

基于相关系数平方和最大的三维点云配准

苗长伟*, 唐志荣, 唐英杰

成都理工大学核技术与自动化工程学院, 四川 成都 610059

摘要 点云配准是三维重建过程的核心问题之一。针对点云散乱、数据存在缺失及噪声干扰情况下配准效率差、精度低的问题,提出了一种基于相关系数平方和最大(MCC)的点云配准算法。分别对目标点云与待配准点云去均值化后进行旋转,使旋转后的两组点云各自满足行向量间相关系数平方和最大;再利用粒子群优化算法分别求解出两组中间态旋转矩阵;最后根据两组中间态旋转矩阵求解出两点云之间的旋转矩阵和平移向量,进而实现点云配准。仿真结果表明,在点云散乱、数据存在缺失以及噪声干扰的情况下,本文算法比现有其他算法的配准速度更快、精度更高,且稳健性良好。

关键词 机器视觉; 点云配准; 相关系数; 粒子群优化; 数据缺失

中图分类号 TP391.9

文献标识码 A

doi: 10.3788/LOP56.221504

Three-Dimensional Point Cloud Registration Based on Maximum Sum of Squares of Correlation Coefficients

Miao Changwei¹*, Tang Zhirong¹, Tang Yingjie¹

¹ College of Nuclear Technology and Automation Engineering, Chengdu University of Technology, Chengdu, Sichuan 610059, China;

Abstract Point cloud registration is a fundamental element of the three-dimensional reconstruction processes. In this study, a point cloud registration algorithm is proposed based on the maximum sum of squares of the correlation coefficients (MCC) to address the issues of scattered point clouds, missing data, and low registration efficiency and accuracy under noise interference. Further, the target point cloud and the point cloud to be registered are de-averaged and rotated, so that the MCC between row vectors of the two sets of point clouds can be achieved after rotation. Subsequently, particle swarm optimization algorithm is used to derive two sets of intermediate-state rotation matrices. Finally, based on these matrix sets, the rotation matrix and translation vector between two point clouds are obtained for registering the point cloud. The simulation results show that the proposed algorithm is faster, more accurate, and more robust compared with the remaining existing algorithms when point clouds are scattered, missing, and interrupted by noise.

Key words machine vision; point cloud registration; correlation coefficient; particle swarm optimization; data missing

OCIS codes 150.6910; 110.6880; 110.3010

1 引言

近些年,在点云配准技术快速发展的趋势下,三维点云配准已被广泛应用于三维重建、产品无损检测和机器视觉等领域。三维点云配准指对同一物体在不同视角下的部分三维点云数据进行拼接和变换,最终得到一个统一坐标系下的完整三维点云的

过程。在现有的诸多配准算法中,由 Besl 等^[1]提出的迭代最近点算法(ICP)最经典,该算法基于四元素运算模型,寻找两点云间最近对应点,并计算其最优变换矩阵。该算法实现简单且精度较高,但收敛性过度依赖初始值。

为有效改善点云配准的速度和准确度,众学者对传统 ICP 算法进行了相应改进,或提出了更好的

收稿日期: 2019-04-02; 修回日期: 2019-05-02; 录用日期: 2019-05-17

* E-mail: 853360040@qq.com

初始配准算法。如: Rusu 等^[2]提出了运用点特征直方图(PFH)提取点云一点处 16D 局部几何特征的配准算法,该算法能得到较好的初始值,但计算复杂;卢章平等^[3]提出了 RANSAC+ICP 配准算法,以共面 4 点集为基础,实现自由曲面的点云配准,该算法提高了 ICP 的配准精度,但对匹配曲面类型有一定的局限性;李仁忠^[4]等提出了基于 ISS 特征点结合改进 ICP 的配准算法,利用 ISS 算法对点云进行特征提取,并以特征直方图描述特征,运用 k 维树近邻搜索法快速寻找对应点,有效提高了 ICP 的配准效率;Ying 等^[5]提出了基于七维空间迭代的 Scale-ICP 算法,该算法能用于不同尺度的配准,但稳健性较差。

此外,也有学者提出了非 ICP 点云配准算法,如混合高斯模型(GMM)^[6-7],该算法稳定性较好,但运算效率较低,且过分依赖初始值。基于关键点的配准算法^[8-11]利用关键点不变的特性实现了点云的精配准,但其配准运算效率较低。Myronenko 等^[12]提出了相干点漂移算法(CPD),在有噪声和异常值、缺失点的情况下,该算法与 GMM 算法结合可以实现对点云的准确配准,但对散乱点云、数据存在遮挡以及缺失的配准较难实现。刘鸣等^[13]提出了基于独立成分分析的三维点云配准算法,利用点云数据与独立分量的对应关系实现点云配准,该算法速度快,且有一定的抗噪声能力,但对匹配数据点数量有一定要求。唐志荣等^[14]提出了基于多维混合柯西分布的点云配准方法,该算法通过数据中心点进行初配准,再利用协方差完成精配准,具有一定的抗噪声能力,且配准精度较高,但效率有待提高。王畅等^[15]提出了利用全局和局部结构特征不变的特性实现点云配准的算法,该算法的配准精度和效率均有所提升,但点云数据缺失严重和交叉数据点不足可能会导致该算法配准效果不佳,甚至失败。

为了提高点云配准的效率和精度,本文提出了基于相关系数平方和最大的点云配准方法。该算法首先将两点云中心化转换到同一坐标下,再以点云各维向量间相关系数平方和最大为约束条件,以粒子群算法作为转动矩阵的求解算法,将两点云分别绕坐标原点转动至此位置,从而求得旋转矩阵与平移向量,实现配准。由于在三维空间中,随着点云的旋转变换,各维度间的相关性会发生改变,同时点云的旋转实质上是乘以一个旋转矩阵,各维度间不会达到平行,故选取在空间中能使各维度相关性达最大的位置作为约束条件,将点云旋转到该位置,从而

实现配准。实验结果表明,与其他算法相比,在点云散乱、数据存在缺失以及有噪声的环境下,本文算法有较高的配准速度和精度。

2 基于相关系数最大的点云配准

传统 ICP 算法中目标点云 $P = (p_1, p_2, \dots, p_n)$ 和待配准点云 $Q = (q_1, q_2, \dots, q_m)$ 的元素均属于三维点云,两者进行刚性几何变换的表达式为

$$p_i = RQ_j + T, i = 1, 2, \dots, n; j = 1, 2, \dots, m, \quad (1)$$

式中: p_i 为第 i 个目标数据点; q_j 为第 j 个待配准数据点; n 为目标点云的点数; m 为待配准点云的点数; R 为旋转矩阵, $R \in \mathbf{R}^{3 \times 3}$ 且 $R^T R = I$; I 为单位矩阵; T 为平移向量, $T \in \mathbf{R}^{3 \times 1}$ 。传统 ICP 算法配准对点云数据的初始值要求较高,只有在合理初配准的基础上,才能实现对点云的精确配准。

为提高点云配准的精度和速度,采用相关系数最大法,以点云各维向量间相关系数平方和最大作为约束条件,对点云进行配准。首先求出目标点云 P 和待配准点云 Q 的均值点,即

$$\bar{P} = \sum_{i=1}^n p_i / n, \bar{Q} = \sum_{j=1}^m q_j / m, \quad (2)$$

然后对目标点云 P 和待配准点云 Q 进行中心化处理,使目标点云和待配准点云的中心均处于原点,从而得到点云 $\hat{P} = (\hat{p}_1, \hat{p}_2, \dots, \hat{p}_n)$ 和点云 $\hat{Q} = (\hat{q}_1, \hat{q}_2, \dots, \hat{q}_m)$, 即

$$\hat{p}_i = p_i - \bar{P}, \hat{q}_j = q_j - \bar{Q}, \quad (3)$$

式中: \hat{p}_i 为去中心化的第 i 个目标数据点; \hat{q}_j 为去中心化的第 j 个待配准数据点。中心化后的点云 \hat{P} 与 \hat{Q} 满足

$$\hat{P} = R\hat{Q}. \quad (4)$$

在三维空间中,随着点云的旋转变换,各维度间的相关性发生改变。从数学模型上看,点云的旋转实质上是乘以一个旋转矩阵,所以各维度间不会达到平行。因此选取空间中能使各维度间相关性达最大的位置作为目标,将两组点云旋转到该位置,从而达到配准的目的。对点云 \hat{P} 和 \hat{Q} 进行旋转,使其满足各自的行向量之间的相关系数最大,即

$$\begin{cases} \bar{P} = R_p \hat{P} = (R_{p_1}^T \hat{P}, R_{p_2}^T \hat{P}, R_{p_3}^T \hat{P})^T = (\bar{p}_1, \bar{p}_2, \bar{p}_3)^T \\ \bar{Q} = R_q \hat{Q} = (R_{q_1}^T \hat{Q}, R_{q_2}^T \hat{Q}, R_{q_3}^T \hat{Q})^T = (\bar{q}_1, \bar{q}_2, \bar{q}_3)^T \end{cases}, \quad (5)$$

式中: \mathbf{R}_p 为正交矩阵, $\mathbf{R}_p = (\mathbf{R}_{p_1}, \mathbf{R}_{p_2}, \mathbf{R}_{p_3})$, 满足 $\mathbf{R}_p \mathbf{R}_p^T = \mathbf{R}_p^T \mathbf{R}_p = \mathbf{I}$; $\mathbf{R}_{p_1}, \mathbf{R}_{p_2}, \mathbf{R}_{p_3}$ 均为 3×1 的列向量; $\tilde{\mathbf{p}}_1, \tilde{\mathbf{p}}_2, \tilde{\mathbf{p}}_3$ 均为 $n \times 1$ 的列向量; \mathbf{R}_Q 为正交矩阵, $\mathbf{R}_Q = (\mathbf{R}_{Q_1}, \mathbf{R}_{Q_2}, \mathbf{R}_{Q_3})$, 满足 $\mathbf{R}_Q \mathbf{R}_Q^T = \mathbf{R}_Q^T \mathbf{R}_Q = \mathbf{I}$; $\mathbf{R}_{Q_1}, \mathbf{R}_{Q_2}, \mathbf{R}_{Q_3}$ 均为 3×1 的列向量; $\tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2, \tilde{\mathbf{q}}_3$ 均为 $m \times 1$ 的列向量。当点云 $\tilde{\mathbf{P}}$ 和 $\tilde{\mathbf{Q}}$ 均满足行相关系数最大时, 可得

$$\tilde{\mathbf{P}} = \tilde{\mathbf{Q}}. \quad (6)$$

综上可得

$$\mathbf{R} = \mathbf{R}_p^{-1} \mathbf{R}_Q, \quad (7)$$

$$\mathbf{T} = \tilde{\mathbf{P}} - \mathbf{R}_p^{-1} \mathbf{R}_Q \tilde{\mathbf{Q}}. \quad (8)$$

2.1 相关系数最大法

点云 $\tilde{\mathbf{P}}$ 各行向量之间的相关系数为

$$\rho_{(\tilde{\mathbf{p}}_\alpha, \tilde{\mathbf{p}}_\beta)} = \frac{\text{Cov}(\tilde{\mathbf{p}}_\alpha, \tilde{\mathbf{p}}_\beta)}{\sigma_{\tilde{\mathbf{p}}_\alpha} \sigma_{\tilde{\mathbf{p}}_\beta}} = \frac{\mathbf{r}_{p_\alpha}^T \boldsymbol{\Sigma}_p \mathbf{R}_{p_\beta}}{\sqrt{\mathbf{r}_{p_\alpha}^T \boldsymbol{\Sigma}_p \mathbf{R}_{p_\alpha}} \sqrt{\mathbf{r}_{p_\beta}^T \boldsymbol{\Sigma}_p \mathbf{R}_{p_\beta}}}, \quad (9)$$

式中: Cov 为协方差; σ 为标准差; $\boldsymbol{\Sigma}_p$ 为点云 \mathbf{P} 的协方差矩阵, $\boldsymbol{\Sigma}_p \in \mathbf{R}^{3 \times 3}$; $\alpha = 1, \beta = 2, 3$; $\alpha = 2, \beta = 3$, 表示行数。为使各行向量相关系数平方和达最大, 需

$$J(\rho_p) = \sum \rho_{(\tilde{\mathbf{p}}_\alpha, \tilde{\mathbf{p}}_\beta)}^2, \quad (10)$$

式中: $J(\cdot)$ 为相关系数平方和; ρ_p 为目标点云; $\rho_{(\tilde{\mathbf{p}}_\alpha, \tilde{\mathbf{p}}_\beta)}$ 为相关系数。

点云 $\tilde{\mathbf{Q}}$ 各行向量之间的相关系数为

$$\rho_{(\tilde{\mathbf{q}}_\alpha, \tilde{\mathbf{q}}_\beta)} = \frac{\text{Cov}(\tilde{\mathbf{q}}_\alpha, \tilde{\mathbf{q}}_\beta)}{\sigma_{\tilde{\mathbf{q}}_\alpha} \sigma_{\tilde{\mathbf{q}}_\beta}} = \frac{\mathbf{r}_{Q_\alpha}^T \boldsymbol{\Sigma}_Q \mathbf{r}_{Q_\beta}}{\sqrt{\mathbf{r}_{Q_\alpha}^T \boldsymbol{\Sigma}_Q \mathbf{r}_{Q_\alpha}} \sqrt{\mathbf{r}_{Q_\beta}^T \boldsymbol{\Sigma}_Q \mathbf{r}_{Q_\beta}}}, \quad (11)$$

式中: $\boldsymbol{\Sigma}_Q$ 为点云 \mathbf{Q} 的协方差矩阵, $\boldsymbol{\Sigma}_Q \in \mathbf{R}^{3 \times 3}$ 。为使各行向量相关系数平方和达最大, 需

$$J(\rho_Q) = \sum \rho_{(\tilde{\mathbf{q}}_\alpha, \tilde{\mathbf{q}}_\beta)}^2, \quad (12)$$

式中: ρ_Q 为待配准点云; $\rho_{(\tilde{\mathbf{q}}_\alpha, \tilde{\mathbf{q}}_\beta)}$ 为相关系数。

2.2 基于粒子群优化算法 (PSO) 的参数确定

粒子群优化算法中没有遗传算法的交叉和变异运算操作, 通过粒子速度迭代更新实现解空间搜索, 并且在迭代进化过程当中, 只有最优解的粒子才会把信息传递给其他粒子, 简单易实现, 且搜索速度快。因此, 基于粒子群优化算法优化(10)式和(12)式, 从而对 \mathbf{R}_p 和 \mathbf{R}_Q 进行求解。

设定在 D 维的目标空间中, 有一个由 N 个粒子构成的群体, 其中第 f 个粒子为一个 D 维的向量, 其 t 时刻的速度更新为

$$\mathbf{V}_{fD}^{(t+1)} = \mathbf{V}_{fD}^{(t)} + c_1 r_1 (\mathbf{p}_{fD}^{(t)} - \mathbf{R}_{fD}^{(t)}) + c_2 r_2 (\mathbf{p}_{gD}^{(t)} - \mathbf{R}_{fD}^{(t)}), \quad f = 1, 2, \dots, N, \quad (13)$$

式中: $\mathbf{V}_{fD}^{(t+1)}$ 和 $\mathbf{V}_{fD}^{(t)}$ 分别为第 f 个粒子在 $t+1$ 和 t 时刻的速度变化; c_1 和 c_2 为学习因子; r_1 和 r_2 为该区域内随机分布的伪随机数, 取值 $[0, 1]$; $\mathbf{p}_{fD}^{(t)}$ 为第 f 个粒子在 t 时刻经过的最优位置; $\mathbf{p}_{gD}^{(t)}$ 为群体内所有粒子在 t 时刻经过的最优位置; $\mathbf{R}_{fD}^{(t)}$ 为第 f 个粒子在 t 时刻的位置。

粒子速度更新过后, 即更新当前时刻的位置后, PSO 位置更新为

$$\mathbf{R}_{fD}^{(t+1)} = \mathbf{R}_{fD}^{(t)} + \mathbf{V}_{fD}^{(t+1)}, \quad (14)$$

更新后, $\mathbf{R}_{fD}^{(t+1)}$ 可能不满足正交的约束条件, 故对其采用奇异值分解, 即

$$\mathbf{R}_{fD}^{(t+1)} = \mathbf{u} \mathbf{v} \mathbf{s}, \quad (15)$$

式中: \mathbf{u} 和 \mathbf{v} 均为酉矩阵, 满足 $\mathbf{I} = \mathbf{u} \mathbf{u}^H = \mathbf{v} \mathbf{v}^H$, H 为共轭转置; \mathbf{s} 为对角矩阵, 除了对角线的元素, 其他均为 0, 主对角线上的每个元素都称为奇异值。重新得到正交矩阵

$$\mathbf{R}_{fD}^{*(t+1)} = \mathbf{u} \mathbf{v}^H = (\mathbf{r}_1^{*(t+1)}, \mathbf{r}_2^{*(t+1)}, \mathbf{r}_3^{*(t+1)}), \quad (16)$$

式中: $\mathbf{R}_{fD}^{*(t+1)}$ 为通过奇异值分解求得的正交矩阵; $\mathbf{r}_1^{*(t+1)}, \mathbf{r}_2^{*(t+1)}, \mathbf{r}_3^{*(t+1)}$ 均为 3×1 的列向量。

结合(16)式可得(10)式和(12)式的展开式, 即

$$J(\rho)^{(t+1)} = \left(\frac{\mathbf{r}_1^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_2^{*(t+1)}}{\sqrt{\mathbf{r}_1^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_1^{*(t+1)}} \sqrt{\mathbf{r}_2^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_2^{*(t+1)}}} \right)^2 + \left(\frac{\mathbf{r}_2^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_3^{*(t+1)}}{\sqrt{\mathbf{r}_2^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_2^{*(t+1)}} \sqrt{\mathbf{r}_3^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_3^{*(t+1)}}} \right)^2 + \left(\frac{\mathbf{r}_3^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_1^{*(t+1)}}{\sqrt{\mathbf{r}_1^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_1^{*(t+1)}} \sqrt{\mathbf{r}_3^{*T(t+1)} \boldsymbol{\Sigma}_p \mathbf{r}_3^{*(t+1)}}} \right)^2, \quad (17)$$

式中: $\boldsymbol{\Sigma}$ 取值分别为 $\boldsymbol{\Sigma}_p, \boldsymbol{\Sigma}_Q$; 通过(13)~(17)式求解出 \mathbf{R}_p 和 \mathbf{R}_Q , 使其分别满足(9)式和(11)式。

本文算法流程如图 1 所示

3 实验仿真

采用 Stanford 大学提供的经典 Bunny(31607) 三维点云数据, 以及一组机械器件(28437)的实物点云数据进行仿真实验。该仿真实验是在 MATLAB 2017a 版本、i7-6700HQ 四核处理器和 GTX965M 下进行的。

在仿真实验中, 粒子群优化算法参数设置^[16]如下: 种群 $N=100$; 搜索空间维度 $D=3$; r_1, r_2 采用介于 $(0, 1)$ 之间的数值, 用 $\text{rand}()$ 函数随机生成, 以

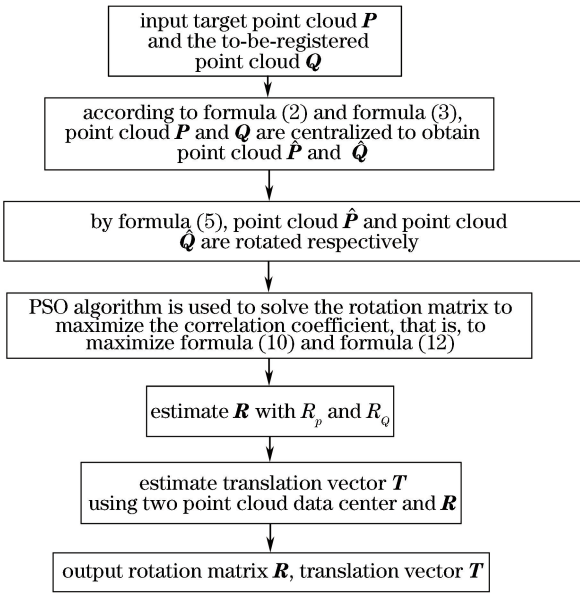


图 1 本文算法流程图

Fig. 1 Flowchart of our algorithm

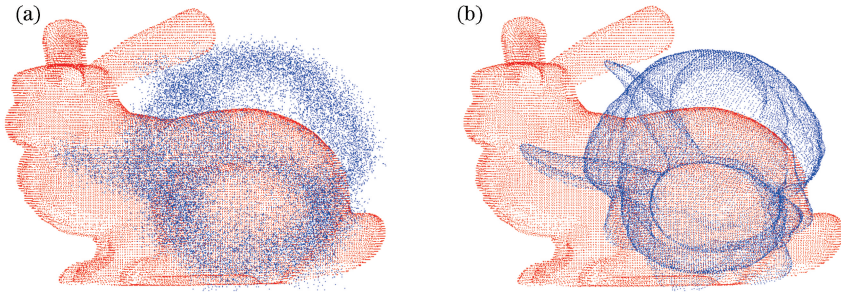


图 2 点云 Bunny 的初始状态。(a)加高斯白噪声;(b)未加高斯白噪声

Fig. 2 Initial state of point cloud for Bunny. (a) With white Gaussian noise; (b) without white Gaussian noise

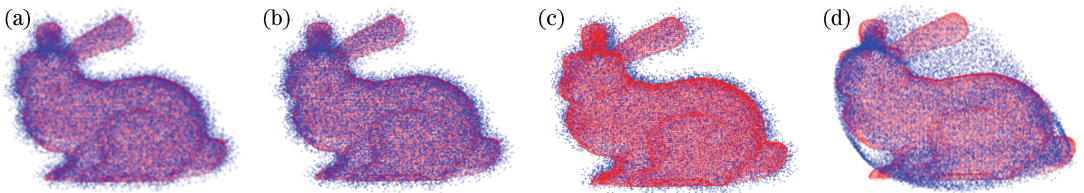


图 3 各算法对 Bunny 点云配准效果。(a) MCC;(b) ICP;(c) Scale-ICP;(d) CPD

Fig. 3 Registration effect of each algorithm for Bunny point cloud. (a) MCC; (b) ICP; (c) Scale-ICP; (d) CPD

各算法的配准时间和误差数据如表 1 所示。

表 1 各算法对 Bunny 点云的配准时间及误差

Table 1 Registration time and error of each algorithm for Bunny point cloud

| Algorithm | Time /s | RMSE /mm |
|-----------|---------|----------|
| MCC | 7.6 | 0.2240 |
| ICP | 174.5 | 0.2239 |
| Scale-ICP | 8.5 | 0.2346 |
| CPD | 135.0 | 0.3654 |

均方根误差定义为

$$E_{\text{RMSE}} =$$

$$\sqrt{\frac{1}{n} \sum_{i=1}^n [(x_i^Q - x_i^P)^2 + (y_i^Q - y_i^P)^2 + (z_i^Q - z_i^P)^2]}, \quad (18)$$

式中: $i=1, 2, \dots, n$; n 为点云的点数; x_i^Q, y_i^Q, z_i^Q 为点云 Q 中第 i 个点的三维坐标; x_i^P, y_i^P, z_i^P 为点云 P 中第 i 个点的三维坐标。

从图 3 和表 1 可知: 在 Bunny 点云数据随机丢失率为 25%、高斯白噪声为 20 dB 的情况下, MCC

保证其随机性; c_1, c_2 为学习因子, 通常等于 2, 不过在文献中也有其他取值。由于本文的目标函数是相关系数的最大值, 且相关系数是不大于 1 的数, 故取值不宜过大, 否则可能会跳过真值, 引起不收敛, 也不宜过小, 否则容易造成收敛过慢。经实验, c_1, c_2 取值 0.5 最合适。

3.1 在有噪声且数据存在丢失情况下的配准

为了验证本文算法在数据存在丢失、遮挡以及噪声干扰情况下的配准效果, 在待配准数据 Bunny 点云添加 20 dB 高斯白噪声, 随机丢失 25% 的数据后进行随机旋转和平移, 目标点云和待配准点云的初始状态如图 2 所示。高斯白噪声的功率谱密度服从均匀分布, 幅度服从高斯分布, 其均值为零; 此外, 加高斯白噪声是为了对目标点云进行偏置。

在同等环境条件下, 用本文算法 (MCC 算法)、ICP 算法、Scale-ICP 算法和 CPD 算法分别配准, 并对结果进行分析。配准后的效果如图 3 所示。

算法的配准速度最快,MCC算法、传统ICP算法和Scale-ICP算法的配准精度几乎相等,且比CPD算法的配准误差低0.1 mm以上;CPD算法的配准效果最差,且容易导致待配准点云的形态失真。

3.2 机械器件点云数据丢失配准

为充分验证MCC算法对实物扫描数据进行配准的有效性,通过MATLAB 2017a进行仿真验证。首先用便携式激光扫描仪(HandySCAN 300™,加拿大)对机械器件进行实物数据采集,然后对扫描数据作相应处理,再进行配准。机械器件有28437点,对数据作25%的随机丢失处理,再进行随机旋转和平移,然后进行配准。目标点云和待配准点云的初始状态如图4所示。

在同等环境条件下,用MCC算法、ICP算法、Scale-ICP算法和CPD算法分别对机械器件点云进行配准,并对配准结果进行分析。配准效果如图5所示。

各算法对机械器件点云的配准时间以及配准后

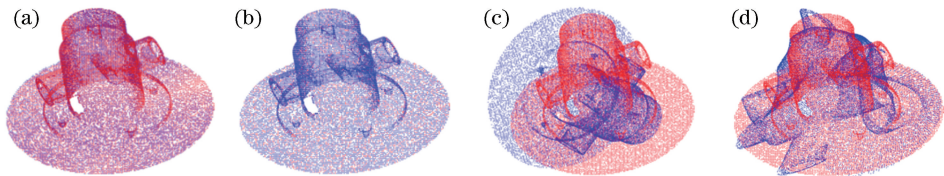


图4 机械器件点云的初始状态。(a)待配准点云;
(b)目标点云

Fig. 4 Initial state of point cloud for mechanical device.

(a) Point cloud to be registered; (b) target point cloud

的误差如表2所示。

由图5和表2可知:在机械器件点云数据随机丢失率为25%的情况下,MCC算法与ICP算法的配准精度几乎相当,且本文MCC算法的配准速度更快;Scale-ICP算法的配准误差最大,不能完成对机械器件的点云配准;CPD算法导致待配准点云形态失真,稳定性较差。

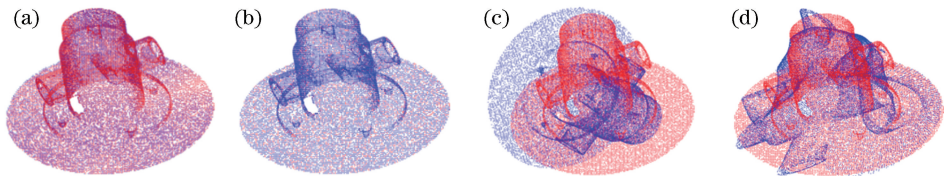


图5 各算法对机械器件点云的配准效果。(a) MCC;(b) ICP;(c) Scale-ICP;(d) CPD

Fig. 5 Registration effect of each algorithm of point cloud for mechanical device. (a) MCC; (b) ICP;

(c) Scale-ICP; (d) CPD

表2 各算法对机械点云的配准时间及误差

Table 2 Registration time and error of each algorithm for mechanical device

| Algorithm | Time /s | RMSE /mm |
|-----------|---------|----------|
| MCC | 6.9 | 0.3508 |
| ICP | 76.3 | 0.3500 |
| Scale-ICP | 5.6 | 20.7587 |
| CPD | 108.6 | 3.0992 |

4 结 论

提出了一种基于相关系数平方和最大的配准算法,该算法能对存在部分缺失、遮挡且带有噪声以及散乱的两组点云数据完成自动配准。分别对目标点云与待配准点云进行去均值化,然后进行旋转,旋转时需满足中间态旋转矩阵为正定矩阵,再利用点云各维向量间相关系数平方和最大作为约束条件,采用粒子群优化算法分别求解出两组中间态旋转矩阵,再由两组中间态旋转矩阵求解出两点云间的旋转矩阵和平移向量,从而实现点云配准。本文算法在配准精度上与经典ICP算法相当,且配准效率更

高。与Scale-ICP算法和CPD算法相比,MCC算法具有较高的配准速度和精度,且稳定性较好。

参 考 文 献

- [1] Besl P J, McKay N D. A method for registration of 3-D shapes [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14 (2): 239-256.
- [2] Rusu R B, Blodow N, Marton Z C, *et al.* Aligning point cloud views using persistent feature histograms[C]// 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, September 22-26, 2008, Nice, France. New York: IEEE, 2008: 3384-3391.
- [3] Lu Z P, Zheng H, Sha C F, *et al.* Point cloud registration algorithm based on free-form surface[J]. Journal of Jiangsu University (Natural Science Edition), 2015, 36(3): 319-323.
卢章平, 郑航, 沙春发, 等. 基于自由曲面的点云配准算法[J]. 江苏大学学报(自然科学版), 2015, 36

- (3): 319-323.
- [4] Li R Z, Yang M, Tian Y, *et al.* Point cloud registration algorithm based on the ISS feature points combined with improved ICP algorithm[J]. *Laser & Optoelectronics Progress*, 2017, 54(11): 111503.
李仁忠, 杨曼, 田瑜, 等. 基于 ISS 特征点结合改进 ICP 的点云配准算法[J]. *激光与光电子学进展*, 2017, 54(11): 111503.
- [5] Ying S H, Peng J G, Du S Y, *et al.* A scale stretch method based on ICP for 3D data registration[J]. *IEEE Transactions on Automation Science and Engineering*, 2009, 6(3): 559-565.
- [6] Li Q S, Xiong R, Vidal-Calleja T. A GMM based uncertainty model for point clouds registration[J]. *Robotics and Autonomous Systems*, 2017, 91: 349-362.
- [7] Jian B, Vemuri B C. Robust point set registration using Gaussian mixture models [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(8): 1633-1645.
- [8] Zhang Z, Xu H L, Yin H. A fast point cloud registration algorithm based on key point selection [J]. *Laser & Optoelectronics Progress*, 2017, 54(12): 121001.
张哲, 许宏丽, 尹辉. 一种基于关键点选择的快速点云配准算法[J]. *激光与光电子学进展*, 2017, 54(12): 121001.
- [9] Prakhya S M, Liu B B, Lin W S, *et al.* B-SHOT: a binary 3D feature descriptor for fast keypoint matching on 3D point clouds [J]. *Autonomous Robots*, 2017, 41(7): 1501-1520.
- [10] Ge X M. Automatic markerless registration of point clouds with semantic-keypoint-based 4-points congruent sets [J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017, 130: 344-357.
- [11] Quan S W, Ma J, Hu F Y, *et al.* Local voxelized structure for 3D binary feature representation and robust registration of point clouds from low-cost sensors[J]. *Information Sciences*, 2018, 444: 153-171.
- [12] Myronenko A, Song X B. Point set registration: coherent point drift [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, 32(12): 2262-2275.
- [13] Liu M, Shu Q, Yang Y X, *et al.* Three-dimensional point cloud registration based on independent component analysis [J]. *Laser & Optoelectronics Progress*, 2019, 56(1): 011203.
刘鸣, 舒勤, 杨赟秀, 等. 基于独立成分分析的三维点云配准算法[J]. *激光与光电子学进展*, 2019, 56(1): 011203.
- [14] Tang Z R, Liu M Z, Wang C, *et al.* Point cloud registration based on multi-dimensional mixed Cauchy distribution[J]. *Acta Optica Sinica*, 2019, 39(1): 0115005.
唐志荣, 刘明哲, 王畅, 等. 基于多维混合柯西分布的点云配准[J]. *光学学报*, 2019, 39(1): 0115005.
- [15] Wang C, Shu Q, Yang Y X, *et al.* Quick registration algorithm of point clouds using structure feature [J]. *Acta Optica Sinica*, 2018, 38(9): 0911005.
王畅, 舒勤, 杨赟秀, 等. 利用结构特征的点云快速配准算法[J]. *光学学报*, 2018, 38(9): 0911005.
- [16] Wang D F, Meng L. Performance analysis and parameter selection of PSO algorithms [J]. *Acta Automatica Sinica*, 2016, 42(10): 1552-1561.
王东风, 孟丽. 粒子群优化算法的性能分析和参数选择[J]. *自动化学报*, 2016, 42(10): 1552-1561.