Postprint: Artificial Bee Colony Point Cloud Registration Algorithm Based on Curvature Information

Authors: Fu Kun, Chen Lei

Date: 2019-01-28T00:00:00+00:00

Abstract

To address the shortcomings of slow correspondence search between two point clouds and slow convergence speed in the optimization process when using only swarm intelligence optimization algorithms and point cloud spatial information for point cloud registration, this paper proposes an artificial bee colony point cloud registration algorithm based on curvature information. The algorithm extracts feature points based on curvature information, and obtains the optimal transformation matrix that can align the two point clouds by optimizing the objective function through an improved artificial bee colony algorithm. During the population optimization process, the search range for corresponding points is constrained based on curvature information, thereby reducing the scale of point clouds participating in the computation. Comparative experiments demonstrate that, compared with registration algorithms that only employ random point selection methods and utilize point cloud spatial coordinate information, the proposed algorithm can effectively accelerate the registration convergence speed and significantly reduce the time required for point cloud registration without compromising registration accuracy.

Full Text

Preamble

Point Cloud Registration Based on Artificial Bee Colony Algorithm and Curvature Information

Fu Kun[1], Chen Lei[1,2] [1] School of Electrical & Information Engineering, Tianjin University, Tianjin 300072, China [2] School of Information Engineering, Tianjin University of Commerce, Tianjin 300134, China

Abstract: Methods for point cloud registration that rely solely on swarm intelligence optimization algorithms and spatial information suffer from slow convergence and high computational costs due to the time-consuming search for corresponding points. To address these limitations, this paper proposes a registration algorithm that incorporates curvature information. The algorithm extracts feature points based on curvature information and obtains the optimal transformation matrix for aligning two point clouds by optimizing the objective function through an improved artificial bee colony algorithm. During population optimization, the search range for corresponding points is constrained using curvature information, thereby reducing the scale of point clouds involved in computation. Comparative experiments demonstrate that, compared with registration algorithms using random point selection and spatial coordinate information alone, the proposed algorithm effectively accelerates convergence and significantly reduces registration time without compromising accuracy.

Key words: point cloud; curvature information; feature point selection; corresponding point search; artificial bee colony algorithm

0 Introduction

Three-dimensional imaging involves acquiring 3D point cloud data of an object from different viewpoints using 3D scanning devices to reconstruct the object' s complete morphology. In this field, point cloud registration is a critical technique that seeks an optimal Euclidean transformation to align partially overlapping point clouds of the same object, captured from different angles, into a unified coordinate system.

Point cloud registration is fundamentally an optimization process. Traditional methods for solving this optimization problem are susceptible to quantization errors, point cloud noise, and initial positioning. For instance, the classic Iterative Closest Points (ICP) algorithm [1] often converges to local optima and fails when the initial positions of point clouds are unfavorable. Although scholars have proposed improved ICP variants—such as Chen et al. [2] who used median point-to-tangent-plane distances rather than point-to-point distances as the objective function, and reference [3] which employed principal component analysis for coarse registration followed by a curvature-based ICP refinement—these approaches still suffer from susceptibility to local convergence.

In recent years, swarm intelligence optimization algorithms have emerged as powerful methods for various engineering optimization problems. References [4-7] applied particle swarm optimization to parameter estimation for permanent magnet synchronous motors in industrial systems, effectively improving performance. Reference [8] introduced particle swarm optimization into speech recognition, enhancing success rates. Reference [9] utilized bee colony optimization in digital image processing to address false positives in watermarking. Reference [10] applied ant colony optimization to embedded system design, achieving successful hardware-software co-design.

In 3D imaging, swarm intelligence-based point cloud registration has become a research hotspot due to the excellent global optimization capabilities of these algorithms, demonstrating clear advantages over traditional methods. For example, Cordón et al. [11] proposed an image registration algorithm based on bacterial foraging, Santamaría [12] developed a CHC algorithm-based approach, and Agrawal et al. [13] introduced an evolutionary rigid-body docking algorithm.

While these swarm intelligence-based methods address the limitations of gradient-based approaches and improve adaptability and accuracy, they require numerous iterations to achieve precise results. Each iteration involves computing objective function values for population individuals, resulting in substantial computational overhead. Consequently, reducing computational complexity while maintaining registration accuracy has become a key research focus.

This paper proposes an artificial bee colony (ABC) [14] point cloud registration algorithm that leverages curvature information. The algorithm first extracts subsets of points through both random sampling and curvature-based sampling. During correspondence search, it excludes points with significantly different curvature ranges, thereby narrowing the search space and reducing computational load. Finally, the optimal transformation is obtained through ABC optimization. Comparative experiments confirm that constraining the correspondence search using curvature information effectively reduces registration time while preserving accuracy.

1 Point Cloud Registration Model

The goal of point cloud registration is to merge static point clouds Q and dynamic point clouds P, acquired from different viewpoints, into a unified 3D coordinate system to obtain a complete geometric model. In spatial coordinates, a rigid transformation can be represented by a rotation-translation matrix T. For a dynamic point cloud $P = \{p_i | i = 1, 2, \dots, N_P\}$ and static point cloud $Q = \{q_j | j = 1, 2, \dots, N_Q\}$, the objective is to find the optimal Euclidean transformation matrix T that aligns corresponding points in the overlapping regions.

The transformation matrix T contains six parameters: translations T_x, T_y, T_z along three axes and rotations α, β, γ about three axes. The matrix is expressed as $T = R_X R_Y R_Z S$, where R_X, R_Y, R_Z are rotation matrices and S is the translation matrix:

$$R_X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R_Y = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$R_Z = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, S = \begin{bmatrix} 1 & 0 & 0 & T_X \\ 0 & 1 & 0 & T_Y \\ 0 & 0 & 1 & T_Z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ideally, for a point $p_i \in P$ transformed by T, the distance to its correspondence $q_j \in Q$ should be zero. In practice, due to measurement errors and noise, this distance is:

$$d_i = \|T(p_i) - q_j\|$$

Thus, registration becomes an optimization problem: find the optimal transformation matrix T that minimizes the Euclidean distance between all corresponding point pairs:

$$\min_T E(T) = \sum_i \|T(p_i) - q_j\|$$

Incorporating the artificial bee colony algorithm, we use the median of squared distances as the objective function:

$$F(T) = \mathrm{MedSE} = \mathrm{Med}(d_i^2)$$

where MedSE represents the median of squared distances between corresponding points in the two point clouds.

2 Improved Artificial Bee Colony Algorithm

The artificial bee colony algorithm, proposed by Karaboga et al., mimics honey bee foraging behavior and has been widely applied to multi-dimensional optimization problems [15-17]. Each solution represents a food source, evaluated by an objective function. The colony consists of three types of bees: employed bees, onlooker bees, and scout bees.

Employed bees search for food sources (solutions) and modify partial information during each iteration, retaining the better solution based on objective function values. They then share information with onlooker bees through a waggle dance. Onlooker bees probabilistically select food sources, make modifications, and also retain superior solutions. If no improvement occurs after a specified number of iterations, onlooker bees abandon their sources and become scouts to search for new ones.

In the basic ABC algorithm, employed and onlooker bees share the same search equation:

$$V_{ij} = X_{ij} + \phi_{ij}(X_{ij} - X_{kj})$$

where ϕ_{ij} is a random number in $[-1, 1]$.

For complex multi-dimensional nonlinear problems, ABC offers advantages including few control parameters, good global convergence, and easy implementation. However, its local exploitation capability is limited, resulting in slow convergence. Addressing this, Gao et al. proposed the Enhanced ABC (EABC) algorithm [19], inspired by differential evolution (DE) [18], which uses separate search equations for employed and onlooker bees:

Employed bee:

$$V_{ij} = X_{ij} + \alpha(X_{r_1 j} - X_{\text{best}j}) + \beta(X_{r_2 j} - X_{r_3 j})$$

Onlooker bee:

$$V_{ij} = X_{ij} + \alpha(X_{r_1 j} - X_{\text{best}j}) + \beta(X_{r_2 j} - X_{r_3 j})$$

where $X_{\text{best}j}$ is the current global best solution, r_1, r_2, r_3 are distinct random integers, α is a random number in $[0, A]$, and β is drawn from a Gaussian distribution with mean μ and standard deviation σ. These equations balance exploration and exploitation by incorporating global best guidance while maintaining diversity through perturbations.

3 Artificial Bee Colony Registration Algorithm Based on Point Cloud Curvature Information

Point cloud registration suffers from high computational complexity, time consumption, and inaccurate correspondence search. Recent work has utilized feature information such as local curvature [20], FPFH [21], and LSP [22] in coarse registration. Curvature information particularly helps reduce mismatches since different surface regions typically exhibit distinct curvature values.

3.1 Feature Point Sampling Based on Curvature Information

For each point in point clouds P and Q, we compute Gaussian curvature k_{Gaussian} and mean curvature k_{Mean} using the Moving Least Squares (MLS) method [23]:

$$k_{\text{Gaussian}} = k_1 \cdot k_2$$
$$k_{\text{Mean}} = \frac{k_1 + k_2}{2}$$

where k_1 and k_2 are the principal curvatures. Following Chen et al. [24], we define curvature information for each point p as:

$$S(p) = \frac{1}{\pi} \arctan\left(\frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)}\right)$$

where $0 < S(p) < 1$. The point cloud can then be represented as $\{p_i, S(p_i) | i = 1, 2, \dots, N_P\}$.

To ensure uniform distribution and avoid local convergence, we combine random sampling with curvature-based sampling. The curvature range $[0, 1]$ is divided into n segments. Dynamic point cloud P is partitioned into subsets P_k where points satisfy:

$$\frac{k-1}{n} < S(p) \le \frac{k}{n}, \quad k = 1, 2, \dots, n$$

Similarly, static point cloud Q is divided into subsets Q_k where:

$$\frac{k-1.5}{n} < S(q) \le \frac{k+0.5}{n}, \quad k = 1, 2, \dots, n$$

[Figure 1: see original paper] illustrates this partitioning for $n = 3$.

The sampling process works as follows: randomly select an integer k from $[1, n]$. If subset P_k is non-empty and its corresponding Q_k is also non-empty, randomly select a point from P_k as a feature point. Repeat until obtaining N_{feature} curvature-based points. To avoid registration failure from local fitting, additionally extract N_{random} points via random sampling. The combined set forms the final registration subset. Empirical results show optimal performance when $N_{\text{feature}} : N_{\text{random}} = 2 : 1$.

[Figure 2: see original paper] shows the sampling procedure schematically.

3.2 Objective Function Constrained by Curvature Information

During optimization, the algorithm continuously searches for correspondences of each point in the transformed dynamic point cloud within the static point cloud, with the median squared distance evaluating registration quality. This correspondence search dominates computational time.

To accelerate this, we propose a curvature-constrained objective function $\psi(T)$. Since corresponding points have similar curvature values, we restrict the search to curvature-similar subsets. After finding correspondences, we compute the median squared distance as the objective function value.

The algorithm flow for $\psi(T)$ is:

a) Transform dynamic point cloud P using matrix T to obtain $T(P)$

b) For each transformed point $T(p_i)$, compute its curvature segment k using:

$$k = \left\lfloor \frac{S(p_i)}{1/n} \right\rfloor + 1$$

c) Search for correspondence q_j only within subset Q_k

d) Compute Euclidean distance $d_i = \|T(p_i) - q_j\|$

e) Calculate objective function value using $\psi(T) = \mathrm{Med}(d_i^2)$

To further accelerate search, a kd-tree data structure is built for each static point cloud subset Q_k.

4 Experiments

4.1 3D Point Cloud Data

Experiments used four point cloud models from the SAMPL database: Tele, Bird, Angel, and Frog. For each model, point clouds from 0° and 40° viewpoints were selected. The 0° views contained 6,841, 9,115, 14,089, and 9,640 points respectively, while the 40° views contained 5,612, 5,872, 13,190, and 7,650 points. During registration, one point cloud remained static while the other was randomly rotated and translated to serve as the dynamic point cloud.

4.2 Experimental Design

Two experiments were conducted to evaluate performance:

Experiment 1: Compared the proposed curvature-based improved ABC registration algorithm (referred to as "our algorithm") against an improved ABC algorithm using only spatial coordinate information (referred to as "spatial algorithm"). summarizes the strategies.

Experiment 2: Compared our algorithm against two recent swarm intelligence-based methods: ABC2016 [25] (improved ISS feature points with ABC) and DE2014 [26] (adaptive invasive differential evolution). Both experiments used identical parameters, platforms, data models, and evaluation criteria.

4.3 Parameter Settings

Experiments ran on an Intel Core i7-4790 processor (3.6 GHz) with 16 GB RAM, Windows 8, and MATLAB R2014a. Through preliminary testing, optimal parameters were determined: population size of 20 (equal employed and onlooker bees), limit = 50, Gaussian distribution parameters $\mu = 0.3$, $\sigma = 0.3$. For each model, the 0° point cloud served as dynamic input P, with 300 sampling points selected. Before registration, P was randomly translated within $[-40\text{mm}, 40\text{mm}]$ and rotated within $[0, 2\pi]$ about all axes. details the EABC parameters.

4.4 Analysis of Experiment 1 Results

Both algorithms converged within approximately 160 seconds, so experiments were limited to 200 seconds. Every 10 seconds, the best transformation vector ρ was recorded and applied to the complete dynamic point cloud, yielding $T(P)$. The median squared distance L_r between $T(P)$ and static cloud Q was computed at each interval $r = 1, 2, \dots, 20$. Convergence curves were plotted using L_r values after 50 seconds for clarity. Registration success was determined by threshold τ_m specific to each model ().

[Figure 3: see original paper] shows convergence curves for all models. Our algorithm typically converged within 100 seconds, while the spatial algorithm required approximately 150 seconds. presents success rates, average objective values, and iteration counts over 30 runs (failed runs excluded). Our algorithm achieved higher success rates (only one failure across all models) and significantly more iterations within the same time frame (e.g., 6,614 vs. 3,891 for Bird model), demonstrating reduced per-iteration computation. [Figure 4: see original paper] visualizes the registration quality.

4.5 Analysis of Experiment 2 Results

Our algorithm was compared against ABC2016 and DE2014. lists the parameters for each method. All algorithms ran 30 times with objective function values recorded every 10 seconds. [Figure 5: see original paper] shows convergence curves, while summarizes success rates and final accuracy.

Our algorithm demonstrated faster convergence and higher stability (fewer failures) compared to both ABC2016 and DE2014, without sacrificing accuracy. The curvature constraint effectively reduced function complexity and accelerated optimization.

5 Conclusion

This paper proposes a curvature-based point cloud registration method that improves sampling and correspondence search. By combining curvature-based and random sampling, the algorithm reduces the search space during optimization, decreasing computational overhead. The improved EABC algorithm then finds the optimal transformation. Experiments confirm that our approach achieves good registration quality with faster convergence and higher success rates compared to spatial-only ABC methods and other recent swarm intelligence approaches, making it highly competitive.

References

[1] Besl P J, Mckay N D. A method for registration of 3-D shapes [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1992, 14 (2): 239-256.

[2] Chen Y, Medioni G. Object modeling by registration of multiple range images [J]. Image and Vision Computing, 1992, 10 (3): 145-155.

[3] Yang Xianhui, Wang Huinan. Application research of ICP algorithm in 3D point cloud alignment [J]. Computer Simulation, 2010, 27 (8): 235-238.

[4] Liu Zhaohua, Wei Hualiang, Li Xiaohua, et al. Global identification of electrical and mechanical parameters in PMSM drive based on dynamic self-learning PSO [J]. IEEE Trans on Power Electronics, 2018, 33 (12): 10926-10937.

[5] Liu Zhaohua, Wei Hualiang, Zhong Qingchang, et al. Parameter estimation for VSI-Fed PMSM based on a dynamic PSO with learning strategies [J]. IEEE Trans on Power Electronics, 2017, 32 (4): 3069-3079.

[6] Liu Zhaohua, Wei Hualiang, Zhong Qingchang, et al. GPU implementation of DPSO-RE algorithm for parameters identification of surface PMSM considering VSI nonlinearity [J]. IEEE Journal of Emerging and Selected Topics in Power Electronics, 2017, 5 (3): 1347-1356.

[7] Liu Zhaohua, Zhang Jing, Zhou Shaowu, et al. Coevolutionary particle swarm optimization using AIS and its application in multi-parameter estimation of PMSM [J]. IEEE Trans on Cybernetics, 2013, 43 (6): 1921-1933.

[8] Yogesh C K, Hariharan M, Ngadiran R, et al. A new hybrid PSO assisted biogeography-based optimization for emotion and stress recognition from speech signal [J]. Expert Systems with Applications, 2017, 69: 149-158.

[9] Ansari I A, Pant M, Ahn C W. ABC optimized secured image watermarking scheme to find out the rightful ownership [J]. International Journal for Light and Electron Optics, 2016, 127 (14): 5714-5723.

[10] Zhang Yudong, Wu Yuenan, Wei Geng. Application of improved ant colony algorithm based on forward/backward feedback in hardware/software partition [J]. Journal of Electronic Measurement and Instrument, 2009, 23 (8): 32-38.

[11] Bermejo E, Cordón O, Damas S, et al. A comparative study on the application of advanced bacterial foraging models to image registration [J]. Information Sciences, 2015, 295: 160-181.

[12] Cordón O, Damas S, Santamaría J. Feature-based image registration by means of the CHC evolutionary algorithm [J]. Image and Vision Computing, 2006, 24 (5): 525-533.

[13] Panda R, Agrawal S, Sahoo M, et al. A novel evolutionary rigid body docking algorithm for medical image registration [J]. Swarm and Evolutionary Computation, 2018, 42: 138-159.

[14] Karaboga D. An idea based on honey bee swarm for numerical optimization, Technical Report-TR06 [R]. Kayseri, Turkey: Erciyes University, 2005.

[15] Cao Yongcun, Pan Xiuqin, Lu Yong. An improved global best guided artificial bee colony algorithm for continuous optimization problems [J]. Cluster

Computer, 2018 (1): 1-9.

[16] Zhu Guopu, Kwong S. Gbest-guided artificial bee colony algorithm for numerical function optimization [J]. Applied Mathematics and Computation, 2010, 217 (7): 3166-3173.

[17] Singh A. An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem [J]. Applied Soft Computing, 2008, 9 (2): 625-631.

[18] Storn R, Price K. Differential Evolution: a simple and efficient heuristic for global optimization over continuous spaces [J]. Journal of Global Optimization, 1997, 11 (4): 341-359.

[19] Gao Weifeng, Liu Sanyang, Huang Lingling. Enhancing artificial bee colony algorithm using more information-based search equations [J]. Information Sciences, 2014, 270 (1): 112-133.

[20] Ge Baozhen, Peng Bo, Tian Qingguo. Registration of three-dimensional point-cloud data based on curvature map [J]. Journal of Tianjin University: Science and Technology, 2013, 46 (2): 174-180.

[21] Rusu R B, Blodow N, Beetz M. Fast point feature histograms (FPFH) for 3D registration [C]//Proc of IEEE International Conference on Robotics and Automation. 2009: 3212-3217.

[22] Chen H, Bhanu B. 3D free-form object recognition in range images using local surface patches [J]. Pattern Recognition Letters, 2007, 28 (10): 1252-1262.

[23] Ding Zhongming, Kawamura T, Sakamoto N, et al. Particle-based multiple irregular volume rendering on CUDA [J]. Simulation Modelling Practice and Theory, 2010, 18 (8): 1172-1183.

[24] Ma Zhongling, Zhou Mingquan, Geng Guohua. Automatic registration algorithm for scattered point clouds based on curvature feature [J]. Application Research of Computers, 2015, 32 (6): 1878-1880.

[25] Ge Baozhen, Zhou Tianyu, Chen Lei, et al. Point clouds registration algorithm based on improved ISS feature points and artificial bee colony algorithm [J]. Journal of Tianjin University: Science and Technology, 2016, 49 (12): 1296-1302.

[26] Falco I D, Cioppa A D, Maisto D, et al. Using an adaptive invasion-based model for fast range image registration [C]//Proc of GECCO' 14 Conference on Genetic and Evolutionary Computation. New York: ACM Press, 2014.

Note: Figure translations are in progress. See original paper for figures.