A method of partially overlapping point clouds registration based on differential evolution algorithm

Xuetao Zhang¹,², Ben Yang¹,², Yunhao Li², Changle Zuo¹,², Xuewei Wang³, Wanxu Zhang¹,²*¹

¹ National Engineering Laboratory for Visual Information Processing and Applications, Xi’an, Shaanxi 710049, China, ² School of Electronic and Information Engineering, Xi’an Jiaotong University, Xi’an, Shaanxi 710049, China, ³ College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang 310027, China, ⁴ School of Information Science and Technology, Northwest University, Xi’an 710127, China

* wxzhang@nwu.edu.cn

Abstract

3D point cloud registration is a key technology in 3D point cloud processing, such as 3D reconstruction, object detection. Trimmed Iterative Closest Point algorithm is a prevalent method for registration of two partially overlapping clouds. However, it relies heavily on the initial value and is liable to be trapped in to local optimum. In this paper, we adapt the Differential Evolution algorithm to obtain global optimal solution. By design appropriate evolutionary operations, the algorithm can make the populations distributed more widely, and keep the individuals from concentrating to a local optimum. In the experiment, the proposed algorithm is compared with existing methods which are based on global optimization algorithm such as Genetic Algorithm and particle filters. And the results have demonstrated that the proposed algorithm is more robust and can converge to a good result in fewer generations.

Introduction

With the development of the depth-sensing technology, such as Kinect and 3D LiDAR, the acquisition of 3D point cloud becomes more convenient. Furthermore, the 3D point cloud processing attracts more and more attention from research and industry fields [1–6]. Point cloud registration is one of the key technologies in 3D point cloud processing. The goal of registration is to obtain the spatial transformation between two different point clouds, which is quite useful in 3D reconstruction and object detection.

Numbers of registration methods are based on the iterative closest point (ICP) [7, 8] algorithm as its high efficiency and accuracy. However, the classical ICP algorithm has some disadvantages which limit its usage in real world applications. Firstly, the speed and accuracy of convergence rely on the initial value heavily. Secondly, when the two point sets are partially overlapped, which means for some points in one set we cannot find corresponding points in the other set, the registration result becomes worse. And the algorithm may fail, if the overlapped portion is small.
Trimmed Iterative Closest Point (TrICP) \cite{9, 10} algorithm has been proposed to handle the partially overlapping point sets registration problem. It estimates the rotation and translation between two sets as well as the overlapping rate. Therefore, in each iteration it finds the correspondences only between overlapped points, and then updates the transformation using the corresponding point pairs. Accurately, when the overlapping rate nears one, the TrICP algorithm approaches the ICP algorithm. However, TrICP still need an appropriate initial value for iterative optimization procedure. Otherwise the algorithm would possibly converge to a local optimum. Moreover, TrICP is time consuming, because it needs to visit all possible overlap rate and search the one which get the best registration result.

The Differential Evolution (DE) \cite{11} algorithm was proposed in 1997 by Rainer Storn and Kenneth Price on the basis of evolutionary ideas such as genetic algorithms. Substantially, it is a multi-objective optimization algorithm for obtaining the optimal solution in a multidimensional space. The source of differential evolution is the early proposed genetic algorithm (Genetic Algorithm, GA) \cite{12}, which simulates crossover, mutation, and reproduction in genetics to design genetic operators. Comparing with GA, the DE algorithm firstly generates the intermediate population based on the difference among the current population, and then obtain the new generation by competing between the intermediate population and the parent population. The DE algorithm has many advantages, such as high converging speed, fewer parameters.

To overcome disadvantages of traditional TrICP algorithm, the current paper proposed a global optimization method for TrICP based on DE algorithm. Through the appropriate design of the evolutionary operation, the population can be distributed widely, and increasing the probability getting the global optimal solution. It is efficient and accurate, even when the overlapping rate is low.

**Related works**

The original ICP algorithm relies heavily on the initial estimation of the transformation between two point clouds. And it also requires the two clouds should be close to each other. Moreover, the algorithm may fail when there the noise is significant or the number of outlier is large \cite{13}. In the literature, numerous efforts have been made to tackle these problems.

Tsin and Kanade \cite{14} proposed a Kernel Correlation algorithm which represented the point clouds as probability distribution, and the distance between the clouds was measured by the similarity of the distribution. However, the computation cost is very high as each point in one cloud should be compared with all points in the other.

The filter-based approaches have also been proposed to solve the local optimal solution problem. Unscented Particle Filter (UPF) is able to register small point clouds \cite{15}. This method requires great number of particles to obtain accurate registration result. Thus, when the amount of points is large, the computational cost will be high. Although the Unscented Kalman Filter (UKF) could overcome this shortcoming, but it is constrained by unimodal distribution assumption of the state vector. Therefore, the algorithm would fail when the distribution is complex. Zhu, et al. proposed to use particle filter for rigid registration \cite{16}. However, this algorithm is time consuming because it needs lots of particles to obtain accurate result.

TrICP introduces an overlapping rate into a least square function in order to improve the robustness to partial overlapping. However, the computation is slow. Phillips et al. \cite{17} presented Fractional ICP (FICP) algorithm to speed up the process of registration. The algorithm computes the best correspondence and overlapping rate simultaneously. But FICP relies heavily on hyper-parameters which may lead registration to failure. Gold \cite{18} proposed the Robust Point Matching (RPM) method, which applies the annealing algorithm to reduce the
exhaustive search time. However, when noise is significant or some structural missing exist, RPM would fail.

**Differential evolution algorithm**

Similar to other evolutionary algorithms, the DE algorithm has several operations, including population initialization, mutation, crossover and selection. The whole process is shown in Fig 1.

**Population initialization**

The initialization operation aims to generate the first generation for the iterative process. If there is no prior knowledge about the optimum, the population can be initialized randomly. In this case, the individuals are widely distributed. Usually, the uniform distribution is used.

**Mutation operation**

In many evolutionary algorithms, the mutation operation generates new vectors via adding disturbance to the individuals in the current generation. In DE, the disturbance is related with the difference between two other randomly selected individuals. The generated vectors are called mutant vectors. This operation enhances the search capability, and it is the key characteristic of the DE algorithm.

**Crossover operation**

The crossover operation is used to increase diversity of the population. It reassigns new values to some components of the mutant vector. The components are selected randomly, and the probability is controlled by a parameter, i.e. crossover rate. The result new vector is named as crossover vector.

**Selection operation**

The selection operation generates population of next generation from the crossover vectors and the original individuals in the current generation. Usually, the DE algorithm utilizes greedy algorithm to select new individuals according to the fitness.

As shown in Fig 1, the mutation, crossover and selection are performed iteratively. When the termination criteria is meet, the process stops. And the algorithm outputs the best individual of the last generation.
Point cloud registration based on differential evolution algorithm

The original TrICP algorithm estimated the rigid transformation and the overlapping rate using an iterative procedure. Such procedure could fall into a local optima easily. Therefore, it needs a good initial value. However, selection of the initial value is a difficult problem. In this paper, we turn to use a global optimization method. Due to efficiency and accuracy, the DE algorithm is increasingly popular in optimization field. Thus, it is adapted for registration partially overlapping point sets.

Formulation

Let $M = \{ m_i \}_{i=1}^{N_M} (m_i \in \mathbb{R}^n, i = 1, \cdots, N_M)$ be the reference point set, and $D = \{ d_j \}_{j=1}^{N_D} (d_j \in \mathbb{R}^n, j = 1, \cdots, N_D)$ be the registration point set. The registration problem is to find the optimal rotation matrix $R$ and the transform vector $t$ to achieve the best alignment with two point clouds. Denote $m_c(j)$ as the corresponding point of $d_j$ in $M$. If the rotation $R$ and the translation $t$ between the two point sets can be obtained, the error between the two matched points can be computed as,

$$e(d_j) = \| Rd_j + t - m_c(j) \|_2$$

Then, the registration process is to minimize total errors.

In the overlapping case, only a part of $D$ can be aligned with a part of $M$. Thus the overlapping rate $r$ need to be estimated as well. Then the overlapped part of $D$ can be defined as

$$D_r = \{ d_j \ | \ \max e(d_j) < \min e(d_i), d_i \in (D \setminus D_r) \}$$

Then the TrICP algorithm find the optimal $R$, $t$ and $r$ by

$$\min_{R,t,r} \frac{1}{N_{D_r} \pi^{\lambda} \lambda} \sum_{d_i \in D_r} \| Rd_i + t - m_c(i) \|_2$$

where $N_r$ is the number of points in $D_r$, and $\lambda$ is a preset parameter (in this paper, $\lambda = 2$).

In the iterative process of DE, the objective function in Eq 2 can be seen as the fitness function which measures the goodness of the individuals. Then, the whole process of registration with DE algorithm is shown in Fig 2.

Initialization

Let $x_i(g)$ represent the $i^{th}$ individual in the population of the $g^{th}$ generation. In this paper, each individual is a 7-dimensional vector, containing a 3D translation vector $t$, a 3D rotation vector $R$ and a trimming parameter. The rotation matrix $R$ can be converted from the rotation vector. And the first generation can be computed using Eq (3).

$$x_{i,j}(0) = x_{U,j} + rand(0, 1) \cdot (x_{L,j} - x_{U,j})$$

where $x_{i,j}(0)$ is the $j^{th}$ gene in the $i^{th}$ individual, and $x_{U,j}$ and $x_{L,j}$ are the upper and lower bound of each gene. $rand(0, 1)$ generates a random number which is uniformly distributed between 0 and 1.

Calculating transformation and fitness

The operations in following steps, such as mutation and selection, are based on the fitness of individuals. Moreover, we also perform optimization like what is done in TrICP, which is
utilized to improve the fitness. Thus, the following operations will be performed on a population which has higher quality.

In this step, we first transform the cloud $M$ to $D$ by initial transformation, and find the correspondence between the points in two sets. Secondly, we discard the point pairs if the distance between corresponding points is out of scope defined by initialized upper and lower threshold. Thirdly, the errors of the left points is calculated and the overlapping rate can be obtained.

Fig 2. Flow chart of the proposed algorithm.

https://doi.org/10.1371/journal.pone.0209227.g002
Finally, we find the index with smallest overlap rate, and update the minimum overlap rate and previous overlap rate. We then replace the two point clouds by the trimmed ones. For new point clouds, the transformation matrix can be calculated. Then the rotation matrix $R'$ and the translation vector $t'$ can be obtained by singular value decomposition (SVD).

**Mutation**

In the mutation operation of original DE algorithm, the involved individuals are selected randomly. And the operation is performed as

$$v_i(g) = x_{i1}(g) + F \cdot (x_{i2}(g) - x_{i3}(g))$$  \hspace{1cm} (4)

where $r1 \neq r2 \neq r3 \neq i$ are indices of individuals which are selected randomly, $F$ is the mutation scale factor, $v_i(g)$ is the mutation vector in $g^{th}$ generation.

Obviously, the operation in the form of Eq 4 does not consider the evolutionary state of the current generation. It has same effect on all the individuals in spite of the fitness. Therefore, the best individual in this generation is introduced. Then the operation has a form as

$$v_i(g) = x_{i1}(g) + F \cdot (x_{i2}(g) - x_{i3}(g)) + F \cdot (x_{best}(g) - x_{i1}(g))$$  \hspace{1cm} (5)

However, this operation has similar effect on all the individuals in spite of the fitness. Actually, the individuals which have good fitness, is more likely to trapped into a local optimum. We should add more disturbance. For the individuals, which has bad fitness, should be changed toward good ones. That is the mutation scale vectors in Eq 5 must adjusted according to the fitness of the individual. In this paper, we introduce a new operator shown in Eq 6

$$v_i(g) = x_{i1}(g) + F \cdot \gamma_1 \cdot (x_{i2}(g) - x_{i3}(g)) + F \cdot \gamma_2 \cdot (x_{best}(g) - x_{i1}(g))$$  \hspace{1cm} (6)

Here, $F$ is the current scale factor. $\gamma_1$ and $\gamma_2$ are the weights, which adjusted according to the fitness of the individuals. They can be computed as

$$\gamma_1 = \frac{f - f_{best}}{f_{worst} - f_{best}}$$  \hspace{1cm} (7)

$$\gamma_2 = 1 - \gamma_1$$  \hspace{1cm} (8)

where $f$ is the fitness of the current selected individual, $f_{best}$ and $f_{worst}$ are the best fitness and worst fitness in the current generation respectively. Since the problem is minimization, $f_{best}$ is the smallest value. Therefore, when the individual is bad, i.e. its fitness value is large, the $\lambda_1$ will be greater than $\lambda_2$. The mutation is more likely toward to the good individuals. Otherwise, when the individual is good, the operation is like the original one which has greater randomness.

Moreover, to keep the population from early-maturing, we change the value adaptively as following,

$$F = F_0 \cdot 2^{\lambda}$$  \hspace{1cm} (9)

and,

$$\lambda = e^{\frac{f_1 - f_{worst}}{f_{worst} - f_{best}}}$$  \hspace{1cm} (10)

$F_0$ is the basic mutation scale factor, $G$ is the number of generations, and $g$ is the index of current generation.
Crossover

In this paper, the uniform crossover is used. For the individuals in the $g^{th}$ generation, the crossover operation is performed as follows.

$$u_{ij}(g + 1) = \begin{cases} v_{ij}(g + 1) & \text{if } \text{rand}(0, 1) < CR \text{ or } j = j_{\text{rand}}, \\ x_{ij}(g) & \text{otherwise} \end{cases}$$

(11)

where $CR \in (0, 1)$ is the crossover rate. It controls the proportion of the components where the crossover occurs. $j_{\text{rand}}$ is the index of randomly selected component. This insures that crossover occurs on at least one of the components. The generated vector is called crossover vector.

Selection

The DE algorithm applies greedy strategy to select individuals which have good fitness for the next generation.

$$x_i(g + 1) = \begin{cases} u_i(g + 1) & \text{if } f(u_i(g + 1)) \leq f(x_i(g)), \\ x_i(g) & \text{otherwise} \end{cases}$$

(12)

where $f(\cdot)$ is the fitness of the individual. The fitness can be obtained by TrICP (Eq 2). This operation generates population for the $g + 1$ generation.

Experiment

In this section, the proposed algorithm was verified on a public and popular 3D range data set, i.e. the Stanford 3D Scanning Repositor [19]. And the proposed algorithm was compared with three related methods: the registration based Particle Filters (PF) [16], the registration based on GA [20], and the registration based on original DE. And we denote these approach as PF, GA, and DE respectively.

Convergence speed

For those global optimization based algorithms, the convergence speed is an important indicator of the algorithm’s performance. Therefore, we selected two shapes from S1 Dataset for the experiment. One was the bunny shape and the other was the dragon shape. To generate data set, a part of the points in the shape were discarded. And after adding white noises to the position of points, the left points formed the data set. Then another part of the shape was cut down. And a randomly generated transformation $(R_r, t_r)$ was applied to the left part. The result point set was the reference set. Then we used the four methods to register the data set and model set.

To compare the performance on convergence speed, we fixed up the overlapping rate $(r = 0.75)$ and varied the population size in different algorithms. We generated 10 pairs sets using the process mentioned in the last paragraph. Then, each algorithm was tested on all pairs. The average number of iterations of each algorithm is shown in Figs 3 and 4. From these results, one can notice that the proposed method need fewest generations than other three methods. This is because in the proposed method, the individuals were evolved according to their own state. So the search ability of the population is better than others. Also, comparing with the original DE, the proposed method can increase the efficiency greatly.
Fig 3. Comparison of number of generations on “Bunny”. (The overlapping rate is 75%.)
https://doi.org/10.1371/journal.pone.0209227.g003

Fig 4. Comparison of number of generations on “Dragon”. (The overlapping rate is 75%.)
https://doi.org/10.1371/journal.pone.0209227.g004
Table 1. Performance comparison under varying overlapping rate. Suc is the percentage of successful trials, and NG is the average number of generations.

| Method | \( r = 1 \) | \( r = 0.85 \) | \( r = 0.7 \) | \( r = 0.6 \) | \( r = 0.5 \) |
|--------|----------|----------|----------|----------|----------|
|        | Suc  | NG     | Suc  | NG     | Suc  | NG     | Suc  | NG     | Suc  | NG     |
| PF     | 100  | 10.3   | 97   | 12.73  | 93   | 14.53  | 80   | 19.47  | 50   | 23.63  |
| GA     | 100  | 5.13   | 100  | 6.17   | 93   | 10.37  | 80   | 15.93  | 50   | 18.73  |
| DE     | 100  | 13.5   | 97   | 16.53  | 67   | 19.7   | 20   | 24.53  | 10   | 28.9   |
| Ours   | 100  | 4.17   | 100  | 4.5    | 97   | 8.1    | 87   | 10.77  | 63   | 14.23  |

https://doi.org/10.1371/journal.pone.0209227.t001

Fig 5. Registration results of the proposed algorithm for shape “Bunny”. The first column is the initial position of two point sets. And the second column is the results after registration using the proposed algorithm.

https://doi.org/10.1371/journal.pone.0209227.g005
Overlapping rate

Then we fixed up the population size for all the methods, and varied the overlapping rate between the data set and the reference set. In this experiment, the population size was set to 100 and the bunny shape from S1 Dataset is used. All the methods were tested with 5 different

Fig 6. Registration results of the proposed algorithm for shape “Dragon”. The first column is the initial position of two point sets. And the second column is the results after registration using the proposed algorithm.

https://doi.org/10.1371/journal.pone.0209227.g006
overlapping rates. And the tests were performed for 30 trials. In this case, the successful rate and the number of iterations are concerned. The result is shown in Table 1. As shown by the results, the propose approach is more robust than other three methods.

In summary, the proposed approach can find optimal result in fewer generations and is more robust for the registration between partially overlapped point cloud. Figs 5 and 6 gives some examples of registration results.

Conclusion
In this paper, we present a new variant of TrICP algorithm based on the differential evolution algorithm. In the proposed algorithm, the DE algorithm was used to obtain a global optimal solution. Via designing appropriate evolutionary operations, the population can be distributed widely. Therefore, it is difficult for the proposed algorithm to be trapped into local optima. The result of the experiment showed that the proposed algorithm can obtain accurate transformation between two point clouds, even when the overlapping rate is low. Meanwhile, the computational cost is relative low comparing with common algorithms such as genetic algorithm and particle filtering.

Supporting information
S1 Dataset. The minimal data set.
(ZIP)

Acknowledgments
This work was supported by the National Science and Technology Major Project of China grant number 2018ZX01008103.

Author Contributions
Conceptualization: Xuetao Zhang, Wanxu Zhang.
Funding acquisition: Xuetao Zhang.
Methodology: Xuetao Zhang, Changle Zuo.
Software: Ben Yang.
Supervision: Wanxu Zhang.
Validation: Xuetao Zhang, Ben Yang, Yunhao Li, Changle Zuo, Xuewei Wang.
Visualization: Ben Yang, Yunhao Li.
Writing – original draft: Xuetao Zhang, Yunhao Li.
Writing – review & editing: Xuetao Zhang, Xuewei Wang, Wanxu Zhang.

References
1. Emambakhs h M, Evans A. Nasal patches and curves for expression-robust 3D face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017 May; 39(5):995–1007. https://doi.org/10.1109/TPAMI.2016.2565473 PMID: 28113700
2. Schmid B, Adhami R. Building descriptors from local feature neighborhoods for applications in semi-dense 3D reconstruction. IEEE Transactions on Image Processing. 2018 Nov; 27(1):5491–5500. https://doi.org/10.1109/TIP.2018.2859594
3. Jiang J, Xue P, Chen S, Liu Z, Zhang X, Zheng N. Line feature based extrinsic calibration of LiDAR and camera. IEEE International Conference on Vehicular Electronics and Safety (ICVES). 2018.
4. Yang Y, Liu Q, He X, Liu Z. Cross-view multi-lateral filter for compressed multi-view depth video. IEEE Transactions on Image Processing. 2019 Jan; 28(1):302–315. https://doi.org/10.1109/TIP.2018.2867740 PMID: 30176589

5. Yang Y, Li B, Li P, Liu Q. A two-stage clustering based 3D visual saliency model for dynamic scenarios. IEEE Transactions on Multimedia. 2018.

6. Wang X, Zhang P, Zhang Y, Ma L, Kwong S, Jiang J. Deep intensity guidance based compression artifacts reduction for depth map. Journal of Visual Communication and Image Representation, 2018 Nov; 57:234–242. https://doi.org/10.1016/j.jvcir.2018.11.008

7. Besl P, McKay N. A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1992 Feb; 14(2):239–256. https://doi.org/10.1109/34.121791

8. Zhang Z. Iterative point matching for registration of free-form curves and surfaces. International Journal of Computer Vision. 1994 Oct; 13(2):119–152. https://doi.org/10.1007/BF01427149

9. Chetverikov D, Stepanov D, Krsek P. Robust Euclidean alignment of 3D point sets: the trimmed iterative closest point algorithm. Image and Vision Computing. 2005 Mar; 23(3):299–309. https://doi.org/10.1016/j.imavis.2004.05.007

10. Chetverikov D, Svirko D, Stepanov D, Krsek P. The trimmed iterative closest point algorithm. International Conference on Pattern Recognition. 2002:545-548.

11. Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization. 1997 Dec; 11:341–359. https://doi.org/10.1023/A:100820821328

12. Melanie M. An introduction to genetic algorithms. The MIT Press, Cambridge, 1999.

13. Zinsse T, Schmidt J, Niemann H. A refined ICP algorithm for robust 3-D correspondence estimation. IEEE International Conference on Image Processing, 2003.

14. Tsin Y, Kanade T. A correlation-based approach to robust point set registration. European Conference on Computer Vision (ECCV), 2004;558-569.

15. Ma B, Ellis RE. Surface-based registration with a particle filter. Medical Image Computing and Computer-Assisted Intervention, 2004:566–573.

16. Zhu J, Du SY, Li ZY, Ma L, Yuan ZJ. Rigid partially registration algorithm for point set with partial filter. SCIENTIA SINICA Informationis. 2014 Jul; 44(7):886–899.

17. Phillips J, Liu R, Tomasi C. Outlier robust ICP for minimizing fractional RMSD. Image and Vision Computing. 2005 Mar; 23(3):299–309.

18. Gold S, Rangarajan A, Lu CP, Pappu S, Mjolsness E. New algorithms for 2D and 3D point matching: pose estimation and correspondence. Pattern Recognition. 1998 Aug; 31(8):1019–1031. https://doi.org/10.1016/S0031-3203(98)00010-1

19. The Stanford 3D Scanning Repository: http://graphics.stanford.edu/data/3Dscanrep

20. Zhu J, Meng D, Li Z, Du S, Yuan Z. Robust registration of partially overlapping point sets via genetic algorithm with growth operator. IET Image Processing. 2014 Oct; 8(10):582–590. https://doi.org/10.1049/iet-ipr.2013.0545