Non-cooperative Target Detection of Spacecraft Objects based on Artificial Bee Colony Algorithm

Xinyu Liu[1], Donghui Li[1], Na Dong[1*], K.L. Yung[2], W.H. Ip[2]

1. School of Electrical Engineering and Information Engineering, Tianjin University, Tianjin 300072, China.
2. Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, China.

Accurate and fast recognition of Non-Cooperative Target (NCT) is an essential technology for Tethered Space Robots (TSR) when implementing on-orbit service. NCT is the spacecraft objects which could not be identified and located through the communication channels in the outer space. One of the approaches to resolve this problem is the use of intelligent computer vision system to track the object and hence to carry out operation and support tasks. By treating multi-object detection as the multi-peak optimization problem, this paper proposed a multi-circle detection method for recognition of circular modules on NCT and a multi-template matching method for other modules. Incorporating the concept of species into the Artificial Bee Colony (ABC) algorithm, a multi-peak optimization algorithm named as Species based Artificial Bee Colony (SABC) was proposed and applied to the above multi-object detection problem. In order to test and verify SABC, benchmark tests consisted of five multi-peak functions were conducted to verify the effectiveness for higher accuracy and shorter execution time in finding multiple optima. Firstly, we applied SABC into circle detection and design complete detection flow. Circular modules of NCT were adopted to the experiments and it was verified that the proposed multi-circle detection method based on SABC can locate circles with higher success rate and better accuracy than other methods. Furthermore, we employed SABC to multi-template matching and conducted experiments using simulated cases of China space missions as well as the Chang'e Camera Point System (CPS) developed by the Hong Kong Polytechnic University with China National Space Agency. Practical applications of multi-objects detection for NCT with kinds of noises, weakened brightness and shoot in continuous flights were evaluated in the experiments, and the results demonstrated that our methods were robust under different kinds of circumstances and thus confirmed its feasibility and effectiveness.

Key words: Non-cooperative Target; Artificial bee colony algorithm; Multiple optima; Species; Multi-circle detection; Multi-template matching

1 Introduction

1.1 Background

In recent decades, the aerospace industry has flourished [1]. Thousands of artificial satellites are running on various types of orbits. Between all kinds of space missions in the past few decades, many spacecrafts failed to enter into the track properly or became invalid in orbit. The geosynchronous orbital resources are limited, that is, the number of geostationary satellites that can be accommodated is limited, therefore, the orbital value is extremely high. Due to the failure of on-orbit devices, exhaustion of fuel, or the expiration of satellite life, some geosynchronous orbit satellites has been useless, but they still occupy precious orbital positions. Therefore, on-orbit service for such invalid (non-cooperative) satellites, such as refueling, component upgrades or arresting the derailments is of great significance [2]. Several on-orbit service tasks [3,4] have already been carried out .

On-orbit service technology has wide applications and commonly includes repairing, upgrading, refueling and re-orbiting spacecraft on-orbit [5]. Traditional space robots, such as the ETS-VII, Canadian Arm and Track Express, lack flexibility due to the rigid body structure and the length of the manipulator. The tethered space robot (TSR) system [6,7], as a new type of space robot, uses space tethers to overcome these shortcomings and is helpful for capturing NCT. The TSR includes a robot platform, a space tether, and an operating robot, as shown in Fig. 1.

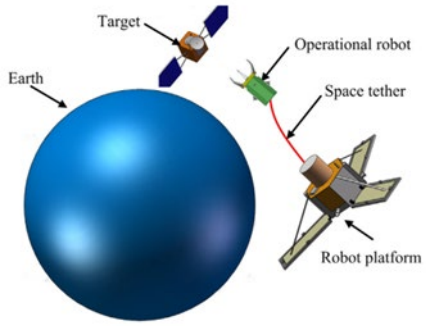

Fig. 1. Tethered space robot (TSR) system

National Academies, Space Studies Board (SSB) and Aeronautics and Space Engineering Board (ASEB) defined NCT in the evaluation report of the maintenance plan of the Hubble Space Telescope as follow [8]: Non-cooperative targets are those space targets without a communication response device or active identification sensor and are not able to be identified and located by other spacecrafts through communication signal feedback. NCT have the following characteristics [9]: (1) No dedicated interface is installed to capture the docking; (2) No suitable reflector or sensor for measurement. (3) The movement of the target satellite cannot be controlled. Because the NCT cannot provide effective cooperative identification information to the tracker, it is difficulty to perform space operations such as grasping by mechanical arm and autonomous rendezvous and docking.

Under the conditions without cooperation information, making full use of the natural structural features and surrounding features of NCT on the photos to recognize target spacecraft is the key technology for space NCT on-orbit service. Compared with sensors such as microwave radar and lidar, the vision-based navigation system has the advantages of higher precision, lower power consumption and cost [10], etc. It has become the main detection method for space missions in short distance. Target recognition can be acquired using the two methods, template-based or feature-based ones. In this paper, recognition based on shape feature and template matching is the emphases for research.

Since the key structures of artificial targets are mostly composed of cubes and cylindrical modules, it is a worth considering method to recognize artificial targets based on simple shape features imaged by optical sensors. Common components installed on NCT are usually selected as the region of interest (ROI) to operate the robot [11], including the bolts that can separate the satellite from the rocket, the docking rings, and the apogee rocket engine injections, the span of solar panels and so on, as shown in Fig. 2. A proper grasping structure should be a necessary equipment for the spacecraft, which ensures the generalization of the grasping system and has a simple

and reliable shape (rectangle, triangle, circle, etc.) Furthermore, another method based on template matching has been widely used for object recognition and tracking problems in aerospace [12-16]. The templates can be selected as a sub-image such as an module in NCT or an object always exists in its surroundings. In view of the above analysis, when charged coupled device (CCD) cameras on operating robots communicate with computer vision system, which consists of above advanced image detection algorithms, TSR can accurately and quickly capture NCT.

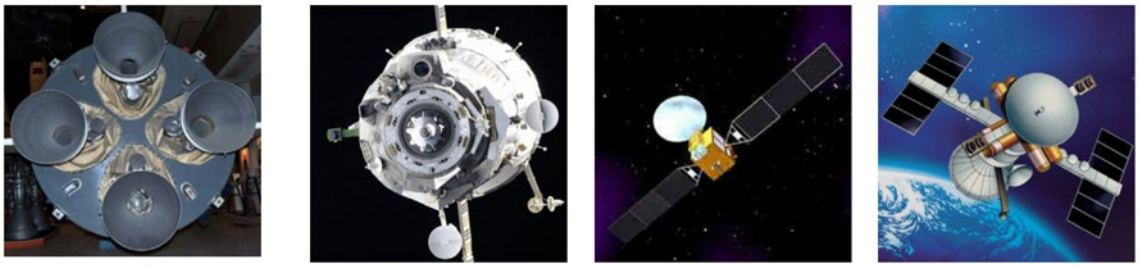

Fig. 2. Common detection components on non-cooperative targets

1.2 Related works

Object localization based on computer vision has been widely used for grasping NCT. For example, a novel template matching algorithm is presented to solve the problem of NCT recognition during short-distance rendezvous events [17]. Chen et al. [18] introduces a novel object localization method for target grasping by predicting object regions before extracting them by using the features of histogram of oriented gradients and support vector machine, which can reduce the search area of targets remarkably and runs fast. Liu et al. [19] shows a practical detection method based on ellipse fitting for the adapter ring on NCT. In this paper, we first adopt circle-detection method to recognize circular modules on NCT. Selecting the circular modules on NCT, such as the circular bolts, docking ring or motor injector as grasping structure, how to accurately detect and identify NCT is viewed as a circle detection problem. Then template matching method is employed to recognize other modules. To recognize multiple objects at one same time, multi-circle detection and multi-template matching are the main work of this paper.

Circle detection problem has been widely studied. Most detection methods can be classified into two categories: technologies based on Hough transform and least square fitting method [20]. Hough transform has been widely used and improved in circle detection [21-25]. For example, an improved Hough transform technique is applied to the development of automatic biometric iris recognition systems [23]. Least square fitting method is also applied to circle detection. Zelniker et al. used convolution-based least square fitting method to estimate the parameter of circular object [26]. Frosio and Borghese [27] employed prior knowledge of foreground and background statistics to estimate the likelihood of circular object parameter. However, contradiction between computation cost and accuracy for the methods based on Hough transform still exists, and the methods based on least squares have low robustness.

The template matching approach is derived from the idea of searching for assigned image sections or given features within an interrelated image [28]. The template can occupy only a limited image area, or it can have the

same size as the search image. The matching function is accomplished by exploiting a correlation relationship. Researchers have proposed many kinds of correlation laws, among which Sum of Hamming Distances (SHD), Normalized Cross Correlation (NCC), Sum of Absolute Difference (SAD), Sum of Squared Difference (SSD), and distance transformation ones are the most commonly used. In view of the above analysis, a template matching algorithm is worth considering to deal with this troublesome problem. Template matching technique has been widely used for object recognition and tracking problems. For example, the multi-template matching method was applied for for cucumber recognition in natural environment [13]. An efficient auto-detection method using a multi-template matching technique for PCB components detection is described [14]. Since most of the template matching methods are so time consuming that they can't be used to many real time applications, Cai et al. [15] use coarse-to-fine searching strategy to improve the matching efficiency and propose a partial computation elimination scheme to further speed up the searching process. Jisung Yoo et al. [16] presents a histogram-based template matching method that copes with the large scale difference between target and template images. The degree of matching is often determined by evaluating the normalized cross correlation (NCC) value. The NCC has long been an effective and simple similarity measurement method in feature matching. The basic idea of template matching is to loop the template through all the pixels in the captured image and compare the similarity. To recognize NCT, the idea of multi-template matching (MTM) is the focus of this work.

Optimization has been widely applied to various fields [29]. In essence, multi-object detection including multi-circle detection and multi-template matching can be viewed as optimization process of multi-peak function [30]. Multi-peak optimization is used to locate all the optima within the search space and has been widely studied by many scholars. A number of intelligent algorithms based on kinds of techniques have been proposed to solve multi-peak optimization problem [31-35]. A "niche" ABC algorithm is proposed since niche technology can maintain swarm diversity and avoid the algorithm from converging only to the global optimal solution [32]. For the shortcomings of insufficient search ability and low optimization accuracy, DENG Tao [33] proposed an improved artificial fish swarm hybrid algorithm (AFSHA) to solve the multi-peak optimization problem. In order to improve the swarm diversity of invasive weed optimization (IWO) algorithm, a niche weed optimization algorithm (NIWO) is proposed [34], which divides weed swarms according to the Euclidean distance among individuals in the swarm and adopts adaptive niche strategy to determine the number of categories. However, most multi-peak optimization methods still exist the defects of lower accuracy and longer running time. Thus, the concept of species is proposed to realize multi-peak optimization [31,35].

Artificial bee colony algorithm was proposed by the Turkish scholar Karaboga [36] in 2005, and the basic idea is inspired by the idea that the bee colony collaborate to take honey through division of labor and information exchange. ABC has the advantages of faster convergence speed, less control parameters, higher searching accuracy, stronger robustness and more simple operation [37, 38]. WANG Jiaoyan [39] pointed out that the quality of solution solved by ABC algorithm is relatively good compared with Genetic Algorithm (GA), Differential Evolution (DE) algorithm and Particle Swarm Optimization (PSO) algorithm. Such characteristics have motivated the use of ABC to solve different sorts of engineering problems such as signal processing [40], flow shop scheduling [41], structural inverse analysis [42], clustering [43], vehicle path planning [44] and electromagnetism [45].

In view of the above analysis, the aim of this paper is to propose a species based artificial bee colony algorithm (SABC) for multi-peak optimization and further to solve the multi-object detection for NCT. The rest of this paper is organized as follows: ABC algorithm will be briefly introduced in Section 2. Section 3 gives detailed description of the novel algorithm SABC and verifies its optimized performance for multi-peak functions. Section 4 applies SABC to circle detection and illustrates the method to evaluation the circular integrity and accuracy. In section 5, we provide contrast experiments on circular modules of NCT in various environments to verify the proposed circle-detection method. Similarly, SABC is employed by MTM and further applied to recognition of NCT in Section 6. Section 7 concludes this work.

2 Artificial Bee Colony Algorithm

The basic model for the bee colony to achieve collective wisdom contains four elements: food sources, employed bees, onlooker bees and scouts, and two basic behaviors: recruitment of bees and the abandonment of food sources [36, 38]. The process that bees search food sources can be summarized as the following three steps: *a)* employed bees find food sources and share the food sources information through waggle dancing; *b)* onlooker bees choose food sources to take honey according to the food sources information provided by employed bees; *c)* if the quality of certain food source is not improved after several trials, the employed bee gives up the food source, and becomes a scout near the hive to continue to search for a new food source. When high-quality food source is found, it will turn into an employed bee.

ABC algorithm is put forward as a swarm intelligence algorithm by imitating the bee's honey-taking behavior. ABC algorithm is different from genetic algorithm and other swarm intelligence algorithms, with role conversion as its unique mechanism [36, 38]. The bees collaborate to find high-quality food sources by the conversion of three different roles: employed bees, onlooker bees and scouts [36]. In the optimization process of ABC algorithm, the roles of the three kinds of bees vary: employed bees maintain the quality of solution; onlooker bees improve the convergence rate; scouts enhance the ability to get rid of the local optimal [38].

The algorithm is to start from a randomly generated initial swarm. Then half of the individuals with better fitness values begin search, and competitive survival strategy is used to reserve individuals, which is called “employed bees search”. The other half with worse fitness values are treated as the onlooker bees and scouts. Each onlooker bee uses the "roulette wheel selection" method to select a good individual, and greedily search round it to form another half of new swarm. This process is called “onlooker bees search”. The individuals generated by employed bees and onlooker bees form a new swarm, meanwhile, “scouts” are introduced to the new swarm to avoid loss of swarm diversity. After the abovementioned processes, the swarm completes the update for once. The algorithm approaches the optimal solution through iterating and calculating continuously, preserving good individuals and giving up inferior individuals.

A nonlinear minimization problem is taken as an example to describe the operation process of ABC algorithm in detail. The problem of minimizing the nonlinear function can be expressed as $min\ f(X), X^L \le X \le X^U$, where X^U and X^L are the upper and lower bounds respectively of the variable $X = (X_1, X_2, ..., X_n)$, and X is the n -dimension vector. To solve a nonlinear minimization problem by ABC algorithm, the first step is to generate the

initial swarm containing NP individuals within the range of X. Assuming maximum cycles number of the algorithm is $maxCycle$, the $i-th$ individual in the $t-th$ iteration swarm can be expressed as $x_i^t = (x_i^t(1),...,x_i^t(n))$, where $i = 1,2,...,NP$.

Steps of the ABC algorithm are described as follows:

a Initialization

Generate the NP individuals randomly satisfying the constraint in the search space of the optimization problem to form an initial swarm. The employed bees are consisted of the first half of the individuals with better fitness values, and the other half performs as the onlooker bees and scouts.

b Employed bees search

Each employed bee generates a new food source in the neighborhood of its present position. A greedy selection process is applied. If the fitness value of the new is better, then old solution is replaced; otherwise, the old remains.

c Onlooker bees search

Each onlooker bee selects one of the new employed bees. The probability of selecting an employed bee is proportional to its fitness. After the employed bee is selected, the onlooker bee will go to the selected employed bee and select a new food source position inside the neighborhood of the selected employed bee. If the fitness of the new solution is better than before, such position is adopted; otherwise, the last solution remains.

d Scouts search

If an individual isn't improved further through for a predetermined number of cycles noted as $Limit$, the corresponding individual change into a scout generated randomly.

3 Species based Artificial Bee Colony Algorithm

Multi-peak optimization is used to find all the optima in the search space. This optimization problem has been widely concerned by scholars. A number of similar algorithms have been proposed [31-35]. Inspired by [35], this paper introduces the concept of species into the ABC algorithm, and proposes a novel algorithm called species based the artificial bee colony (SABC) to solve multi-peak optimization problem.

3.1 Species

The introduction of species to swarm intelligent optimization algorithm has been proven to be a feasible method to solve multi-peak optimization problem [31, 35]. The species technique aims to generate multiple species in parallel by dividing the large swarm and search synchronously for multiple optima. The core of the SABC algorithm is the concept of species. A species can be defined as a group of individuals with similar characteristics. This similarity can be determined by the European distance. The shorter the European distance between the two individuals, the greater their similarity is. A distance parameter γ_s is introduced to donate the distance between the species center and the edge. The center of a species, also called species seed, is usually

defined as the individual with the highest fitness value. All individuals within γ_s from the seed are considered to belong to the same species.

Steps of dividing the swarm into species can be summarized as follows:

a Rank all individuals by fitness values (in descending order)

b Find the individual with the highest fitness value as the first species seed

c Assign each individual to different species according to Euclidean distance and determine the seeds of each species.

d Repeat steps *c* until all individuals are divided.

The Fig. 3 illustrates the iterative process of dividing species.

Taking the optimization objective function with four global optima for example, the swarm distributions in iterative order are shown in Fig. 3. Fig. 3 a is the swarm distribution after initialing, and it is evenly distributed. Then the swarm is divided into four species and searches within every species for several times. Fig. 3 b shows the updated swarm, and the four species and their seeds are indicated. After several iterations, as seen from Fig. 3 c, the swarm converged to the four best values gradually and the species are more obvious. The final distribution is as Fig. 3 d, and all individuals are almost around the four optima. In this process, the bee colony evolves gradually as iteration, and finally the algorithm locates all the optima successfully.

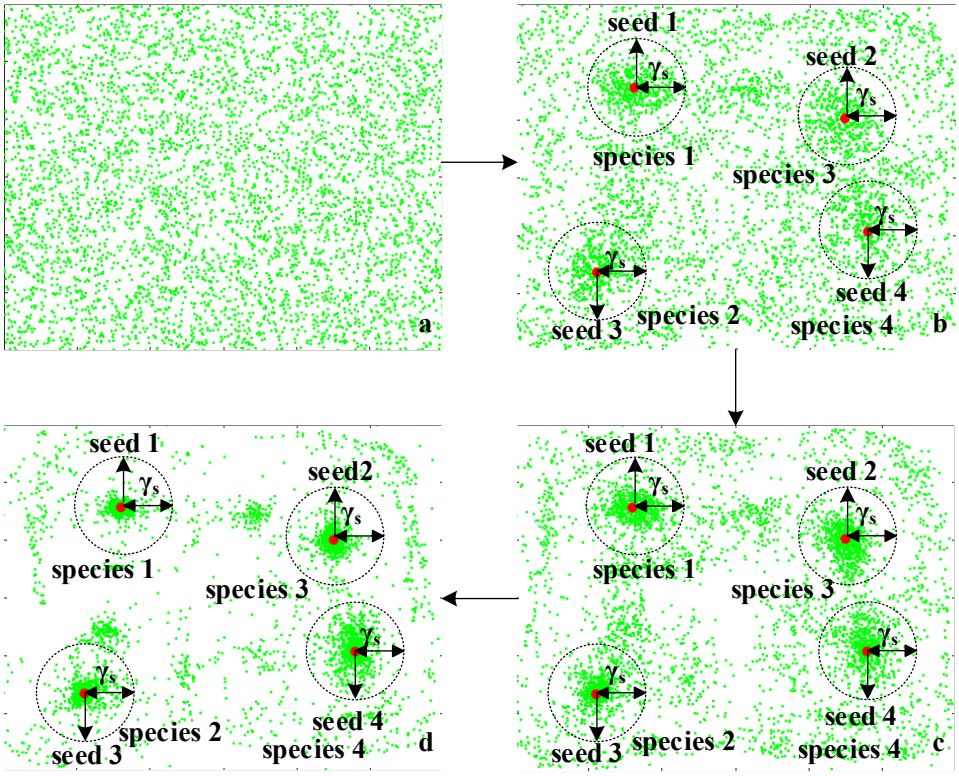

Fig. 3. Iterative process of dividing species

3.2 Species Based Artificial Bee Colony Algorithm

Here, we propose a new multi-peak optimization algorithm SABC by incorporating the concept of species into ABC. The novelty of SABC lies that it searches according to ABC mechanism within species instead of the whole swarm. In SABC, after initializing and ranking the fitness values of all individuals, the swarm is divided into some species according to the Euclidean distance as described in section 3.1, and each species searches independently as ABC mechanism.

Initialization is given in formula (1), where X^U and X^L are the upper and lower bounds respectively, NP and D are the individuals number and dimension, and $Foods$ is the position vector storing the position of all individuals. At the same time, a vector Bas is initialized to record the number that each individual stays in the same location continuously.

$$Foods = X^L + (X^U - X^L) \cdot rand(NP, D) \tag{1}$$

Referred to Figure. 4, the principle of SABC is illustrated in detail. As seen from Fig. 4 a, the individuals are distributed evenly in the search space after initialing. Then all individuals are divided into some species and the seeds are determined. The locations of individuals within every species are stored in $sFoods\{q\}$ separately, and the corresponding flags are stored in $sBas\{q\}$, where the parameter q is used as an index to distinguish species. Fig. 4 b illustrates how to divide the swarm into species, where the discrete points donate individuals. Here, the number of species is assumed as three, but in fact, the number and scale of the species are also related to γ_s and the range of search space. The locations of the seeds are also pointed out in Fig. 4 b, and it can be seen that every species is composed of the individuals within the globe whose center is the seed and the radius is γ_s. After determining the species, the individuals within species are divided into three the kinds of bees to start search. As shown in Fig. 4 c, three different colors are used to show three kinds of bees. The method of determining three kinds of bees has been described in section 2.

Employed bee: within species, every employed bee will search a new food source according to formula (2).

$$sol\{q\}(i, j) = sFoods\{q\}(i, j) + (sFoods\{q\}(i, j) - sFoods\{q\}(neibour, j)) \cdot (rand - 0.5) \cdot 2 \tag{2}$$

Where i and j respectively represent index of a randomly chosen employed bee from the current species and its random dimension, $neibour$ represents a randomly chosen employed bee in addition to i. $sol\{q\}(i, j)$ and $sFoods\{q\}(i, j)$ respectively represent new food source and the current food source. Different from ABC, since SABC searches within species, all formulas are added the $'q'$. Then fitness values of $sol\{q\}(i, j)$ and $sFoods\{q\}(i, j)$ are compared and the better is reserved using greedy selection. If the employed bee keeps the old food source, the flag vector should be updated as $sBas\{q\}(i) = sBas\{q\}(i) + 1$. This is “employed bees search” of SABC.

Then, probability vector $P\{q\}$ is obtained by formula (3) for employed bees in every species, which is proportional to honey in food source.

$$P\{q\} = (0.9 \cdot sFitness\{q\} / max(sFitness\{q\}) + 0.1 \quad (3)$$

Onlooker bee: every onlooker bee selects a good food source $sFoods\{q\}(i, j)$ from the updated employed bees using the "roulette wheel selection" method. As known from the formula (3), the food source with greater fitness has more probability to be selected, so that the swarm constantly converges. Then, the onlooker bee searches a new food source $sol\{q\}(t, j)$ in the neighborhood of $sFoods\{q\}(i, j)$, as formula (2). Finally, $sFoods\{q\}(i, j)$ and $sol\{q\}(t, j)$ are compared to reserve the better as a new onlooker bee. This is the "onlooker bees search" of SABC.

Scout : after all employed bees and onlooker bees search, if some bee i keeps in a same location for $Limit$ times, that is $sBas\{q\}(i) > Limit$, the current food source will be abandoned and the bee will become a scout. The scout will randomly search a new food source in the search space as formula (1), so that the swarm diversity won't be lost. This is "onlooker bees search" of SABC.

In one cycle, all species independently search once according to the above process. Finally, both the location $sFoods\{q\}$ and flag $sBas\{q\}$ of every species are restored respectively in $Foods$ and Bas to prepare for next iteration.

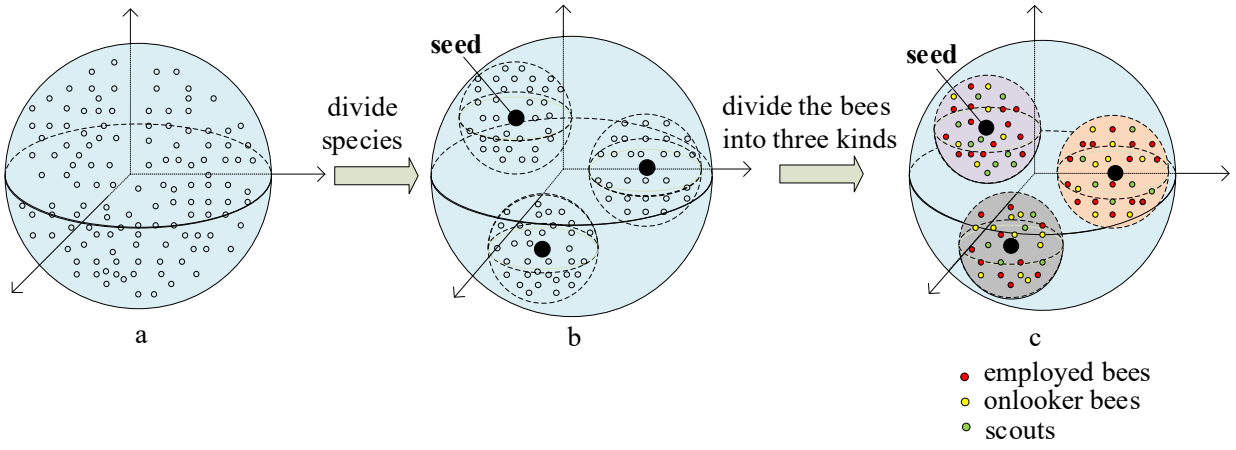

Fig. 4. Principle of SABC

According to principle, the implementation of SABC can be summarized by the following steps:

Step 1. Initialize following parameters: the number of individuals NP, the maximum of keeping in the same food source continuously $Limit$, maximum cycles $maxCycle$, the current cycle $iter$, the number of scouts $SearchNumber$ and the radius of species γ_s. The location and flag vector of swarm are named as $Foods$ and Bas respectively.

Step 2. Divide species and determine the seeds. Rank all individuals in descending order and divide them into species according to steps in section 3.1. Individuals belonged to species q are stored $sFoods\{q\}$ and their flags vector is called as $sBas\{q\}$. For species q, the following steps $a-e$ are executed:

a. Within the species q, individuals are classified into three kinds of bees according to the fitness values, and

the flag vector $sBas\{q\}$ is used to record how many times the bees keeping in the same food source continuously.

b. Every employed bee i searches for a new food source. If the new is better than the current food source, the current location of the food source is replaced, and $sBas\{q\}(i)$ is reset as zero, otherwise update the flag vector as $sBas\{q\}(i) = sBas\{q\}(i) + 1$.

c. Calculate probability probability vector $P\{q\}$ according to fitness of employed bees. Every onlooker bee chooses an employed bee in probability $P\{q\}$ and search around it to generate a new food source, then reserve the better one.

d. If the number of certain employed bee or onlooker bee keeping in same location continuously more than $Limit$, the bee will give up the current food source and become a scout to search a new food source.

e. If the species q achieves the above search process, the updated location $sFoods\{q\}$ and the flag vector $sBas\{q\}$ are restore into the swarm.

Step 3. Two stop criteria are employed: either all optima have been found or the number of cycles $iter$ attains an upper limit $maxCycle$. Output locations and fitness corresponding to seeds of all species if a termination criterion is met.

The procedure of SABC is exhibited in Fig. 5.

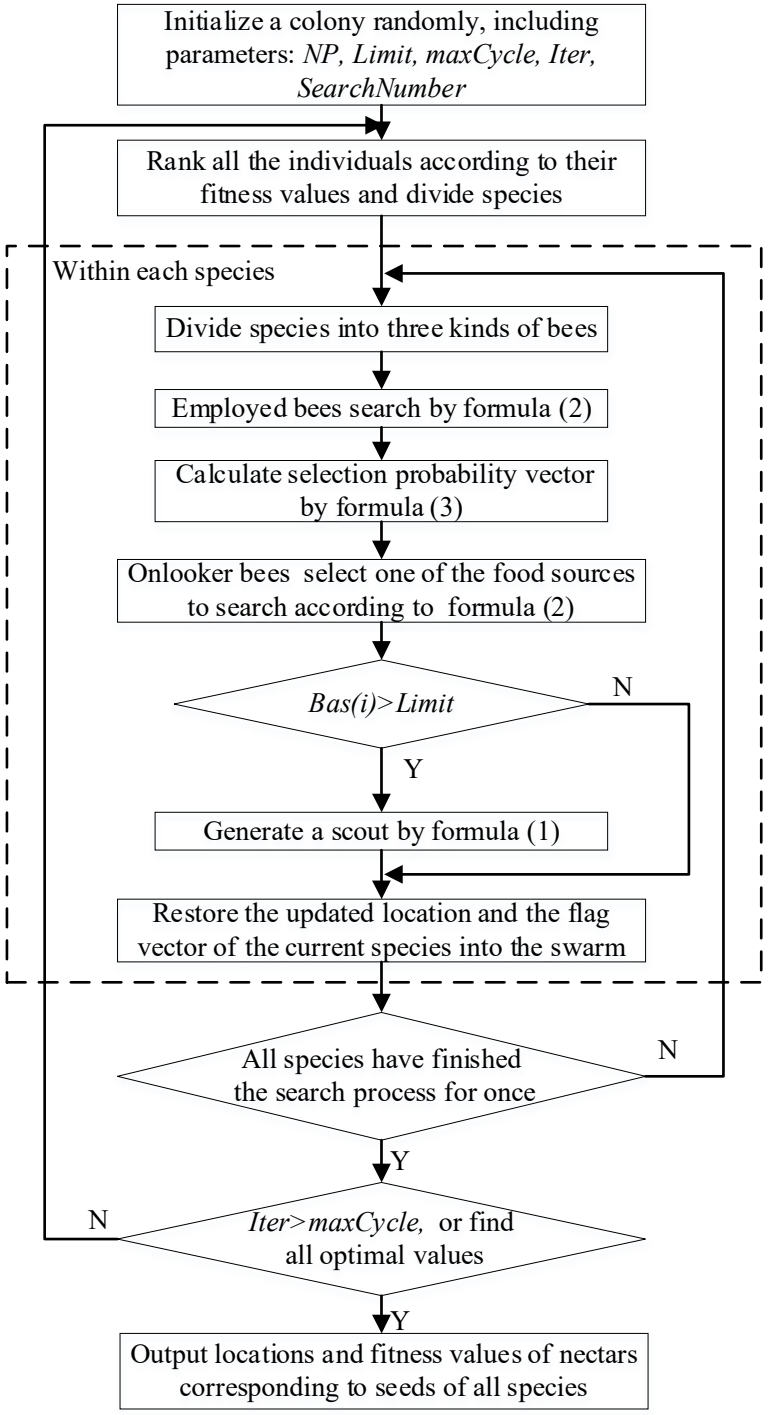

Fig. 5 Flowchart of SABC

3.3 Benchmark test

Five benchmark functions are used to test the performance of the newly proposed method. Table 1 shows definition, range of variables and comment of the test functions. All above benchmarks are maximization of objectives. An Intel (R) Core(TM) i5-6500 CPU @ 3.20GHz 8.00G RAM computer is used to execute all experiments under MATLAB programming.

Table 1 Benchmark test functions

Function	Range	comment
$F1(x) = sin^6(5\pi x)$	[0,1]	Five global optima with equal heights
$F2(x) = exp\left(-2log(2)\cdot\left(\frac{x-0.1}{0.8}\right)^2\right)\cdot sin^6(5px)$	[0,1]	One global optimum and four local optima
$F3(x) = sin^6(5\pi(x^{\frac{3}{4}} - 0.05))$	[0,1]	Five global optima unevenly spaced
$F4(x,y) = 200 - (x^2 + y - 11)^2 - (x + y^2 - 7)^2$	[-6,6]	Four global optima with equal heights
$F5(x,y) = (1 - 2y - \frac{sin(4\pi y - x)}{20})^2 + (y - \frac{sin(2\pi x)}{2})^2$	[-10,10]	Five global optima with equal heights

Table 2 Benchmark test results

Function	Measurement	γ_s	SGA	SPSO	SABC
F1	Accuracy	0.1	-7.92 e-05±2.58 e-04	0±0	**0±0**
	Time (s)		0.5851	0.2252	**0.0622**
	Success rate* (%)		100	100	**100**
F2	Accuracy	0.1	-3.60 e-04±3.09 e-05	-1.56 e-05±2.02 e-04	**-7.23 e-06±1.17 e-06**
	Time (s)		0.6265	0.0726	**0.0385**
	Success rate* (%)		96	100	**100**
F3	Accuracy	0.1	-4.58 e-05±6.39 e-05	0±0	**0±0**
	Time (s)		0.6646	0.1171	**0.0693**
	Success rate* (%)		94	100	**100**
F4	Accuracy	3	-0.0763±7.94 e-03	-2.39 e-07±8.94 e-07	**-1.80 e-09±1.00 e-09**
	Time (s)		0.6375	0.6654	**0.2155**

	Success rate* (%)		100	100	**100**
	Accuracy		0.0815±4.79 e-03	7.62 e-03±6.37 e-04	**5.05 e-04±5.45 e-05**
F5	Time (s)	0.15	0.6931	0.3614	**0.1503**
	Success rate* (%)		100	100	**100**

* The success rate represents the percentage of finding all the optima successfully in 50 trials

In order to enhance the algorithm analysis, the concept of species is also introduced to GA and PSO to compare with SABC [30]. The experimental parameters are fixed as: $maxCycle = 2000$, $NP = 60$, $Limit = 20$. The inertia weight ω and the acceleration coefficient c_1, c_2 of SPSO are set as 2 and 0.85 respectively. Crossover probability and mutation probability of SGA are set as 0.75 and 0.1 respectively. Other parameters are the same as SABC. Each algorithm runs 50 times, the related indictors are recorded in Table 2 to show the optimization performance.

As can be seen from Table 1, $F1$, $F2$ and $F3$ are univariate functions, while $F4$ and $F5$ are bivariate functions. And $F1$ $F4$ and $F5$ all have global optima with equal heights, while global optima of $F2$ and $F3$ are unevenly distributed. It can be observed from Table 2, SPSO and SABC can locate all the optima every time for all functions, while SGA has 96% and 94% success rate for $F2$ and $F3$, which means SGA has 2 and 3 misdetections out of 50 trials. Considering execution-time needed in these three algorithms, SABC works faster for each function. Then, comparing the optimization accuracy, optima solved by SABC are the most stable and closest to the theoretical values.

After experiments and comparison, it can be concluded that the proposed algorithm SABC has excellent performance in multi-peak optimization problem.

4 The application of SABC in circle detection

In this section, multi-circle detection is considered to be a multi-peak optimization problem, and we design a multi-circle detection method by employing SABC. The three-edge-point positioning method [30] is used to represent a circle, which can effectively narrow the search space and eliminate the infeasible location.

4.1 Representation of the circle

In this representation method, all edge points in the image are stored as an index to their relative position in the edge array V after extracting edges. Three edge points stored in the form of a sequence number corresponding to their coordinates V_i, V_j, V_k can determine a circle passing through them. Each circle is represented by center (x_0, y_0) and radius. Through coordinates of three points on the edge map, the circle C passing through them can be determined by the following formulas:

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \tag{4}$$

$$x_0 = \frac{\begin{vmatrix} x_j^2 + y_j^2 - (x_i^2 + y_i^2) & 2(y_j - y_i) \\ x_k^2 + y_k^2 - (x_i^2 + y_i^2) & 2(y_k - y_i) \end{vmatrix}}{4\left((x_j - x_i)(y_k - y_i) - (x_k - x_i)(y_j - y_i)\right)} \tag{5}$$

$$y_0 = \frac{\begin{vmatrix} 2(x_j - x_i) & x_j^2 + y_j^2 - (x_i^2 + y_i^2) \\ 2(x_k - x_i) & x_k^2 + y_k^2 - (x_i^2 + y_i^2) \end{vmatrix}}{4\left((x_j - x_i)(y_k - y_i) - (x_k - x_i)(y_j - y_i)\right)} \tag{6}$$

Thus, the parameters of circle can be represented by sequence numbers of three edge points i, j, k:

$$[x_0, y_0, r] = T(i, j, k) \tag{7}$$

Where T is the transformation composed of the computations formula (4-6) .

The test set for the points is $S = \{s_1, s_2, ..., s_{Ns}\}$, where N_s is the number of tested points on the circle edge, and the test set of points S is sampled uniformly from the circle edge. Each point s_i is a two-dimensional vector, and its coordinates (x_i, y_i) is obtained according to the following formula:

$$\begin{cases} x_i = x_0 + r \cdot \cos\dfrac{2\pi i}{N_s} \\ y_i = y_0 + r \cdot \sin\dfrac{2\pi i}{N_s} \end{cases} \tag{8}$$

4.2 Assessment of circular accuracy

Each circle corresponds to an individual in the swarm, and the algorithm finds the optimal solution, that is, the circle is detected. Generally, the fitness function is used to assess individuals. Here, for assessment of circular accuracy, the fitness function indicates the existence situation of tested points over the edge in the actual circle, and it is defined as:

$$F(\mathrm{C}) = \left(\sum_{i=0}^{N_s - 1} E(x_i, y_i)\right) / N_s \tag{9}$$

Where $E(x_i, y_i)$ is gray value of the coordinates (x_i, y_i) on the image, so $F(\mathrm{C}) \in [0 \quad 255]$. The fitness value reflects the degree of overlap between the circle constructed by the test points and the circle presented in the actual image. Therefore, the greater $F(\mathrm{C})$ implies better accuracy. For clarity, a parameter P_C is defined to evaluate the percentage of coinciding with the actual circle.

$$P_c = N_t / N_s \tag{10}$$

Where N_t is the number of test points that are actually present in the edge image, so $P_C \in [0 \quad 1]$. The greater P_C also implies better accuracy. Guided by this fitness function, the set of candidate circles corresponding to all seeds is evolved using the SABC algorithm so that the multiple optimal candidate circles can fit into actual circles. A circle will be seen as being correctly detected if the corresponding individual fitness meets

the set value (180), and its center x_0, y_0 and radius r are determined. Then differences with the theoretical values of the actual circle $\Delta x, \Delta y, \Delta r$ [46] are computed to describe detection accuracy.

The complete process of the proposed multi-circle detection algorithm based on SABC is exhibited in Fig. 6.

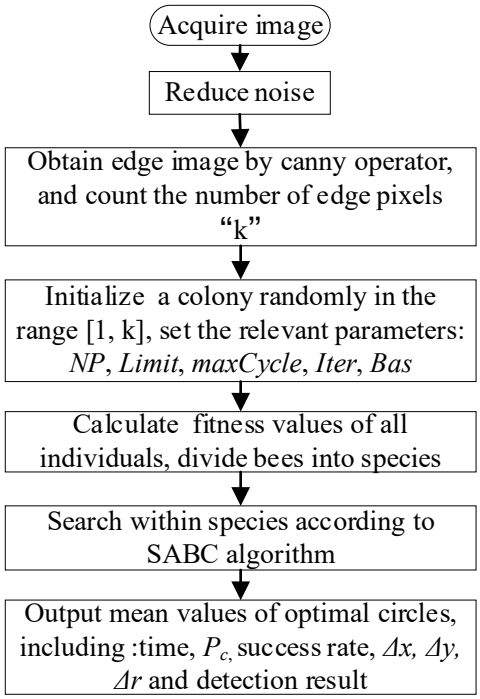

Fig. 6 Flowchart of the circle detection method based on SABC

It is worth mentioning that the number of detected circles is a pre-set value, and the algorithm searches until all circles has been correctly detected or the cycle reaches $maxCycle$.

5 The application of SABC in multi-circle detection

In this section, two test images are firstly introduced for experimental tests to verify SABC's performance in circle detection, then the method is further applied to detect circular modules on NCT.

5.1 Test experiments on drawn sketches

Two test images with randomly distributed circles are firstly used to test the proposed performance of multi-circle detection method. The distance between two individuals in a species is defined as the distance between the centers of the circles corresponding to the two individuals. The number of circles in each test image and the distance between the circles are known, and the species radius γ_s is set as a smaller value than the distance between the two nearest circles. Fig. 7 exhibits the test images, detection results in edge maps, detection results in original images, and the detected circles and its centers are marked with the same color.

The algorithm parameters are set as: $NP = 500$, $maxCycle = 2000$, $Limit = 30$, $N_s = 200$, γ_s of two images are considered as 170 and 100 respectively. The inertia weight ω of SPSO is adjusted between 0.2 and 0.9, and acceleration coefficient c_1, c_2 are set as 2. Crossover probability and mutation probability of SGA are set as 0.75 and 0.1 respectively.

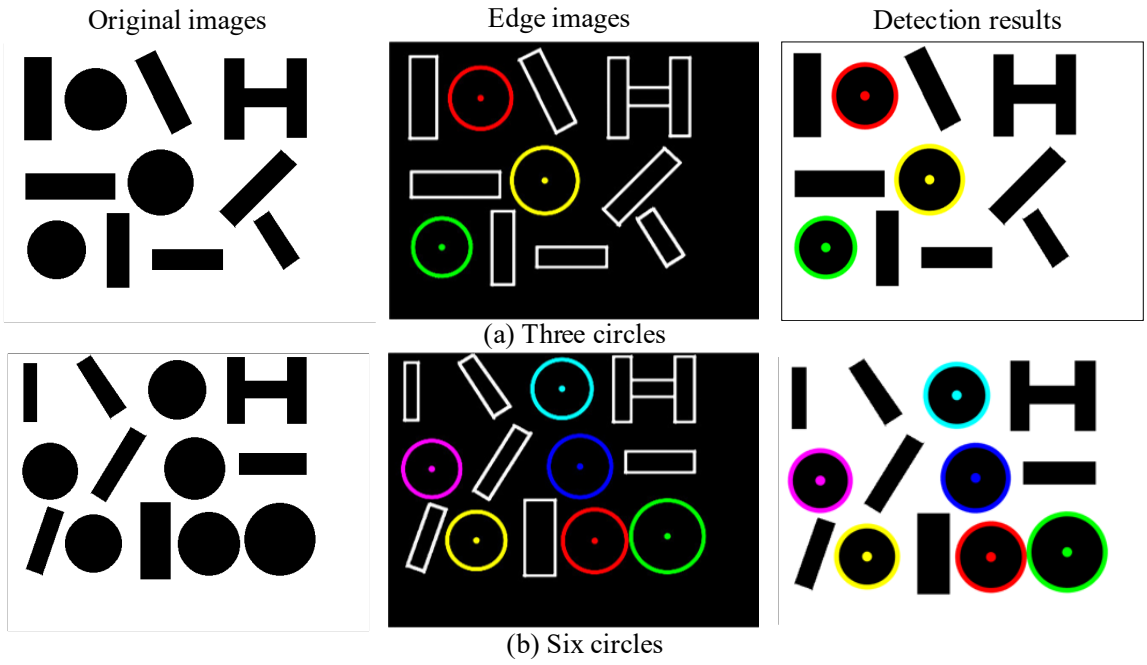

Fig. 7. Test images and the detection results

Each algorithm runs independently for 50 times. The running results including average execution time, the average and standard deviation of the P_C, the success rate and $(\Delta x, \Delta y) / \Delta r$ are given to show performance of SABC in Table 3.

As can be seen from the running results, for Fig. 7, only SABC algorithm can locate all circles every time, while SGA and SPSO algorithm have several failures. Specifically, for Fig. 7. b, the success rate of SGA and SPSO algorithm only can reach 82% and 80%, which means that 9 and 10 times out of 50 experiments fails to locate all circles. As far as execution time, the new algorithm SABC works fastest and time consumption is stable, while the SPSO and SGA use too long time. Furthermore, P_C of SABC can reach 96% and 93%, which is higher than that of SPSO and SGA and indicates higher coinciding rate and detection accuracy.

From the results, SABC can locate multiple circles with high success rate and accuracy as well as less computational time. All the above demonstrates good performance of SABC in multi-circle detection.

Table 3 The running results on the two test images out of 50 runs

Image	Method	SGA	SPSO	SABC
Figure. 7 a	Average time(s)	3.2966	2.6328	**1.8633**

	P_C (A. v and S. d)	0.9120±0.0503	0.9285±0.0363	**0.9637±4.56 e-04**
	Success rate (%)	88	90	**100**
	$(\Delta x, \Delta y) / \Delta r$	(0.32,0.68)/0.46	(0.22, 0.39)/0.34	**(0.11, 0.35)/0.25**
Figure. 7 b	Average time(s)	3.9778	3.1350	**2.0709**
	P_C (A. v and S. d)	0.9076±4.32 e-02	0.9183±1.05 e-04	**0.9391±2.69 e-05**
	Success rate (%)	92	94	**100**
	$(\Delta x, \Delta y) / \Delta r$	(1.01, 0.52)/0.53	(0.42, 0.24)/0.42	**(0.39, 0.19)/0.31**

* The success rate represents the percentage of finding all circles successfully in 50 trials.

5.2 Detection for circular modules on non-cooperative targets

In this section, the results of the application of SABC for circular modules on NCT were reported and compared with SGA and SPSO. Moreover, to fully verify our method, another approach GHC [46] was used as contrast, which is based on region-growing of gradient and histogram distribution of Euclidean distance and has been proved to be better than many other methods. As classical circle-detection method: Randomized Hough transform (RHT) [47] was also introduced for comparison.

For us, photographs of real rockets, spacecraft or space stations are difficult to obtain limited to confidentiality and cost. So it is a desirable method to use photographs of real models of these spacecraft. In our experiments, the models of spacecraft "Shenzhou 8" and space station "Tiangong 1" from China were used as shown in Fig. 8 [46]. As shown in Fig. 8, there are several circular components on the model, such as a hatch gate, docking rings, nozzles of engines, antenna of radar, etc.

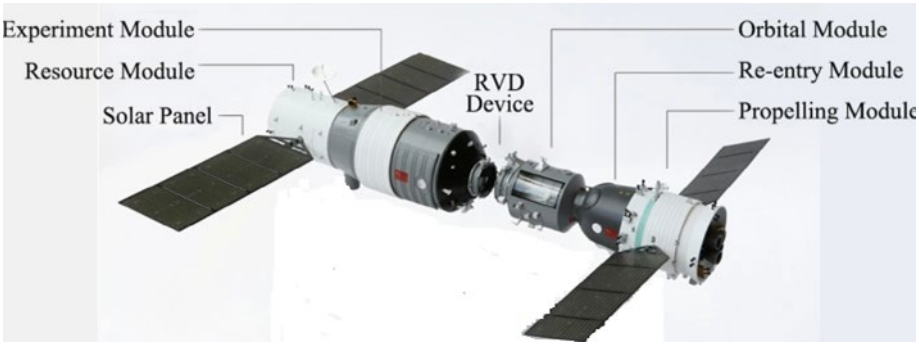

Fig. 8 The models of spacecraft "Shenzhou 8" and space station "Tiangong"

As shown in Fig. 9(a)–(d), four circular modules were used to test performance of the five methods. (a) is a circular antenna. (b) is the image of docking ring. (c) and (d) show the motor injectors. After running all methods on four images, the detected results were also shown in Fig. 9. Among the four images, (a) has a single circle, others have multiple circles, on which the multiple-circle recognition capability can be tested. It is observed that, for the simple image (a), the five methods all can detect the circle successfully. With the number of circles increasing in (b) and (c), detection by the first four methods stills were correct, while the RHT began misdetection

and missed detection. Specifically, RHT detected false circle in (b) and missed the correct circles in (b) and (c). Then in (d), GHC was also defeated by the three methods based on optimization algorithms due to a missed detected circle. So, as far as the correctness of detection results, the SGA, SPSO, and SABC work better than GHC and RHT. Moreover, the experiments (b), (c) and (d) also reflect the good performance of our method in multiple-circle recognition.

Table 4 showed the comparison results between SGA, SPSO and SABC dealing with the above four images. From the table, the data intuitively indicated that method based on SABC performs best from execution-time, accuracy, success rate, deviation with theoretical values, which once again confirmed the previous results.

The contrast between SABC, GHC and RHT was shown in Table 5. As it shown, we can find that the average time consumption of our method is much less than others'. Further, the detection deviation $(\Delta x, \Delta y), \Delta r$ by SABC were much smaller.

Fig. 9 Detection results for circular modules on non-cooperative targets

Table 4 The running results of the three algorithms on Fig. 10

Image	Algorithm	SGA	SPSO	SABC
Figure. 9 a	Average time(s)	2.0754	1.2046	**0.9523**
	P_C (A. v and S. d)	0.8427±0.7376	0.8893±0.4823	**0.9273±6.28 e-04**
	Success rate (%)	92	96	**100**
	$(\Delta x, \Delta y) / \Delta r$	(0.36,0.49)/0.57	(0.32, 0.27)/0.48	**(0.11, 0.19)/0.35**
Figure. 9 b	Average time(s)	2.9658	2.0134	**1.1731**
	P_C (A. v and S. d)	0.8936±6.85 e-02	0.9073±6.42 e-04	**0.9264±5.53 e-05**
	Success rate (%)	94	94	**100**
	$(\Delta x, \Delta y) / \Delta r$	(0.57, 0.61)/0.63	(0.50, 0.54)/0.73	**(0.49, 0.35)/0.41**
Figure. 9 c	Average time(s)	4.0738	3.1692	**2.3764**
	P_C (A. v and S. d)	0.8274±4.74 e-02	0.8536±2.85 e-04	**0.9163±7.23 e-04**
	Success rate (%)	90	92	**100**
	$(\Delta x, \Delta y) / \Delta r$	(0.61, 0.52)/0.68	(0.73, 0.44), 0.62	**(0.58, 0.39), 0.43**
Figure. 9 d	Average time(s)	4.3854	3.9435	**2.6852**
	P_C (A. v and S. d)	0.8037±2.89 e-02	0.8463±3.64 e-04	**0.8846±6.93 e-04**

	Success rate (%)	92	90	**100**
	$(\Delta x, \Delta y)$ / Δr	(1.25, 0.63),0.76	(0.97, 0.56),0.82	**(0.82, 0.46),0.63**

* The success rate represents the percentage of finding all circles successfully in 50 trials.

Table 5 The running results by SABC compared with GHC and RHT

Image	Algorithm	SABC	GHC	RHT
Figure. 9 a	Average time(s)	**0.9523**	1.56	1.09
	$(\Delta x, \Delta y)$ / Δr	**(0.11, 0.19)/0.35**	(0.36,0.47)/0.52	(0.25, 0.34)/0.37
Figure. 9 b	Average time(s)	**1.1731**	1.80	2.42
	$(\Delta x, \Delta y)$ / Δr	**(0.49, 0.35)/0.41**	(0.65,0.58)/0.61	NAN
Figure. 9 c	Average time(s)	**2.3764**	3.7	0.91
	$(\Delta x, \Delta y)$ / Δr	**(0.58, 0.39), 0.43**	(0.72,0.63)/0.72	NAN
Figure. 9 d	Average time(s)	**2.6852**	2.74	3.53
	$(\Delta x, \Delta y)$ / Δr	**(0.82, 0.46),0.63**	NAN	NAN

5.3 Detection performance with noise

The first step in our method is to denoise the images, here a further set of experiments was performed in order to evaluate the tolerance of the our method if the original image was directly affected by noise. In the tests, we added gaussian noises, speckle noises and salt & pepper noises to the same image separately. Noises of different parameters were added to original images and then they were detected by our method.

Fig. 10 Detection results with noise

Fig. 10 exhibits two original images: a circular component with two circles. With noise parameters gradually increasing from the default value, the detection results are demonstrated in Fig. 10. As observed in Fig. 10, our method acquired a good detection result when default parameter. Then the noise intensity was gradually increasing. When the images were seriously interfered with noise, the results were not ideal. Specifically, the method failed to find the small circle in Fig. 10 when the noise was big enough. Focusing on the misdetections, we can discover the undetected circles are indeed too blurry to recognize through human vision, so the misdetections by our method can be accepted.

Based on the results, we can conclude that our method acquires a good detection result under default parameter, and Gaussian noises, Speckle noises or Salt and Pepper noises didn't affect the detection process and thus, the proposed method is robust and stable to moderate noises.

5.4 Detection performance in different light intensity

Actually, the images photographed under weak light may be not clear. To test the detection performance in weaken light, Fig. 9. b and Fig. 9. d were dimmed to compare with the detection results in normal. Fig. 11 shows the detection results with the gradually weaken light intensity.

From the edge maps, we can find that the extracted edges become less when light weakens, which increases difficulty for detection. Observing all results, all circles were detected except for only one misdetection in the darkest image of Fig. 11. a, in which two small circles were undetected, while detection results on Fig. 11. b were all correct. In fact, the missed circles are almost impossible to be identified, which makes the failure of our method understandable.

All the above, we can obtain satisfied results by the method in most cases. Thus, we come to a conclusion that the proposed method is effective for detection when light intensity weakens.

Fig. 11 Detection results with gradually weakened brightness

5.5 Detection performance during continuous flight

In the above experiments, images are some static circular components. Although our method performed well on these images, it is worth considering the detection performance during continuous flight. Hence, other experiments were designed to test the performance during continuous flight. In reality, the circle may be deformed and the size of circle may change dramatically due to the movement. Consider the problem of circular deformation, the detected circle is perfect no matter how the actual circle deforms, which means the coincidence degree between both lowers when serious deformation. In our method, when their coincidence degree meets the pre-set value, the circle is considered to be correctly detected. Therefore, to ensure that the circle is still detected, lowering the set value is a feasible method when the circle is slightly deformed. However, if deformation is severe, our method is also ineffective. Furthermore, our method limits the radius for detected circle. So during continuous flight, expanding the range of radius also is helpful for successful detection.

Here, photos taken during continuous flight of spacecraft were collected from Apollo 9 Magazine for experiments. Detection results on two spacecrafts from four continuous views are shown in Fig. 12. From the images, we can find that the spacecrafts are gradually become larger with the approaching, and the circles are not perfect, which transform to be a shape more like an ellipses. Despite this, our method always finds the two circles in Fig. 12. a and the one circle in Fig. 12. b successfully. The slightly transformation does not affect the detection results.

It can be seen obviously that the proposed method is capable of dealing with slightly deformed and gradually approaching circles. From all the above experiments, it is enough to prove that our method can achieve excellent performance during continuous flight of spacecraft.

Fig. 12 Detection results during continuous flights

6 The application of SABC in multi-template matching

To enrich multi-object detection, SABC was further introduced into multi-template matching. Similar to multi-circle detection, here, we employed SABC to search multiple templates on edge images at one time. The multi-template matching problem is formulated as in [48].

Here, the impact of the SABC algorithm was also evaluated using the open source data set from Chang'e 3 space mission [49]. The Chang'e 3 space mission from China was successfully launched and the rover "Yutu" landed on the moon's surface in December 2013. However, due to certain technical problem with the solar control panel, the rover was not able to travel to the pre-setting destination. At the limited driving range, the rover had been operated for over three years on the moon and thousands of photos were obtained in this mission. Using the improved Tethered Space Robot in future would help operate and repair the rover and overcome the problem of such NCT recognitions and control.

We used several images from the China's Chang'e 3 Lander to test our new algorithm. The results provided various insights to the development of the Camera Pointing System (CPS) which was mounted on the Lander for capturing images of the moon and rover for Chang'e space mission. The CPS developed by the Hong Kong Polytechnic University in the early 2013 was used to capture images of the moon as well as the movement of the rovers [49-52]. It was capable of 360 degree for image capturing as well as positioning and navigating of the rover. In future, the TSR embedded with ABC algorithm can be used for the operation and repair of the rover or equipment in the Chang'e missions. In this paper, we tested the proposed method and evaluated the feasibility for future space missions of Chang'e. Based on our past experience of the CPS, the proposed SABC algorithm could enhance the operation and repair services using TSR system in the future.

6.1 Multi-template matching by SABC

Multi-template matching based on the proposed algorithm SABC was applied on three images obtained from Chang'e 3, including two rovers and one lander. To be clear, we recorded the template position detected in the edge images through MATLAB, and then marked the corresponding detection results on the original images, as shown in Fig. 13. Both detected templates were 50×50 pixel in size, and the detected areas were marked with the same color as the templates. For comparison, SPSO-based MTM algorithm and SGA-based MTM algorithm [48] were also introduced. The performance of the three algorithms is exhibited in Table. 6.

Fig. 13 Multi-template matching results in the Chang'e missions

As shown in Table 6, the proposed SABC outperformed other two template matching methods. In case of the running time, the SABC-based template matching needed shorter time than SGA and SPSO. Meanwhile, among the three images, templates were correctly recognized and positioned in different frames. The targets were correctly identified and matching by SABC only failed twice in fifty trials in Fig. 14. b. In general, the correct recognition rate was more than 96%, which was quite acceptable.

Table. 6 Matching performance of SABC, SPSO and SGA

Image	Algorithm	SABC	SPSO	SGA
Figure. 13 a	Maximum execution time (s)	**0.6825**	0.9672	1.3974
	Maximum execution time (s)	**2.2408**	3.0895	3.0137
	Average time(s)	**1.1498**	1.4973	1.8804
	Success rate (%)	**100**	98	96
Figure. 13 b	Maximum execution time (s)	**0.7317**	0.9738	1.0368
	Maximum execution time (s)	**3.0325**	3.2870	3.4693
	Average time(s)	**1.3684**	1.6505	2.0737
	Success rate (%)	**96**	92	90
Figure. 13 c	Maximum execution time (s)	**0.3026**	0.5964	1.0568
	Maximum execution time (s)	**0.8286**	1.9685	2.4502
	Average time(s)	**0.4194**	1.0875	1.7653
	Success rate (%)	**100**	96	94

* The success rate represents the percentage of finding all circles successfully in 50 trials.

6.2 Multi-template matching for blurred images

Considering that the captured image may be blurred during the movement of the target, here the blurred images were introduced to verify the MTM based on SABC in blurred images and then the defuzzified images were detected again. Fig. 14 gives the results, and the two templates are found in the area with same color frame of original images.

As depicted in Fig. 14, although the templates were very blurred, the matching results all were right. The average time consumption for blurred images is respectively 1.8 s and 1.2 s, which is a little longer than the clear images. And their success rate for the two blurred images can achieve higher than 96%. Thus, conclusions can be drawn that our multi-template matching is also applicable for moving targets.

(a) Detection for blurred Lunar Rover

(b) Detection for blurred Lander

Fig. 14 Multi-template matching results for blurred images

6.3 Multi-template matching for images with noises

Experiments were further conducted to show whether the detection performance are affected by noise. To simulate the actual situation, two images added three kinds of noises were used for experiment, while the

templates were no noise added. Fig. 15 shows the matching results by MTM based on SABC.

As shown in Fig. 15, the original images have been added noises, while the templates were the same as that in Fig. 15. Despite of this, it can be seen that the template images were matched relatively well. During the experiment process, the noise parameters all were default values, and we find that mismatching will occur when the noises are big enough. Moreover, despite of the added noises, time needed is still around 1-2 s, which has no much increase, and the success rate stays more than 90%. So it can be concluded that our method is robust to general noises.

(a) Detection for Lunar Rover with noises

(b) Detection for Lander with noises

Fig. 15 Multi-template matching results for images with noises

7 Conclusions

With the exploration and use of human beings in outer space, there are more and more kinds of discarded spacecraft left in space. But the orbital resources are limited and invalid satellites have become obstacles to current and future space activities. Therefore, the implementation of on-orbit service for space debris is essential. Such non-specific identifiers and failure to communicate are often called non-cooperative targets (NCT). Accurate and fast recognition of NCT is an essential technology for Tethered Space Robots (TSR) when implementing on-orbit service. For such targets, recognition based on intelligent computer vision system has more advantages than other methods. This paper proposes a multi-object detection method of NCT based on Artificial Bee Colony Algorithm.

In this paper, incorporating ABC with the concept of species, a novel algorithm SABC for the multi-peak optimization problem was firstly proposed. Its novelty is that multiple species generated in parallel search synchronously as ABC algorithm, so that all multiple optima can be found. The update mechanism and iterative process of SABC were given in detail. In contrast to ABC algorithm finding only one optimal solution in one iteration, the proposed SABC algorithm is able to find single or multiple optima by running only one optimization cycle. In order to evaluate the performance of the proposed algorithm, experiments over five benchmark test functions were executed. After comparison with SGA and SPSO, the results showed that the new SABC algorithm can solve multi-peak optimization problem with higher accuracy and better success rate in shorter time.

Further, considering the multi-object detection problem to be a multi-peak optimization process. SABC was applied to the multi-circle detection and the multi-template matching. The total flow of the circle detection can be summarized three main steps: reducing noise, extracting edge of the original image, then SABC searching for circles in the entire edge-map by using a combination of three edge points as candidate circles. A function was defined to measure the existence of a candidate circle over the actual circle. Guided by this function, candidate circles in every species are evolved using the ABC algorithm so that multiple circles can fit into actual circles synchronously. Similarly, SABC was also employed by multi-template matching, so the multiple templates can be detected at once. Then, our proposed multi-object detection method, including multi-circle detection and

multi-template matching, were used as an efficient detector for TSR's recognition of modules fixed on NCT during rendezvousing and capturing process. Simulating different kinds of circumstances, experiments were performed on lots of images of modules on NCT. The results verified that our multi-object detection method outperformed other related methods in speed and accuracy, and has robustness to light intensity and noise.

The aim of this study is to show that by incorporating with the concept of species, the proposed SABC can effectively serve as an attractive method to successfully solve multi-peak optimization problem. On the basis of this, a multi-object detection method is proposed, and experiments using "shenzhou 8" and "Chang'e 3" space missions verified good applicability of our method for NCT. In future works, we will make effects to expand our method applied to detect more spacecrafts objects and implement more space missions which could enhance the operation and repair services using TSR system.

Acknowledgement

This work is supported by the Natural Science Foundation (NNSF) of China under Grant 61773282.

References：

[1] Weiss P, Leung W, Yung K L. Feasibility study for near-earth-object tracking by a piggybacked micro-satellite with penetrators[J]. Planetary & Space Science, 2010, 58(6):913-919.

[2] Wang X H. Space-in-orbit Service Technology and Its Development Status and Trends[J]. Satellite and Network, 2016(3):70-76.

[3] Jia P, Liu H Y, Li H. Analysis of Development of the German Track Mission Service System[J]. Chinese Spaceflight, 2016(6):24-29.

[4] Tipaldi M, Glielmo L. A Survey on Model-Based Mission Planning and Execution for Autonomous Spacecraft[J]. IEEE Systems Journal, 2017, PP(99):1-13.

[5] Xu W, Liang B, Li B, et al. A universal on-orbit servicing system used in the geostationary orbit[J]. Advances in Space Research, 2011, 48(1):95-119.

[6] Huang P, Wang D, Meng Z, et al. Adaptive Postcapture Backstepping Control for Tumbling Tethered Space Robot–Target Combination[J]. Journal of Guidance Control & Dynamics, 2015, 39(1):1-7.

[7] Wang D, Huang P, Cai J, et al. Coordinated control of tethered space robot using mobile tether attachment point in approaching phase[J]. Advances in Space Research, 2014, 54(6):1077-1091.

[8] Liang B, Du X D, Li C, et al. Research Progress of On-orbit Service for Space Robots Non-cooperative Spacecraft[J]. Robot, 2012, 34(2):242-256.

[9] Du X, Liang B, Xu W, et al. Pose measurement of large non-cooperative satellite based on collaborative

cameras[J]. Acta Astronautica, 2011, 68(11–12):2047-2065.

[10] Wan L, Han G, Shu L, et al. The Critical Patients Localization Algorithm Using Sparse Representation for Mixed Signals in Emergency Healthcare System[J]. IEEE Systems Journal, 2018, PP(99):1-12.

[11] Sansone F, Branz F, Francesconi A, et al. 2D Close-Range Navigation Sensor for Miniature Cooperative Spacecraft[J]. Aerospace & Electronic Systems IEEE Transactions on, 2014, 50(1):160-169.

[12] Opromolla R, Fasano G, Rufino G, et al. A Model-Based 3D Template Matching Technique for Pose Acquisition of an Uncooperative Space Object[J]. Sensors, 2015, 15(3):6360-6382.

[13] Bao G, Cai S, Qi L, et al. Multi-template matching algorithm for cucumber recognition in natural environment[J]. Computers & Electronics in Agriculture, 2016, 127:754-762.

[14] Wang D Z, Wu C H, Ip A, et al. Fast Multi-template Matching Using a Particle Swarm Optimization Algorithm for PCB Inspection[J]. 2008, 4974(4):365-370.

[15] Cai H, Zhu F, Wu Q, et al. A new template matching method based on contour information[J]. International Symposium on Optoelectronic Technology & Application Image Processing & Pattern Recognition, 2014, 9301:930109-930109-7.

[16] Jisung Yoo, Sung Soo Hwang, Seong Dae Kim, et al. Scale-invariant template matching using histogram of dominant gradients[J]. Pattern Recognition, 2014, 47(9):3006-3018.

[17] Cai J, Huang P, Zhang B, et al. A TSR Visual Servoing System Based on a Novel Dynamic Template Matching Method[J]. Sensors, 2015, 15(12):32152-32167.

[18] Chen L, Huang P, Cai J, et al. A Non-cooperative Target Grasping Position Prediction Model for Tethered Space Robot[J]. Aerospace Science & Technology, 2016, 58:571-581.

[19] Liu Y, Xie Z, Wang B, et al. A practical detection of non-cooperative satellite based on ellipse fitting[C]// IEEE International Conference on Mechatronics and Automation. IEEE, 2016:1541-1546.

[20] Pan L, Chu W S, Saragih J M, et al. Fast and Robust Circular Object Detection With Probabilistic Pairwise Voting[J]. IEEE Signal Processing Letters, 2011, 18(11):639-642.

[21] Dong N, Wu C H, Ip W H, et al. An opposition-based chaotic GA/PSO hybrid algorithm and its application in circle detection[J]. Computers & Mathematics with Applications, 2012, 64(6):1886-1902.

[22] Luo W, Sun J, Bu C, et al. Species-based Particle Swarm Optimizer enhanced by memory for dynamic optimization[J]. Applied Soft Computing, 2016, 47:130-140.

[23] Djekoune A O, Messaoudi K, Amara K. Incremental circle hough transform: An improved method for circle detection[J]. Optik - International Journal for Light and Electron Optics, 2017, 133:17-31.

[24] Zhang H, Wiklund K, Andersson M. A fast and robust circle detection method using isosceles triangles sampling[J]. Pattern Recognition, 2016, 54:218-228.

[25] Marco T D, Cazzato D, Leo M, et al. Randomized circle detection with isophotes curvature analysis[J]. Pattern Recognition, 2015, 48(2):411-421.

[26] Zelniker E E, Clarkson I V L. Maximum-likelihood estimation of circle parameters via convolution[J]. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 2006, 15(4):865-76.

[27] Frosio I, Borghese N A. Real-time accurate circle fitting with occlusions[J]. Pattern Recognition, 2008, 41(3):1041-1055.

[28] Gonzalez, R.C.; Woods, R.E. Digital Image Processing, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2002.

[29] Santos L R R D, Durand F R, Abrão T. Adaptive PID Scheme for OCDMA Next Generation PON Based on Heuristic Swarm Optimization[J]. IEEE Systems Journal, 2018, PP(99):1-11.

[30] Dong N, Wu C H, Ip W H, et al. An opposition-based chaotic GA/PSO hybrid algorithm and its application in circle detection[J]. Computers & Mathematics with Applications, 2012, 64(6):1886-1902.

[31] Luo W, Sun J, Bu C, et al. Species-based Particle Swarm Optimizer enhanced by memory for dynamic optimization[J]. Applied Soft Computing, 2016, 47:130-140.

[32] Xiao-Jun B I, Wang Y J. Niche artificial bee colony algorithm for multi-peak function optimization[J]. Systems Engineering & Electronics, 2011, 33(11):2564-2568.

[33] Deng T, Yao H, Du J. Improved artificial fish swarm hybrid algorithm for multi-peak function optimization [J].Journal of Computer Applications, 2012, 32 (10): 2904-2906. DOI: 10.3724/SP.J.1087.2012.02904

[34] Jia P, Tian X. Improved Invasive Weed Optimization Based on Adaptive Niche Algorithm[J]. Journal of Shanghai Dianji University, 2012.

[35] Dong N, Wu C H, Ip W H, et al. Species-Based Chaotic Hybrid Optimizing Algorithm and its Application in Image Detection[J]. Applied Artificial Intelligence, 2014, 28(7):647-674.

[36] Karaboga D. An Idea Based on Honey Bee Swarm for Numerical Optimization[J]. 2005.

[37] Qin Q D, Cheng S, Li L. A review on artificial bee colony algorithm[J].Journal of Intelligent Systems, 2014,9 (2):127-135.

[38] Zhang C Q, Zheng J G, Wang X. A review of research on bee colony algorithm[J]. Application Research of Computers, 2011, 28 (9): 3201-3205.DOI:10.3969/j.issn.1001-3695.2011.09.001

[39] Wang J Y. Research and application of artificial bee colony algorithm[J]. Journal of Harbin Engineering University, 2013.

[40] Chen Y, Lü S X, Wang M J, et al. A blind source separation method for chaotic signals based on artificial bee colony algorithm[J]. Acta Phys Sin, 2014, 63(23):0-0.

[41] Li J Q, Pan Q K. Solving the large-scale hybrid flow shop scheduling problem with limited buffers by a hybrid artificial bee colony algorithm[M]. Elsevier Science Inc. 2015.

[42] Kang F, Li J, Xu Q. Structural inverse analysis by hybrid simplex artificial bee colony algorithms[J]. Computers & Structures, 2009, 87(13–14):861-870.

[43] Ozturk C, Hancer E, Karaboga D. Dynamic clustering with improved binary artificial bee colony algorithm[J]. Applied Soft Computing, 2015, 28:69-80.

[44] Li B, Chiong R, Gong L G. Search-evasion path planning for submarines using the Artificial Bee Colony algorithm[C]// Evolutionary Computation. IEEE, 2014:528-535.

[45] Kitagawa W, Takeshita T. Optimum design of electromagnetic solenoid by using Artificial Bee Colony (ABC) algorithm[C]// Xxth International Conference on Electrical Machines. IEEE, 2012:1393-1398.

[46] Cai J, Huang P, Chen L, et al. An efficient circle detector not relying on edge detection[J]. Advances in Space Research, 2016, 57(11):2359-2375.

[47] Xu L, Oja E. Randomized Hough Transform (RHT): Basic Mechanisms, Algorithms, and Computational Complexities[J]. Cvgip Image Understanding, 1993, 57(2):131-154.

[48] Dong N, Wu C H, Ip W H, et al. Chaotic species based particle swarm optimization algorithms and its application in PCB components detection[J]. Expert Systems with Applications, 2012, 39(16):12501-12511.

[49] Chinese Academy of Sciences ,China National Space Administration , The Science and Application Center for Moon and Deepspace Exploration. Chang'e 3 data: Rover Panoramic camera[DB]. http://planetary.s3.amazonaws.com/data/change3/pcam.html, 2018-04-14.

[50] Polyu.edu.hk. Life on Mars? Award-wining PolyU device could dig up the answer[OL]. http://www.polyu.edu.hk/openingminds/en/story.php?sid=3, 2017-11-10.

[51] Polyu.edu.hk. PolyU 70th Anniversary[OL].

https://www.polyu.edu.hk/cpa/70thanniversary/memories_achievement10.html, 2017-11-03

[52] Polyu.edu.hk. The Hong Kong Polytechnic University - The Space Exploration Journey[OL].

https://www.polyu.edu.hk/web/filemanager/en/content155/1960/Appendix_Milest_ones_SpaceProjects_.pdf, 2017-11-24.