的初始条件要求较高，且容易陷入局部最优。在 ICP 算法的基础上，众多学者对 ICP 算法进行了改进：BOUAZIZ 等人提出的稀疏 ICP（sparse iterative closest

基金项目：国家重点研发计划资助项目（2018YFF0300802）
作者简介：陈斯祺（1994-），男，硕士研究生，现主要从事激光点云处理、3 维场景重建的研究。
*通讯联系人。E-mail：ocean@bit.edu.cn
收稿日期：2019-11-25；收到修改稿日期：2019-12-19

point,SICP)算法[13],削弱了离群点对配准效果的影响,提高了配准精度;RUSINKIEWICZ 等人提出利用点-面对应来代替点-点对应[14],可以大大提高算法的稳健性和收敛精度。在优化算法方面,WU 等人尝试使用二进制编码的标准遗传算法(standard genetic algorithm,SGA)进行配准,但这一算法搜索空间大,匹配时间长[15];DUAN 等人使用粒子群优化算法寻找对应点后使用 ICP 算法实现散乱点云配准[16],速度较快但易陷入局部最优。

现阶段点云自动配准方法多数采用寻找特征点的方式进行匹配,优化特征点间的对应关系来实现点云配准,此方法计算量较大且仅在特征点较明显的场景下可以获得较好的结果。在优化算法层面,粒子群算法(particle swarm optimization,PSO)搜索速度快、效率高,算法简单,但对于离散优化问题处理效率不高,容易陷入局部最优[17]。天牛须搜索算法(beetle antennae search,BAS)在全局搜索最优解方面具有优势,可弥补粒子群算法易陷入局部最优的问题[18]。为解决以上问题提出一种基于天牛须改进的粒子群算法(particle swarm optimization algorithm improved by beetle antennae algorithm, BAS-PSO),以优化点云分布熵(spatial distribution entropy,SDE)为目标,寻找能够使点云分布熵最小的空间坐标变化关系,对于任意未知对应关系的两组点云数据,获得其旋转矩阵以实现点云粗配准,为精配准提供良好的初始值。

1　点云分布熵原理

本文中使用八叉树建立 3-D 立方栅格。将初始两组点云数据放至同一空间中,对两组点云数据分别进行中心化处理,将质心移至坐标原点后对合成的点云数据建立 3-D 立方栅格。由于每个栅格大小一致,而包含在栅格中的点云数量不同,即可通过同一栅格中点云数量的大小来表示在该区域内点云的稀疏程度,点云数量大表明该区域点云密集,数量小则表明该区域点云稀疏。

点云分布熵可用于描述点云在空间中的位置关系,通过密集程度反映点云的分布情况[19]。点云数据的采集往往是在不同视角的条件下进行的,点云数据之间并没有明显的对应关系[20],在经历配准操作后,点云数据会相对集中而非表现出配准前的不确定状态,这表现在两组点云数据相对距离最小,且空间中的位置关系唯一。点云信息熵可准确反映两组点云在配准过程中位置变换关系,故适用于点云配准过程中的优化对象。求解点云分布熵的具体步骤如图 1 所示。

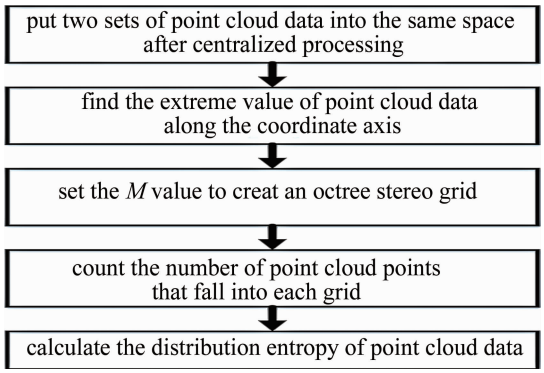

Fig. 1　Specific steps for calculating point cloud distribution entropy

在点云空间中,按照栅格数量切分,将同一空间下的两个点云数据切分为 $2^M \times 2^M \times 2^M$ 个栅格,即每个栅格的大小为:

$$\begin{cases} x = (X_{\max} - X_{\min})/2^M \\ y = (Y_{\max} - Y_{\min})/2^M \\ z = (Z_{\max} - Z_{\min})/2^M \end{cases} \tag{1}$$

式中,$X_{\max}$,$Y_{\max}$,$Z_{\max}$为点云数据坐标轴方向的最大值,$X_{\min}$,$Y_{\min}$,$Z_{\min}$为最小值,x,y,z 为每一个立体栅格沿坐标轴方向的长度。

将点云数据按照边界以及栅格数量进行栅格化后,统计落入每一个栅格中点云数量 $n(i,j,k)$,求得其分布概率 $p(i,j,k)$:

$$p(i,j,k) = n(i,j,k)/N \tag{2}$$

式中,N 为两组点云的总点云数量。点云分布熵的表达形式为:

$$E_{\mathrm{SDE}} = \sum_{i}^{2^M} \sum_{j}^{2^M} \sum_{k}^{2^M} p(i,j,k) \ln[p(i,j,k)] \tag{3}$$

相比 ICP 算法通过求算均值平方差(mean square error,MSE)来评价点云配准的精度,点云分布熵的求算方式时间复杂度更小。MSE 的求算方法需要为待配准点云中每一个点在目标点云中找寻与之距离最小的点,这需要多次全局遍历,而点云分布熵的评价方式只需进行一次遍历,运算速度有较大提高。

对完全重合的两组 bunny 点云数据 P,Q 进行点云分布熵计算操作。在 $-180° \sim 180°$范围内将 Q 绕 x 轴旋转,每 1°生成一个新的点云数据 Q_m,Q_m 与 P 共同组成新的点云数据 T_m,计算 T_m 的点云分布熵 E_{SDE} 和均值平方差 E_{MSE}。下标 m 表示旋转角 α,β,γ 的角度。以旋转角度 α 为横坐标,E_{SDE} 与 E_{MSE} 为纵坐标建立与旋转角度的对应关系,如图 2 所示。Q 绕 x 轴与 y 轴旋转计算两个维度变换的点云分布熵。

如图 3 所示,SDE 与 MSE 都能很好地反映两组点

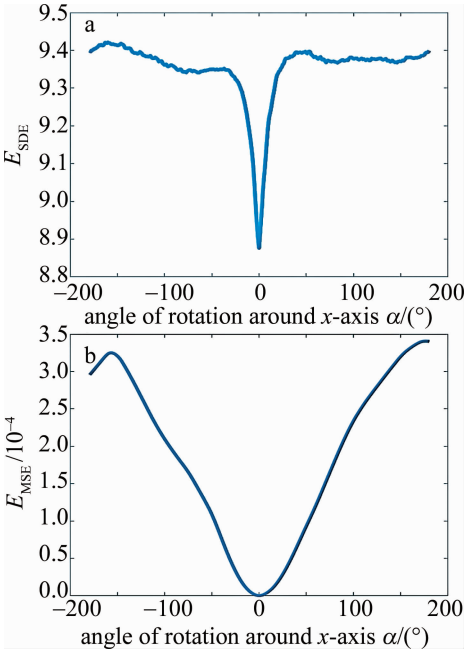

Fig. 2 SDE and MSE curve with rotation angle α

a—correspondence between SDE and rotation angle b—correspondence between MSE and rotation angle

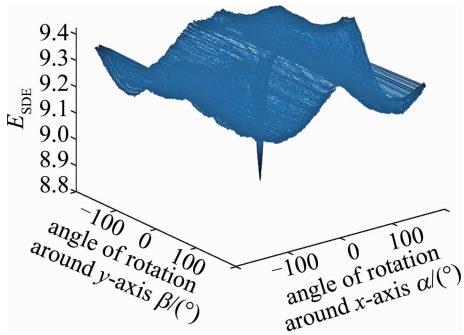

Fig. 3 SDE curve with rotation angle α and β

云的重合程度,当旋转角度为 0°时,SDE 与 MSE 均为最小值,适合作为点云配准的评价标准。

在运算速度方面,对不同大小的点云数据,点云分布熵的计算方式都明显优于均值平方差的计算方式,实验数据如表 1 所示。

Table 1 Compare between SDE and MSE calculation time

data(point number)	time of MSE/s	time of SDE/s
bunny(35947)	52.4695	0.9349
horse(48485)	86.6042	1.2046
dragon(109411)	262.6325	2.6982

2 基于 BAS-PSO 算法的点云配准方法

2.1 点云配准

针对点云配准问题,其实质就是寻找能使两组点云数据尽可能重合的旋转矩阵[21]。由激光雷达采集获得的激光点云数据因采集视角不同,同一物体的点云数据在空间上存在着刚性变换关系,即两组点云数据中对应点都可通过同一种空间坐标变换方式使之尽可能重合,数学表达式如下式所示:

$$\begin{bmatrix} x_{i,p} \\ y_{i,p} \\ z_{i,p} \end{bmatrix} = \boldsymbol{R}\begin{bmatrix} x_{i,q} \\ y_{i,q} \\ z_{i,q} \end{bmatrix} + \boldsymbol{T} \tag{4}$$

式中,p 表示目标点云数据,q 表示待配准点云数据,$\boldsymbol{R}$ 为旋转矩阵,$\boldsymbol{T}$ 为平移矩阵。在点云粗配准中,平移矩阵可通过中心化的方式,将两组点云数据的质心移至坐标原点来尽可能的消除平移误差,即:

$$\begin{bmatrix} x_{i,p}' \\ y_{i,p}' \\ z_{i,p}' \end{bmatrix} = \boldsymbol{R}\begin{bmatrix} x_{i,q}' \\ y_{i,q}' \\ z_{i,q}' \end{bmatrix} \tag{5}$$

旋转矩阵 $\boldsymbol{R}$ 可通过点绕 x 轴、y 轴、z 轴旋转的角度 α,β,γ 确定,其表达方式为:

$$\boldsymbol{R} = \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\beta \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix} \tag{6}$$

寻找点云配准的最优旋转矩阵即寻找最优的旋转角度 α,β,γ。

2.2 BAS-PSO 算法

粒子群优化算法是模拟鸟群捕食行为的一种寻优算法[22],它的基本思想是将每一个个体视为 n 维空间中没有质量和体积的粒子,粒子包含位置与飞行速度两个属性,每一个粒子用一个 n 维矢量表示,可以视为 n 维空间中的一个寻优解,在寻优过程中,每个粒子以自身的飞行速度更新自己的位置,通过记录每个位置的适应度来确定个体极值 p_{best} 和粒子群体的群体极值 g_{best},找到全局最优解并由此来调整自己的位置与速度,适应度由目标函数决定。粒子群优化算法通过群体寻优的正反馈机制,根据个体与群体的对目标函数的适应度不断调整个体状态,将个体逐步迁移至较优区域,从而获得目标函数的最优解[23]。

粒子群优化算法中粒子速度 v 与位置 x 更新的数学表达如下式所示:

$$\begin{cases} v_i(t+1) = v_i(t) + c_1 r_1(t)[p_{i,best} - x_i(t)] + \\ \qquad c_2 r_2(t)[g_{best} - x_i(t)] \\ x_i(t+1) = x_i(t) + v_i(t+1) \end{cases} \tag{7}$$

式中，c_1 和 c_2 为非负的学习参量；r_1，r_2 为取值范围在(0,1)之间的随机数，以保证群体的多样性；t 表示迭代次数；$p_{i,\mathrm{best}}$ 为粒子群中第 i 个粒子所记录的个体适应度极值；g_{best} 为当前整个粒子群中记录的适应度极值，通过设置适合的学习参量并不断迭代逐步获得问题的最优解。

粒子群优化算法虽然在收敛速度上具有优势，但由于缺乏粒子速度的动态调节，容易陷入局部最优，导致在收敛后期的收敛精度低和不易收敛[24]。针对以上问题，可使用天牛须搜索算法为粒子速度调整提供自主寻优的能力。

天牛须搜索算法是 2017 年提出的一种新型仿生优化算法，在搜索速度和全局搜索方面有一定的优势，可用于粒子群算法粒子速度调节优化[24]，其仿生学原理为：自然界里天牛在觅食的过程中，由于不知道食物位置，只能通过气味来寻找食物的大致方向。天牛有左右两个触须，它可通过左右两触须所探测到的气味强度来判断食物在自身位置的左右方位，例如当左须探测到的气味比右触须探测到的气味更强时，天牛向左触须方向移动，移动一段距离后再次使用左右触须探测当前位置食物气味直至找到气味最强即食物的位置。在天牛须搜索算法中，食物的具体位置即为寻优的极值点，食物气味相当于寻优函数本身，通过逐步逼近的方式获得最优解。

天牛须搜索算法的数学表达如下：

(1)在 n 维空间中设置质心位置为 x，其左触须位置为 x_1，右触须位置为 x_r，用 d_0 表示两须之间的距离，根据步长与两须间距离成正比这一特点，可设置步长为：

$$s = cd_0 \tag{8}$$

式中，c 是一个设定比例。由于质心的方向是随机的，即右触须指向左触须的向量也是随机的，所以用一个随机 n 维向量来表示右触须指向左触须的方向，即：

$$\boldsymbol{d} = \mathrm{rands}(n,1) \tag{9}$$

(2)左右触须长度相同，且连线方向的法向量指向质心，则可以根据质心位置 x，两须间距 d_0 以及右触须指向左触须的向量 $\boldsymbol{d}$ 表达左右触须的位置。将 $\boldsymbol{d}$ 归一化处理后可以得到：

$$x_1 - x_r = d_0\boldsymbol{d}/\|\boldsymbol{d}\| \tag{10}$$

$$x_1 = x + d_0\boldsymbol{d}/(\|\boldsymbol{d}\|2) \tag{11}$$

$$x_r = x - d_0\boldsymbol{d}/(\|\boldsymbol{d}\|2) \tag{12}$$

(3)对于目标函数 f，分别求得 x_1 和 x_r 两个位置的值 f_1 和 f_r，并比较两个值的大小，为寻求最小值，则当 $f_1 < f_r$ 时，质心向左须方向移动距离 s，当 $f_1 > f_r$ 时，质心向右须方向移动距离 s，可表示为：

$$\begin{cases} x = x + s(x_1 - x_r), (f_1 < f_r) \\ x = x - s(x_1 - x_r), (f_1 > f_r) \end{cases} \tag{13}$$

质心移动后，按照比例改变步长大小：

$$s = \theta s \tag{14}$$

式中，θ 为设置的比例系数，一般取值为 0.95，循环(2)步、(3)步即可获得最优解。

基于天牛须改进的粒子群算法的基本思路是将粒子群中个体最优适应度值的比较过程改为天牛须搜索算法寻优，通过这种方式更新个体与群体的最优适应度值。使用 BAS-PSO 算法，以点云分布熵作为优化目标求解获得最佳的空间变换关系，设置旋转角度[α, β, γ]为目标解，通过寻找点云分布熵值最小时对应的解[α, β, γ]来获得点云配准时最优的旋转矩阵实现点云粗配准。

该算法的操作步骤如图 4 所示。

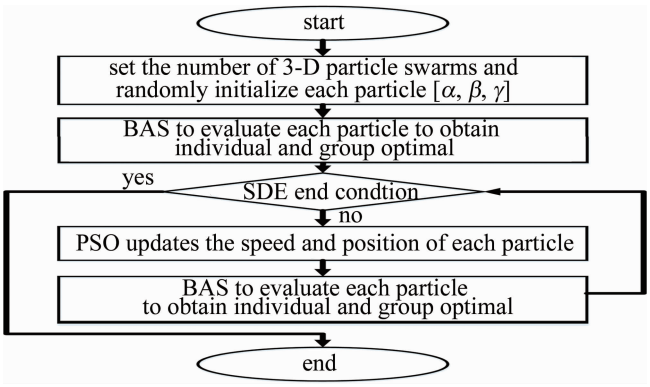

Fig. 4　The flowchart of BAS-PSO

3　实验结果与分析

为验证算法可行性及优化效果，作者在 Intel Core-i7，8 GB 内存的 Windows 10 操作系统上使用 MATLAB R2018a 软件对斯坦福大学点云数据库中的 bunny，horse，dragon 点云进行点云配准实验，以不同视角下的点云数据 P，Q 为操作对象，使用以点云分布熵为优化目标，基于 BAS-PSO 算法进行点云粗配准。基于经验对算法中的参量设置如表 2 所示。

为观测 BAS-PSO 算法中每一代更新后 SDE 的优化效果，提取出每一代配准后点云的 SDE，以 bunny 模型为例，构建了 SDE 随进化次数变化的曲线图，如图 5 所示。

从图 5 中可以看出，随着粒子种群的进化，SDE 不断减小并向优化方向进行，在第 30 次更新种群后，SDE 趋于平稳，其值接近于两点云重合时计算获得的

Table 2 Algorithm parameter setting

parameter	number
c_1	1.4995
c_2	1.4995
θ	0.95
population size	100
number of evolutions	100
step	π

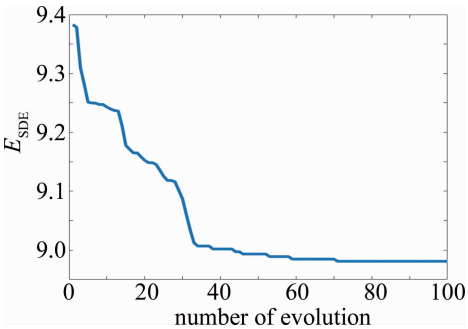

Fig. 5 Evolutionary curve with SDE as the optimization goal

SDE 值,证明 BAS-PSO 算法是一种有效的优化算法。

针对部分缺失的点云数据以及含有噪声的点云数据,本文中的算法依旧有较好的鲁棒性,配准效果如图 6 所示。

使用 BAS-PSO 算法配准效果与主成分分析法(principal component analysis,PCA)、四点集法(4-point congruent sets,4PCS)以及基于遗传算法的粗配准算法进行对比,配准后的模型如图 7 所示。对比实验结果如表 3 所示。

Table 3 Registration time of different point cloud coarse registration method

	4PCS	SGA	PCA	BAS-PSO
bunny	62.39s	78.62s	1.02s	47.83s
horse	89.74s	109.25s	1.33s	53.55s
dragon	146.53s	167.82s	2.11s	95.64s

由仿真结果可知,在粗配准效果上,BAS-PSO 算法与 4PCS 算法以及基于遗传算法的粗配准方法均明显优于 PCA 算法,在配准速度上,虽然 PCA 算法速度快,但在考虑配准效果的前提下,BAS-PSO 算法优于 4PCS 算法与基于遗传算法的粗配准方法,如图 8 所示。针对不同大小的点云数据,BAS-PSO 算法在配准速度上同样具有优势,且针对数据量大的点云数据,配准速度优势越大。

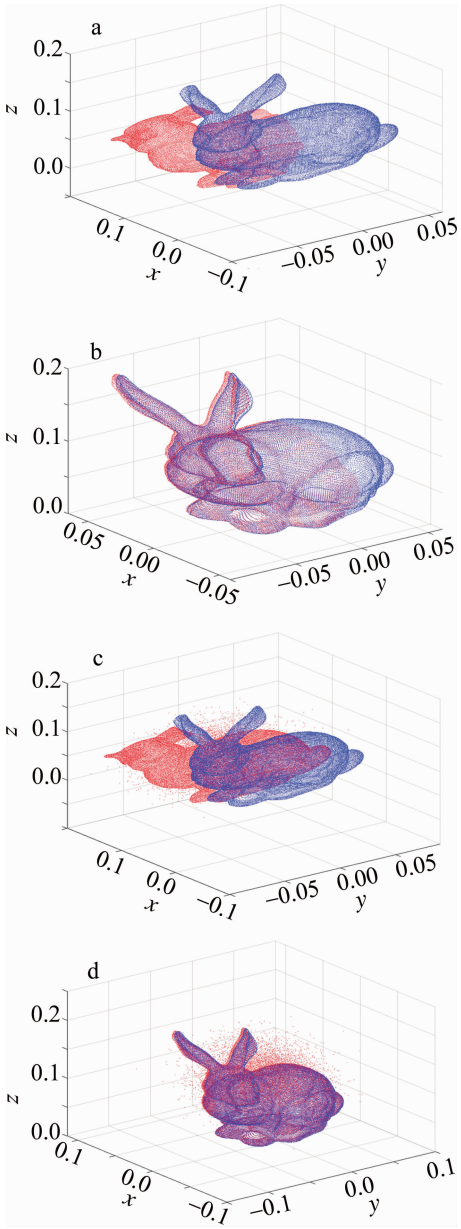

Fig. 6 Abnormal point cloud data registration effect

a—partial cloud missing before registration b—post-registration effect c—point cloud with noise before registration d—post-registration effect

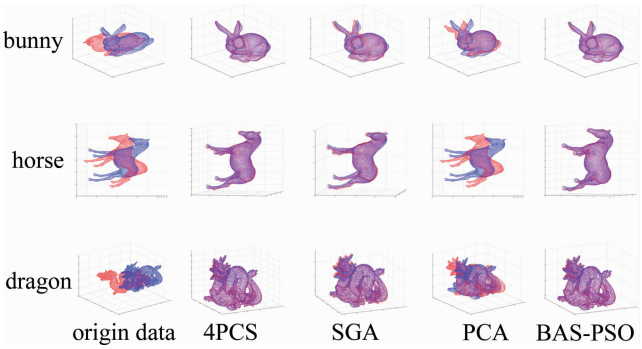

Fig. 7 Registration effect of different point cloud coarse registration method

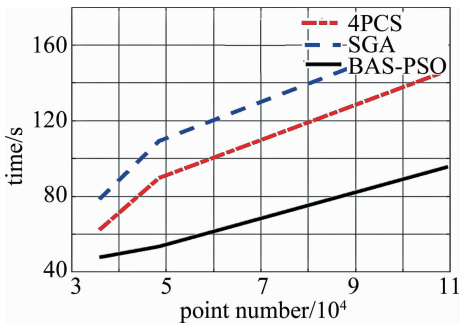

Fig. 8　The curve of the registration time with the number of point clouds

4　结　论

提出了一种以点云分布熵为优化目标，基于天牛须改进的粒子群算法用于激光点云数据的粗配准。在优化目标上，点云分布熵计算量明显小于传统 ICP 算法所使用的均值平方差，且点云分布熵可以准确反映点云配准效果，可作为配准算法中的寻优目标。在寻优算法方面，BAS-PSO 算法可以有效弥补天牛须搜索算法个体单一，寻优步长收敛过快，以及粒子群算法易陷入局部最优的问题，实现既精准又快速的点云配准。本文中通过对基于 BAS-PSO 算法的点云粗配准方法与 PCA 算法、4PCS 算法以及基于遗传算法的点云粗配准方法进行了实验对比，证实了 BAS-PSO 算法在配准速度上的优势，在点云数据量较大的条件下，BAS-PSO 算法也能实现点云的粗配准，为 ICP 算法提供良好的初始状态。

针对算法中关键参量如搜索步长和学习参量的设定为基于经验的人工手动设定，无法确定是否达到算法最优，寻找一种自动化设置关键参量的方法，保证算法执行效率，提升点云配准效率是下一步需要继续改进的方向。

参　考　文　献

[1] HE Zh B, TIAN Y R. Filtering algorithm for non-ground point from airborne laser scanner data[J]. Journal of Geodesy and Geodynamics, 2009, 29(4): 97-101(in Chinese).

[2] HUA L, HUANG H Y, CHEN Ch Ch, *et al*. Three dimensional modeling of Hakka earth buildings based on the laser scanned point cloud data[J]. Remote Sensing Technology and Application, 2015, 30(1): 115-122(in Chinese).

[3] ZHAO H Y, ZHANG Zh P, *et al*. The application of 3-D laser point cloud data in the city modeling[J]. Urban Geotechnical Investigation & Surveying, 2009(1): 69-72(in Chinese).

[4] CHEN X D, ZHANG J Ch, PANG W S, *et al*. Key technology and application algorithm of intelligent driving vehicle LiDAR[J]. Opto-Electronic Engineering, 2019, 46(7): 34-46(in Chinese).

[5] SHI L, GUO T, PENG Ch, *et al*. Segmentation of laser point cloud and safety detection of power lines[J]. Laser Technology, 2019, 43(3): 341-346(in Chinese).

[6] WU J J, CHEN L, LI L, *et al*. Power line extraction and reconstruction from airborne LiDAR point cloud[J]. Laser Technology, 2019, 43(4): 64-69(in Chinese).

[7] HU X, LI X, ZHANG Y. Fast filtering of LiDAR point cloud in urban areas based on scan line segmentation and GPU acceleration[J]. IEEE Geoscience and Remote Sensing Letters, 2013, 10(2): 308-312.

[8] KEDZIERSKI M, FRYSKOWSKA A. Methods of laser scanning point clouds integration in precise 3-D building modelling[J]. Measurement, 2015, 74: 221-232.

[9] VáZQUEZ-ARELLANO M, REISER D, PARAFOROS D S, *et al*. 3-D reconstruction of maize plants using a time-of-flight camera[J]. Computers and Electronics in Agriculture, 2018, 145: 235-247.

[10] DAI J L, CHEN Zh Y, YE X Z. The application of ICP algorithm in point cloud alignment[J]. Journal of Image and Graphics, 2007, 12(3): 517-521(in Chinese).

[11] LI T Sh, WANG Y M, HU Ch M. Research on automatic point cloud registration based on laser reflection intensity[J]. Bulletion of Surveying and Mapping, 2014(s2):143-145.

[12] BESL P J, McKAY N D. A method for registration of 3-D shapes[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 1992, 14(2): 239-256.

[13] BOUAZIZ S, TAGLIASACCHI A, PAULY M. Sparse iterative closest point[J]. Computer Graphics Forum, 2013, 32(5):12-14.

[14] RUSINKIEWICZ S, LEVOY M. Efficient variants of the ICP algorithm[J]. Proceedings of the SPIE, 2001,1497:149701.

[15] WU D L, HUANG H L, DING Y Ch, *et al*. Surfaces matching algorithm based on genetic algorithm and least square criterion[J]. Acta Aeronautica ET Astronautica Sinica, 2002, 23(3): 285-288(in Chinese).

[16] DUAN D Q, LI J F, SHEN P P, *et al*. Registration of scattered cloud point data based on PSO[J]. Journal of Guangxi Normal University (Natural Science Edition), 2008, 26(3): 226-229(in Chinese).

[17] SHE X X, XU B. Improved particle swarm opetimization algorithm based on genetic thought[J]. Journal of Yangtze University(Natural Science Edition), 2016, 13(22): 4-8(in Chinese).

[18] XIANG Y J, SHUAI L. Beetle antennae search algorithm for optimization problems[J]. International Journal of Robotics and Control, 2018, 1(1):37-42.

[19] CHEN J, CAI Y, ZHANG J Sh. Point cloud registration based on entropy criterion genetic algorithm[J]. Application Research of Computers, 2019, 36(1): 322-326(in Chinese).

[20] LI H X. Lidar point cloud feature analysis and data segmentation[D]. Xi'an: Xidian University, 2014:57-64(in Chinese).

[21] TAO H Q, DA F P. Automatic registration algorithm for the point clouds based on the normal vector[J]. Chinese Journal of Lasers, 2013,40(8):809001(in Chinese).

[22] SHI Y. A modified particle swarm optimizer[J]. Proceedings of the SPIE, 1998, 7803:699146.

[23] UGOLOTTI R, CAGNONI S. Multi-objective parameter tuning for PSO-based point cloud localization[M]. Berliln, Germary: Springer International Publishing, 2014:75-85.

[24] CHEN T T, YIN H, JINAG H L, *et al*. Particle swarm optimization algorithm based on beetle antennae search for solving portfolio problem[J]. Computer Systems & Applications, 2019, 28(2): 171-176(in Chinese).