Research Article

An Enhanced Artificial Bee Colony-Based Support Vector Machine for Image-Based Fault Detection

Guijun Chen,[1] Xueying Zhang,[1] Zizhong John Wang,[1,2] and Fenglian Li[1]

[1]*College of Information Engineering, Taiyuan University of Technology, Taiyuan, Shanxi 030024, China*
[2]*Department of Mathematics and Computer Science, Virginia Wesleyan College, Norfolk, VA 23502, USA*

Correspondence should be addressed to Xueying Zhang; tyzhangxy@163.com

Fault detection has become extremely important in industrial production so that numerous potential losses caused from equipment failures could be saved. As a noncontact method, machine vision can satisfy the needs of real-time fault monitoring. However, image-based fault features often have the characteristics of high-dimensionality and redundant correlation. To optimize feature subsets and SVM parameters, this paper presents an enhanced artificial bee colony-based support vector machine (EABC-SVM) approach. The method is applied to the image-based fault detection for the conveyor belt. To improve the optimized capability of original ABC, the EABC algorithm introduces two enhanced strategies including the Cat chaotic mapping initialization and current optimum based search equations. Several UCI datasets have been used to evaluate the performance of EABC-SVM and the experimental results show that this approach has better classification accuracy and convergence performance than the ABC-SVM and other ABC variants-based SVM. Furthermore, the EABC-SVM can achieve a significant detection accuracy of 95% and reduce the amount of features about 65% in the conveyor belt fault detection.

1. Introduction

In industrial production, to improve the reliability and reduce the possible loss due to equipment failures, fault detection has become extremely important. For example, in modern coal mining enterprises, the tear fault of conveyor belt has often occurred due to the impact of falling sharp materials in operation [1]. If the fault can be detected early, the damage of transmission devices caused by belts breakdown can be minimized or even avoided. As a noncontact method, the machine vision can capture the rich image-based information of detected regions in real time by the CCD camera deployed on the equipment. Therefore, image-based fault detection can satisfy the needs of modem industrial production including manufacturing processes [2], electrified railways [3], and defect detection [4].

Generally, the procedure of image-based fault detection consists of image acquisition, image preprocessing, feature extraction and analysis, and alarm control. After the images are acquired, the key step is feature extraction and analysis.

The extracted features usually include colors, textures, shapes, points, and edges, which are combined into a multidimensional feature vector. However, existing methods cannot fully utilize the feature vector due to its complex characteristics including the high-dimensionality and redundant correlation [5]. Feature selection is an effective approach to eliminate irrelevant or redundant features by picking a feature subset from the original features [6]. The selection methods of feature subset include forward selection, backward elimination, and bidirectional search [7]. These methods usually start from an initial subset (empty set or complete feature set) and generate a new subset by adding or discarding a feature. Then, the generated feature subset is evaluated by the classification accuracy. Thus, an efficient global search technique is needed for the large feature space.

At present, some intelligent optimization algorithms have been proposed for feature selection, such as the genetic algorithm (GA) [8], particle swarm optimization (PSO) [9], ant colony optimization (ACO) [10], and artificial bee colony (ABC) [6]. These methods can select some informative or

actual useful feature variables and improve efficiency and accuracy of the data analysis. Moreover, due to few control parameters and superior optimization performance, the ABC algorithm has attracted much attention [11, 12]. In [13], a hybrid approach based on ABC algorithm and artificial neural networks (ANN) was presented to select feature subset effectively. Schiezaro and Pedrini [6] proposed a feature selection method using the ABC algorithm to classify different UCI datasets and the selected feature set could provide better classification accuracy. In [14], a feature selection technique based on the ABC and k-Nearest Neighbor (k-NN) was employed for image steganalysis. Compared with ANN and k-NN classifier, support vector machine (SVM) is a more powerful classification method due to its excellent classification accuracy and generalization performance [15, 16].

On the other hand, some variants of ABC algorithm [17] have been proposed to improve the global search performance and the convergence speed, which mainly focus on the population initialization and the solution search strategy [18–24]. Zhu and Kwong [21] proposed the gbest-guide ABC (GABC) algorithm by incorporating the information of global best solution into the solution search equation to improve the exploitation. Gao and Liu [22] presented a modified ABC (MABC) algorithm by introducing the chaotic and opposition-based initialization and the best solution of the previous iteration. To improve the exploitation and keep the exploration of ABC, Zhang and Liu [19] proposed a novel ABC (NABC) algorithm by incorporating the global best solution and the random solution into the search equations of the onlookers and the employed bees, respectively. He et al. [20] introduced a modified ABC (SDABC) algorithm from three aspects including the search space division initialization, disruptive selection strategy, and the improved scout bee phase.

In the paper, an enhanced artificial bee colony-based support vector machine (EABC-SVM) classifier is proposed for image-based fault detection. To improve the convergence of the ABC, two enhanced strategies including the Cat chaotic mapping initialization and the current optimum based search equations are presented in the enhanced ABC (EABC) algorithm. The EABC algorithm is employed to optimize the feature subset and parameters of SVM. In order to assess EABC-SVM's capability, six benchmark datasets from UCI are first used. And then the EABC-SVM is applied to detect image-based tear fault of the conveyor belt.

This paper is organized as follows. Fundamentals about the ABC algorithm and SVM are introduced in Section 2. In Section 3, the proposed EABC-SVM classification method is given in detail. In Section 4, experimental results are presented to demonstrate the capability of the EABC-SVM using six benchmark datasets from UCI and show the detection performance applied to the image-based fault detection for conveyor belt. Finally, the conclusion is drawn in Section 5.

2. Fundamentals

This section will briefly introduce the fundamentals about artificial colony bee algorithm and support vector machine.

2.1. Artificial Colony Bee (ABC) Algorithm. Artificial bee colony (ABC) algorithm [25] is an intelligent system inspired by foraging behavior of a bee colony and widely used to solve continuous numerical optimization problems. In the ABC algorithm, the bee colony consists of three kinds of bees: employment bees, onlooker bees, and scout bees. During the process of optimization, the position of a food source represents a possible solution to the optimization problem. The operation procedure of ABC is an iterative process, as shown in Algorithm 1, which refers to repeated searches for the solution with employment bees, onlooker bees, and scout bees until the maximum cycle number, MAXcycles, or the allowable minimum error reaches [26, 27]. The details of the four phases are described as follows.

(1) The Initialization Phase. The population of the ABC algorithm is initialized randomly as

$$x_{i,j} = x_{\min,j} + \text{rand}(0,1)\left(x_{\max,j} - x_{\min,j}\right), \qquad (1)$$

where $i = 1, 2, \ldots, SN$ is the number of population, $j = 1, 2, \ldots, D$ is the dimension of population, rand$(0, 1)$ is a random number in $[0, 1]$, and $x_{\min,j}$ and $x_{\max,j}$ are the lower and upper bounds of the jth optimization parameter, respectively.

After the population initialization, food sources are evaluated by the fitness function. The greater the fitness value, the better quality the food source. For the maximum optimization problem, the fitness function is the objective function obj$_i$. For the minimum optimization problem, the fitness function is defined as

$$\text{fit}_i = \begin{cases} \dfrac{1}{1 + \text{obj}_i}, & \text{obj}_i \geq 0 \\ 1 + |\text{obj}_i|, & \text{obj}_i < 0. \end{cases} \qquad (2)$$

(2) The Employed Bee Phase. In this phase, each employed bee generates a new food source with the following search equation:

$$v_{i,j} = x_{i,j} + \varphi_{i,j}\left(x_{i,j} - x_{k,j}\right), \qquad (3)$$

where $k \in \{1, 2, \ldots, SN\}$, $k \neq i$, and $\varphi_{i,j}$ is a random number in $[-1, 1]$.

When all employed bees complete the search for new food sources, the fitness values of new food sources are calculated and compared to the old ones according to the greedy selection mechanism of

$$v_i = \begin{cases} v_i, & \text{fit}(v_i) > \text{fit}(x_i) \\ x_i, & \text{fit}(v_i) \leq \text{fit}(x_i). \end{cases} \qquad (4)$$

(3) The Onlooker Bee Phase. After all employed bees complete the updated process, they share their information about

The operation procedure of ABC algorithm
- - Initialization phase - -
(1) Initialize the population of solutions and assign the population to employed bees
(2) **while** (cycle = MAXcycles) **do**
        - - Employed bee phase - -
(3)   **for** $i = 1$ to SN **do**
(4)   Produce a new solution v_i for employed bees and calculate its fitness value
(5)   Apply the greedy selection mechanism between v_i and x_i, select the better one
(6)   If the solution x_i does not update, the non-updated number $trial_i = trial_i + 1$; otherwise $trial_i = 0$
(7)   **end for**
        - - Onlooker bee phase - -
(8)   Calculate the selection probability p_i
(9) $t = 0, i = 1$
(10)   **while** ($t < SN$) **do**
(11)   **if** random $< p_i$ **then**
(12) $t = t + 1$
(13)   Produce a new solution v_i for onlooker bee of the solution x_i and calculate its fitness value
(14)   Apply the greedy selection mechanism between v_i and x_i, select the better one
(15)   If the solution x_i does not update, $trial_i = trial_i + 1$; otherwise $trial_i = 0$
(16)   **end if**
(17) $i = i + 1$
(18)   If $i = SN + 1, i = 1$
(19)   **end while** ($t = SN$)
        - - Scout bee phase - -
(20)   **if** $trial_i >$ LIMIT **then**
(21)   Replace x_i with a new random solution v_i
(22)   **end if**
(23)   Memorize the current optimal solution
(24)   cycle = cycle + 1
(25) **end while** (cycle = MAXcycles)

ALGORITHM 1: The pseudocode of ABC algorithm.

the amounts and the positions of food sources with onlooker bees. An onlooker bee evaluates all food sources from employed bees and selects a good one to update the new solution based on the probability calculated with the roulette wheel selection scheme in

$$p_i = \frac{\text{fit}_i}{\sum_{i=1}^{SN} \text{fit}_i}, \tag{5}$$

where fit_i is the fitness value of the solution i.

(4) The Scout Bee Phase. If the food source x_i cannot be further updated through a predetermined number of trials LIMIT, the employed bee is converted to scout bee, which abandons the old food source and searches the new one using (1).

Among the operation procedure, there are some drawbacks in the ABC algorithm. Firstly, due to the randomness of the initialization equation (1), the diversity of initial population may be limited, which would affect the convergence efficiency. Secondly, in the phases of employed bee and onlooker bee, the new solution is searched out around the current solution according to (3) by moving the current solution near to or away from another random solution of the population. However, the generation process does not

consider the current optimal solution as a guidance of the search. So the search strategy is considered to have good exploration but poor exploitation capacity [21]. To overcome these drawbacks, an enhanced ABC algorithm is proposed in Section 3.1 in detail.

2.2. Support Vector Machine (SVM). The SVM classifier is briefly described as follows. Given the training set $T = \{(x_1, y_1), \ldots, (x_l, y_l)\}$, where $x_i \in R^n$ and $y_i \in \{-1, 1\}$, $i = 1, \ldots, l$, the goal is to find a separating hyperplane and classify the training data into two categories accurately based on the principle of margin maximization [28]. The following optimization problem is constructed:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{l} \xi_i \\ \text{s.t.} \quad & y_i \left((w \cdot \phi(x_i)) + b \right) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \ i = 1, \ldots, l, \end{aligned} \tag{6}$$

where w and b are the normal vector and the offset of the separating hyperplane, respectively; $C > 0$ is the penalty parameter of error term ξ; $\phi(\cdot)$ is the mapping function that makes sample x_i mapped onto a high-dimension space.

Equation (6) can be translated to the following Lagrange dual problem:

$$\min_{\alpha} \quad \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j \left(\phi(x_i) \cdot \phi(x_j) \right) - \sum_{j=1}^{l} \alpha_j$$

$$\text{s.t.} \quad \sum_{i=1}^{l} y_i \alpha_i = 0,$$

$$0 \le \alpha_i \le C, \ i = 1, \ldots, l. \tag{7}$$

The Lagrange multipliers $\alpha_i \in (0, C)$ are obtained from (7), and then we can construct the classification decision function $f(x)$ as follows:

$$f(x) = \text{sign} \left(\sum_{i=1}^{l} \alpha_i y_i \left(\phi(x_i) \cdot \phi(x) \right) + b \right). \tag{8}$$

Generally, $K(x_i, x) = (\phi(x_i) \cdot \phi(x))$ is defined as the kernel function. This paper mainly discusses the Gaussian kernel function because the Gaussian kernel can approximate most kernel functions if the kernel parameter is chosen appropriately [29]. Its form is expressed as follows:

$$K(x_i, x_j) = \exp \left(-\gamma \|x_i - x_j\|^2 \right), \tag{9}$$

where γ is the kernel parameter.

The parameters of SVM with the Gaussian kernel function refer to the penalty parameter C and the kernel parameter γ, which should be optimized by the user. The optimal SVM parameters have important influence on the classification performance.

3. Methodology

In this section, the enhanced artificial bee colony-based support vector machine (EABC-SVM) classifier is proposed, which can effectively improve the classification accuracy by feature selection and parameters optimization for SVM simultaneously.

3.1. Enhanced Artificial Bee Colony. To solve the above drawbacks of the ABC algorithm, two enhanced strategies are introduced as follows.

(1) The Cat Chaotic Mapping Based Initialization. Similar to other intelligent algorithms, the initialized population plays an important role in convergence to the global optimal solution of the ABC algorithm. As one of the useful initialization techniques, the chaotic mapping method can generate the random sequence with the ergodicity and nonperiodicity. Currently, the chaotic maps mainly include the Logistic mapping function, the Tent mapping function, and the Cat mapping function. According to the analysis on chaotic characteristics of the mapping functions [30], the Cat mapping function has better ergodic uniformity and does not easily fall into the minor cycle compared with other mapping functions. So this paper will employ the Cat mapping function to
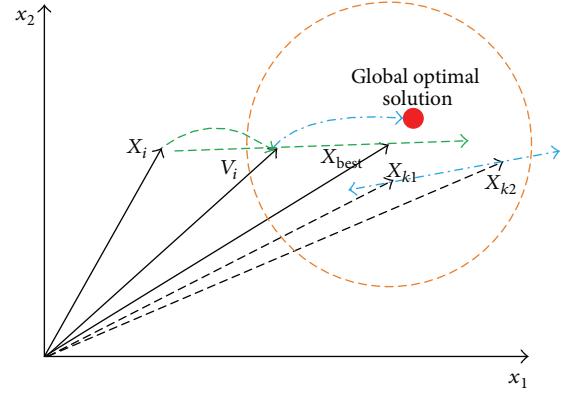

FIGURE 1: The principle of the current optimum based search.

initialize the population. The Cat mapping function is shown as

$$z_{i+1} = (z_i + y_i) \bmod 1$$

$$y_{i+1} = (z_i + 2y_i) \bmod 1, \tag{10}$$

where $i = 0, 1, 2, \ldots, l$, l is the iteration number of the Cat chaotic sequence, $z_i, y_i \in (0, 1)$, and mod is the modulus operator. When $z_0 \in (0.25, 0.5, 0.75)$, $y_0 \notin (0.25, 0.5, 0.75)$, and vice versa. And then the new initialized equation of population is expressed as

$$x_{i,j} = x_{\min,j} + y_{i,j} \left(x_{\max,j} - x_{\min,j} \right). \tag{11}$$

(2) The Current Optimum Based Search Equations. To improve the search efficiency, many studies have been conducted to modify the search equation, most of which introduced the current optimal solution to search process [21–23]. However, since the guidance of the current optimal solution, the search process may cause an "oscillation" phenomenon [24]. Therefore, to overcome the defect of the above method, we introduce two search equations for employed bees and onlooker bees, respectively, in (12) and (13). The equations can take full advantage of the current optimal solution and the random solutions of population based on the characteristics of different search phases:

$$v_{i,j} = x_{i,j} + 2 \cdot \text{rand}(0, 1) \cdot \left(x_{\text{best},j} - x_{i,j} \right), \tag{12}$$

$$v_{i,j} = x_{i,j} + \varphi_{i,j} \left(x_{k1,j} - x_{k2,j} \right), \tag{13}$$

where $x_{\text{best},j}$ is the current optimal solution, the indices $k1$ and $k2$ are random integers chosen from $\{1, 2, \ldots, SN\}$ that are different from the index i, and $\varphi_{i,j}$ is a random number in $[-1, 1]$.

The principle of the current optimum based search is shown in Figure 1. In the employed bee phase, each employed bee can find out a better solution around the current optimal solution x_{best} with (12). That is, all employed bees will move toward the space centered on x_{best}, as the dotted circle in
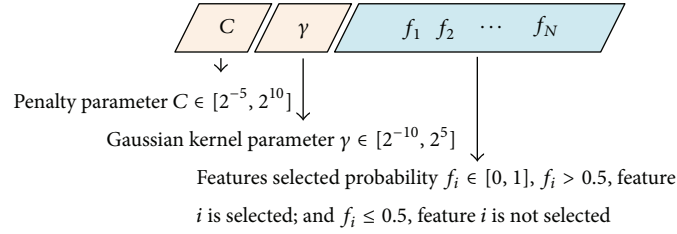
FIGURE 2: Optimization parameters.

Figure 1. In the onlooker bee phase, two different current solutions, x_{k1} and x_{k2}, are randomly selected and must be located in the dotted circle because they come from the employed bees. In this way, the onlooker bees can complete the search of the global optimal solution in a fairly small solution space with (13). Therefore, the different foraging processes of employed bees and onlooker bees can be simulated well with the two search equations. As a result, employed bees can implement the directional search with the guidance of current optimal solution, and onlooker bees will further complete the wide range search in the confined solution space provided by employed bees. A detailed convergence analysis of the EABC algorithm based on the Markov chain theory is presented in the Appendix.

3.2. EABC-SVM Classifier

3.2.1. Classifier Initialization. The classifier initialization involves two parts: the optimization parameters and the fitness function. The optimization parameters are expressed as the combination of SVM parameters (C, γ) with the Gaussian kernel function and the selection probability on each feature, as shown in Figure 2.

The fitness function is used to evaluate the selected feature subsets and SVM parameters. Two factors that are usually considered include the classification accuracy and the number of selected features [8]. The fitness function is defined as (14). When the classification accuracy is high and the number of selected features is small, the fitness value will be large:

$$\text{fit} = w \cdot \text{Acc} + (1 - w) \cdot \left(1 + \sum_{i=1}^{n_f} F_i \right). \tag{14}$$

In (14), w is a predefined weight, Acc is the classification accuracy, and n_f is the total number of features; "$F_i = 1$" represents that feature i is selected while "$F_i = 0$" represents that feature i is not selected. The weight w can be adjusted to balance between classification accuracy and feature selection. In general, w can be chosen from 0.7 to 0.9 according to the different datasets.

3.2.2. Classifier Architecture. The architecture of the proposed EABC-SVM classifier is shown in Figure 3. The main implemented steps can be described as follows.
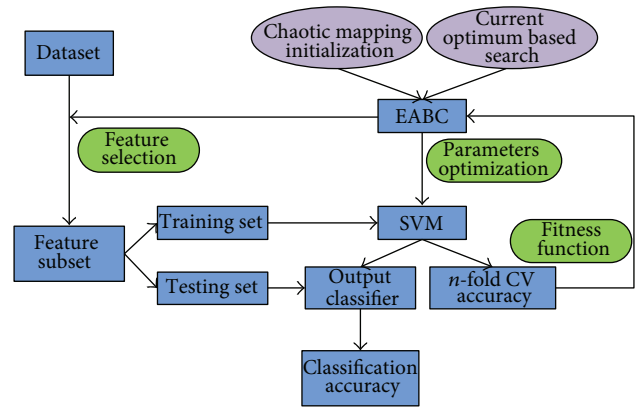

FIGURE 3: The architecture of proposed EABC-SVM system.

Step 1 (preparing the dataset). The feature dataset discards the feature elements with the selected probability of "$f_i < 0.5$" and becomes the actually used feature subset. And then the feature subset is divided into the training set and the testing set. Meanwhile, the training set is used to train the SVM model by n-fold cross validation (CV).

Step 2 (training the classifier). The SVM classifier is trained with the training set and the initialized parameters (C, γ). The optimal classifier parameters and feature subset are evaluated with the fitness function.

Step 3 (testing the classifier). The testing set is used to verify the trained SVM classifier and feature subset for their performance of classification. Finally, the output SVM classifier and the regulation of feature selection are obtained.

4. Experimental Results and Analysis

The numerical experiments were performed in a PC with Intel Pentium (R) G630, 2.7 GHz CPU, 2G RAM, 32-bit Windows 7 operating systems. The development environment is MATLAB R2010a. The simulation toolbox of SVM is Libsvm [31]. In the experiment, the SVM classifier with the Gaussian kernel function was used, in which the searching range of parameter C is $[2^{-5}, 2^{10}]$ and that of parameter γ is $[2^{-10}, 2^5]$.

4.1. UCI Datasets. To examine the capability of the proposed EABC-SVM, we used the UCI benchmark datasets [32]

for classification and compare the results with the original ABC and other four ABC variants including GABC [21], MABC [22], NABC [19], and SDABC [20]. For the ABC, GABC, MABC, NABC, SDABC, and EABC algorithms, the parameters were set as follows: size of employed bee SN = 50, size of onlooker bee SN = 50, maximum number of iterations MAXcycles = 30, and control parameters LIMIT = 10. For the GABC algorithm, C = 1.5; for the MABC algorithm, K = 300. The experiment adopted six real world datasets: Breast Cancer Wisconsin (WDBC), Ionosphere, Musk1, Sonar, Vehicle, and Wine. Their number of classes, number of instances, and number of features (dimensions) are shown in Table 1.

For the above six different datasets, the range of each feature value was scaled to the range of $[-1, +1]$. In the experiment, the 10-fold cross validation method was used to divide the dataset into the training set and the testing set. Each experiment was run 20 times and then the average accuracy was taken.

Table 2 shows the results of optimal accuracy (Opt Acc), average accuracy (Ave Acc), standard deviation of accuracy (Acc SD), p value for t-test, and selected feature dimensions (average ± standard deviation) from the experiment of each dataset with the comparison of ABC-SVM and EABC-SVM. The EABC-SVM method yields the higher classification accuracy in six datasets, especially in Musk1 and Vehicle. Meanwhile, the standard deviation of accuracy of EABC-SVM method is smaller than that of ABC-SVM method in each dataset, which means the EABC-SVM is more stable. Moreover, the independent sample t-test was implemented to verify whether the difference is significant or not. Through the p value for t-test, the proposed EABC-SVM method outperforms ABC-SVM for six datasets under 95% confidence levels. In addition, the EABC-SVM method can select fewer features with smaller standard deviation than the ABC-SVM. Due to fewer features and higher accuracy, EABC-SVM is capable of selecting more appropriate features with the global search.

In order to illustrate convergence of EABC-SVM, the convergence curves are showed in Figure 4. From the figure, we see that EABC-SVM can reach optimal solution with fewer iterations than ABC in six datasets. Meanwhile, for each dataset, the initial solution in EABC-SVM is superior to that in ABC-SVM, which further verifies the ergodicity of the Cat chaotic mapping function.

Tables 3 and 4 show the experimental comparisons including the classification accuracy and the selected feature dimensions (average ± standard deviation) of our proposed EABC-SVM with other ABC variants-SVM. From Table 3, we can find that EABC-SVM yields higher classification accuracy in 6 datasets than other four ABC variants-SVM. For the feature selection, as shown in Table 4, EABC-SVM can produce a moderate-sized feature subset, while NABC-SVM has fewer features in five datasets. Thus, the proposed method can perform better feature selection with high classification accuracy, which is suitable for the redundant feature reduction and classification application, such as the image-based dataset. In particular, when the redundant dimensions

Table 1: UCI datasets.

Number	Dataset	Classes	Instances	Dimensions
1	WDBC	2	569	30
2	Ionosphere	2	351	34
3	Musk1	2	476	166
4	Sonar	2	208	60
5	Vehicle	4	846	18
6	Wine	3	178	13

of feature vectors are very large, the reduced effect is more apparent and the classification accuracy would be improved greatly.

4.2. Image-Based Conveyor Belt Fault Dataset. In this section, we use EABC-SVM to implement feature selection and parameters optimization for SVM, and then the fault detection of the conveyor belt is used for empirical analysis. Firstly, the visual monitoring system of conveyor belt was established to acquire the image-based belt fault signal and create a fault feature dataset. The schematic diagram of monitoring system is shown as in Figure 5. A series of CCD cameras and light sources were installed at the bottom of the conveyor belt for image acquisition, and then the images were transmitted to the industrial computer for online fault detection. According to the failure analysis of conveyor belt in [1], the fault dataset is classified into four different types, which include normal, rope fracture, scratches, and tear as shown in Figure 6.

The dataset comprises 180 gray belt images with size of 640×480 and each class has 45 images. Due to the irregular shape of the scratches or tear, we extracted the gray histogram features and texture features as the basis of fault detection. Figure 7 illustrates the flowchart of image preprocessing, feature extraction, and analysis. First, the belt image was enhanced with the Gaussian filter, and then the belt fault features were extracted. The details are shown as follows.

(i) *Gray Histogram.* The image in the tearing part of belt has a much lower gray value than that in the background. So we chose the first 30 grayscale histogram values as features. Figure 8 shows the grayscale histogram for each image of Figure 6. From Figure 8, we can find that the first 30 grayscale values have obvious difference corresponding to the different fault images.

(ii) *Texture Features.* As a good texture descriptor, gray level cooccurrence matrix (GLCM) can reflect the grayscale difference of the direction and adjacent interval effectively. As it is described in [33], the GLCM in this paper was calculated in four different directions of $0°$, $45°$, $90°$, and $135°$ and the interpixel distance d = 4 to improve the computational efficiency. The corresponding energy, entropy, contrast, and correlation were then selected to describe the belt texture.

On the above feature extraction process, the size of feature vector is $30 + 16 = 46$. Next, the EABC-SVM method

TABLE 2: Experimental results for EABC-SVM and ABC-SVM method.

Dataset	Method	Opt Acc	Ave Acc	Acc SD	p value for t-test[*]	Selected dimensions
WDBC	ABC-SVM	96.13	94.57	1.05	<0.001	13.70 ± 1.73
	EABC-SVM	98.24	97.25	0.71		12.15 ± 1.42
Ionosphere	ABC-SVM	95.44	93.93	1.24	<0.001	14.95 ± 1.80
	EABC-SVM	96.30	95.63	0.38		12.55 ± 0.86
Musk1	ABC-SVM	92.65	87.20	2.98	<0.001	83.40 ± 3.17
	EABC-SVM	95.59	93.04	1.70		76.30 ± 3.10
Sonar	ABC-SVM	89.90	87.86	1.20	<0.001	29.00 ± 2.73
	EABC-SVM	93.27	91.54	0.78		26.80 ± 1.69
Vehicle	ABC-SVM	80.26	77.06	2.23	<0.001	12.00 ± 1.52
	EABC-SVM	85.58	85.30	0.50		10.50 ± 0.81
Wine	ABC-SVM	98.88	95.86	1.73	<0.001	6.20 ± 1.08
	EABC-SVM	100	99.07	1.04		5.95 ± 0.59

[*]Confidence level: 95%.

TABLE 3: Classification accuracy of the proposed EABC-SVM and other ABC variants-SVM.

Dataset	Methods				
	EABC-SVM	GABC-SVM	MABC-SVM	NABC-SVM	SDABC-SVM
WDBC	97.25 ± 0.71	94.92 ± 2.06	96.09 ± 0.94	96.22 ± 0.97	96.01 ± 1.06
Ionosphere	95.63 ± 0.38	94.27 ± 1.43	94.52 ± 0.48	94.47 ± 2.15	94.00 ± 1.60
Musk1	93.04 ± 1.70	90.18 ± 2.58	92.27 ± 1.82	91.35 ± 1.78	87.69 ± 2.15
Sonar	91.54 ± 0.78	87.98 ± 2.03	88.70 ± 2.01	88.25 ± 2.40	88.10 ± 0.92
Vehicle	85.30 ± 0.50	78.34 ± 2.55	79.41 ± 2.74	81.53 ± 2.36	78.84 ± 2.23
Wine	99.07 ± 1.04	98.74 ± 0.59	98.79 ± 0.67	98.57 ± 0.70	98.55 ± 0.99

TABLE 4: Selected feature dimensions of the proposed EABC-SVM and other ABC variants-SVM.

Dataset	Methods				
	EABC-SVM	GABC-SVM	MABC-SVM	NABC-SVM	SDABC-SVM
WDBC	12.15 ± 1.42	10.10 ± 2.34	10.10 ± 1.58	8.70 ± 2.00	19.30 ± 2.66
Ionosphere	12.55 ± 0.86	14.20 ± 2.36	13.75 ± 3.75	13.95 ± 3.00	12.40 ± 1.98
Musk1	76.30 ± 3.10	78.20 ± 3.76	79.60 ± 4.03	73.5 ± 4.41	89.45 ± 4.42
Sonar	26.80 ± 1.69	27.05 ± 2.25	26.85 ± 2.73	25.05 ± 2.65	35.20 ± 2.20
Vehicle	10.50 ± 0.81	11.80 ± 1.50	11.90 ± 1.61	10.00 ± 1.22	13.35 ± 1.68
Wine	5.95 ± 0.59	5.00 ± 1.14	4.70 ± 1.05	4.35 ± 0.65	8.20 ± 1.21

was employed to reduce the number of features and detect the fault types. The 180 samples were divided into training set (25 × 4 = 100) and testing set (20 × 4 = 80) randomly. Moreover, the 5-fold cross validation method was used in the training phase. Table 5 gives the final results of EABC-SVM and another ABC-related method about the feature selection and classification accuracy. Results in Table 5 show that EABC-SVM can achieve high classification accuracy of 95.0%, while the MABC-SVM yields the same classification accuracy. Moreover, the proposed EABC-SVM selected merely 16 features and the obtained features only cost 34.78% (16/46) of the memory needed for the original 46 features. The selected array of feature variables includes 4, 8, 13, 15, 21, 22, 25, 32, 34, 37, 38, 41, 43, 44, 45, and 46, in which 7 features belong to

TABLE 5: Fault detection results of EABC-SVM and other ABC-related methods.

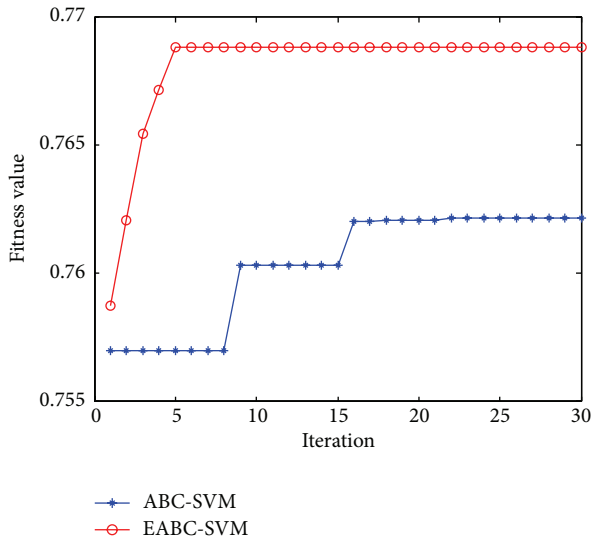| Methods	C	γ	Selected dimensions	Accuracy (%)
ABC-SVM	540.3200	2.1327	21	92.50
GABC-SVM	515.1092	3.6465	17	93.75
MABC-SSVM	544.7164	2.4870	23	95.00
NABC-SVM	805.7588	1.8308	12	91.25
SDABC-SVM	855.1585	2.4785	12	91.25
EABC-SVM	267.3419	1.1891	16	95.00

gray histogram and 9 features belong to GLCM texture features.
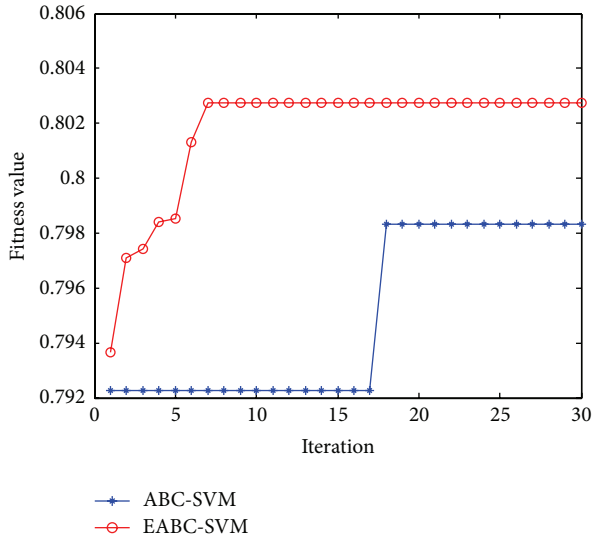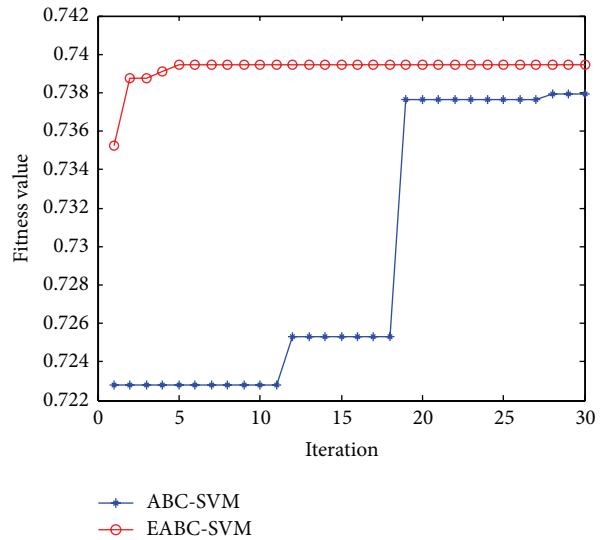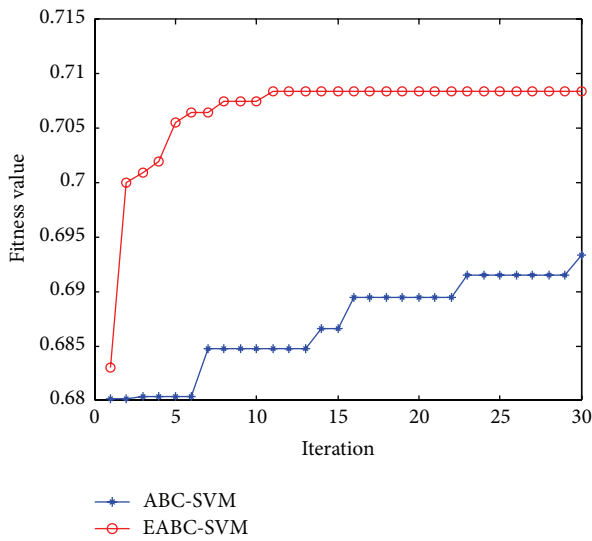
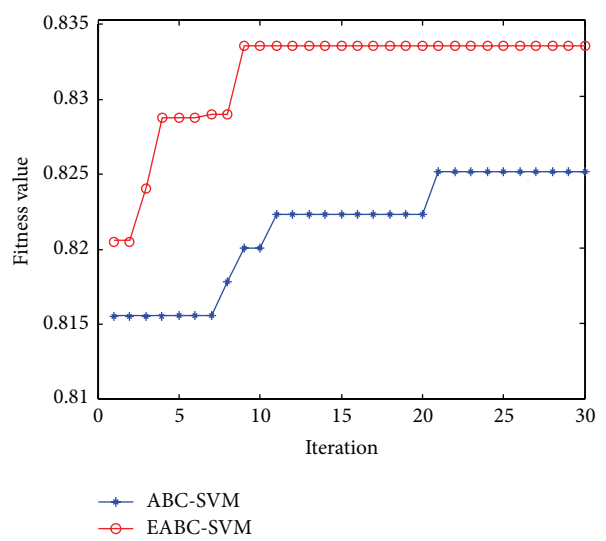(a) WDBC
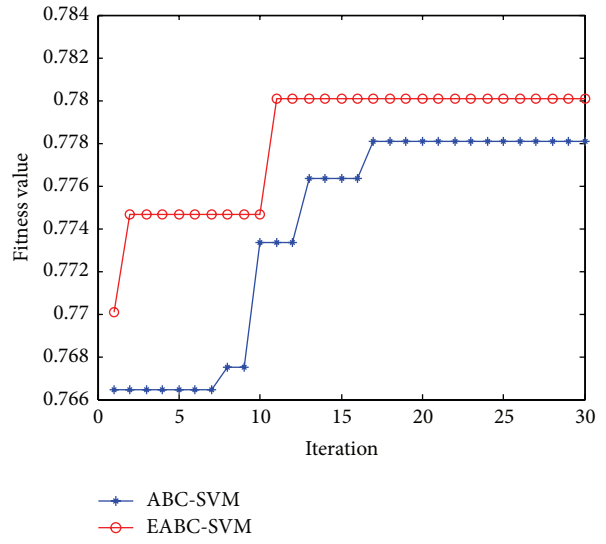
(b) Ionosphere

(c) Musk1

(d) Sonar

(e) Vehicle

(f) Wine

FIGURE 4: Convergence curves for ABC-SVM and EABC-SVM method.

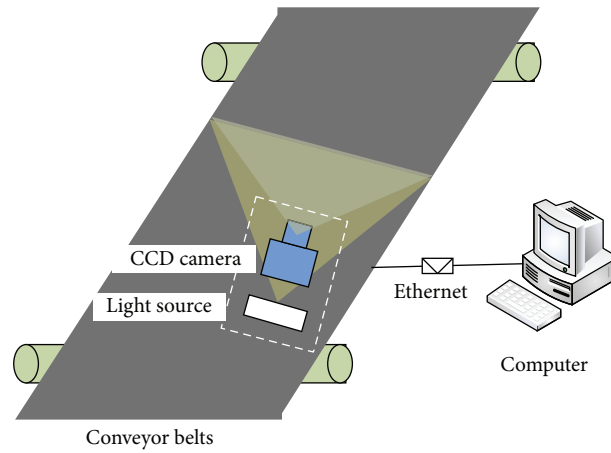FIGURE 5: Schematic diagram of monitoring system.

(a) Normal

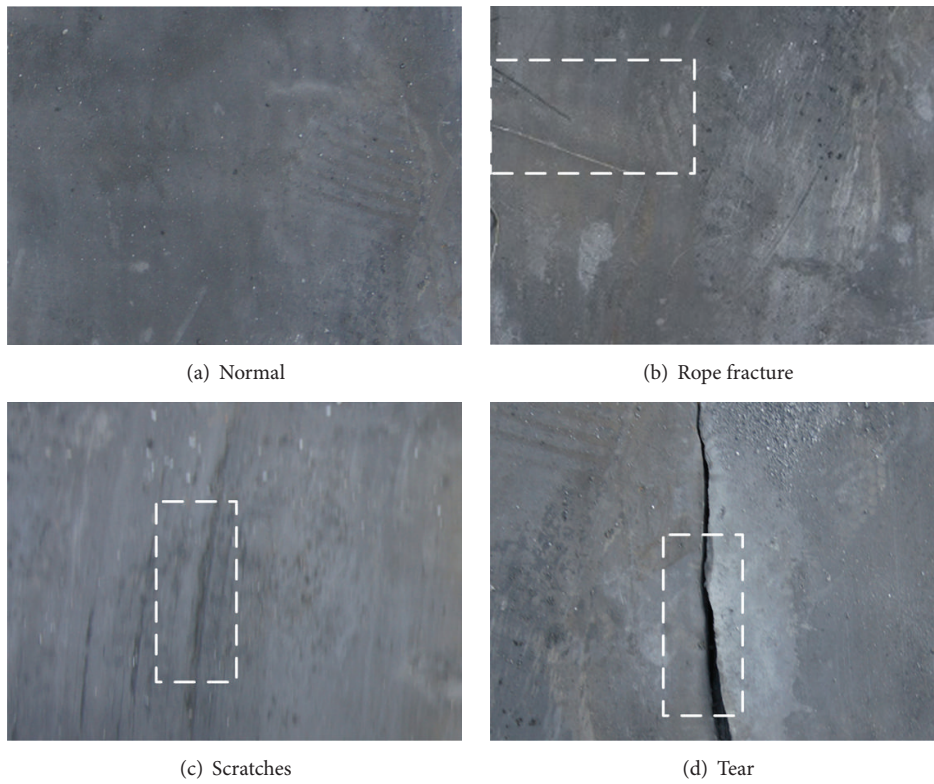(b) Rope fracture

(c) Scratches

(d) Tear

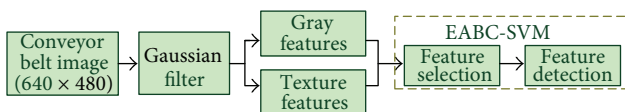FIGURE 6: Fault types of conveyor belt.

FIGURE 7: Flowchart of feature extraction and analysis.

The confusion matrix of fault detection for EABC-SVM is shown in Figure 9. Each row represents the predicted class (output), and each column represents the actual class (target). The number in ith row and jth column represents the rate of samples whose target is the ith class that is classified as jth class. From Figure 9, we find that the faults of rope fracture and tear can be detected to be completely correct as 100% accuracy, while other two types between normal and scratches would be easily misclassified. Because of nonuniform illumination and dusts in real environment, the detection results of these two classes are influenced at different levels. So how the two classes can be recognized effectively remains one of our future tasks.
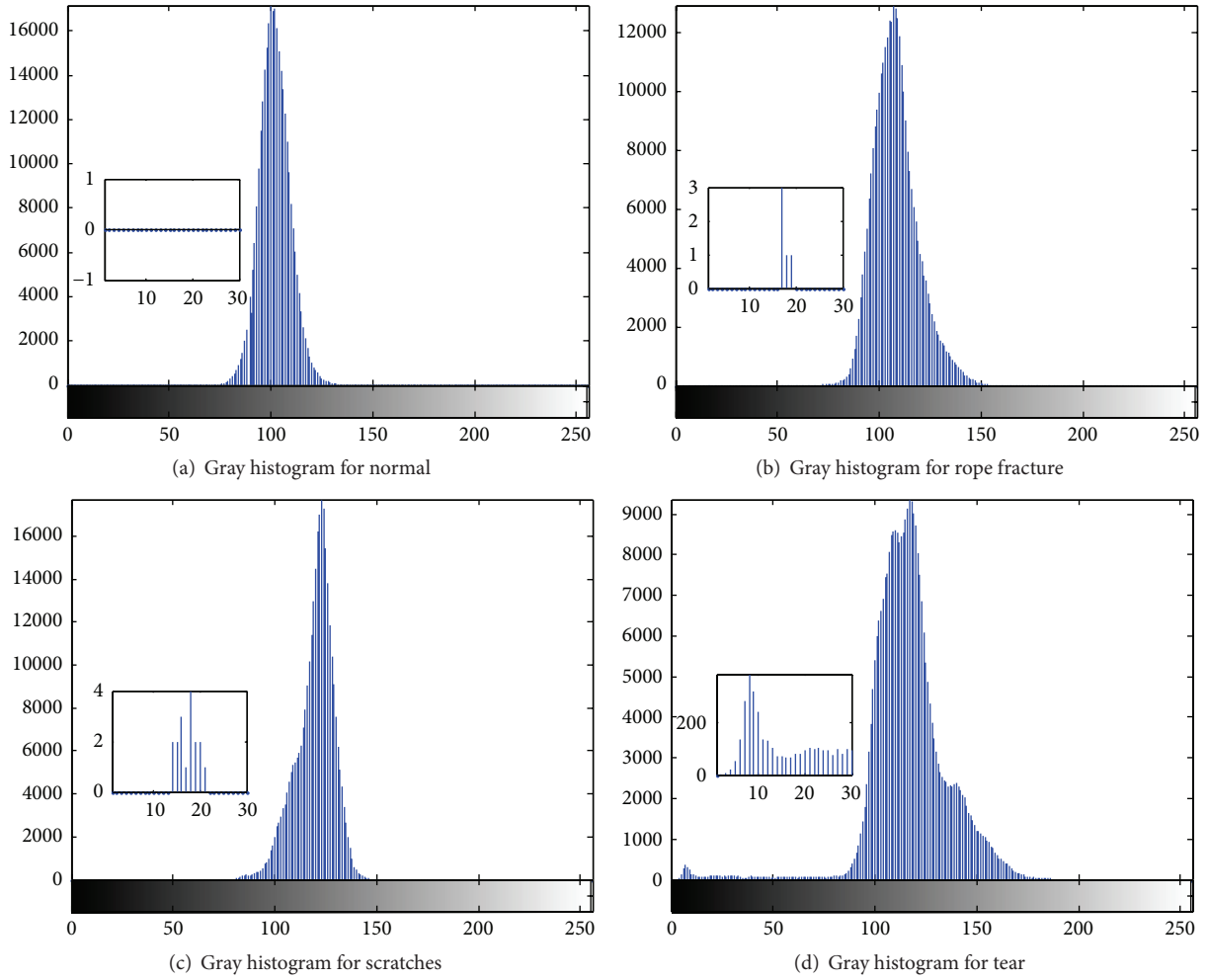
(a) Gray histogram for normal

(b) Gray histogram for rope fracture

(c) Gray histogram for scratches

(d) Gray histogram for tear

FIGURE 8: Grayscale histogram for each image of Figure 6.

5. Conclusions

In this paper, we proposed an enhanced ABC (EABC) algorithm to search for the optimal feature subset and parameters of SVM simultaneously. The experimental results demonstrate that the proposed EABC-SVM approach has better classification accuracy and convergence performance than ABC-SVM and other four ABC variants-SVM on six UCI datasets. Furthermore, the EABC-SVM approach was applied to the image-based fault detection for the conveyor belt. Through the combination of gray histogram and GLCM texture features, the EABC-SVM can achieve a significant classification accuracy of 95% and reduce the amount of feature storage about 65%.

The proposed EABC algorithm has the following characteristics: (1) the algorithm possesses two enhanced strategies including the Cat chaotic mapping initialization and current optimum based search equations to improve the convergence speed and global optimization performance; (2) the algorithm is employed to optimize the feature subset and parameters for SVM simultaneously, which can improve overall classification accuracy and reduce the computation complexity.

FIGURE 9: Confusion matrix of EABC-SVM for fault detection.

Further research plan is to (1) reduce the belt image noise and optimize fault features to increase the detection accuracy and robustness and (2) take some other classification

problems with a high number of feature variables to test and extend the proposed EABC-SVM approach.

Appendix

Convergence Analysis of the EABC Algorithm

According to the enhanced strategies described in the paper, the convergence of the EABC algorithm will be analyzed based on the Markov chain theory. Herein, we will theoretically prove that the EABC algorithm can converge in probability to the global optimum.

Definition A.1 (convergence in probability [34]). Let the set $\{X_t, t > 0, \ t = 1, 2, \ldots\}$ be a population sequence generated with a population-based stochastic algorithm; the stochastic sequence $\{X_t\}$ weakly converges in probability to the global optimum, if and only if

$$\lim_{t \to \infty} p\left\{X_t \cap X^* \neq \Phi\right\} = 1, \tag{A.1}$$

where X^* is the set of the global optima of an optimization problem. Convergence in probability of a population sequence $\{X_t\}$ means that the probability of the individual population from $\{X_t\}$ converging to the global optimum approximates to 1 with the increase of iteration number t. The convergence of EABC algorithm is proved in Lemma A.2 as follows.

Lemma A.2. *Suppose that $\{X_t, t > 0, t = 1, 2, \ldots\}$ is the population sequence generated with the EABC algorithm; then,*

(i) *the set $\{X_t, t > 0, \ t = 1, 2, \ldots\}$ is a finite homogeneous Markov chain on the state space;*

(ii) *the set $\{X_t, t > 0, \ t = 1, 2, \ldots\}$ converges in probability to the global optimum.*

Proof. (i) Let $X_k = \{x_1^k, x_2^k, \ldots, x_{SN}^k\}$ denote the population (i.e., food source position) generated with the EABC algorithm in the kth iteration, and SN is the number of population. The dimension of x_i is usually finite in the optimization problem, and its value is set to m. So the state space of $\{X_t, t > 0, \ t = 1, 2, \ldots\}$ is finite, and its size is $m \times SN$ in each iteration. The search equations of new food sources in the EABC algorithm are independent of the iteration time t and dependent only on the current state X_k ($x_{\text{best}}, x_{k1},$ or x_{k2}) that is based on (12) and (13). Thus, the stochastic sequence $\{X_t, \ t > 0, \ t = 1, 2, \ldots\}$ is a finite homogeneous Markov chain.

(ii) Since the best food source would be retained in each iteration of the EABC algorithm, the optimal fitness sequence $\text{fit}(X_t) = \max \text{fit}(x_i^t), i = 1, 2, \ldots, SN$, is nondecreasing, and the property is shown as follows:

$$\text{fit}\left(X_{t+1}\right) \geq \text{fit}\left(X_t\right), \quad \forall t > 0, \ t \in \{1, 2, \ldots\}. \tag{A.2}$$

Let X^* denote the set of the global optima, and then the sequence of optimal population is $I = \{i \mid x_i^t \cap X^* \neq \Phi\}$.

Let $p_{ij}(t) = p(X_t^i \to X_{t+1}^j)$ denote the transition probability from the state i to j in a step iteration, $p_i(t)$ denote the probability of X_t in the state i, and then the probability of X_{t+1} in the state $j \notin I$ based on Markov chain theory is $p_{t+1} = \sum_{i \in I} \sum_{j \notin I} p_i(t) p_{ij}(t) + \sum_{i \notin I} \sum_{j \notin I} p_i(t) p_{ij}(t)$.

Define $p_t = \sum_{i \notin I} p_i(t) = \sum_{i \notin I} \sum_{j \in I} p_i(t) p_{ij}(t) + \sum_{i \notin I} \sum_{j \notin I} p_i(t) p_{ij}(t)$, and then $p_{t+1} = \sum_{i \in I} \sum_{j \notin I} p_i(t) p_{ij}(t) + p_t - \sum_{i \notin I} \sum_{j \in I} p_i(t) p_{ij}(t)$.

Suppose $i \in I, j \notin I$, and then $\text{fit}(X_{t+1}^j) < \text{fit}(X_t^i)$; this contradicts the property in (A.2).

In this case, $\sum_{i \in I} \sum_{j \notin I} p_i(t) p_{ij}(t) = 0$; that is, $p_{t+1} = p_t - \sum_{i \notin I} \sum_{j \in I} p_i(t) p_{ij}(t)$.

For $i \notin I, j \in I$, then $\sum_{i \notin I} \sum_{j \in I} p_i(t) p_{ij}(t) \geq 0$, so $0 \leq p_{t+1} \leq p_t$; that is, $\lim_{t \to \infty} p_t = 0$.

We can get $\lim_{t \to \infty} p\{X_t \cap X^* \neq \Phi\} = \lim_{t \to \infty} \sum_{i \in I} p_i(t) = 1 - \lim_{t \to \infty} \sum_{i \notin I} p_i(t) = 1 - \lim_{t \to \infty} p_t = 1$.

From (A.2), the EABC algorithm converges in probability to the global optimum. □

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] G. Fedorko, V. Molnar, D. Marasova et al., "Failure analysis of belt conveyor damage caused by the falling material. Part II: application of computer metrotomography," *Engineering Failure Analysis*, vol. 34, pp. 431–442, 2013.

[2] F. M. Megahed and J. A. Camelio, "Real-time fault detection in manufacturing environments using face recognition techniques," *Journal of Intelligent Manufacturing*, vol. 23, no. 3, pp. 393–408, 2012.

[3] I. Aydin, M. Karaköse, and E. Akin, "A new contactless fault diagnosis approach for pantograph-catenary system using pattern recognition and image processing methods," *Advances in Electrical and Computer Engineering*, vol. 14, no. 3, pp. 79–88, 2014.

[4] Y. Yang, C. Miao, X. Li, and X. Mei, "On-line conveyor belts inspection based on machine vision," *Optik*, vol. 125, no. 19, pp. 5803–5807, 2014.

[5] H. Yan, K. Paynabar, and J. Shi, "Image-based process monitoring using low-rank tensor decomposition," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 216–227, 2015.

[6] M. Schiezaro and H. Pedrini, "Data feature selection based on artificial bee colony algorithm," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, article 47, 8 pages, 2013.

[7] M. Reif and F. Shafait, "Efficient feature size reduction via predictive forward selection," *Pattern Recognition*, vol. 47, no. 4, pp. 1664–1673, 2014.

[8] M. Zhao, C. Fu, L. Ji, K. Tang, and M. Zhou, "Feature selection and parameter optimization for support vector machines: a new approach based on genetic algorithm with feature chromosomes," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5197–5204, 2011.

[9] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimisation for feature selection in classification: novel initialisation and updating mechanisms," *Applied Soft Computing*, vol. 18, pp. 261–276, 2014.

[10] B. Chen, L. Chen, and Y. Chen, "Efficient ant colony optimization for image feature selection," *Signal Processing*, vol. 93, no. 6, pp. 1566–1576, 2013.

[11] B. Akay and D. Karaboga, "A survey on the applications of artificial bee colony in signal, image, and video processing," *Signal Image & Video Processing*, vol. 9, no. 4, pp. 967–990, 2015.

[12] D. L. Jia, X. T. Duan, and M. K. Khan, "Modified artificial bee colony optimization with block perturbation strategy," *Engineering Optimization*, vol. 47, no. 5, pp. 642–655, 2014.

[13] M. Shokouhifar and S. Sabet, "A hybrid approach for effective feature selection using neural networks and artificial bee colony optimization," in *Proceedings of the 3rd International Conference on Machine Vision (ICMV '10)*, pp. 502–506, December 2010.

[14] F. G. Mohammadi and M. S. Abadeh, "A new metaheuristic feature subset selection approach for image steganalysis," *Journal of Intelligent and Fuzzy Systems*, vol. 27, no. 3, pp. 1445–1455, 2014.

[15] X. Zhang, X. Liu, and Z. J. Wang, "Evaluation of a set of new ORF kernel functions of SVM for speech recognition," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 10, pp. 2574–2580, 2013.

[16] A. Moosavian, H. Ahmadi, and A. Tabatabaeefar, "Fault diagnosis of main engine journal bearing based on vibration analysis using Fisher linear discriminant, K-nearest neighbor and support vector machine," *Journal of Vibroengineering*, vol. 14, no. 2, pp. 894–906, 2012.

[17] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review*, vol. 42, no. 1, pp. 21–57, 2014.

[18] H. Duan and Q. Luo, "New progresses in swarm intelligence-based computation," *International Journal of Bio-Inspired Computation*, vol. 7, no. 1, pp. 26–35, 2015.

[19] S. Zhang and S. Liu, "A novel artificial bee colony algorithm for function optimization," *Mathematical Problems in Engineering*, vol. 2015, Article ID 129271, 10 pages, 2015.

[20] Z.-A. He, C. Ma, X. Wang et al., "A modified artificial bee colony algorithm based on search space division and disruptive selection strategy," *Mathematical Problems in Engineering*, vol. 2014, Article ID 432654, 14 pages, 2014.

[21] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, 2010.

[22] W.-F. Gao and S.-Y. Liu, "A modified artificial bee colony algorithm," *Computers & Operations Research*, vol. 39, no. 3, pp. 687–697, 2012.

[23] A. Banharnsakun, T. Achalakul, and B. Sirinaovakul, "The best-so-far selection in artificial bee colony algorithm," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2888–2901, 2011.

[24] W.-F. Gao, S.-Y. Liu, and L.-L. Huang, "A novel artificial bee colony algorithm based on modified search equation and orthogonal learning," *IEEE Transactions on Cybernetics*, vol. 43, no. 3, pp. 1011–1024, 2013.

[25] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. Rep. tr06, Computer Engineering Department, Engineering Faculty, Erciyes University, Kayseri, Turkey, 2005.

[26] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, 2008.

[27] B. Akay and D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization," *Information Sciences*, vol. 192, pp. 120–142, 2012.

[28] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[29] S. S. Keerthi and C.-J. Lin, "Asymptotic behaviors of support vector machines with gaussian kernel," *Neural Computation*, vol. 15, no. 7, pp. 1667–1689, 2003.

[30] M.-W. Li, W.-C. Hong, and H.-G. Kang, "Urban traffic flow forecasting using Gauss-SVR with cat mapping, cloud model and PSO hybrid algorithm," *NeuroComputing*, vol. 99, pp. 230–240, 2013.

[31] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.

[32] C. L. Blake and C. J. Merz, *UCI Repository of Machine Learning Databases*, vol. 55, Department of Information and Computer Science, University of California, Irvine, Calif, USA, 1998, http://www.ics.uci.edu/~mlearn/MLRepository.html.

[33] J. L. Raheja, S. Kumar, and A. Chaudhary, "Fabric defect detection based on GLCM and Gabor filter: a comparison," *Optik International Journal for Light & Electron Optics*, vol. 124, no. 23, pp. 6469–6474, 2013.

[34] Z. Hu, S. Xiong, Q. Su, and Z. Fang, "Finite Markov chain analysis of classical differential evolution algorithm," *Journal of Computational & Applied Mathematics*, vol. 268, pp. 121–134, 2014.