

Article

Coarse–Fine Registration of Point Cloud Based on New Improved Whale Optimization Algorithm and Iterative Closest Point Algorithm

Yunsheng Tian, Xiaofeng Yue * and Juan Zhu

School of Mechanical and Electrical Engineering, Changchun University of Technology, Changchun 130012, China; 1202001008@stu.ccut.edu.cn (Y.T.); zhujuan@ccut.edu.cn (J.Z.)

* Correspondence: yuexf@ccut.edu.cn

Abstract: In this paper, a point cloud coarse–fine registration method based on a new improved version of the whale optimization algorithm (NIWOA) and iterative closest point (ICP) algorithm is proposed; we use three strategies to improve the whale optimization algorithm (WOA). Firstly, circle chaotic mapping is used to initialize the whale population to increase the diversity of the population and make the search space more comprehensively explored. In addition, a Newton inertia weight is proposed to flexibly adjust the proportion of global exploration and local optimization in order to achieve the balance between the exploitation performance and exploration ability of the algorithm. At the same time, we introduce the nonlinear convergence factor that can adjust the size adaptively so that the algorithm can find the global optimal solution faster and more accurately, allowing it to avoid falling into the local optimal solution to a certain extent. The NIWOA algorithm is used to optimize the objective function of point cloud coarse registration to obtain the optimal coordinate transformation, and the rotation and translation operation is carried out on the registered point cloud. The obtained position is used as the initial pose of the ICP fine registration, and the final registration is achieved through ICP iteration. We conduct coarse registration experiments on multiple model point clouds and scene point clouds using the Stanford 3D Scanning Repository dataset and Princeton 3Dmatch dataset, respectively. The experimental results confirm that the NIWOA algorithm can not only find the initial position that is closer to the target point cloud, but also provide reliable initial values for the ICP algorithm. Meanwhile, the NIWOA algorithm combined with ICP experiment results show that the method has a higher registration accuracy and operation efficiency.

Keywords: point cloud registration; whale optimization algorithm; iterative closest point algorithm



Citation: Tian, Y.; Yue, X.; Zhu, J. Coarse–Fine Registration of Point Cloud Based on New Improved Whale Optimization Algorithm and Iterative Closest Point Algorithm. *Symmetry* **2023**, *15*, 2128. <https://doi.org/10.3390/sym15122128>

Academic Editor: Silvio Pardi

Received: 30 October 2023

Revised: 21 November 2023

Accepted: 24 November 2023

Published: 29 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Reverse engineering technology is widely used in modern industrial design, material production processes, product modeling and analysis, and other advanced manufacturing fields [1–3]. Three-dimensional (3D) reconstruction is an important research topic in reverse engineering technology [4,5]. In order to improve the accuracy of reconstruction and improve the operation efficiency of algorithms, many researchers have conducted in-depth exploration and research on 3D laser scanning systems [6], and point cloud registration technology, as an important part of 3D reconstruction and laser scanning, has become a hot topic for many scholars [7,8].

Due to the influence of the point cloud acquisition environment and target self-occlusion, the establishment of a complete 3D model of an object needs to measure multiple sets of point cloud data from multiple angles and accurately splice these point cloud data. The splicing process can usually be divided into registration problems of multiple two point cloud data, so it is of great significance to conduct in-depth research on registration problems of two point cloud data. The purpose of point cloud registration is to unify the overlapping point cloud data into the same coordinate system through coordinate

transformation so as to obtain the complete 3D point cloud model of the target object. Point cloud registration is one of the key steps in 3D laser scanning, and the quality of registration directly determines the effect of 3D model reconstruction. At present, the most widely used point cloud registration method is the classical ICP algorithm [9], which calculates the optimal coordinate transformation between the two point cloud data through multiple iterations. However, the ICP algorithm needs to calculate all corresponding points between two point cloud data in each iteration, which adds a lot of computational burden to the algorithm, resulting in slow convergence and making it easy to fall into the local optimal solution when solving the optimal objective function value. To address these issues, researchers usually perform a coarse registration of the point cloud data before executing the ICP algorithm so that the two sets of point clouds have closer initial positions.

In recent years, with the popularization and wide application of artificial intelligence technology in various disciplines, swarm intelligence optimization algorithms have been receiving attention more and more by scholars and have been applied to the field of 3D imaging technology. Swarm intelligence is a class of algorithms inspired by the combination of natural intelligence and human intelligence, and the proposed algorithms only involve some basic mathematical calculations, which have the advantages of being easy to implement and being adaptable to various types of complex optimization problems. Representative algorithms in the field of swarm intelligence research include the genetic algorithm (GA) [10], particle swarm optimization (PSO) [11], bat algorithm (BA) [12], ant lion optimizer (ALO) [13], cuckoo search (CS) [14], invasive weed optimization (IWO) [15], etc. These classical algorithms and various forms of improved algorithms can achieve excellent results in different engineering optimization problems.

- Optimization problems and related research in point cloud registration

The process of 3D point cloud registration is employed to solve the spatial coordinate transformation so that the distance between the corresponding points of the source point cloud and the target point cloud is zero. However, the measured point cloud in the real scene is affected by noise, error, and other factors, and the actual registration result cannot reach the ideal value after the coordinate transformation. Therefore, the essence of point cloud registration can be transformed into solving the global optimization problem, that is, solving the rigid body transformation matrix that minimizes the Euclidean distance between all corresponding points of two sets of point clouds in 3D space. Because the swarm intelligence optimization algorithm has good optimization performance for solving optimization problems, and because this kind of algorithm has a good global search and local optimization ability for complex spatial optimization problems, it has full research value and broad application prospects for optimizing the objective function of point cloud registration to achieve fast and accurate global registration. In this study, Shi et al. [16] proposed a point cloud coarse registration method by combining the filtering and adaptive fireworks algorithm [17], which showed a good performance in error analysis and stability analysis. Zhan et al. [18] proposed a 3D point cloud registration method based on entropy and the PSO algorithm and proved through experiments that their method can effectively improve the registration accuracy. Feng et al. [19] used the grey wolf optimizer (GWO) [20] algorithm to solve various parameters in the rotation matrix, which has great potential to improve the calculation speed and registration accuracy compared with other traditional registration methods. Liu et al. [21] used GA to optimize the HSV color information of a point cloud and applied it to point cloud registration to reduce registration errors. Chen et al. [22] introduced a new search equation and enhanced artificial bee colony (ABC) [23] algorithm to alternately search for the optimal solution, which effectively shortened the calculation time of registration.

- Research on the whale optimization algorithm (WOA) and related improvements

The whale optimization algorithm is a novel optimization algorithm proposed in recent years by Mirjalili et al. [24] inspired by the hunting behavior of humpback whales. This algorithm simulates the “spiral bubble net”, the contraction enveloping mechanism,

and the spiral position updating mechanism of humpback whales for foraging and has the characteristics of a simple structure, few adjustment parameters, and easy implementation. However, it was later found that the algorithm still has problems such as falling into local optimal and slow convergence. Therefore, many scholars have proposed various forms of improved WOA algorithms to overcome these shortcomings. Chakraborty et al. [25] introduced a unique selection parameter to balance the global and local search process of the algorithm, improved the adjustment vector, and introduced inertia weights in the exploitation stage, which greatly improved the search performance of the original algorithm. Liu et al. [26] introduced differential evolution operators to adjust the method of whale location update in the exploration and exploitation stages and improved the global exploration and local exploitation capabilities of the algorithm. Luo et al. [27] proposed a hybrid WOA named MDE-WOA, which not only improved the diversity of the population, but also made the algorithm easily jump out of the local optimal by embedding an improved differential evolution operator. Li et al. [28] added a tent chaos graph to the original WOA algorithm and adopted a tournament selection strategy to improve the optimization accuracy of the algorithm during the algorithm execution. Anitha et al. [29] proposed modified WOA (MWOA), which controls the position of the whale by adjusting the cosine function and introduces a correction factor to adjust the position update of the search agent during the motion process, effectively balancing the exploration and exploitation capabilities of the algorithm. Lin et al. [30] proposed a niching hybrid heuristic WOA (NHWOA), which introduced niche technology in the initialization to improve population diversity and inhibit premature convergence. Meanwhile, it flexibly adjusted algorithm parameters and carried out design disturbances on all search agents to improve the search performance of the algorithm and avoid falling into local optimum. Saha et al. [31] proposed cosine adapted modified WOA (CamWOA) with cosine adaptive correction. They adjusted the control parameters and used correction factors to reduce the step size. Yang et al. [32] improved WOA by introducing four strategies: chaotic mapping, adaptive weight and dynamic convergence factor, Levy flight mechanism, and evolutionary population dynamics, which showed certain advantages in benchmark test functions and actual optimization problems. Chakraborty et al. [33] proposed hunger search-based WOA (HSWOA) in 2022, which combined the hunger games search concept with the whale hunting process, and adaptively designed hunger games search (HGS) weights according to the whale's hunger level to balance the overall search process of the algorithm. The above improved methods have different degrees of contribution to improve the search accuracy and convergence speed of the WOA, which provide valuable ideas for the further improvement of the algorithm's performance. However, the problems of not being able to stably converge to the global optimum for some high-dimensional multi-peak test functions, the lack of a certain degree of robustness, and the high computational complexity are still more obvious in different improved algorithms. Therefore, it is of considerable practical significance to further study the WOA algorithm in terms of its operational theory and calculation process.

- Coarse–fine registration of point cloud based on new improved whale optimization algorithm (NIWOA) and iterative closest point (ICP) algorithm

WOA and its improved algorithms have been applied in many engineering optimization problems and show good performance. In the field of 3D imaging and computer vision, WOA has been used as a basic tool to optimize the objective function. However, for the objective function optimization problem in 3D point cloud registration, only applying the original WOA to the search and optimization of rotation and translation parameters can no longer meet the requirements of accuracy and speed of registration operation. Therefore, some scholars have begun to improve WOA and use the improved WOA to optimize the registration process. For example, Li et al. [34] proposed an improved WOA based on nonlinear convergence factor and adaptive weight coefficient, which was combined with the RANSAC algorithm to realize initial registration, and used the obtained transformation matrix as the initial pose estimation for fine registration. In order to achieve higher registration accuracy and running speed, based on the original WOA algorithm, three strategies

were proposed to improve the algorithm, which were circle chaotic mapping, Newton inertia weight, and nonlinear convergence factor. The new improved WOA (NIWOA) is used to optimize the objective function of the coarse registration to obtain the global optimal coordinate transformation to realize the initial pose estimation of the two point clouds, which provides a reliable initial value for the fine registration based on the ICP algorithm. Finally, the accuracy of registration is improved and the convergence speed of the algorithm is accelerated.

NIWOA's contribution to point cloud registration and its comparison with classical methods are briefly outlined in Table 1. The specific proof is given in the experimental section.

Table 1. Contribution of NIWOA and its comparison with classical methods.

Registration Method	Method Description	Accuracy and Efficiency
ICP	The registration of the point cloud data is performed directly without initial pose estimation	low
WOA + ICP	WOA is used for coarse registration optimization followed by ICP fine registration	medium
NIWOA + ICP	NIWOA is used to optimize the coarse registration to obtain a more accurate initial pose of the point cloud, and then ICP fine registration is performed	high

2. Principle and Processing Operation of Point Cloud Registration

2.1. Principle of Point Cloud Registration

For two sets of point cloud data P and Q , $P = \{p_i | p_i \in R^3, i = 1, 2, \dots, m\}$ is the point cloud to be registered, $Q = \{q_i | q_i \in R^3, i = 1, 2, \dots, n\}$ is the target point cloud, and m and n are the number of points in the two sets of point cloud; the purpose of registration is to solve the space coordinate transformation matrix V between the point cloud P and Q so as to minimize the Euclidean distance between the two corresponding points. V can be expressed as Equation (1), where V contains three rotation angles α, β, γ around the x, y, z axes and three translation vectors v_x, v_y, v_z along the x, y, z axes.

$$V = t \cdot R_x \cdot R_y \cdot R_z \quad (1)$$

where

$$t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ v_x & v_y & v_z & 1 \end{bmatrix}, R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix},$$

$$R_y = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R_z = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 & 0 \\ -\sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

Theoretically, the corresponding points of point cloud P and Q should coincide completely after the coordinate transformation of the rotation matrix and translation vector; that is, the Euclidean distance between the corresponding points should be zero. However, due to the measurement error in the process of point cloud acquisition and the influence of noisy environment, there will be some inevitable errors in the registration results. The task of point cloud registration is to minimize the Euclidean distance between the point cloud to be registered and the corresponding point of the target point cloud through the optimal rigid body coordinate transformation matrix V , which is essentially a global optimization problem. Point cloud fine registration using the ICP algorithm usually requires that two point clouds have relatively close initial relative positions between them. Therefore, before fine registration, two sets of point clouds are usually coarse-registered to obtain a more effective initial pose. This paper proposes using the new improved WOA (NIWOA) algorithm to optimize the rotation and translation matrix in the coarse registration stage so as

to make the Euclidean distance of the corresponding points between the point cloud to be registered and the target point cloud closer.

2.2. Point Cloud Data Preprocessing

The point cloud data obtained by laser scanning equipment usually contains a large number of spatial points. The statistics and processing of all points in the point cloud will undoubtedly increase the computational burden of the registration process, which cannot ensure the final registration accuracy and affect the overall registration efficiency. In this paper, before the point cloud registration operation, the uniform sampling method is used to reduce the amount of data processing in the subsequent operation, and the registration speed is accelerated to some extent in the early stage.

In order to further improve the accuracy and efficiency of the algorithm, we perform a feature point extraction operation on the uniformly sampled point cloud. In this paper, the classical intrinsic shape signature (ISS) [35] feature point extraction algorithm is used to screen out the representative points. This algorithm has the advantages of a simple principle and easy implementation, especially suitable for point cloud data with uniform distribution. The main idea of the ISS algorithm is to calculate the eigenvalues of the covariance matrix of the neighborhood of each point and determine the key points through the linear relationship between the eigenvalues. The process can be expressed as follows:

- (a) Suppose there are n points in point cloud P , and the coordinate of any point is $p_i(x_i, y_i, z_i)$, $i = 0, 1, \dots, n - 1$;
- (b) The local coordinate system is established for each point p_i in the point cloud, and the search radius r of each point is set;
- (c) Query all points p_{ij} of each point p_i in the point cloud data within the radius r ; j is the number of neighborhood points, and calculate the weight as follows:

$$w_{ij} = \frac{1}{\left| \left\{ p_i : \|p_i - p_{ij}\|_2 < r \right\} \right|} \quad (2)$$

- (d) Calculate the covariance matrix for each point p_i :

$$Cov(p_i) = \frac{\sum_{j=1}^k \|p_i - p_{ij}\|_2 < r w_{ij} (p_i - p_{ij})(p_i - p_{ij})^T}{\sum_{j=1}^k \|p_i - p_{ij}\|_2 < r w_{ij}} \quad (3)$$

- (e) Compute the eigenvalues $\{\lambda_i^1, \lambda_i^2, \lambda_i^3\}$ of the covariance matrix for each point p_i and sort them in descending order;
- (f) Set two thresholds γ_{21} and γ_{32} as not greater than 1; the points satisfying the following equation will be marked as ISS feature points.

$$\begin{cases} \lambda_i^2 / \lambda_i^1 \leq \gamma_{21} \\ \lambda_i^3 / \lambda_i^2 \leq \gamma_{32} \end{cases} \quad (4)$$

2.3. Feature Description and Matching

The preprocessed point cloud data effectively retains the feature points that can represent its spatial geometric characteristics and filters out most of the useless points in the original point cloud. On the basis of feature extraction, we use the classical PFH [36] feature description method to represent the mathematical model of spatial features for points within the neighborhood of each feature point. The spatial feature description of

PFH first constructs a local coordinate system for the point pairs composed of feature points and their neighborhood points, and the construction method can be expressed as follows:

$$\begin{aligned} u &= h_s \\ v &= u \times \frac{(p_i - p'_i)}{\|p_i - p'_i\|^2} \\ w &= u \times v \end{aligned} \quad (5)$$

where h_s is the normal vector of feature point p_i , \times represents the outer product between vectors, and u , v , and w are all unit vectors. Based on this local coordinate system, PFH describes the spatial features of points as follows:

$$\begin{cases} \alpha = v \cdot h_t \\ d = \|p_i - p'_i\|_2 \\ \phi = u \cdot \frac{(p_i - p'_i)}{d} \\ \theta = \arctan(w \cdot h_t, u \cdot h_t) \end{cases} \quad (6)$$

where h_t is the normal vector of the neighborhood point that forms a local point pair with point p_i , \cdot represents the inner product between vectors, and α , ϕ , θ , d together constitute the feature description of PFH.

The similarity of the feature point description in the point cloud to be registered and the target point cloud is used as the matching basis to find the spatially symmetric corresponding points in the two groups of point clouds. In this paper, the Euclidean distance between feature vectors is used to measure the similarity of matching point pairs. Here, the distance between feature vectors is denoted by $|f_n(p_i) - f_n(q_i)| / f_n(p_i) + f_n(q_i)$. Then, a threshold τ is set, and the point pairs whose distance between feature vectors is less than or equal to τ are selected as matching point pairs. The selection conditions are as follows:

$$|f_n(p_i) - f_n(q_i)| / f_n(p_i) + f_n(q_i) \leq \tau \quad (7)$$

where n is the number of feature descriptors, f is the feature descriptions of PFH, p_i and q_i are the feature points in the point cloud to be registered and the target point cloud, respectively, $i = 1, 2, \dots, N$, and N is the number of points in the point cloud after feature extraction.

To further confirm the accuracy of the corresponding point pairs, we use the RANSAC method [37] to eliminate the mismatching point pairs. The algorithm randomly selects three pairs of matching points as samples, calculates a model parameter from the point cloud to be registered to the target point cloud, then checks the deviation between all remaining matching point pairs and the model and compares the deviation with the threshold set according to experience. When the deviation is less than the threshold, the matching point pair belongs to the sample point in the model, that is, the correct matching point pair. Otherwise, it is identified as a wrong matching point pair. The current number of correct point pairs is recorded, and the process is repeated continuously. In each repetition, the parameter with the largest number of correctly matched point pairs is retained as the best model parameter. The number of iterations N required by the algorithm to satisfy all corresponding point pairs for at least one randomly selected sample model is computed as:

$$N = \left\lceil \frac{\log(1 - k)}{\log(1 - (1 - e)^s)} \right\rceil \quad (8)$$

where k is the confidence that all randomly selected sample models are interior points (that is, correctly matched point pairs) at least once. s denotes the number of corresponding point pairs in the minimal sampling set. In 3D point cloud registration, three pairs of non-collinear matching points are needed to calculate the rigid transformation matrix, so s is set to 3. e is the outlier rate, which is the probability that a point pair does not match. After N iterations, the best model parameters are the estimated values of the desired

model parameters, which are used as the correct rigid body transformation matrix, and the point pairs calculated from the transformation matrix are used as the set of exact matching point pairs.

3. The Classic Whale Optimization Algorithm

The WOA algorithm consists of three phases: encircling prey, bubble net attack, and searching prey; the performance of the algorithm depends on the balance between the global exploration phase and the local exploitation phase. In WOA, these three strategies are used to achieve a dynamic balance between exploration and exploitation processes.

3.1. Mathematical Model of Encircling Prey

Humpback whales can accurately identify the location of their prey and surround them. As the position of the optimal solution in the search space is unknown, WOA assumes that the best alternative solution in the current population (the optimal whale position) is the position of the target prey. After determining the position of the prey, the rest of the search agents (individual whales) will move to the best search agent (the prey) in different ways. At this stage, other whale individuals in the population approach the optimal position and update their positions using Equations (9) and (10):

$$\vec{D} = \left| \vec{C} \cdot \vec{X}^*(t) - \vec{X}(t) \right| \quad (9)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D} \quad (10)$$

where \vec{D} represents the bounding step size and t is the current iteration number. \vec{A} and \vec{C} are coefficient vectors; \vec{X}^* is the position vector of the current prey; \vec{X} is the position vector of the remaining search agents; $||$ is the absolute value; and \cdot is a multiplication between elements. The vectors \vec{A} and \vec{C} can be calculated by the following equations:

$$\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a} \quad (11)$$

$$\vec{C} = 2 \cdot \vec{r} \quad (12)$$

where \vec{a} is the coefficient in the iterative process (linearly decreasing from 2 to 0 in the iterative process); \vec{r} is a random vector between 0 and 1. During the computation, the position of \vec{X} around the optimal solution can be adjusted by adjusting the values of vectors \vec{A} and \vec{C} .

3.2. Mathematical Model of Bubble Net Attack

3.2.1. Shrinking Encircling Mechanism

During the iteration, this behavior is achieved by decreasing the value of a from 2 to 0 in Equation (11), in which case \vec{A} will fluctuate within $[-a, a]$, and when \vec{A} is a random value within $[-1, 1]$, the whale can be located anywhere between the original position and the current optimal position.

3.2.2. Spiral Updating Position Mechanism

Firstly, the distance between the position of the search agent (individual whale) and the position of the prey is calculated, and then the position of the individual whale is updated by using Equation (13) by simulating the hunting method that the whale spits out the bubble net along the spiral line and approaches the prey.

$$\vec{X}(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) \quad (13)$$

where $\vec{D}' = \left| \vec{X}^*(t) - \vec{X}(t) \right|$ represents the distance between the whale and the prey, b is a constant controlling the shape of the logarithmic spiral, and l is a random number in $[-1, 1]$.

3.2.3. Mathematical Model of the Search Prey Phase

In the stage of searching prey, the individual whale does not choose the current best individual as the target for position update but randomly selects an individual from the current population for position update. The purpose is to increase the search range and find the optimal solution while maintaining the diversity of the population. The mathematical model at this stage is as follows.

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_{rand} - \vec{X} \right| \quad (14)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D} \quad (15)$$

where \vec{X}_{rand} is the position of a randomly selected individual whale from the current population, $||$ is the absolute value, and \cdot is a multiplication between elements.

In general, humpback whales swim around their prey in a shrank circle following a spiral trajectory. In order to simulate this behavior, suppose that there is a 50% probability to choose the shrinking bounding mechanism or the spiral mechanism to update the position of the whale in the optimization process, that is, p is a random number of $[0, 1]$ interval. When $p \geq 0.5$, the bubble net attack mechanism is entered, and the position is updated using Equation (13). When $p < 0.5$, the system enters the stage of searching prey or encircling prey according to the absolute value of the coefficient vector \vec{A} . If $|\vec{A}| < 1$, then it enters the stage of encircling the prey (the search agent moves in the direction of the prey), and the position is updated using Equation (10). When $|\vec{A}| \geq 1$, it enters the stage of searching prey (the search agent moves in the direction of the randomly selected search agent), and the position is updated using Equation (15). Overall, WOA updates its position in the following way:

$$\vec{X}(t+1) = \begin{cases} \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D} & |\vec{A}| < 1 \\ \vec{X}_{rand} - \vec{A} \cdot \vec{D} & |\vec{A}| \geq 1 \end{cases} & \text{if } p < 0.5 \\ \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \vec{X}^*(t) & \text{if } p \geq 0.5 \end{cases} \quad (16)$$

4. New Improved Whale Optimization Algorithm

4.1. Population Initialization Based on Circle Chaotic Map

Chaos is a special motion form of a nonlinear dynamic system, which is a pseudo-random phenomenon between deterministic and random phenomena. In the latest research of the swarm intelligence algorithm, chaos is widely used in the initialization process to improve the optimization performance of the algorithm; a chaotic sequence has the characteristics of randomness and ergodicity. The main idea is to map the variables into the value interval of the chaotic variable space, and the obtained solution is linearly transformed into the optimization variable space.

The convergence accuracy and speed of the WOA algorithm are largely affected by the diversity of the initial population distribution. The classical WOA algorithm establishes the initial population by a random method, and the algorithm does not have any prior knowledge of the solution space of the optimization problem, which is easy to make the distribution of individual whales in the population uneven, and the spatial position of the optimization is not extensive enough. To address such problems, in order to produce a

uniform population distribution during the initialization process of the algorithm and to improve the speed of the algorithm to search for the global optimal solution, this paper adds circle mapping in the process of population initialization, and the improved algorithm is able to conduct a comprehensive exploration of the search space, which increases the diversity of the initial population location of whales. To a certain extent, the optimization efficiency of the algorithm is increased. The circle mapping can be expressed as follows:

$$z_{t+1} = \text{mod}(z_t + 0.2 - \left(\frac{0.5}{2\pi}\right) \sin(2\pi z_t), 1) \quad (17)$$

where mod is the residual function. Based on the chaotic variables obtained from the circle mapping, we set the upper and lower boundaries of the search space, ub and lb ; the improved algorithm produces a high-quality initial whale population [32]:

$$X_i^j = lb + z_j \times (ub - lb) \quad (18)$$

where X_i^j is the j th dimension coordinate of the i th search agent and z_j is the j th dimension coordinate of z after internal random ordering.

4.2. Newton Inertia Weight

In order to achieve a better balance between the global search and local optimization ability of the algorithm, the inertia weight is widely used in the optimization process of intelligent algorithms. The commonly used inertia weight includes constant inertia weight, linear decreasing inertia weight, and nonlinear adaptive inertia weight. Setting the constant inertia weight as larger or smaller will affect the performance of the algorithm in different degrees, resulting in the decline of the accuracy of the algorithm. Although the linearly decreasing inertia weight can adjust the global and local search ability of the algorithm to a certain extent, its local search ability is weak in the early iteration, and it often misses the global optimum. In the later iteration, the global search ability becomes weak, and it is easy to fall into the local optimum value. The nonlinear adaptive inertia weight can adjust the search direction to the better area by comparing the objective function value, but it is not flexible enough to adjust the exploration and exploitation ability of the algorithm.

The classical whale optimization algorithm lacks adaptability in the process of updating the whale position. Therefore, this paper improves Equation (16) by designing an effective inertia weight. In view of the shortcomings of the traditional inertia weight, this section proposes a Newton inertia weight based on Newton interpolation formula according to the number of iterations of the algorithm, which can flexibly adjust the weight and better distribute the relationship between the exploration and exploitation capabilities of the original whale optimization algorithm. The weights corresponding to 0 times; intermediate iteration times T_{mid} and maximum iteration times T_{max} are set as ω_{init} , ω_{mid} , and ω_{end} , respectively. According to the three points $(0, \omega_{init})$, (T_{mid}, ω_{mid}) , and (T_{max}, ω_{end}) , the Newton interpolation formula is obtained as follows:

$$\omega(t) = \omega_{init} + Mt + Nt(t - T_{mid}) \quad (19)$$

where t is the current iteration number, $M = \frac{\omega_{mid} - \omega_{init}}{T_{mid}}$, $N = \frac{1}{T_{max}} \left(\frac{\omega_{end} - \omega_{mid}}{T_{max} - T_{mid}} - \frac{\omega_{mid} - \omega_{init}}{T_{mid}} \right)$.

In order to make the algorithm produce a relatively small weight when approaching the optimal point and accelerate the convergence speed of the algorithm, on this basis, we add the random disturbance part to the Newton interpolation formula and obtain the expression of the Newton inertia weight:

$$\omega(t) = \omega_{init} + \frac{\omega_{mid} - \omega_{init}}{T_{mid}} t + \frac{t}{T_{max}} \left(\frac{\omega_{end} - \omega_{mid}}{T_{max} - T_{mid}} - \frac{\omega_{mid} - \omega_{init}}{T_{mid}} \right) (t - T_{mid}) + \mu \cdot rand. \quad (20)$$

where μ is an adjustable parameter; in general, the inertia weight will decrease with the increase in the number of iterations; therefore, $\omega_{init} > \omega_{mid} > \omega_{end}$ is set in this paper.

Combined with the Newton inertia weight, the position update method of the whale is improved by the following equation:

$$\vec{X}(t+1) = \begin{cases} \begin{cases} \omega(t) \times \vec{X}^*(t) - \vec{A} \cdot \vec{D} & \left| \frac{\vec{A}}{A} \right| < 1 \\ \omega(t) \times \vec{X}_{rand} - \vec{A} \cdot \vec{D} & \left| \frac{\vec{A}}{A} \right| \geq 1 \end{cases} & \text{if } p < 0.5 \\ \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + \omega(t) \times \vec{X}^*(t) & \text{if } p \geq 0.5 \end{cases} \quad (21)$$

In the process of updating the whale position, by adjusting the intermediate iteration number T_{mid} and weight value ω_{mid} , Newton inertia weight can flexibly adjust the proportion of the exploration and exploitation ability of the algorithm. In the early stage of iteration, we set a larger weight to improve the global search ability of the algorithm; in the later stage of iteration, setting a smaller weight helps to improve the ability of local optimization. Under the action of Newton inertia weight, the whale will approach the prey in the later iteration and adjust it into an adaptive weight to make the population around the pilot whale so that the algorithm can carry out a more refined local search and greatly avoid the algorithm falling into local optimum.

4.3. Nonlinear Convergence Factor

In the classical WOA algorithm, the search performance of the algorithm greatly depends on the value of the coefficient vector \vec{A} , and \vec{A} changes linearly with the decrease in the convergence factor a ; moreover, the coefficient vector \vec{A} is controlled by the coefficient a in the iterative process. However, with the increase in iterations, a is a linearly decreasing function, which cannot show the actual search state and the actual iterative process of the algorithm. It has the same decreasing speed during the execution of the algorithm, which easily affects the diversity of the whale population and reduces the overall optimization ability of the algorithm and makes it easy for the algorithm to fall into the local optimum [24]. In order to further make the algorithm achieve a better balance between the global search in the early stage and the local optimization in the later stage, this paper proposes a nonlinear decreasing convergence factor, which can be expressed as:

$$a = 2 \times (1 - e^{\frac{1}{t}-1}) \quad (22)$$

In the early stage of the algorithm execution, the convergence factor is adaptively adjusted to a large value and rapidly decreases so that the algorithm can carry out optimal exploration in the global scope. With the increase in the number of iterations, the value of a decreases slowly in the later stage of the algorithm operation, which makes it so that the improved whale optimization algorithm can search the optimal solution more accurately and effectively improve the convergence accuracy of the algorithm.

5. Point Cloud Registration Based on NIWOA and ICP

5.1. Optimization Model for Coarse Registration of Point Clouds Based on NIWOA

In this section, we use the optimization objective function of point cloud coarse registration as the fitness function of NIWOA. The corresponding point set that overlaps with the target point cloud Q is found in the point cloud P to be registered. NIWOA is used to optimize the rotation and translation matrix, which aims to minimize the average Euclidean distance between the corresponding points of the two point clouds. The rotation and translation matrix contains six parameters (rotation angles α, β, γ and translations v_x, v_y, v_z). These six parameters are used as optimization variables for the NIWOA algorithm.

In this paper, we assume that the rotation matrix is \mathbf{R} and the translation vector is \mathbf{T} ; the fitness function can be expressed as follows:

$$d = \min \frac{1}{c} \sum_{i=1}^c \|q_i - (\mathbf{R} \cdot p_i + \mathbf{T})\|^2 \quad (23)$$

where c represents the number of matched point pairs and p_i and q_i represent the matched points in the point cloud P to be registered and the target point cloud Q , respectively.

5.2. Fine Registration of Point Clouds Based on ICP Algorithm

Through NIWOA, we can obtain the initial transformation parameters of the two point clouds, and some overlap has been achieved in these matching points. To achieve a more accurate registration effect, the most simple and effective method is to continue to iteratively calculate the two point clouds on the basis of the initial rotation and translation transformation to make more corresponding point pairs coincide. Therefore, this paper uses the classic ICP algorithm to calculate the final rigid body coordinate transformation. After calculating the transformation parameters each time, the algorithm transforms the coordinates of the point clouds obtained by the previous transformation so that the two point clouds obtain a closer pose; then, the algorithm continues to calculate the next transformation until the distance between the corresponding point pairs is small enough or the pose of the two iterations does not change significantly. The procedure of ICP can be viewed as a robust estimation problem, which aims to further minimize the mean square objective function as follows:

$$f(T) = \frac{1}{c} \sum_{i=1}^c \left(\|q_i^* - (r p_i^* + t)\|^2 \right) \quad (24)$$

where T contains the coordinate transformation of rotation and translation, r and t are the rotation matrix and translation vector during the ICP iteration, and p_i^* and q_i^* are the spatially symmetric corresponding points of the two point clouds after coarse registration transformation, respectively.

Fine registration using the ICP algorithm has the advantages of a simple implementation process and no need for segmenting point clouds. As NIWOA provides effective initial registration parameters for ICP, the overall registration operation largely avoids the problem of the algorithm falling into local optimum.

6. Registration Experiments and Result Analysis

6.1. Experimental Instructions and Setup

In this section, we apply the proposed NIWOA to the objective function optimization of point cloud registration. At the same time, we conduct registration experiments on several representative point cloud datasets, and then, through the comparison of multiple evaluation indicators, we prove the effectiveness and robustness of the NIWOA algorithm in point cloud registration. In the experiment, we compare the accuracy and running efficiency of the coarse registration using only WOA and the coarse registration using NIWOA; we also verify the performance of the coarse–fine registration combined with ICP. The datasets selected for the experiment include the classic point cloud models of Stanford University [38] (Bunny, Dragon, Armadillo, HappyBuddha) and the Princeton 3Dmatch dataset [39,40] (sun3d-hotel_umd-maryland_hotel3). The Stanford University dataset is the most common dataset for point cloud processing, and the point cloud models in it are also widely used as experimental objects for point cloud registration. The Princeton 3Dmatch dataset is often used for keypoint detection, feature descriptors, point cloud registration, and other point cloud tasks.

These datasets contain some model point clouds and indoor scene point clouds, which can verify the applicability of the registration algorithm for different application objects. The point clouds under different viewpoints are shown in Figures 1–5.



Figure 1. (a) bun000; (b) bun045.



Figure 2. (a) dragonBk1_0; (b) dragonBk4_0.



Figure 3. (a) ArmadilloOnHeadMultipleOffset_15; (b) ArmadilloOnHeadMultipleOffset_45.



Figure 4. (a) happyBkgd3_0; (b) happyBkgd4_0.

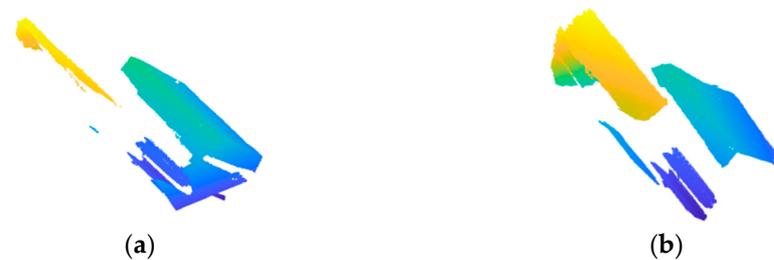


Figure 5. (a) sun3d-hotel_umd-maryland_hotel3 cloud_bin_35; (b) sun3d-hotel_umd-maryland_hotel3 cloud_bin_36.

The sizes of different point cloud models are shown in Table 2.

Table 2. Size of the point cloud model.

Point Cloud Model	Number of Points
bun000	39,937
bun045	39,974
dragonBk1_0	47,747
dragonBk4_0	44,832
ArmadilloOnHeadMultipleOffset_15	32,208
ArmadilloOnHeadMultipleOffset_45	24,764
happyBkgd3_0	83,327
happyBkgd4_0	79,802
sun3dhotel_umdmaryland_hotel3 cloud_bin_35	75,521
sun3dhotel_umdmaryland_hotel3 cloud_bin_36	151,564

The parameter setting in each stage of the registration process is particularly important, and different parameter values have a great impact on the accuracy and efficiency of the registration results. Therefore, after numerous tests and verifications, we set the most appropriate parameters for each link to conduct the experiment. Firstly, for the preprocessing stage, the uniform sampling scale of the point cloud directly determines the calculation amount of the subsequent point cloud processing. Too high a sampling rate will increase the computational burden of the algorithm, and too low a sampling rate will lead to inaccurate local feature expression and cause the algorithm to be prone to large registration error. On the basis of appropriate uniform sampling, we determined the parameters of the ISS algorithm through multiple sets of feature extraction experiments and finally selected the search radius $r = 0.04$ and 0.2 ; the values of γ_{21} and γ_{32} were 0.5 . This set of parameters can better retain the inherent shape features of the point cloud after preprocessing and has good robustness for point cloud data with noise and outliers.

In addition, in this experiment, the number of iterations of WOA, NIWOA, and ICP were all set to 100, the size of the whale population was set to 20, and the dimension of the solution was set to 6. The experimental operation was programmed using MATLAB R2022a on a 3.60 GHz Intel(R) Core (TM) i9-9900KF CPU configured with 32 GB of RAM.

6.2. Experiments on Coarse Registration of Point Clouds

In this section, we apply the proposed NIWOA algorithm to the coarse registration process of point clouds to verify its effectiveness in improving the accuracy of coarse registration. In the experiment, we chose the registration method based on the original WOA for comparison to verify the contribution of the proposed NIWOA algorithm to the improvement of coarse registration performance.

In this experiment, the registration accuracy was tested by the mean square error (MeanSE) between the corresponding points of the two point clouds after coarse registration so as to measure the error of coarse registration. The smaller the value of MeanSE, the higher the consistency of the two point clouds after coarse registration. The mathematical expression of MeanSE is shown in Equation (25). We calculated the error of coarse registration using two different methods respectively, and the comparison results are shown in Table 3 below:

$$\text{MeanSE} = \text{Mean}_{R,t} \|Rp_i^* + t - q_i^*\|^2 \quad (25)$$

It can be seen from the MeanSE values in the table that NIWOA has a smaller registration error. Compared with the coarse registration method of the original WOA algorithm, NIWOA can provide a more accurate coarse registration guarantee for point cloud data of different scales so that the relative positions of the two point clouds are closer. In addition, the coarse registration results of five groups of point cloud data achieved by using these two methods are shown in Figures 6–10. It can be clearly seen from the figures that the coarse registration based on the NIWOA algorithm can make the point cloud be registered and have the target obtain a closer distance, regardless of if it is used on the model point

cloud data of the Stanford dataset or on the large indoor scene point cloud data of Princeton. It can ensure the smooth execution of the subsequent fine registration more effectively.

Table 3. MeanSE of coarse registration based on WOA and NIWOA.

Point Cloud Model	WOA	NIWOA
bun000 and bun045	54.971×10^{-3}	4.3481×10^{-3}
dragonBk1_0 and dragonBk4_0	3.6750×10^{-3}	1.7964×10^{-3}
ArmadilloOnHeadMultipleOffset_15 and 45	2.9120×10^{-3}	0.9347×10^{-3}
happyBkgd3_0 and happyBkgd4_0	49.537×10^{-3}	1.3966×10^{-3}
sun3d-hotel_umd-maryland_hotel3 cloud_bin_35 and 36	93.479×10^{-3}	48.141×10^{-3}



Figure 6. Coarse registration results based on WOA (a) and NIWOA (b) for point cloud bun000 and bun045.



Figure 7. Coarse registration results based on WOA (a) and NIWOA (b) for point cloud dragonBk1_0 and dragonBk4_0.



Figure 8. Coarse registration results based on WOA (a) and NIWOA (b) for point cloud ArmadilloOnHeadMultipleOffset_15 and ArmadilloOnHeadMultipleOffset_45.

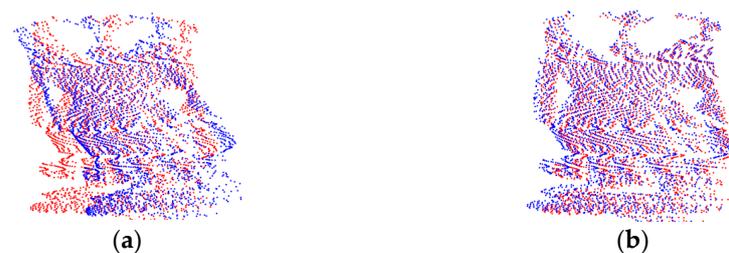


Figure 9. Coarse registration results based on WOA (a) and NIWOA (b) for point cloud happyBkgd3_0 and happyBkgd4_0.



Figure 10. Coarse registration results based on WOA (a) and NIWOA (b) for point cloud sun3d-hotel_umd-maryland_hotel3_cloud_bin_35 and sun3d-hotel_umd-maryland_hotel3_cloud_bin_36.

6.3. Experiments on Coarse–Fine Registration of Point Clouds

The test results of multiple sets of coarse registration show that the proposed NIWOA algorithm can achieve a better coarse registration effect for the point cloud data after feature extraction, which also indicates that the coarse registration based on the NIWOA algorithm can provide a more reliable initial registration position for the next step of fine registration. To verify the final registration results, we further test the effectiveness and robustness of this improved algorithm by combining the ICP algorithm. In the experiment, we use the relative positions of point clouds obtained by coarse registration methods based on WOA and NIWOA as the initial values of the ICP algorithm. In the case of increasing iterations, we use these methods to calculate the final registration accuracy on point cloud models of different datasets. To this end, this paper provides two accuracy evaluation indicators, including the root mean square error (RMSE) and mean absolute error (MAE) of the Euclidean distance between the corresponding points of the two point clouds after coarse–fine registration, which are expressed in the following equations:

$$RMSE(P, Q) = \sqrt{\frac{1}{M} \sum_{i=1}^M \|Rp_i^* + t - q_i^*\|^2} \quad (26)$$

$$MAE(P, Q) = \frac{1}{M} \sum_{i=1}^M \|Rp_i^* + t - q_i^*\| \quad (27)$$

where R and t represent the rotation matrix and translation vector from point p_i^* to point q_i^* , respectively, and M is the number of corresponding point pairs after feature point extraction. Both RMSE and MAE can reflect the effect of registration, and the smaller their values are, the higher the accuracy of registration is. As can be seen from Table 4, compared with the classical WOA algorithm, our improved method has a great improvement in the registration accuracy, and the NIWOA can also reduce the final registration error to a certain extent.

Table 4. Coarse and fine registration accuracy combined with the ICP algorithm.

Point Cloud Model	WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE
bun000 and bun045	0.2137×10^{-3}	0.1796×10^{-3}	0.1830×10^{-3}	0.1596×10^{-3}
dragonBk1_0 and dragonBk4_0	0.4564×10^{-3}	0.3835×10^{-3}	0.3991×10^{-3}	0.3557×10^{-3}
ArmadilloOnHeadMultipleOffset_15 and 45	2.2423×10^{-3}	1.7998×10^{-3}	1.6007×10^{-3}	1.4047×10^{-3}
happyBkgd3_0 and happyBkgd4_0	1.2529×10^{-3}	0.9981×10^{-3}	1.5877×10^{-3}	0.8737×10^{-3}
sun3d-hotel_umd-maryland_hotel3 cloud_bin_35 and 36	6.0036×10^{-3}	5.8355×10^{-3}	1.8908×10^{-3}	1.7002×10^{-3}

After verifying the effectiveness of the NIWOA algorithm for point cloud registration, this paper further considers the registration performance of the proposed improved algorithm for initial position changes. In order to better prove the robustness of the NIWOA algorithm for arbitrary initial pose registration, we perform varying degrees of rotation and

translation transformations on the point clouds to be registered in these point cloud data, as shown in Table 5. A total of 10 groups of transform positions are given in the table, in which the rotation angle refers to the angle of rotation along the x, y, and z axes, respectively, and the translation parameter represents the translation distance along the three coordinate axes. In the experiment, the NIWOA+ICP, WOA+ICP, and direct ICP registration methods were compared. RMSE and MAE were also used as evaluation indexes. The comparison results are shown in Tables 6–10.

Table 5. Different initial position changes.

Transformation	Rotation Angle	Translation Parameter
1	$\pi/18, \pi/18, \pi/18$	0.02, 0.02, 0
2	$-\pi/18, -\pi/18, \pi/18$	0.02, 0.02, 0
3	$\pi/18, \pi/9, \pi/18$	0.02, 0.02, 0.02
4	$-\pi/9, -\pi/18, \pi/18$	0.04, 0.04, 0.04
5	$-\pi/9, -\pi/9, \pi/18$	-0.02, -0.02, 0.02
6	$\pi/18, \pi/18, -\pi/9$	0.04, -0.04, -0.04
7	$-\pi/9, \pi/9, -\pi/9$	0.02, 0.02, 0.02
8	$-\pi/6, \pi/6, \pi/6$	0.02, 0, 0.02
9	$\pi/3, \pi/4, \pi/5$	0.02, 0.02, 0.02
10	$\pi/4, \pi/5, -\pi/3$	0.02, 0.02, 0.02

Table 6. Accuracy comparison of different methods under initial position transformation of point cloud Bunny.

Transformation	ICP		WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	2.1002×10^{-3}	1.4861×10^{-3}	0.2007×10^{-3}	0.1604×10^{-3}	0.1594×10^{-3}	0.1260×10^{-3}
2	3.4861×10^{-3}	1.7831×10^{-3}	1.2827×10^{-3}	1.0576×10^{-3}	1.1262×10^{-3}	0.9321×10^{-3}
3	3.1411×10^{-3}	2.3184×10^{-3}	0.9909×10^{-3}	0.7848×10^{-3}	0.4054×10^{-3}	0.3247×10^{-3}
4	3.9570×10^{-3}	3.4569×10^{-3}	3.3988×10^{-3}	2.7422×10^{-3}	2.8368×10^{-3}	2.3830×10^{-3}
5	1.9144×10^{-3}	1.6427×10^{-3}	0.3501×10^{-3}	0.2812×10^{-3}	0.2022×10^{-3}	0.1631×10^{-3}
6	2.8302×10^{-3}	2.4801×10^{-3}	1.6946×10^{-3}	1.3569×10^{-3}	1.6486×10^{-3}	1.3116×10^{-3}
7	18.126×10^{-3}	10.026×10^{-3}	5.2826×10^{-3}	4.3323×10^{-3}	4.1156×10^{-3}	3.4269×10^{-3}
8	20.061×10^{-3}	14.061×10^{-3}	2.9983×10^{-3}	2.5001×10^{-3}	1.9489×10^{-3}	1.6364×10^{-3}
9	29.314×10^{-3}	24.731×10^{-3}	7.4075×10^{-3}	5.9711×10^{-3}	7.5001×10^{-3}	6.1138×10^{-3}
10	66.117×10^{-3}	61.217×10^{-3}	37.331×10^{-3}	30.152×10^{-3}	22.662×10^{-3}	17.913×10^{-3}

Table 7. Accuracy comparison of different methods under initial position transformation of point cloud Dragon.

Transformation	ICP		WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	2.8112×10^{-3}	1.4781×10^{-3}	0.4685×10^{-3}	0.3926×10^{-3}	0.4039×10^{-3}	0.3458×10^{-3}
2	71.331×10^{-3}	69.913×10^{-3}	44.146×10^{-3}	36.296×10^{-3}	1.0666×10^{-3}	0.8857×10^{-3}
3	50.096×10^{-3}	41.562×10^{-3}	30.267×10^{-3}	25.145×10^{-3}	0.4070×10^{-3}	0.3437×10^{-3}
4	15.584×10^{-3}	10.015×10^{-3}	3.1386×10^{-3}	2.5825×10^{-3}	2.2677×10^{-3}	1.9539×10^{-3}
5	17.146×10^{-3}	11.291×10^{-3}	8.8337×10^{-3}	7.6719×10^{-3}	5.1751×10^{-3}	4.4633×10^{-3}
6	15.001×10^{-3}	10.687×10^{-3}	12.501×10^{-3}	9.8141×10^{-3}	1.6681×10^{-3}	1.3077×10^{-3}
7	67.839×10^{-3}	57.630×10^{-3}	34.551×10^{-3}	28.232×10^{-3}	9.1667×10^{-3}	7.2391×10^{-3}
8	66.110×10^{-3}	61.561×10^{-3}	47.292×10^{-3}	39.386×10^{-3}	22.095×10^{-3}	18.644×10^{-3}
9	36.001×10^{-3}	30.088×10^{-3}	19.637×10^{-3}	16.394×10^{-3}	6.3658×10^{-3}	5.2960×10^{-3}
10	34.892×10^{-3}	29.939×10^{-3}	23.121×10^{-3}	19.864×10^{-3}	19.764×10^{-3}	15.716×10^{-3}

Table 8. Accuracy comparison of different methods under initial position transformation of point cloud Armadillo.

Transformation	ICP		WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	18.641×10^{-3}	10.124×10^{-3}	1.5373×10^{-3}	1.2586×10^{-3}	0.2062×10^{-3}	0.1553×10^{-3}
2	18.712×10^{-3}	11.349×10^{-3}	2.1173×10^{-3}	1.7448×10^{-3}	1.5942×10^{-3}	1.3149×10^{-3}
3	17.130×10^{-3}	13.177×10^{-3}	3.5651×10^{-3}	2.8069×10^{-3}	2.5926×10^{-3}	2.0991×10^{-3}
4	13.577×10^{-3}	10.006×10^{-3}	5.2117×10^{-3}	4.3217×10^{-3}	1.2529×10^{-3}	1.0146×10^{-3}
5	14.519×10^{-3}	12.508×10^{-3}	6.9152×10^{-3}	5.6668×10^{-3}	2.5551×10^{-3}	2.0391×10^{-3}
6	11.667×10^{-3}	9.6474×10^{-3}	7.6794×10^{-3}	6.2240×10^{-3}	2.0824×10^{-3}	1.6497×10^{-3}
7	10.969×10^{-3}	9.9773×10^{-3}	7.1197×10^{-3}	5.9673×10^{-3}	4.9126×10^{-3}	4.1194×10^{-3}
8	17.478×10^{-3}	14.302×10^{-3}	5.9541×10^{-3}	4.8468×10^{-3}	1.9536×10^{-3}	1.5721×10^{-3}
9	70.447×10^{-3}	61.347×10^{-3}	31.051×10^{-3}	24.896×10^{-3}	3.9365×10^{-3}	3.0721×10^{-3}
10	77.893×10^{-3}	74.622×10^{-3}	36.490×10^{-3}	29.492×10^{-3}	36.263×10^{-3}	29.271×10^{-3}

Table 9. Accuracy comparison of different methods under initial position transformation of point cloud Happy Buddha.

Transformation	ICP		WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	1.3479×10^{-3}	1.1371×10^{-3}	0.4246×10^{-3}	0.3392×10^{-3}	0.1132×10^{-3}	0.0903×10^{-3}
2	1.1748×10^{-3}	1.0030×10^{-3}	1.1835×10^{-3}	0.9645×10^{-3}	1.0259×10^{-3}	0.8322×10^{-3}
3	6.0047×10^{-3}	5.4641×10^{-3}	3.1674×10^{-3}	2.5306×10^{-3}	0.2740×10^{-3}	0.2330×10^{-3}
4	3.9903×10^{-3}	3.2616×10^{-3}	2.3172×10^{-3}	1.9182×10^{-3}	2.1463×10^{-3}	1.7688×10^{-3}
5	5.3781×10^{-3}	5.1791×10^{-3}	5.4066×10^{-3}	4.5087×10^{-3}	3.9862×10^{-3}	3.3243×10^{-3}
6	1.1792×10^{-3}	1.0463×10^{-3}	0.5406×10^{-3}	0.4141×10^{-3}	0.2620×10^{-3}	0.2222×10^{-3}
7	14.300×10^{-3}	11.413×10^{-3}	9.0206×10^{-3}	7.4120×10^{-3}	5.0598×10^{-3}	4.2153×10^{-3}
8	12.588×10^{-3}	11.647×10^{-3}	2.5597×10^{-3}	2.1557×10^{-3}	0.9920×10^{-3}	0.7820×10^{-3}
9	17.493×10^{-3}	13.414×10^{-3}	4.4930×10^{-3}	3.7820×10^{-3}	4.3744×10^{-3}	3.7186×10^{-3}
10	29.157×10^{-3}	24.937×10^{-3}	15.620×10^{-3}	12.618×10^{-3}	14.347×10^{-3}	11.438×10^{-3}

Table 10. Accuracy comparison of different methods under initial position transformation of point cloud Sun3d-hotel.

Transformation	ICP		WOA + ICP		NIWOA + ICP	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
1	8.9401×10^{-3}	8.4811×10^{-3}	5.7463×10^{-3}	5.0161×10^{-3}	1.5710×10^{-3}	1.3297×10^{-3}
2	10.257×10^{-3}	10.043×10^{-3}	9.5630×10^{-3}	8.0642×10^{-3}	8.9771×10^{-3}	7.1357×10^{-3}
3	2.1486×10^{-3}	1.8326×10^{-3}	1.4353×10^{-3}	1.2796×10^{-3}	0.2486×10^{-3}	0.2102×10^{-3}
4	60.011×10^{-3}	57.561×10^{-3}	13.508×10^{-3}	11.683×10^{-3}	10.366×10^{-3}	8.9703×10^{-3}
5	9.1137×10^{-3}	8.8991×10^{-3}	5.9154×10^{-3}	5.1550×10^{-3}	4.3545×10^{-3}	3.8034×10^{-3}
6	35.593×10^{-3}	27.911×10^{-3}	17.320×10^{-3}	13.497×10^{-3}	9.3239×10^{-3}	8.0980×10^{-3}
7	29.316×10^{-3}	26.633×10^{-3}	15.143×10^{-3}	12.344×10^{-3}	7.7929×10^{-3}	6.5554×10^{-3}
8	94.538×10^{-3}	89.441×10^{-3}	25.555×10^{-3}	18.465×10^{-3}	27.143×10^{-3}	20.060×10^{-3}
9	88.807×10^{-3}	83.218×10^{-3}	71.708×10^{-3}	58.281×10^{-3}	32.818×10^{-3}	26.988×10^{-3}
10	91.861×10^{-3}	86.144×10^{-3}	74.896×10^{-3}	60.091×10^{-3}	40.155×10^{-3}	28.103×10^{-3}

According to the values of RMSE and MAE in Tables 6–10, it can be seen that in most cases, the NIWOA algorithm proposed in this paper can obtain higher registration accuracy in the coarse and fine registration that is combined with ICP, which has a significant performance improvement compared with the registration directly using ICP and also has a certain breakthrough compared with WOA+ICP. This is because our proposed NIWOA algorithm uses circle chaotic mapping to explore the registration space in a more comprehensive way. Under the adjustment of Newton’s inertia weight, the global search

ability and local exploration level of the algorithm are flexibly kept within a reasonable range of relations. Under the action of nonlinear convergence factors, the algorithm can be used to explore the registration space. In the later stage of the algorithm search, the global optimal solution is more accurately found, that is, the optimal coordinate transformation of coarse registration is completed. Therefore, the NIWOA algorithm obtains a more accurate coarse registration position. The registration result is used as the initial pose of ICP for iterative optimization, and the final registration error is also reduced as expected, which fully proves the effectiveness and robustness of the improved method.

In order to clearly show the effectiveness and robustness of the overall registration strategy in this paper, the experiments in this section are presented in a complete visualization on the above four model point clouds and one scene point cloud, respectively, as shown in Figures 11–15, where Figure 11a demonstrates the original input point cloud, Figure 11b is the coarse registration based on the NIWOA, Figure 11c is the fine registration using ICP on the basis of coarse registration, and Figure 11d is the final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered. As can be seen from the figures, thanks to the reliable coordinate transformation calculated by NIWOA for the initial point cloud pose, the fine registration algorithm has a good initial value, and the overall registration method used in this paper obtains a more accurate registration effect and also achieves the expected improvement effect. Figure 11e is the error convergence curve of the coarse registration by using NIWOA, and Figure 11f is the convergence curve of the error transformation with the number of iterations in the ICP fine registration process. In the figures, the blue arrow represents the flow guidance of point cloud registration, and the orange arrow represents the correspondence with the registration result and the convergence curve. From the final error performance, it can be seen that NIWOA effectively reduces the coarse registration error of two point clouds to a certain extent, and the ICP algorithm further improves the registration accuracy on this basis.

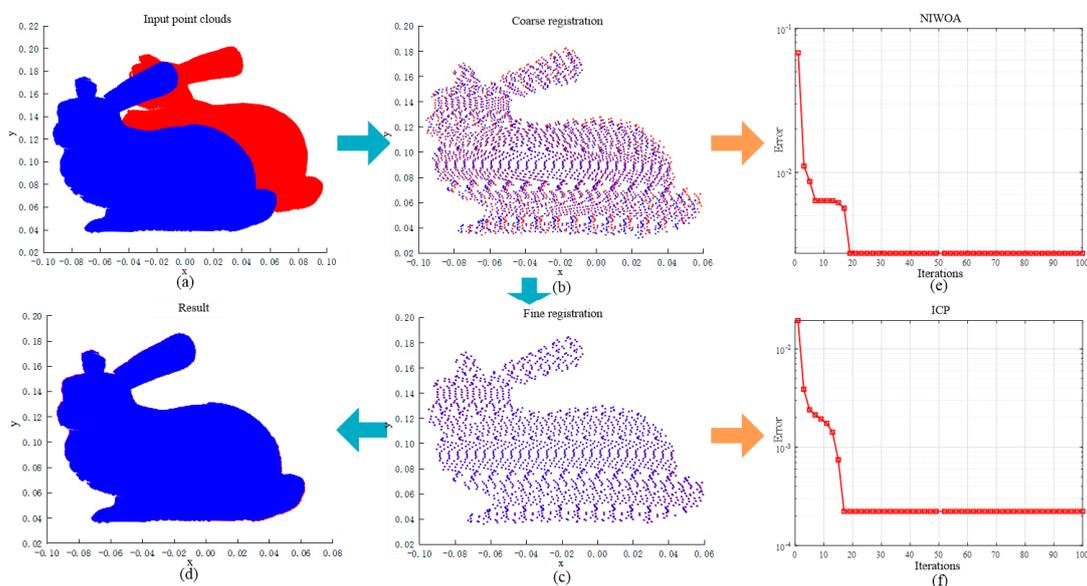


Figure 11. Coarse–fine registration process of point cloud bun000 and bun045 and convergence curve of the registration error of the NIWOA and ICP algorithms. (a) original input point cloud, (b) coarse registration based on the NIWOA, (c) fine registration using ICP on the basis of coarse registration, (d) final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered, (e) error convergence curve of the coarse registration by using NIWOA, (f) convergence curve of the error transformation with the iterations in the ICP fine registration process.

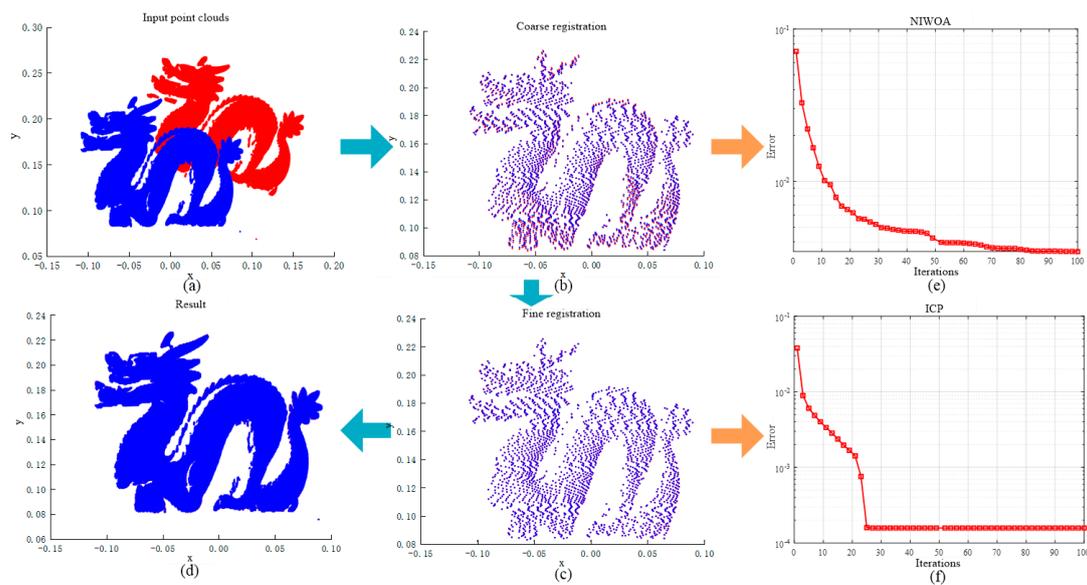


Figure 12. Coarse–fine registration process of point cloud dragonBk1_0 and dragonBk4_0 and convergence curve of the registration error of the NIWOA and ICP algorithms. (a) original input point cloud, (b) coarse registration based on the NIWOA, (c) fine registration using ICP on the basis of coarse registration, (d) final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered, (e) error convergence curve of the coarse registration by using NIWOA, (f) convergence curve of the error transformation with the iterations in the ICP fine registration process.

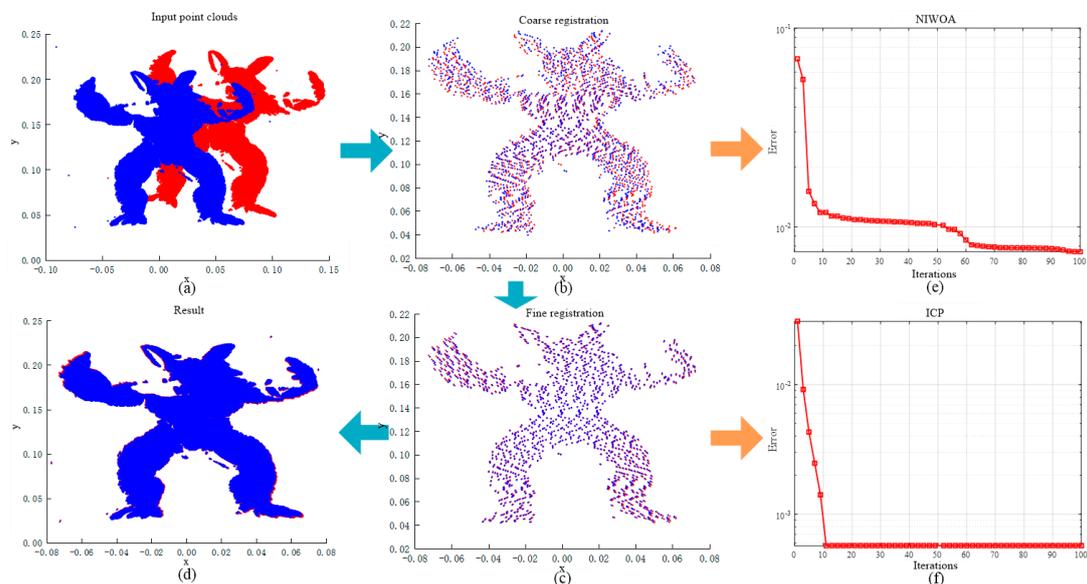


Figure 13. Coarse–fine registration process of point cloud ArmadilloOnHeadMultipleOffset_15 and ArmadilloOnHeadMultipleOffset_45 and convergence curve of the registration error of the NIWOA and ICP algorithms. (a) original input point cloud, (b) coarse registration based on the NIWOA, (c) fine registration using ICP on the basis of coarse registration, (d) final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered, (e) error convergence curve of the coarse registration by using NIWOA, (f) convergence curve of the error transformation with the iterations in the ICP fine registration process.

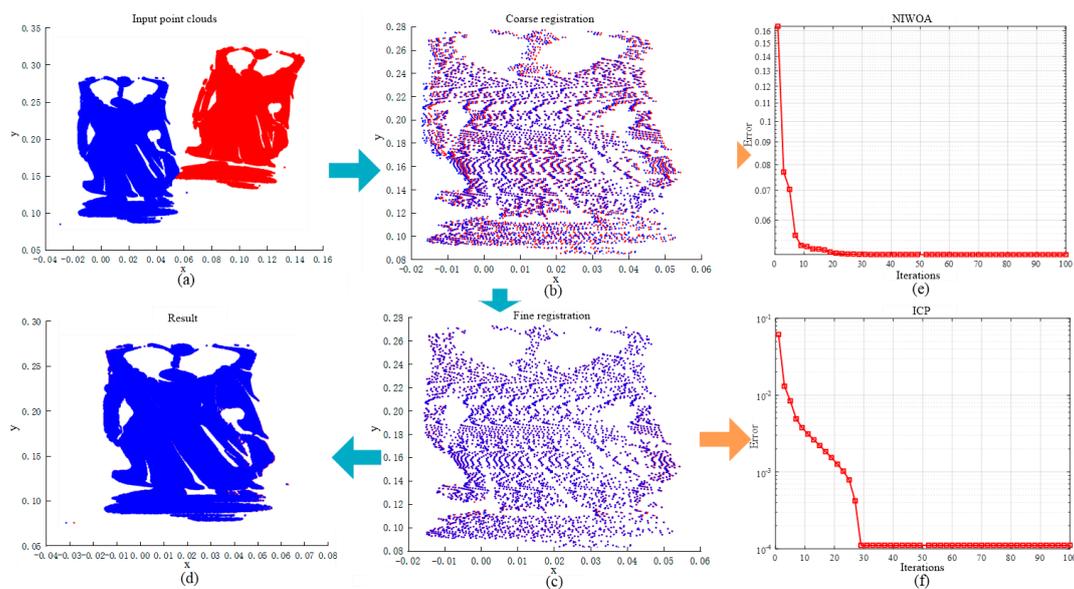


Figure 14. Coarse–fine registration process of point cloud happyBkgd3_0 and happyBkgd4_0 and convergence curve of the registration error of the NIWOA and ICP algorithms. (a) original input point cloud, (b) coarse registration based on the NIWOA, (c) fine registration using ICP on the basis of coarse registration, (d) final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered, (e) error convergence curve of the coarse registration by using NIWOA, (f) convergence curve of the error transformation with the iterations in the ICP fine registration process.

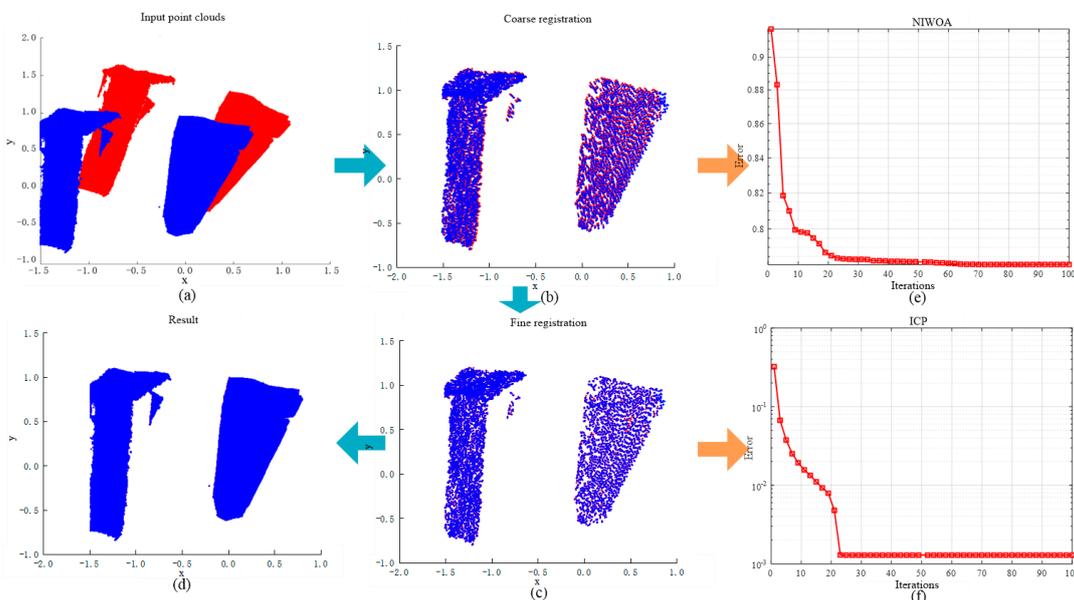


Figure 15. Coarse–fine registration process of point cloud sun3d-hotel_umd-maryland_hotel3 cloud_bin_35 and sun3d-hotel_umd-maryland_hotel3 cloud_bin_36 and convergence curve of the registration error of the NIWOA and ICP algorithms. (a) original input point cloud, (b) coarse registration based on the NIWOA, (c) fine registration using ICP on the basis of coarse registration, (d) final result of mapping the final calculated coordinate transformation parameters to the input point cloud to be registered, (e) error convergence curve of the coarse registration by using NIWOA, (f) convergence curve of the error transformation with the iterations in the ICP fine registration process.

The execution efficiency of the algorithm is another important index to evaluate the performance of point cloud registration. Therefore, after determining the effective improvement of the accuracy of the registration method proposed in this paper, we record the time consumption of different methods to complete registration. The experimental test was conducted on the point cloud data Bunny, with the initial position of the point cloud to be registered being rotated $\pi/18, \pi/18, \pi/18$ and the translation distance being 0.02, 0.02, 0. The statistics are shown in Table 11, which shows the registration times using only the ICP algorithm, the original WOA algorithm combined with ICP registration, and the NIWOA algorithm combined with ICP registration. All time units in the table are in seconds (s). It can be clearly seen from the table that the registration time of the proposed NIWOA algorithm combined with ICP algorithm is less, which is significantly reduced compared with the ICP algorithm only. It also has a certain competitive advantage compared with WOA + ICP and has a higher time efficiency because the improved algorithm can search the global optimal value more accurately and quickly. In the coarse registration stage, the position of the target point cloud is found in general, which provides a reliable initial pose for further iterative searches of ICP, and it reduces the computational burden of the ICP algorithm.

Table 11. Registration times for different methods (s).

Point Cloud Model	ICP		WOA + ICP		NIWOA + ICP		
	t_{ICP}	t_{WOA}	$t_{ICP'}$	t_{sum}	t_{NIWOA}	$t_{ICP''}$	t_{sum}
bun000 and bun045	9.62	2.08	0.78	2.86	1.58	0.73	2.31
dragonBk1_0 and dragonBk4_0	11.06	2.60	0.74	3.34	2.47	0.54	3.01
ArmadilloOnHeadMultipleOffset_15 and 45	8.57	1.61	0.39	2.00	1.49	0.41	1.90
happyBkgd3_0 and happyBkgd4_0	27.39	9.04	1.14	10.18	8.36	1.22	9.58
sun3d-hotel_umdmaryland_hotel3cloud_bin_35 and 36	13.26	6.55	0.79	7.34	3.91	0.57	4.48

7. Conclusions

In this paper, a point cloud coarse and fine registration method based on the new improved version of whale optimization algorithm (NIWOA) and ICP algorithm is proposed. A circle chaotic map, Newton inertia weight, and nonlinear convergence factor are integrated in WOA to enhance the global exploration and local exploitation ability of the algorithm. The improved algorithm is used to optimize the objective function of the coarse registration of the point cloud to obtain a more accurate initial registration position, and this initial registration result is used as the initial value of the ICP algorithm to iteratively compute the globally optimal coordinate transformations to achieve the final registration. Through the coarse registration experiments on different model point clouds and the scene point cloud, it is verified that NIWOA can effectively improve the coarse registration accuracy. At the same time, in the coarse and fine registration experiment combined with the ICP algorithm, by changing the initial positions of multiple sets of point clouds to be registered, the superiority of NIWOA+ICP for improving the registration performance and the robustness of different registration conditions is proven. In addition, in terms of the execution efficiency of registration, the proposed method is also proven to have better performance.

In future work, according to the characteristics of the solution space of the optimization objective function and in combination with more advanced improvement strategies, we will focus on developing improved algorithms with faster optimization speeds and higher search accuracies based on this study, with the aim of being able to calculate more accurate registration parameters and provide higher registration efficiency for larger and more complex models and scene point clouds.

Author Contributions: Conceptualization, Y.T.; Methodology, Y.T.; Software, Y.T.; Validation, Y.T.; Formal analysis, Y.T.; Investigation, Y.T.; Resources, X.Y. and J.Z.; Data curation, X.Y. and J.Z.; Writing—original draft, Y.T.; Writing—review & editing, X.Y. and J.Z.; Visualization, Y.T.; Supervision, X.Y. and J.Z.; Project administration, X.Y. and J.Z.; Funding acquisition, X.Y. and J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Jilin Province Development and Reform Commission (2020C018-3), the Department of Science and Technology of Jilin Province (20220203091SF).

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Buonomici, F.; Carfagni, M.; Furferi, R.; Governi, L.; Lapini, A.; Volpe, Y. Reverse engineering of mechanical parts: A template-based approach. *J. Comput. Des. Eng.* **2018**, *5*, 145–159. [\[CrossRef\]](#)
2. Yanamandra, K.; Chen, G.L.; Xu, X.; Mac, G.; Gupta, N. Reverse engineering of additive manufactured composite part by toolpath reconstruction using imaging and machine learning. *Compos. Sci. Technol.* **2020**, *198*, 108318. [\[CrossRef\]](#)
3. Raffo, A.; Barrowclough, O.J.D.; Muntingh, G. Reverse engineering of CAD models via clustering and approximate implicitization. *Comput. Aided Geom. Des.* **2020**, *80*, 101876. [\[CrossRef\]](#)
4. Zheng, L.; Chen, S.; Chen, X.; Ji, S. Reverse Engineering-Inspired Parametric 3D Geometry Model of Marine Propeller. *Pol. Marit. Res.* **2023**, *30*, 35–47. [\[CrossRef\]](#)
5. Dong, Z.; Wu, X.; Ma, Z. Research on 3D model reconstruction based on a sequence of cross-sectional images. *Mach. Vis. Appl.* **2021**, *32*, 92. [\[CrossRef\]](#)
6. Song, L.; Jiang, Q.; Zhong, Z.; Dai, F.; Wang, G.; Wang, X.; Han, G.; Zhang, D. Technical path of model reconstruction and shear wear analysis for natural joint based on 3D scanning technology. *Measurement* **2022**, *188*, 110584. [\[CrossRef\]](#)
7. Peng, Y.; Lin, S.; Wu, H.; Cao, G. Point Cloud Registration Based on Fast Point Feature Histogram Descriptors for 3D Reconstruction of Trees. *Remote Sens.* **2023**, *15*, 3775. [\[CrossRef\]](#)
8. Zou, Z.; Lang, H.; Lou, Y.; Lu, J. Plane-based global registration for pavement 3D reconstruction using hybrid solid-state LiDAR point cloud. *Autom. Constr.* **2023**, *152*, 104907. [\[CrossRef\]](#)
9. Besl, P.J.; McKay, N.D. A method for registration of 3D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1992**, *14*, 239–256. [\[CrossRef\]](#)
10. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73. [\[CrossRef\]](#)
11. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.
12. Yang, X.S. A new metaheuristic bat-inspired algorithm. In *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 65–74.
13. Mirjalili, S. The ant lion optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98. [\[CrossRef\]](#)
14. Yang, X.S.; Deb, S. Engineering Optimisation by Cuckoo Search. *Int. J. Math. Model. Numer. Optim.* **2010**, *1*, 330–343. [\[CrossRef\]](#)
15. Mehrabian, A.R.; Lucas, C. A novel numerical optimization algorithm inspired from weed colonization. *Ecol. Inform.* **2006**, *1*, 355–366. [\[CrossRef\]](#)
16. Shi, X.; Liu, T.; Han, X. Improved Iterative Closest Point (ICP) 3D point cloud registration algorithm based on point cloud filtering and adaptive fireworks for coarse registration. *Int. J. Remote Sens.* **2020**, *41*, 3197–3220. [\[CrossRef\]](#)
17. Ma, Y.; Zhao, H.D.; Zhang, W. Research on Intelligent Minefield Attack Decision Based on Adaptive Fireworks Algorithm. *Arab. J. Sci. Eng.* **2019**, *44*, 2487–2496.
18. Zhan, X.; Cai, Y.; He, P. A three-dimensional point cloud registration based on entropy and particle swarm optimization. *Adv. Mech. Eng.* **2018**, *10*, 1687814018814330. [\[CrossRef\]](#)
19. Feng, Y.; Tang, J.; Su, B.; Zhou, Z. Point cloud registration algorithm based on the grey wolf optimizer. *IEEE Access* **2020**, *8*, 143375–143382. [\[CrossRef\]](#)
20. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [\[CrossRef\]](#)
21. Liu, D.; Hong, D.; Wang, S.; Chen, C. Genetic Algorithm-Based Optimization for Color Point Cloud Registration. *Front. Bioeng. Biotechnol.* **2022**, *10*, 923736. [\[CrossRef\]](#)
22. Chen, L.; Kuang, W.; Fu, K. A resample strategy and artificial bee colony optimization-based 3d range imaging registration. *Comput. Vis. Image Underst.* **2018**, *175*, 44–51. [\[CrossRef\]](#)
23. Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **2007**, *39*, 459–471. [\[CrossRef\]](#)
24. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [\[CrossRef\]](#)
25. Chakraborty, S.; Saha, A.K.; Chakraborty, R.; Saha, M. An enhanced whale optimization algorithm for large scale optimization problems. *Knowl.-Based Syst.* **2021**, *233*, 107543. [\[CrossRef\]](#)

26. Liu, M.; Yao, X.; Li, Y. Hybrid whale optimization algorithm enhanced with Lévy flight and differential evolution for job shop scheduling problems. *Appl. Soft Comput.* **2020**, *87*, 105954. [[CrossRef](#)]
27. Luo, J.; Shi, B. A hybrid whale optimization algorithm based on modified differential evolution for global optimization problems. *Appl. Intell.* **2019**, *49*, 1982–2000. [[CrossRef](#)]
28. Li, Y.; Han, M.; Guo, Q. Modified whale optimization algorithm based on tent chaotic mapping and its application in structural optimization. *KSCE J. Civ. Eng.* **2020**, *24*, 3703–3713. [[CrossRef](#)]
29. Anitha, J.; Pandian, S.I.A.; Agnes, S.A. An efficient multilevel color image thresholding based on modified whale optimization algorithm. *Expert Syst. Appl.* **2021**, *178*, 115003. [[CrossRef](#)]
30. Lin, X.; Yu, X.; Li, W. A heuristic whale optimization algorithm with niching strategy for global multi-dimensional engineering optimization. *Comput. Ind. Eng.* **2022**, *171*, 108361. [[CrossRef](#)]
31. Saha, N.; Panda, S. Cosine adapted modified whale optimization algorithm for control of switched reluctance motor. *Comput. Intell.* **2022**, *38*, 978–1017. [[CrossRef](#)]
32. Yang, W.; Xia, K.; Fan, S.; Wang, L.; Li, T.; Zhang, J.; Feng, Y. A multi-strategy Whale optimization algorithm and its application. *Eng. Appl. Artif. Intell.* **2022**, *108*, 104558. [[CrossRef](#)]
33. Chakraborty, S.; Saha, A.K.; Chakraborty, R.; Saha, M.; Nama, S. HSWOA: An ensemble of hunger games search and whale optimization algorithm for global optimization. *Int. J. Intell. Syst.* **2022**, *37*, 52–104. [[CrossRef](#)]
34. Li, G.; Cui, Y.; Wang, L.; Meng, L. Automatic registration algorithm for the point clouds based on the optimized RANSAC and IWOA algorithms for robotic manufacturing. *Appl. Sci.* **2022**, *12*, 9461. [[CrossRef](#)]
35. Zhong, Y. Intrinsic shape signatures: A shape descriptor for 3D object recognition. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, Kyoto, Japan, 27 September–4 October 2009; pp. 689–696.
36. Rusu, R.B.; Blodow, N.; Marton, Z.C.; Beetz, M. Aligning Point Cloud Views using Persistent Feature Histograms. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 3384–3391.
37. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [[CrossRef](#)]
38. The Stanford 3D Scanning Repository. Available online: <http://graphics.stanford.edu/data/3Dscanrep/> (accessed on 19 August 2014).
39. Zeng, A.; Song, S.; Nießner, M.; Fisher, M.; Xiao, J.; Funkhouser, T. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In Proceedings of the of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1802–1811.
40. Halber, M.; Funkhouser, T. Fine-to-coarse global registration of rgb-d scans. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1755–1764.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.