IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 23, 2019, accepted December 25, 2019, date of publication December 30, 2019,
date of current version January 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2963163

A Novel Support Vector Machine Based on Hybrid Bat Algorithm and Its Application to Identification of Low Velocity Impact Areas

QI LIU[1], LEI WU[1], FENGDE WANG[2], AND WENSHENG XIAO[1]
[1]College of Mechanical and Electronic Engineering, China University of Petroleum, Qingdao 266580, China
[2]College of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Qingdao 266590, China

Corresponding author: Wensheng Xiao (xiaows@upc.edu.cn)

This work was supported in part by the High-Tech Shipping Research Project from Ministry of Industry and Information Technology of China under Grant 2018GXB01-02-003.

ABSTRACT The standard support vector machine (SVM) performs poorly on the identification problem of low velocity impact areas due to its lower accuracy rate. Improving SVM's performance using the bat algorithm (BA) is feasible, but BA has the premature convergence problem. In this study, a hybrid bat algorithm with double mutation operations (DMBA), in which the Cauchy mutation operator and the extremal optimization mutation operator are integrated into BA, is proposed to enhance BA's ability to jump out of the local optima. Then, a novel SVM based on this hybrid BA, which is called SVM_DMBA, is developed to address the identification problem. Compared with the standard SVM and twelve improved SVM methods which are combined with the standard algorithms, advanced algorithms, and bat variants, the significant performance of SVM_DMBA is validated using UCI datasets. Moreover, to identify low velocity impact areas, SVM_DMBA is applied to the low velocity impact localization system based on fiber Bragg grating (FBG) sensors. The statistical results indicate that SVM_DMBA is a significantly effective method for identifying the low velocity impact areas and generates higher identification accuracy than comparative methods. For 64 low velocity impact areas of 30 mm × 30 mm on an aluminium plate, the average identification error obtained by SVM_DMBA is 1.615%.

INDEX TERMS Extremal optimization, fiber Bragg grating sensor, hybrid bat algorithm, low velocity impact areas, support vector machine.

I. INTRODUCTION

Low velocity impact damages often occur on the ship because of accidents which include collisions with floating ice, stranding, and explosion. Therefore, the identification of low velocity impact areas on the ship is important. The location of impact source can be identified according to impact signals which are acquired from the sensing system in the structure. Generally, sensor types in the sensing system for structural health monitoring (SHM) include piezoelectric sensors (PZT), fiber optic sensors, and other types of sensors. Ships always sail in the harsh marine environment, which requires that sensors have better performance. Compared with the electrical sensor, the fiber Bragg grating (FBG) sensor is more suitable for the SHM of ships due to its flexibility, multiplexing capability, corrosion resistance, small size, and embedding capability [1]. Several researchers have reported that the FBG sensor is feasible for SHM using strain monitoring [2] and acoustic emission (AE) detection [3], [4].

There have been many studies on impact localization methods for structures with various materials by using different types of sensors. Among various kinds of methods available, the triangulation method [5], the neural network algorithm [6], the time reversal method [7], and the reference database method [8] are widely used. Support vector machine (SVM) has been adopted to identify low velocity impact areas in the plate structure [9], but it presents some limitations due to its lower accuracy rate [10].

The accuracy rate, which mainly depends on the training process of SVM, is a crucial value for evaluating SVM's

The associate editor coordinating the review of this manuscript and approving it for publication was Shuihua Wang.

performance. To improve the performance of SVM, a variety of methods such as gradient descent method [11] and grid search method [12] have been successfully used. Evolutionary computation (EC), as a powerful stochastic optimization method, has aroused the attention of scientists and begun to be combined with SVM. For instance, particle swarm algorithm (PSO) was widely employed to tune SVM's parameters for the classification of spot welded joint strength [13], the recognition of anchor rod [14], and the recognition of multiple fault condition of rolling bearing [15]. The combination of genetic algorithm (GA) and SVM, in which GA was utilized to optimize SVM's parameters, was applied to the power transformer fault identification [16]. In addition, other metaheuristic algorithms such as simulated annealing (SA) [17] and fruit fly optimization algorithm (FOA) [18] were utilized to acquire optimal parameters of SVM in classification tasks.

Recently, the bat algorithm (BA), inspired by microbats' echolocation behavior, was proposed by Yang [19]. Based on their echolocation capability, microbats can judge the prey's distance, shape, and location. A metaheuristic searching algorithm depends on two crucial components: exploration and exploitation. Exploration is the search for diverse solutions in new and undiscovered regions. Exploitation is to look for the best solution among the explored neighbors. The standard BA has been proven to be effective and robust on low-dimensional problems and several real world applications because of its great global search capability, but it still has the premature convergence problem. Thus, this study aims to overcome the disadvantage of the standard BA and combine improved BA with SVM to identify low velocity impact areas.

In this study, a hybrid bat algorithm with double mutation operations (DMBA) is proposed and incorporated into the support vector machine (SVM) to enhance SVM's performance. To overcome BA's defect of trapping in the local optima, the Cauchy mutation operator and the extremal optimization mutation operator are introduced into the standard BA. Subsequently, DMBA is applied to the optimization of kernel parameter g and of penalty parameter c in the SVM, which is called SVM_DMBA. The performance of SVM_DMBA is evaluated by real world benchmark datasets from UCI data repository and compared with that of the standard SVM and that of twelve improved SVM methods. Moreover, SVM_DMBA is applied to the low velocity impact localization system based on fiber Bragg grating (FBG) sensors to identify 64 low velocity impact areas on an aluminium plate.

The rest of this paper is given as follows. Section II provides relative theories about SVM_DMBA and low velocity impact localization system. The details about the hybrid bat algorithm and SVM_DMBA are given in Section III. Results obtained by SVM_DMBA and comparative methods in the numerical experiment and identification problem of low velocity impact areas are discussed in Section IV. Finally, Section V draws the conclusion and provides the suggestion for future work.

II. RELATIVE THEORY

A. PRINCIPLE OF FBG SENSOR

The fiber Bragg grating (FBG) sensor is used as a filter to select and reflect back a part of the input light. The center wavelength of the reflected light is given by Eq. (1).

$$\lambda = 2n_{neff}\Lambda \tag{1}$$

where λ is the center-reflecting wavelength of an FBG sensor. Here, n_{neff} is the average refractive index of optic fiber. Λ is the Bragg grating spacing.

The refractive index and Bragg grating spacing vary as the temperature and stress-strain change, which results in the shift of the center-reflecting wavelength. Under normal room temperature, the experiment is completed in a short time, so the variation of the center-reflecting wavelength is caused only by stress-strain. Thus, the shift of the center-reflecting wavelength is expressed by Eq. (2).

$$\Delta\lambda_{\varepsilon} = \lambda\varepsilon\left(1 - P_e\right) \tag{2}$$

where P_e denotes the photo-elastic constant concerned with the Pockel and Poisson's ratio. ε is the stress-strain.

As shown in Eq. (2), the center-reflecting wavelength is linearly correlated with the stress-strain. The center-reflecting wavelength of FBG sensors shifts in response to the stress-strain, which varies with the low velocity impact process on an aluminium plate.

B. WAVELET TRANSFORM METHOD

In the literature [20], the wavelet transform method is introduced as follows:

$$W_f\left(a, b\right) = |a|^{-\frac{1}{2}} \int_{-\infty}^{\infty} f\left(t\right)\psi^*\left(\frac{t-b}{a}\right) \mathrm{d}t \tag{3}$$

where $a \in R_+$ and $b \in R$ denote the scale parameter and translation parameter, respectively. Here, ψ^* is the complex conjugation of wavelet function ψ. In this entire study, Daubechies 4 (DB4) with great orthogonality is chosen as the mother wavelet.

To discretize the continuous wavelet, a and b are introduced as follows:

$$a = a_0^j, \quad b = kb_0^j b_0 \tag{4}$$

where k and j are integers, and $a_0 > 1$ is the constant value.

The discrete wavelet transform is defined as Eq. (5).

$$W_{j,k}\left(t\right) = a_0^{-\frac{1}{2}} \int_{-\infty}^{\infty} f\left(t\right)\psi^*\left(a_0^{-j}t - kb_0\right) \mathrm{d}t \tag{5}$$

Then, the reconstruction formula is given by Eq. (6).

$$f\left(t\right) = C \sum_{-\infty}^{\infty}\sum_{-\infty}^{\infty} W_{j,k} a_0^{-\frac{j}{2}} \psi\left(a_0^{-j}t - kb_0\right) \tag{6}$$

where C is the constant value that is not concerned with the signal.

Algorithm 1 The Standard Bat Algorithm

```
Define the objective function $f(X)$, $X = (x_1, x_2, \cdots, x_d)^T$
Initialize the bat population $X_i$ and $V_i$ $(i = 1, 2, \cdots, N)$
Define pulse frequencies $f_i$ at $X_i$
Initialize loudness $A_i$ and pulse emission rate $r_i$
while ($t \leq$ Maximum number of iterations)
    Calculate frequency Eq. (8)
    Calculate velocities Eq. (9)
    Calculate positions/solutions Eq. (10)
    if ($rand > r_i$)
        Choose a solution from the best solutions
        Obtain a new solution Eq. (11)
    end if
    if ($rand < A_i \& f(X_i) < f(X^*)$)
        Adopt the new solutions
        Reduce $A_i$ Eq. (12)
        Increase $r_i$ Eq. (13)
    end if
    Rank bats and search the current best $X^*$
    $t = t + 1$
end while
Post-process results and visualization
```

C. BAT ALGORITHM

The bat algorithm (BA), whose idea comes from microbats' echolocation capability and social behavior, was developed for handling global optimization problems [19]. The echolocation process of microbats is idealistically regarded as three characteristics which are expressed as follows [19]:

- Using their echolocation capability, bats are able to discriminate food/prey and avoid some barriers.
- Bats flight randomly with velocity V_i at position X_i with a fixed frequency f_{min} and vary the wavelength and loudness of echolocation pulse to seek food. The wavelength and pulse emission rate $r \in [0, 1]$ can automatically change through the distance between bats and their targets.
- The loudness of the pulse is adjusted from a maximum (positive) value A_0 to a minimum constant value A_{min}.

According to these idealizations, the pseudocode of BA is described in Algorithm 1. The purpose of an algorithm is to find high-quality food in a region where food sources are in. According to Eq. (7), the initial population is acquired from Nd-dimensional vectors. Subsequently, the quality of food for each bat is evaluated.

$$x_{ij} = x_{mind} + \mu \left(x_{maxd} - x_{mind}\right) \tag{7}$$

where $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, d$. $\mu \in [0, 1]$ is a random value. x_{maxd} is the upper limit and x_{mind} is the lower limit in d-dimensional space.

The position (X_i) and velocity (V_i) of each bat (i) varies with the number of iterations. For a bat at time step (t), the new position and velocity can be given as follows:

$$f_i = f_{min} + (f_{max} - f_{min})\, beta \tag{8}$$

$$V_i^t = V_i^{t-1} + \left(X_i^{t-1} - X^*\right) f_i \tag{9}$$

$$X_i^t = X_i^{t-1} + V_i^t \tag{10}$$

where f_i is the frequency of the ith bat. f_{min} is the minimum frequency and f_{max} is the minimum frequency. $\beta \in [0, 1]$ is the random value. X^* is the current global best solution achieved by comparing solutions among N bats.

The new solution of each bat in the exploitation process is obtained by Eq. (11).

$$x_{new} = x_{old} + \varepsilon A^t \tag{11}$$

where $\varepsilon \in [0, 1]$ is a random value and $A^t = \left\langle A_i^t \right\rangle$ is the average loudness at this time step.

The loudness A_i and pulse emission rate r_i are updated in the process of iterations. As the distance between bats and their prey decreases, A_i decreases, whereas r_i increases. They are calculated in terms of Eq. (12) and Eq. (13), respectively.

$$A_i^{t+1} = \alpha A_i^t \tag{12}$$

$$r_i^{t+1} = r_i^0 \left[1 - \exp\left(-\gamma t\right)\right] \tag{13}$$

where α and γ are both fixed values. For any $0 < \alpha < 1$ and $\gamma > 0$, $A_i^t \to 0$ and $r_i^t \to r_i^0$ as $t \to \infty$. In general, the initial loudness A_i^0 belongs to [1, 2] and the initial pulse emission rate r_i^0 belongs to [0, 1].

D. SUPPORT VECTOR MACHINE

The support vector machine (SVM) was proposed based on statistical learning theory, which is an approximate structural risk minimization method and a powerful tool for classification tasks [21], [22]. Compared with other classification methods, SVM shows greater efficiency and accuracy in dealing with the large quantity of sample data because there is no distinct effect of sample size on SVM's performance.

Consider the sample data and sample category are $\{x_i, y_i\}$, where $i = 1, 2, \cdots, l$, $y_i \in \{-1, 1\}$, and l is the number of sample data. For the soft margin classification, the primal problem of SVM is defined as follows:

$$\begin{cases} \min \dfrac{1}{2} \left\|\omega\right\|^2 + c \displaystyle\sum_{i=1}^{l} \varepsilon_i \\ s.t.\ y_i \left(\omega \cdot x + b^*\right) \geq 1 - \varepsilon_i \quad (\varepsilon_i \geq 0) \end{cases} \tag{14}$$

where ω and b^* denote the weight vector and the bias vector, respectively. Here, ε_i and c denote a slack factor and the penalty parameter, respectively.

Then, using SVM to handle classification problems can be presented as the dual optimization problem which is given by Eq. (15).

$$\begin{cases} \min \dfrac{1}{2} \displaystyle\sum_{i=1}^{j} \sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j K\left(x_i, x_j\right) - \sum_{i=1}^{l} \alpha_i \\ s.t.\ \displaystyle\sum_{i=1}^{l} \alpha_i y_i = 0 \quad (0 \leq \alpha_i \leq c) \end{cases} \tag{15}$$

where α_i denotes the Lagrange multipliers. $K\left(x_i, x_j\right)$ denotes the kernel function which commonly involves linear function, polynomial function, Gaussian function, and radial basis function (RBF). In this paper, we choose RBF as the kernel function because RBF has a faster learning speed and greater classification capability. The RBF is given by Eq. (16).

$$K\left(x_i, x_j\right) = \exp\left(-g\left\|x_i - x_j\right\|^2\right) \tag{16}$$

where g is the kernel parameter.

The classification results of sample data are calculated according to the decision function expressed as follows:

$$f\left(x\right) = \operatorname{sgn}\left(\sum_{i=1}^{l} \alpha_i^* y_i K\left(x_i, x_j\right) + b^*\right) \tag{17}$$

III. NOVEL SVM BASED ON HYBRID BAT ALGORITHM

A. HYBRID BAT ALGORITHM WITH DOUBLE MUTATION OPERATIONS

The standard bat algorithm may encounter the premature convergence problem in solving complex optimization problems [23]. In this study, we have introduced two mutation operators into BA to enhance its performance in jumping out of the local optima. These two mutation operators involve the Cauchy mutation operator and the extremal optimization mutation operator. The pseudocode of the hybrid bat algorithm with double mutation operations (DMBA) is displayed in Algorithm 2.

1) CAUCHY MUTATION OPERATOR

A Cauchy mutation operator based on the Cauchy probability distribution has been introduced in designing a fast evolutionary programming [24]. Other than the Gaussian probability distribution, the Cauchy probability distribution has a longer two-tail, and thus the population generated by the Cauchy mutation operator is significantly different from its parents [25]. In other words, the bound of the random value is wider, which means that this mutation operator can offer more chances for solutions to escape from the local optima.

The definition of Cauchy density function is given by Eq. (18) [26].

$$f\left(x\right) = \frac{1}{\pi}\frac{t}{t^2 + x^2} \tag{18}$$

where $x \in [-\infty, \infty]$, and $t > 0$ is the scale factor.

Then, the Cauchy distributed function is described by Eq. (19) [26].

$$F_t\left(x\right) = \frac{1}{2} + \frac{1}{\pi} arctan\left(\frac{x}{t}\right) \tag{19}$$

During early iterations, the global search capability of BA must be enhanced to enlarge bats' search space. Thus, the Cauchy mutation operator is embedded into BA to compulsively perform the mutation of the position of each bat using Eq. (20).

$$X_i^t = X_i^t\left(1 + Cauchy\left(0, 1\right)\right) \tag{20}$$

Algorithm 2 DMBA

```
Define the objective function $f(X), X = (x_1, x_2, \cdots, x_d)^T$
Initialize the bat population $X_i$ and $V_i$ $(i = 1, 2, \cdots, N)$
Define pulse frequencies $f_i$ at $X_i$
Initialize loudness $A_i$ and pulse emission rate $r_i$
Define $INV$-iteration interval
while ($t \leq$ Maximum number of iterations)
    Calculate frequency Eq. (8)
    Calculate velocities Eq. (9)
    Calculate positions/solutions Eq. (10)
    Perform the Cauchy mutation operator on the new
    solution Eq. (20)
    if (rand $> r_i$)
        Choose a solution from the best solutions
        Obtain a new solution Eq. (11)
    end if
    if (rand $< A_i \& f(X_i^t) < f(X_i^{t-1})$)
        Adopt the new solutions
        Reduce $A_i$ Eq. (12)
        Increase $r_i$ Eq. (13)
    end if
    if ($f(X_i^t) < f(X^*)$)
        Update the current best $X^*$
    end if
    if(iteration mod INV=0)
      Evaluate local fitness $C_{ij}$ for each component
      $(i = 1, 2, \cdots, N; j = 1, 2, \cdots, d)$
      Find $k$ satisfying $C_{ik} \leq C_{ij}$ for all $i$ and $j$
      $(k = 1, 2, \cdots, d)$
      Choose $X' \in Neigh(X)$ such that $x_{ik}$ must change its
      state
      if ($f(X') < f(X_i^t)$)
          Update $X_i^t$
      end if
      if ($f(X_i^t) < f(X^*)$)
          Update the current best $X^*$
      end if
    end if
    $t = t + 1$
end while
Post-process results and visualization
```

where *Cauchy* (0,1) is the random value achieved by Eq. (19) in which t is equal to 1.

2) EXTREMAL OPTIMIZATION MUTATION OPERATOR

During later iterations, bats are around the global best solutions as the diversity of the population decreases, which means that it is necessary to reinforce the local search capability of BA to search for the best solutions. Additionally, the premature convergence problem may occur when there is no change in the global best solution after a certain number of iterations. Thus, to achieve better quality solutions, the extremal optimization (EO) algorithm with a great local

Algorithm 3 The Extremal Optimization Algorithm

```
Randomly initialize the configuration $X$
Set the optimal solution $X_{best} = X$
while ($t \leq$ Maximum number of iterations)
    Evaluate local fitness $C_i$ for each variable $x_i$ $(i = 1, 2, \cdots, d)$
    Find $j$ satisfying $C_j \leq C_i$ for all $i$, i. e., $x_j$ has the "worst fitness"
    Choose $X' \in Neigh(X)$ such that $x_j$ changes its state
    Adopt $X = X'$ unconditionally
    if ($f(X) < f(X_{best})$)
        $X_{best} = X$
        $f(X_{best}) = f(X)$
    end if
    $t = t + 1$
end while
Return $X_{best}$ and $f(X_{best})$
```

search capability is chosen as the mutation operator to combine with BA.

Based on Bak-Sneppen model of self-organized criticality, Boettcher and Percus developed the EO algorithm [27]. Unlike other intelligent algorithms, EO is a local search method where poor individuals are partly eliminated and replaced to produce new individuals in the population evolution. The EO algorithm to solve the unconstrained minimization problem is described in Algorithm 3.

In this method, the new best solution is obtained by searching near the current best solution. Due to the small mutation size of the EO mutation operator, BA's performance in escaping from the local optima is greatly enhanced. This method is suitable for exploitation during later iterations. Therefore, as the second mutation operator, the EO algorithm is embedded into BA.

To handle d-dimensional unconstrained minimization problems, the local fitness is calculated. Then, the worst component in each component of a solution is determined. For the jth component of a solution, its fitness value is generally regarded as the mutation cost [28]. Therefore, in the DMBA, the local fitness C_{ij} of each component $x_{ij}(i = 1, 2, \cdots, N; j = 1, 2, \cdots, d)$ is given by Eq. (21).

$$C_{ij} = f\left(X_{ij}\right) - f\left(X_{best}\right) \tag{21}$$

where X_{ij} is the new solution achieved executing the mutation operation only on x_{ij} and fixing the other components. $f(X_{ij})$ is the fitness value of X_{ij}, whereas $f(X_{bset})$ is the fitness value of the global best solution X_{best}.

Note that executing the EO mutation operator will increase the computation time if BA is not stuck at the local optima. To control the frequency of launching the EO mutation operator in the DMBA, a strategy that the EO mutation operator is executed every *INV*-iterations is utilized in this paper [29].

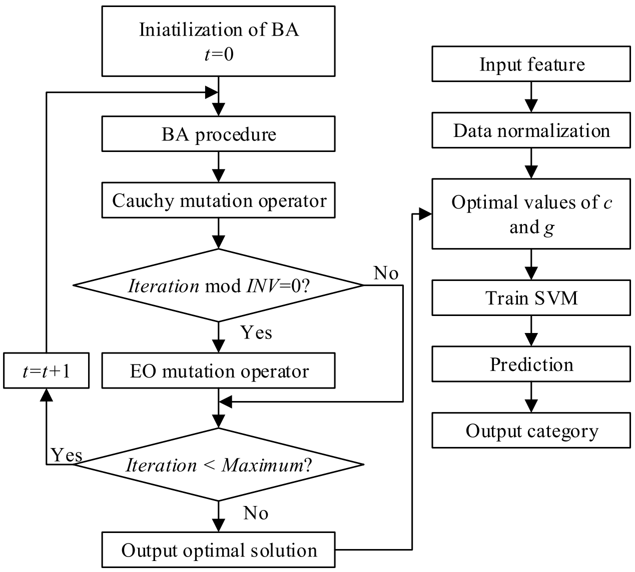

FIGURE 1. The flowchart of the proposed SVM_DMBA.

B. COMBINATION OF SVM AND DMBA

The determination of parameters has a significant influence on the identification accuracy and the convergence rate of SVM. In the SVM, there are two parameters to be optimized, including the kernel parameter g and the penalty parameter c [30]. The kernel parameter g has an effect on the kernel mapping distribution of the sample, and the penalty parameter c has an effect on the clash between algorithm complexity and identification accuracy [15].

Generally, in the SVM, as the kernel parameter g is smaller and the penalty parameter c is larger, the nonlinear fitting capability is better, whereas the training time is longer. Small kernel parameter g means less smoothness, and the generalization capability decrease as the penalty parameter c increase. In other words, determining the appropriate combination of g and c is important because it significantly affects the use of SVM to perform great identification accuracy for impact areas.

Since the time consumption is too long when common techniques such as grid search and random search are used to handle a large number of sample data [31], these techniques are not applicable to the identification of low velocity impact areas. To obtain better identification performance, DMBA is employed to optimize g and c during the training process of SVM, which is called SVM_DMBA. The correct identification rate is regarded as the fitness value in DMBA, which is obtained by the 5-fold cross validation on the training sample set [15]. Fig. 1 shows the flowchart of SVM_DMBA.

IV. EXPERIMENTS AND DISCUSSIONS

In this work, two experiments were carried out. The first numerical experiment based on benchmark datasets from the UCI data repository was executed to evaluate SVM_DMBA's

TABLE 1. Classical benchmark datasets from the UCI data repository.

No.	Dataset	Instances	Number of Features	Number of Classes
1.	Zoo	101	16	7
2.	Iris	150	4	3
3.	Wine	178	13	3
4.	Segmentation	210	18	7
5.	Glass	214	9	6
6.	Ecoli	336	7	8
7.	Vowel	528	10	11
8.	Diabetes	768	8	2
9.	Abalone	4177	7	28
10.	Robot Navigation	5456	24	4

performance on real world application problems. Additionally, the second experiment was implemented to emphasize the feasibility and effectiveness of SVM_DMBA on the identification problem of low velocity impact areas. These two experiments were performed through MATLAB R2015b on Intel (R) Core (TM) i7-8700 CPU @ 3.20 GHz with 16.00 GB of RAM and Windows 10 64-bit operating system.

A. PERFORMANCE ANALYSIS OF SVM_DMBA

To validate the performance of SVM_DMBA on the real world application problem, the publicly available real world benchmark datasets chosen from the UCI data repository [32] (http://archive.ics.uci.edu/ml/index.php) were used in this experiment. Table 1 lists the details of ten benchmark datasets. Each dataset was randomly divided into two parts that 70% of each dataset was employed for the training sample set and 30% of each dataset was used as the test sample set. The appropriate parameters were selected by using the training sample set during the training process, meanwhile, the classification error was measured by using the test sample set during the test process.

In this experiment, to analyze the contribution of different modifications, BA was combined with Cauchy mutation operator, which was called BACA; BA was combined with EO mutation operator, which was called BAEO. These two comparative algorithms were utilized to optimize parameters of SVM and compare with DMBA. Four standard algorithms including BA, GA [33], PSO [34], and differential evolution (DE) [35] were embedded into SVM, which were called SVM_BA, SVM_GA, SVM_PSO, and SVM_DE. Additionally, three advanced algorithms, which were composed of chicken swarm optimization (CSO) [36], ant lion optimizer (ALO) [37], and grey wolf optimizer (GWO) [38], were adopted. These advanced algorithms were combined with SVM as follows: SVM_CSO, SVM_ALO, and SVM_GWO. Since a majority of BA variants were proposed to improve BA's performance, three improved bat algorithms were also chosen as the comparative methods to optimize the SVM. These BA variants were hybrid bat algorithm (HBA) [39], bat algorithm based on iterative local search and stochastic inertia weight (ILSSIWBA) [40], and directional bat algorithm (dBA) [41]. For the HBA, three effective modifications were proposed. For the ILSSIWBA, the iterative local search and stochastic inertia weight were incorporated with BA. For the dBA, the directional echolocation was introduced into BA. Then, these improved SVM methods were called SVM_HBA, SVM_ILSSIWBA, and SVM_dBA.

For a fair comparison, the population size N was set to 10 and the maximum number of iterations t_{max} was set to 100 for DMBA and comparative algorithms. Other parameters of these algorithms were set as follows:

- DMBA: The proposed algorithm was implemented with $\alpha = \gamma = 0.9, f_{min} = 0, f_{max} = 1, A_0 = 0.95, r_0 = 0.85$, and $INV = 10$.
- BA: As mentioned in the literature [42], the standard BA was executed with $\alpha = \gamma = 0.9, f_{min} = 0, f_{max} = 1$, $A_0 = 0.95$, and $r_0 = 0.85$.
- GA: The crossover probability p_c was set to 0.95 and the mutation probability p_m was set to 0.05 [33].
- PSO: The acceleration coefficients c_1 and c_2 were set to 1.5 and 1.2. The inertia weight $\omega \in [0.4, 0.9]$ was the monotone decreasing function [34].
- DE: As described in the literature [35], this standard algorithm was carried out with the cross rate $CR \in [0.2, 0.9]$ and the scale factor $F \in [0.4, 1]$.
- CSO: We executed this algorithm with the number of the roosters $RN = 0.2 * N$, the number of the hens $HN = 0.6 * N$, the number of the chicks CN $= N\text{-}RN\text{-}HN$, the number of the mother hens $MN = 0.1 * N$, the time steps $G = 10$, and the parameter $FL \in [0.5, 0.9]$ [36].
- HBA: It was applied with $\alpha = \gamma = 0.9, f_{min} = 0, f_{max} = 1, A_0 = 0.95, r_0 = 0.85$, the limiting threshold $LT = 5$, and the maximum limiting threshold$MLT = 2$ [39].
- ILSSIWBA: This algorithm described in the literature [40] was considered with $f_{min} = 0, f_{max} = 2, A_0 = 0.9$, the lower limit of loudness$A_\infty = 0.6$, $r_0 = 0.1$, the upper limit of pulse rate $r_\infty = 0.7$, the minimum factor of the stochastic inertia weight $\mu_{min} = 0.4$, the maximum factor of the stochastic inertia weight $\mu_{max} = 0.9$, and the deviation between the stochastic inertia weight and its mean $\sigma = 0.2$.
- dBA: The considered algorithm was implemented as it was shown in the literature [41] with $r_0 = 0.1, r_\infty = 0.7, A_0 = 0.9, A_\infty = 0.6, f_{min} = 0$, and $f_{max} = 2$.
- The parameters of BAEO and BACA were the same as those of DMBA and BA, respectively.
- ALO has no special parameters besides the population size and the maximum number of iterations [37], and GWO has two random generated parameters [38].

For parameters of the standard SVM, the penalty parameter c was set to 1, and the kernel parameter g was equal to $1/d$ (d was the dimension of sample data) [15]. During the optimization process, the penalty parameter c and the kernel parameter g, whose lower bounds and upper bounds were [0.1, 0.01] and [100, 1000], were optimized by the above thirteen optimization algorithms. For each dataset, the procedures of all methods were carried out 30 times. The mean value and the standard deviation of classification errors are recorded in Tables 2-5.

TABLE 2. Comparison of the standard SVM with improved SVM methods based on two mutation operators on classical benchmark datasets. The "*p*-value" is the result of the Wilcoxon test at a 0.05 level of significance (Time in seconds).

Dataset	SVM_BA		SVM_BACA		SVM_BAEO		SVM_DMBA		SVM	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
Zoo	1.655E-01	1.246E-01	1.655E-01	1.246E-01	**6.897E-02**	1.170E-16	**6.897E-02**	**1.170E-16**	**6.897E-02**	4.517E-16
Iris	2.222E-03	7.027E-03	2.222E-03	7.027E-03	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
Wine	1.962E-01	2.813E-01	1.358E-01	2.466E-01	1.887E-02	2.466E-03	**1.698E-02**	5.967E-03	1.887E-02	**4.517E-16**
Segmentation	1.206E-01	3.604E-02	1.083E-01	9.010E-03	1.095E-01	5.019E-03	**1.048E-01**	1.110E-03	1.111E-01	**3.388E-16**
Glass	4.508E-01	4.414E-02	4.215E-01	7.648E-02	4.112E-01	3.841E-02	4.092E-01	4.822E-02	**3.077E-01**	**0.000E+00**
Ecoli	1.545E-01	8.349E-03	1.535E-01	1.069E-02	1.584E-01	1.170E-16	**1.505E-01**	1.670E-02	1.584E-01	**0.000E+00**
Vowel	5.117E-01	1.493E-01	4.623E-01	1.397E-02	4.065E-01	1.422E-02	**3.896E-01**	1.015E-02	4.481E-01	**3.388E-16**
Diabetes	2.543E-01	5.274E-02	2.313E-01	2.599E-02	2.297E-01	3.963E-03	**2.274E-01**	6.166E-03	2.304E-01	**1.129E-16**
Abalone	7.088E-01	1.306E-03	7.088E-01	2.174E-03	7.073E-01	3.597E-03	**7.068E-01**	2.407E-03	7.091E-01	**1.694E-16**
Robot Navigation	1.296E-01	4.109E-02	1.270E-01	4.275E-02	1.159E-01	1.553E-02	**1.135E-01**	2.462E-02	1.271E-01	**5.646E-16**
+/−/≈	2/8/0		4/6/0		5/1/4		7/1/2			
p-value	**0.021824**		0.168807		0.345448		0.161429			
Computation time	4.803E+03		5.018E+03		5.209E+03		5.488E+03		**1.716E+00**	

TABLE 3. Comparison of SVM_DMBA with improved SVM methods based on the standard algorithms on classical benchmark datasets. The "*p*-value" is the result of the Wilcoxon test at a 0.05 level of significance (Time in seconds).

Dataset	SVM_GA		SVM_PSO		SVM_DE		SVM_DMBA	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
Zoo	2.793E-01	7.529E-02	9.310E-02	7.365E-02	2.724E-01	1.090E-02	**6.897E-02**	**1.170E-16**
Iris	2.222E-02	1.170E-16	1.482E-03	5.638E-03	2.000E-02	7.027E-03	**0.000E+00**	**0.000E+00**
Wine	4.245E-01	2.735E-01	1.943E-01	2.726E-01	4.472E-01	2.581E-01	**1.698E-02**	5.967E-03
Segmentation	2.032E-01	4.220E-02	**9.947E-02**	2.318E-02	2.138E-01	8.066E-02	1.048E-01	**1.110E-03**
Glass	4.769E-01	4.644E-02	**4.064E-01**	**3.468E-02**	4.338E-01	1.630E-02	4.092E-01	4.822E-02
Ecoli	**1.455E-01**	3.269E-02	1.584E-01	**0.000E+00**	1.532E-01	4.834E-02	1.505E-01	1.670E-02
Vowel	5.708E-01	1.629E-01	3.961E-01	3.865E-02	4.799E-01	1.910E-01	**3.896E-01**	1.015E-02
Diabetes	2.974E-01	4.080E-02	2.443E-01	4.739E-02	2.561E-01	2.671E-02	**2.274E-01**	6.166E-03
Abalone	7.146E-01	3.028E-02	7.091E-01	1.100E-02	7.106E-01	8.772E-03	**7.068E-01**	2.407E-03
Robot Navigation	1.256E-01	9.058E-02	1.461E-01	2.194E-02	1.260E-01	9.786E-02	**1.135E-01**	2.462E-02
+/−/≈	1/9/0		2/8/0		0/10/0			
p-value	**0.006910**		**0.036658**		**0.005062**			
Computation time	5.057E+03		**4.208E+03**		5.360E+03		5.488E+03	

TABLE 4. Comparison of SVM_DMBA with improved SVM methods based on the advanced algorithms on classical benchmark datasets. The "*p*-value" is the result of the Wilcoxon test at a 0.05 level of significance (Time in seconds).

Dataset	SVM_ALO _		SVM_CSO		SVM_GWO		SVM_DMBA	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
Zoo	9.310E-02	7.633E-02	**6.897E-02**	**1.170E-16**	**6.897E-02**	**1.170E-16**	**6.897E-02**	**1.170E-16**
Iris	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
Wine	7.547E-02	1.857E-01	1.887E-02	**0.000E+00**	2.075E-02	5.967E-03	**1.698E-02**	5.967E-03
Segmentation	1.111E-01	**2.341E-16**	1.095E-01	5.019E-03	1.079E-01	6.693E-03	**1.048E-01**	1.110E-03
Glass	**3.831E-01**	4.002E-02	4.000E-01	3.768E-02	3.923E-01	**3.497E-02**	4.092E-01	4.822E-02
Ecoli	1.584E-01	**1.170E-16**	1.531E-01	1.820E-02	1.545E-01	1.252E-02	**1.505E-01**	1.670E-02
Vowel	3.857E-01	1.341E-02	3.870E-01	1.270E-02	**3.831E-01**	1.104E-02	3.896E-01	**1.015E-02**
Diabetes	2.343E-01	3.998E-02	**2.209E-01**	**1.833E-03**	2.230E-01	4.125E-03	2.274E-01	6.166E-03
Abalone	7.110E-01	2.626E-03	7.093E-01	1.954E-03	7.077E-01	**1.035E-03**	**7.068E-01**	2.407E-03
Robot Navigation	**1.063E-01**	1.271E-03	1.070E-01	**1.095E-03**	1.075E-01	1.271E-03	1.135E-01	2.462E-02
+/−/≈	3/6/1		4/4/2		4/4/2			
p-value	0.313938		0.123025		0.262618			
Computation time	4.712E+03		3.110E+03		**2.726E+03**		5.488E+03	

In this study, the nonparametric Wilcoxon test with a level of significance of 0.05 [43] was employed to compare the mean values of the classification errors achieved by the standard SVM and different improved SVM methods. The *p*-values of a two-sided Wilcoxon signed rank test are illustrated in Tables 2-5. At the bottom of these Tables, the computation time obtained by SVM_DMBA and comparative algorithms for all problems is listed. The best results are bold.

In Table 2, SVM_BA obtained better results than the standard SVM only for Ecoli and Abalone, which indicates that BA has a limited ability to optimize SVM due to BA's premature convergence problem. However, improved SVM methods based on two mutation operators showed better performance than SVM_BA for a majority of problems, especially SVM_DMBA. Compared with the standard SVM, SVM_BACA had better results for four problems (Segmentation, Ecoli, Abalone, and Robot Navigation); SVM_BAEO had better results for five problems (Segmentation, Vowel, Diabetes, Abalone, and Robot Navigation), and it was similar to SVM for four problems (Zoo, Iris, Wine, and

TABLE 5. Comparison of SVM_DMBA with improved SVM methods based on the bat variants on classical benchmark datasets. The "*p*-value" is the result of the Wilcoxon test at a 0.05 level of significance (Time in seconds).

Dataset	SVM_HBA		SVM_dBA		SVM_ILSSIWBA		SVM_DMBA	
	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
Zoo	**6.897E-02**	**1.170E-16**	**6.897E-02**	**1.170E-16**	**6.897E-02**	**1.170E-16**	**6.897E-02**	**1.170E-16**
Iris	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
Wine	1.887E-02	2.466E-03	1.887E-02	3.020E-03	1.869E-02	2.837E-03	**1.698E-02**	5.967E-03
Segmentation	1.060E-01	7.483E-03	1.111E-01	2.341E-16	1.111E-01	**2.341E-16**	**1.048E-01**	1.110E-03
Glass	**3.785E-01**	**3.339E-02**	4.969E-01	8.972E-02	4.231E-01	5.487E-02	4.092E-01	4.822E-02
Ecoli	**1.495E-01**	1.779E-02	1.556E-01	1.275E-02	1.526E-01	4.314E-02	1.505E-01	**1.670E-02**
Vowel	3.859E-01	1.137E-02	4.234E-01	1.633E-02	**3.825E-01**	1.080E-02	3.896E-01	**1.015E-02**
Diabetes	2.291E-01	3.911E-03	2.312E-01	2.880E-02	2.301E-01	6.743E-02	**2.274E-01**	6.166E-03
Abalone	7.075E-01	2.515E-03	7.093E-01	2.593E-03	7.082E-01	1.655E-03	**7.068E-01**	**2.407E-03**
Robot Navigation	1.079E-01	6.039E-03	1.144E-01	1.203E-02	**1.075E-01**	**1.573E-03**	1.135E-01	2.462E-02
+/−/≈	4/4/2		0/8/2		2/6/2			
p-value	0.483840		**0.011719**		0.400814			
Computation time	7.155E+03		7.278E+03		6.437E+03		**5.488E+03**	

"+", "−", and "≈" denote that the performance of the comparative method is significantly superior to, inferior to, and similar to that of SVM_DMBA, respectively.

Ecoli); SVM_DMBA achieved better results for seven problems (Wine, Segmentation, Ecoli, Vowel, Diabetes, Abalone, and Robot Navigation), whereas the performance of SVM_DMBA was not significantly different from that of SVM and that of SVM_BAEO for Zoo and Iris.

The Wilcoxon signed rank test results showed that there was a significant difference between the standard SVM and SVM_BA, whereas SVM was similar to other improved SVM methods. Thus, we can note that the proposed modifications can effectively overcome BA's disadvantage, and the combination of two mutation operators compared with the single mutation operator has a remarkable influence on improving the performance of BA. Additionally, EO mutation operator has the strongest contribution among these two mutation operators.

As shown in Table 3, compared with improved SVM methods based on three standard algorithms, SVM_DMBA obtained better results for seven classification problems (Zoo, Iris, Wine, Vowel, Diabetes, Abalone, and Robot Navigation). SVM_DMBA outperformed significantly SVM_DE for all test datasets, whereas it was worse than SVM_GA only for Ecoli and SVM_PSO for Segmentation and Glass.

Table 4 indicated that SVM_DMBA was superior to improved SVM methods based on three advanced algorithms for four problems (Wine, Segmentation, Ecoli, and Abalone). In addition, the performance of SVM_DMBA was similar to that of all comparative methods for Iris and that of comparative methods expect for SVM_ALO for Zoo. SVM_ALO obtained better results for Glass and Robot Navigation, SVM_CSO obtained better results only for Diabetes, and SVM_GWO obtained better results only for Vowel.

In Table 5, for Zoo and Iris, there was no significant difference between SVM_DMBA and improved SVM methods based on three bat variants. For four test datasets (Wine, Segmentation, Diabetes, and Abalone), SVM_DMBA achieved better results than SVM_HBA, SVM_dBA, and SVM_ILSSIWBA. Nevertheless, SVM_DMBA was worse than SVM_HBA for Glass and Ecoli, and it was worse than SVM_ILSSIWBA for Vowel and Robot Navigation.

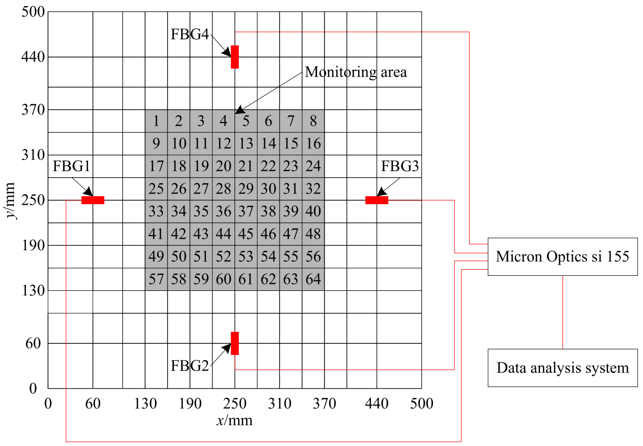

FIGURE 2. Sketch of the low velocity impact localization system.

TABLE 6. Wavelengths and locations of four FBG sensors.

Sensor	Wavelength	Location
FBG1	1570.339 nm	(60 mm, 250 mm)
FBG2	1549.794 nm	(250 mm, 60 mm)
FBG3	1555.937 nm	(440 mm, 250 mm)
FBG4	1562.021 nm	(250 mm, 440 mm)

Through analyzing the *p*-values in Tables 3-5, for solving classification problems, SVM_DMBA was significantly different from SVM_GA, SVM_PSO, SVM_DE, and SVM_dBA, but similar to other improved SVM methods. By analyzing computation time obtained by SVM_DMBA and all comparative methods, we can note that combining SVM with optimization algorithms will strongly increase the complexity of SVM. In addition, SVM_DMBA required the shortest computation time compared with SVM_HBA, SVM_dBA, and SVM_ILSSIWBA, but SVM_DMBA's computation time was longer than that of other nine SVM variants.

In summary, compared with the standard algorithms, advanced algorithms, and bat variants, DMBA has greater ability to optimize the performance of SVM in classification

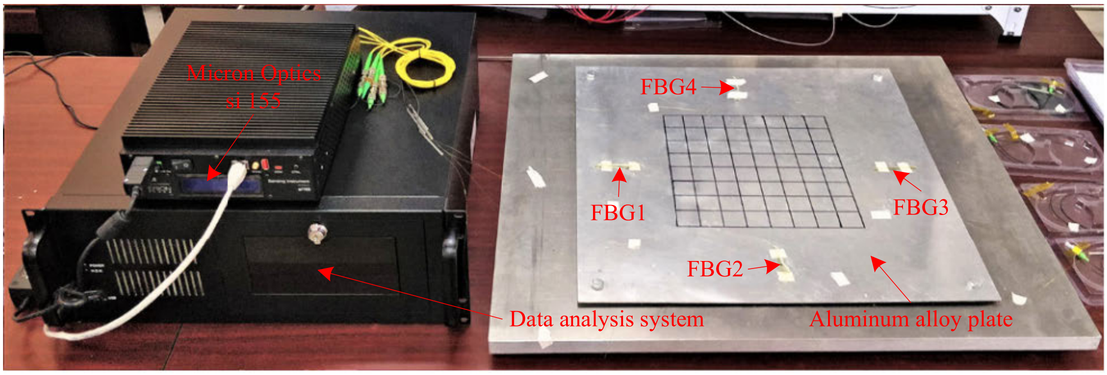

FIGURE 3. Diagram of the low velocity impact localization system.

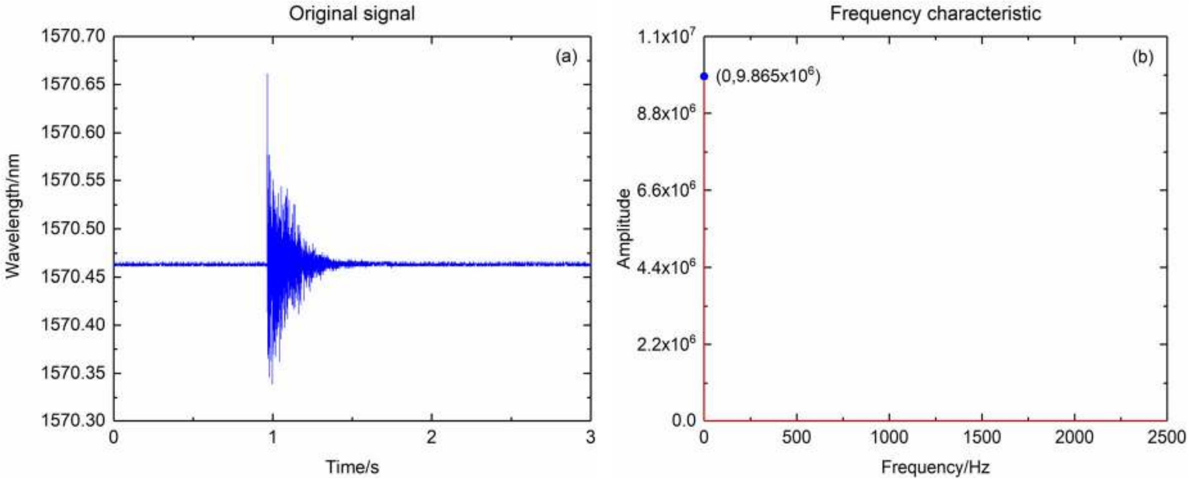

FIGURE 4. Original impact signal received by the sensor FBG1 and frequency characteristics in the area 19.

tasks. Furthermore, SVM_DMBA compared with the standard SVM and other improved SVM methods can be considered as the best performing method for solving real world classification problems.

B. APPLICATION OF SVM_DMBA ON THE IDENTIFICATION OF LOW VELOCITY IMPACT AREAS

1) LOW VELOCITY IMPACT LOCALIZATION SYSTEM

The low velocity impact localization system consists of an optical sensing interrogator, an aluminium plate, four FBG sensors, a low velocity impact device, and a data analysis system. The optical sensing interrogator is the Micron Optics si155, which has the wavelength accuracy of 2 pm/3 pm and the full spectrum at 5 kHz with 80 nm on 4 parallel channels simultaneously. An aluminium plate with dimensions of 500 mm × 500 mm × 2 mm was chosen as the test specimen and fixed on the experiment table by the fixture. In the center of the test specimen, there was a square monitoring area of 240 mm × 240 mm which was divided into 64 square areas of 30 mm × 30 mm. Each square area as a category was marked as $S = (1, 2, \cdots, 64)$.

Four FBG sensors with the same grating length (10 mm) were symmetrically located on the centerline of the monitoring area. The center-reflecting wavelength and locations of four FBG sensors are listed in Table 6. To generate the low velocity impact sources, a steel ball with the weight of 36 g was chosen as the low velocity impact device. The steel ball fell down in free fall from the impact height of 200 mm, which resulted in the impact energy of 0.07056 J and the impact velocity of 1.98 m/s. The sketch and diagram of the low velocity impact localization system are shown in Figs. 2 and 3, respectively.

2) FEATURE EXTRACTION

In the experiment of identifying the low velocity impact areas, the target feature to be extracted is the frequency characteristic of the low velocity impact signals. To verify the effectiveness of the extracted feature, a steel ball was used to sequentially impact the square areas marked 19, 22, 43, and 46 on the aluminium plate. The impact signals received by the sensor FBG1 were selected to analyze the relationship between impact areas and extracted features.

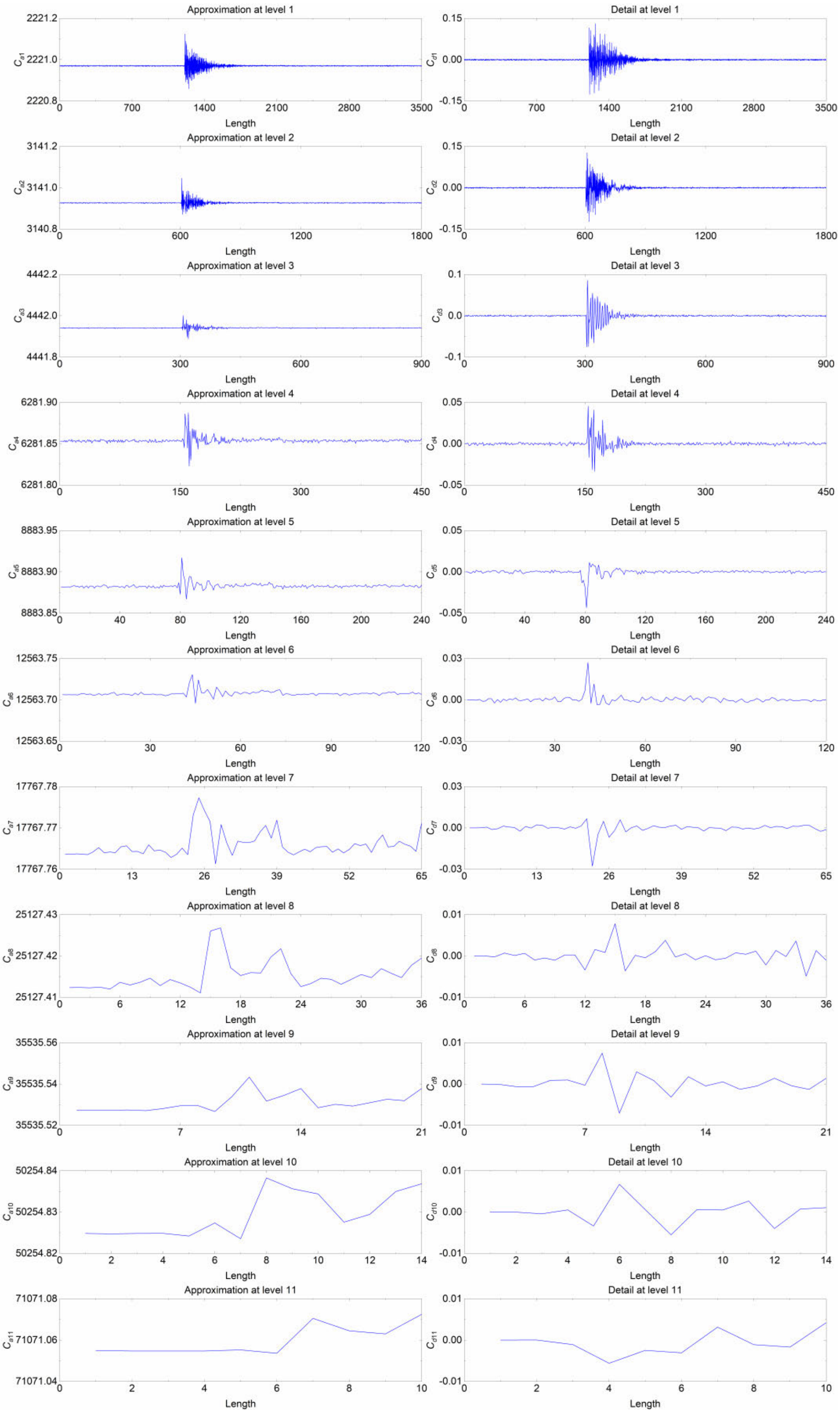

FIGURE 5. Approximation coefficients and detail coefficients of impact signal received by the sensor FBG1 in the area 19.

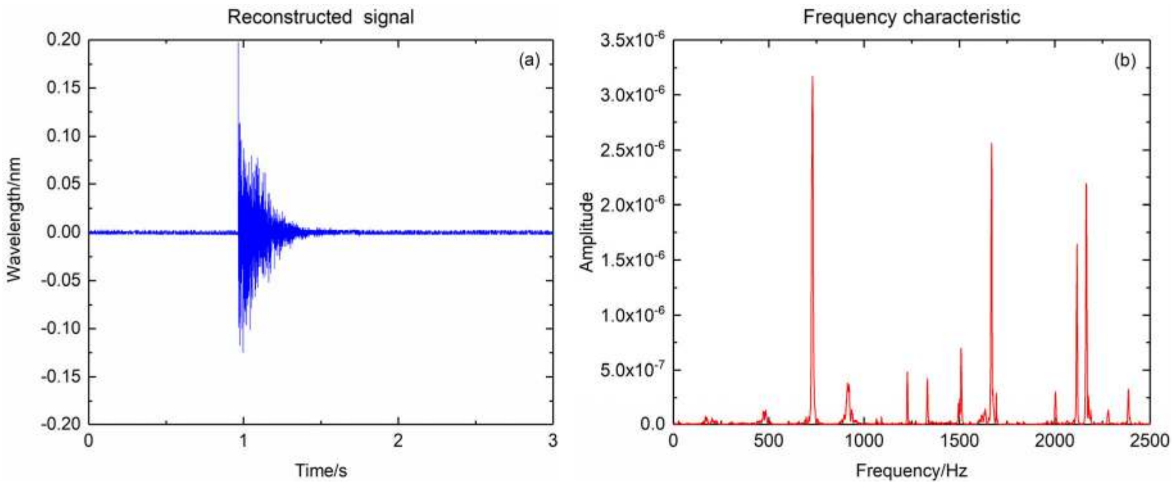

FIGURE 6. Reconstructed signal without the baseline and frequency characteristics in the area 19.

TABLE 7. Comparison of SVM with improved SVM methods on the identification of low velocity impact areas (Time in seconds).

Method	c	g	Mean	Std.	Time
SVM	1	0.0156	1.563E-01	**0.000E+00**	**4.262E-02**
SVM_BA	17.876	0.0105	1.875E-02	5.923E-03	1.720E+02
SVM_BACA	19.156	0.0113	1.823E-02	4.588E-03	1.773E+02
SVM_BAEO	19.338	0.01	1.763E-02	3.178E-03	1.864E+02
SVM_GA	63.037	158.541	1.414E-01	3.185E-02	1.655E+02
SVM_PSO	17.369	0.01	3.125E-02	**0.000E+00**	1.758E+02
SVM_DE	0.550	0.8675	1.406E-01	1.637E-02	1.935E+02
SVM_CSO	13.937	0.0135	2.344E-02	8.235E-03	1.397E+02
SVM_GWO	19.701	0.01	1.719E-02	4.941E-03	1.405E+02
SVM_ALO	21.616	0.01	2.031E-02	7.548E-03	1.770E+02
SVM_HBA	17.940	0.01	1.703E-02	2.895E-02	2.004E+02
SVM_dBA	21.797	0.0103	3.281E-02	1.928E-02	2.155E+02
SVM_ILSSIWBA	0.316	128.8989	7.969E-02	5.031E-02	1.534E+02
SVM_DMBA	17.719	0.0117	**1.615E-02**	2.853E-03	1.923E+02

Fig. 4 (a) shows the low velocity impact signal received by the sensor FBG1 in the monitoring area marked 19, and Fig. 4 (b) shows its frequency characteristic achieved by Fourier transform. We noted that the low velocity impact signal received by the sensor FBG1 was deteriorated by the baseline of 1570.339 nm. Consequently, a wavelet transform method mentioned in Section 2.2 was used to remove the interference of the baseline in the signal.

The original impact signal with the baseline was decomposed through the wavelet transform method in which the wavelet decomposition scale was set to 11 to obtain the high-frequency detail coefficients Cd_i $(i = 1, 2, \cdots, 11)$ and the low-frequency approximation coefficients $Ca_i(i = 1, 2, \cdots, 11)$. The approximation coefficients and the detail coefficients were depicted in Fig. 5. As shown in Fig. 5, the interference of the baseline was involved in the low-frequency wavelet coefficients. Therefore, only the approximation coefficients were set to zero, and the detail coefficients were maintained. Then, the low velocity impact signal without the baseline was obtained by executing the operation of wavelet reconstruction. The low velocity impact signal without the baseline and its frequency characteristic are presented in Fig. 6 (a) and Fig. 6 (b), respectively.

To clearly analyze the relationship between the impact area and the frequency characteristic, the frequency characteristics, which were transformed from the impact signals received by the sensor FBG1 in four symmetric impact areas (19, 22, 43, and 46) and five adjacent impact areas (11, 18, 19, 20, and 27), were depicted in Fig. 7 and Fig. 8. As shown in Fig. 7, the frequency characteristic curves obtained from four symmetric impact areas were similar, but the amplitude values in different impact areas under the same frequency were slightly different. Additionally, Fig. 8 shows the difference between the frequency characteristic curves obtained from five adjacent impact areas, and the difference between the amplitude values in different impact areas under the same frequency can be clearly observed. Thus, extracting the frequency characteristic of the low velocity signal as the feature value is effective for identifying the low velocity impact areas.

3) RESULTS AND DISCUSSION OF IDENTIFYING LOW VELOCITY IMPACT AREAS

There are two crucial components for SVM: the training process and the test process. The training process is to establish the identification model, and the test process is to validate the accuracy of the identification model. Four groups of low velocity impact signals were obtained by using the steel ball to sequentially impact 64 square monitoring areas on the aluminium plate for 4 independent times. Among these groups of signals, a group of signals was randomly selected as the test sample set, whereas the rest of signals were chosen as the training sample set. The frequency characteristics of impact signals received by FBG sensors were used as the input of SVM, and the categories of low velocity impact areas were used as the output of SVM. Since a large number of feature values increased the computational time, the frequency characteristics were extracted from the middle part of impact signals received by four FBG sensors at regular intervals. Then, the number of input feature values was 256.

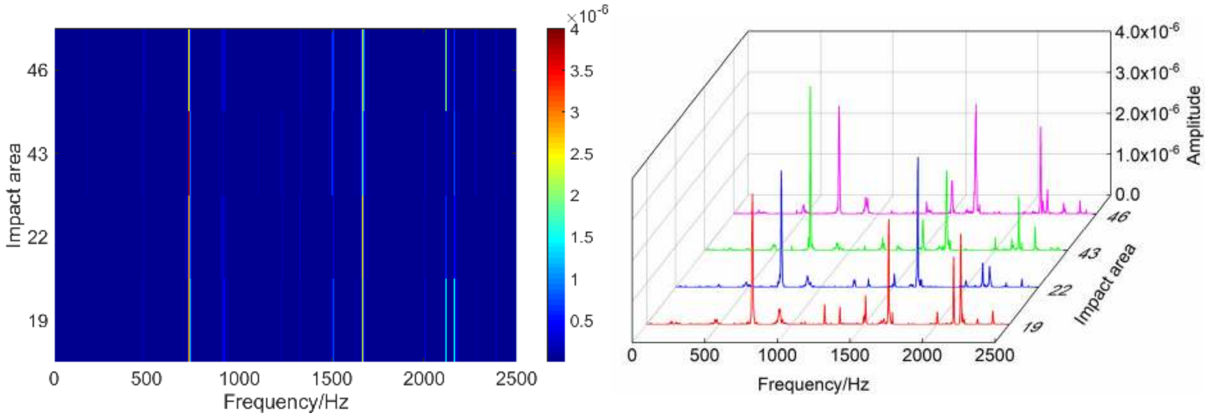

FIGURE 7. Comparison of frequency characteristics of impact signals received by the sensor FBG1 in areas 19, 22, 43, and 46.

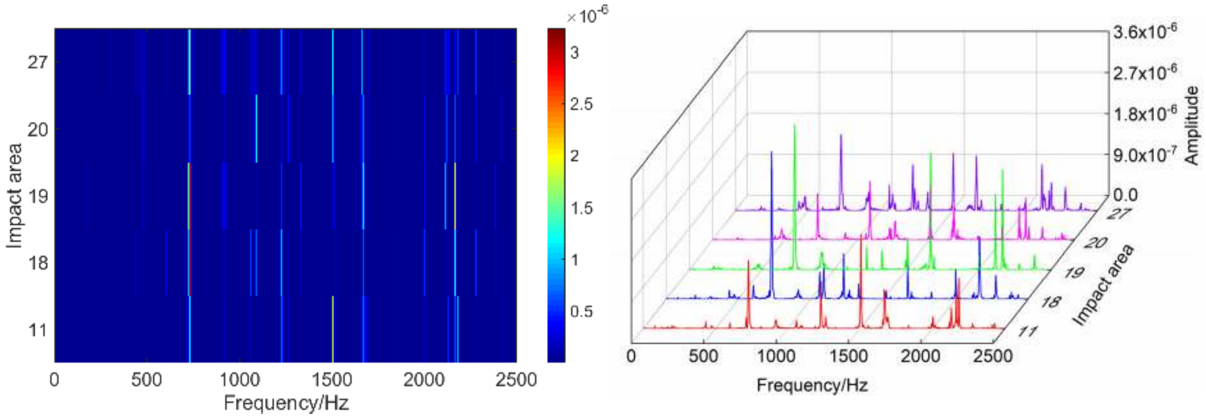

FIGURE 8. Comparison of frequency characteristics of impact signals received by the sensor FBG1 in areas 11, 18, 19, 20, and 27.

To evaluate the performance of SVM_DMBA, the standard SVM, SVM_BA, SVM_BACA, SVM_BAEO, SVM_DE, SVM_PSO, SVM_GA, SVM_CSO, SVM_GWO, SVM_ALO, SVM_HBA, SVM_dBA, and SVM_ILSSIWBA were chosen as the comparative methods. The parameter settings of these methods were the same as mentioned in Section 4.1. The procedures of all methods were carried out 30 times. Table 7 shows the mean value and the standard deviation (Std.) of identification errors. The computation time obtained by these methods is listed at the bottom of Table 7. The best results are bold.

As illustrated by Table 7, compared with the standard SVM, all improved SVM methods obtained better results whereas they took much longer computation time. The identification accuracy and robustness of SVM_DMBA are superior to those of twelve comparative SVM variants, especially SVM_GA and SVM_DE, which demonstrates that DMBA can significantly improve the performance of SVM on identifying the low velocity impact areas. Nevertheless, SVM_DMBA required much longer runtime than SVM, SVM_BA, SVM_BACA, SVM_BAEO, SVM_GA, SVM_PSO, SVM_CSO, SVM_GWO, SVM_ALO, and SVM_ILSSIWBA because of the introduction of BA and two mutation operations.

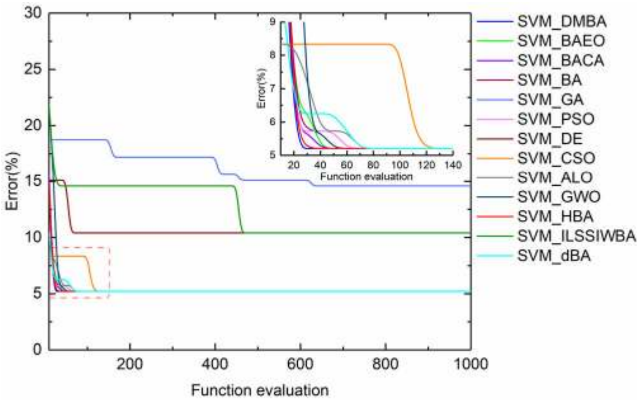

FIGURE 9. Process of optimizing SVM by DMBA and comparative algorithms.

Fig. 9 describes the process of optimizing SVM by using DMBA, three standard algorithms, three advanced algorithms, and six improved bat algorithms. As we observed previously, SVM_DMBA with $c = 17.719$ and $g = 0.0117$

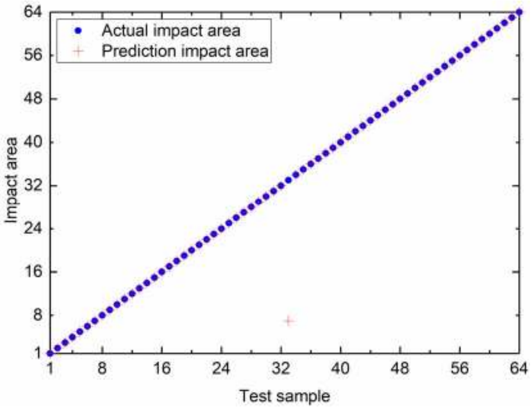

FIGURE 10. Result of identifying the low velocity impact areas by using SVM_DMBA.

converges faster than other improved SVM methods. The identification results of the test sample set obtained by SVM_DMBA are described in Fig. 10, which indicates that SVM_DMBA is an effective method for identifying the low velocity impact areas on the aluminium plate and the average identification error is 1.615%.

V. CONCLUSION

In this paper, a hybrid bat algorithm with double mutation operations (DMBA) is proposed to reinforce the performance of the standard bat algorithm (BA). Two mutation operations are introduced into BA: the Cauchy mutation operator and the extremal optimization mutation operator. The Cauchy mutation operator with the large mutation size focus on extending the search space of the bat during the early iterations, whereas the extremal optimization mutation operator with the small mutation size focus on searching the best solutions during the later iterations. To enhance the accuracy rate of support vector machine (SVM) in the identification problem of low velocity impact areas, a novel SVM based on this hybrid bat algorithm (SVM_DMBA) is further developed. A numerical experiment based on real world benchmark datasets from the UCI data repository is executed to evaluate SVM_DMBA's performance. The standard SVM and twelve improved SVM methods based on the standard algorithms, advanced algorithms, and bat variants are selected as comparative methods. The statistical result of the two-sided Wilcoxon test illustrates that SVM_DMBA is superior to other comparative methods. Furthermore, in a low velocity impact localization system based on FBG sensors, SVM_DMBA is utilized to identify impact areas on an aluminium plate, and the frequency characteristics are extracted as the feature value and used as the input of SVM_DMBA. The statistical results indicate that the proposed SVM_DMBA has the best performance, and the average identification error is 1.615% for 64 low velocity impact areas of 30 mm $\times$ 30 mm on the aluminium plate.

According to the above analysis, there are still a few problems to be solved. SVM_DMBA shows better performance on both the numerical experiment and the identification problem of low velocity impact areas, but it requires longer runtime than the standard SVM. We should enhance SVM's performance without increasing the complexity in the future work. Extracting appropriate features of the low velocity signal is important for improving identification accuracy, and less number of features can effectively reduce the runtime of the methods. The better feature extraction method should be further researched to obtain less and useful feature values. Additionally, compared with the identification problem of single impact source, the identification problem of multiple impact sources is more complex because of mixture signals. Thus, the future work should also focus on the effective signal processing method and the modification of SVM to identify the multiple impact source.

REFERENCES

[1] Q. Wu, F. Yu, Y. Okabe, K. Saito, and S. Kobayashi, "Acoustic emission detection and position identification of transverse cracks in carbon fiber-reinforced plastic laminates by using a novel optical fiber ultrasonic sensing system," *Struct. Health Monit.*, vol. 14, no. 3, pp. 205–213, May 2015, doi: 10.1177/1475921714560074.

[2] S. W. Park, D. H. Kang, H. Bang, S. Park, and C. G. Kim, "Strain monitoring and damage detection of a filament wound composite pressure tank using embedded fiber Bragg grating sensors," *Adv. Nondestruct. Eval. Key Eng. Mater.*, vols. 321–323, pp. 182–185, Dec. 2008, doi: 10.4028/www.scientific.net/KEM.321-323.182.

[3] B. Liu, Y. Ruan, Y. Yu, J. Xi, Q. Guo, J. Tong, and G. Rajan, "Laser self–mixing fiber Bragg grating sensor for acoustic emission measurement," *Sensors*, vol. 18, no. 6, p. 1956, Jun. 2018, doi: 10.3390/s18061956.

[4] D. Pang, Q. Sui, M. Wang, Y. Sai, R. Sun, and Y. Wang, "Acoustic emission source localization system using fiber Bragg grating sensors and a barycentric coordinate–based algorithm," *J. Sensors*, vol. 2018, pp. 1–8, Aug. 2018, doi: 10.1155/2018/9053284.

[5] T. Fu, Y. Liu, K.-T. Lau, and J. Leng, "Impact source identification in a carbon fiber reinforced polymer plate by using embedded fiber optic acoustic emission sensors," *Compos. B, Eng.*, vol. 66, pp. 420–429, Nov. 2014, doi: 10.1016/j.compositesb.2014.06.004.

[6] B. W. Jang, S. O. Park, Y. G. Lee, C. G. Kim, and C. Y. Park, "Detection of impact damage in composite structures using high speed FBG interrogator," *Adv. Compos. Mater.*, vol. 21, no. 1, pp. 29–44, Jul. 2012, doi: 10.1163/156855111X620874.

[7] F. Ciampa and M. Meo, "Impact detection in anisotropic materials using a time reversal approach," *Struct. Health Monitor.*, vol. 11, no. 1, pp. 43–49, Jan. 2012, doi: 10.1177/1475921710395815.

[8] B.-W. Jang, Y.-G. Lee, C.-G. Kim, and C.-Y. Park, "Impact source localization for composite structures under external dynamic loading condition," *Adv. Compos. Mater.*, vol. 24, no. 4, pp. 359–374, Jul. 2015, doi: 10.1080/09243046.2014.917239.

[9] J. Lu, B. Wang, and D. Liang, "Wavelet packet energy characterization of low velocity impacts and load localization by optical fiber Bragg grating sensor technique," *Appl. Opt.*, vol. 52, no. 11, p. 2346, Apr. 2013, doi: 10.1364/ao.52.002346.

[10] S. Lu, M. Jiang, Q. Sui, Y. Sai, and L. Jia, "Low velocity impact localization system of CFRP using fiber Bragg grating sensors," *Opt. Fiber Technol.*, vol. 21, pp. 13–19, Jan. 2015, doi: 10.1016/j.yofte.2014.07.003.

[11] C. Gold and P. Sollich, "Model selection for support vector machine classification," *Neurocomputing*, vol. 55, nos. 1–2, pp. 221–249, Sep. 2003, doi: 10.1016/s0925-2312(03)00375-8.

[12] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 27, Jan. 2011. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[13] X. Wang, S. Guan, L. Hua, B. Wang, and X. He, "Classification of spot-welded joint strength using ultrasonic signal time-frequency features and PSO–SVM method," *Ultrasonics*, vol. 91, pp. 161–169, Jan. 2019, doi: 10.1016/j.ultras.2018.08.014.

[14] H. Zheng, S. Zhang, and X. Sun, "Classification recognition of anchor rod based on PSO–SVM," in *Proc. 29th Chin. Control Decis. Conf. (CCDC)*, Chongqing, China, May 2017, pp. 2207–2212.

[15] X. Yan and M. Jia, "A novel optimized SVM classification algorithm with multi-domain feature and its application to fault diagnosis of rolling bearing," *Neurocomputing*, vol. 313, pp. 47–64, Nov. 2018, doi: 10.1016/j.neucom.2018.05.002.

[16] S.-W. Fei and X.-B. Zhang, "Fault diagnosis of power transformer based on support vector machine with genetic algorithm," *Expert Syst. Appl.*, vol. 36, no. 8, pp. 11352–11357, Oct. 2009, doi: 10.1016/j.eswa.2009.03.022.

[17] M. L. Dantas Dias and A. R. Rocha Neto, "Training soft margin support vector machines by simulated annealing: A dual approach," *Expert Syst. Appl.*, vol. 87, pp. 157–169, Nov. 2017, doi: 10.1016/j.eswa.2017.06.016.

[18] L. Shen, H. Chen, Z. Yu, W. Kang, B. Zhang, H. Li, B. Yang, and D. Liu, "Evolving support vector machines using fruit fly optimization for medical data classification," *Knowl.-Based Syst.*, vol. 96, pp. 61–75, Mar. 2016, doi: 10.1016/j.knosys.2016.01.002.

[19] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization*, J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Eds. Berlin, Germany: Springer, 2010, pp. 65–74.

[20] M. Eckstein, M. Vrabeľ, and I. Maňková, "Application of discrete wavelet decomposition in monitoring of hole-making inconel 718," *Procedia CIRP*, vol. 62, pp. 250–255, Jan. 2017, doi: 10.1016/j.procir.2016.06.023.

[21] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.

[22] S. Ding and L. Chen, "Intelligent optimization methods for high–dimensional data classification for support vector machines," *Intell. Inf. Manage.*, vol. 2, no. 6, pp. 354–364, 2010, doi: 10.4236/iim.2010.26043.

[23] X.-B. Meng, X. Gao, Y. Liu, and H. Zhang, "A novel bat algorithm with habitat selection and Doppler effect in echoes for optimization," *Expert Syst. Appl.*, vol. 42, nos. 17–18, pp. 6350–6364, Oct. 2015, doi: 10.1016/j.eswa.2015.04.026.

[24] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Trans. Evol. Comput.*, vol. 3, no. 2, pp. 82–102, Jul. 1999, doi: 10.1109/4235.771163.

[25] X. J. Yang and Z. G. Huang, "Artificial bee colony with dynamic Cauchy mutation for numerical optimization," *J. Inf. Comput. Sci.*, vol. 8, no. 15, pp. 3371–3376, Dec. 2011.

[26] F. A. P. Paiva, C. R. M. Silva, I. V. O. Leite, M. H. F. Marcone, and J. A. F. Costa, "Modified bat algorithm with cauchy mutation and elite opposition-based learning," in *Proc. IEEE Latin Amer. Conf. Comput. Intell. (LA-CCI)*, Nov. 2017.

[27] S. Boettcher and A. G. Percus, "Extremal optimization: An evolutionary local-search algorithm," in *Computational Modeling and Problem Solving in the Networked World* (Operations Research/Computer Science Interfaces Series), H. K. Bhargava and N. Ye, Eds. Boston, MA, USA: Springer, 2003, pp. 61–77.

[28] M.-R. Chen, J. Weng, X. Li, and X. Zhang, "Handling multiple objectives with integration of particle swarm optimization and extremal optimization," in *Foundations of Intelligent Systems* (Advances in Intelligent Systems and Computing), vol. 277, Z. Wen, and T. Li, Eds. Berlin, Germany: Springer, 2014, pp. 287–297.

[29] M.-R. Chen, X. Li, X. Zhang, and Y.-Z. Lu, "A novel particle swarm optimizer hybridized with extremal optimization," *Appl. Soft Comput.*, vol. 10, no. 2, pp. 367–373, Mar. 2010, doi: 10.1016/j.asoc.2009.08.014.

[30] A. Shintemirov, W. Tang, and Q. Wu, "Power transformer fault classification based on dissolved gas analysis by implementing bootstrap and genetic programming," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 1, pp. 69–79, Jan. 2009, doi: 10.1109/tsmcc.2008.2007253.

[31] X. Tang, X. Xie, B. Fan, and Y. Sun, "A fault–tolerant flow measuring method based on PSO–SVM with transit–time multipath ultrasonic gas flowmeters," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 5, pp. 992–1005, May 2018, doi: 10.1109/tim.2018.2795298.

[32] Y. Kokkinos and K. G. Margaritis, "Simulating parallel scalable probabilistic neural networks via exemplar selection and EM in a ring pipeline," *J. Comput. Sci.*, vol. 25, pp. 260–279, Mar. 2018, doi: 10.1016/j.jocs.2017.07.008.

[33] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA, USA: MIT Press, 1992.

[34] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA, USA: Springer, 2011, pp. 760–766.

[35] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011, doi: 10.1109/tevc.2010.2059031.

[36] X. B. Meng, Y. Liu, X. Gao, and H. Zhang, "A new bio-inspired algorithm: Chicken swarm optimization," in *Advances in Swarm Intelligence* (Lecture Notes in Computer Science), vol. 8794, Y. Tan, Y. Shi, and C. A. C. Coello, Eds. Cham, Switzerland: Springer, 2014, pp. 86–94.

[37] S. Mirjalili, "The ant lion optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015, doi: 10.1016/j.advengsoft.2015.01.010.

[38] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[39] Q. Liu, L. Wu, W. Xiao, F. Wang, and L. Zhang, "A novel hybrid bat algorithm for solving continuous optimization problems," *Appl. Soft Comput.*, vol. 73, pp. 67–82, Dec. 2018, doi: 10.1016/j.asoc.2018.08.012.

[40] C. Gan, W. Cao, M. Wu, and X. Chen, "A new bat algorithm based on iterative local search and stochastic inertia weight," *Expert Syst. Appl.*, vol. 104, pp. 202–212, Aug. 2018, doi: 10.1016/j.eswa.2018.03.015.

[41] A. Chakri, R. Khelif, M. Benouaret, and X.-S. Yang, "New directional bat algorithm for continuous optimization problems," *Expert Syst. Appl.*, vol. 69, pp. 159–175, Mar. 2017, doi: 10.1016/j.eswa.2016.10.050.

[42] S. Yılmaz and E. U. Küçüksille, "A new modification approach on bat algorithm for solving optimization problems," *Appl. Soft Comput.*, vol. 28, pp. 259–275, Mar. 2015, doi: 10.1016/j.asoc.2014.11.029.

[43] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011, doi: 10.1016/j.swevo.2011.02.002.

QI LIU received the B.S. degree in mechanical engineering from the China University of Petroleum, Qingdao, China, in 2016, where he is currently pursuing the Ph.D. degree in mechanical engineering. His research interests include structural health monitoring, neural network algorithm, and intelligent algorithm.

LEI WU received the B.S. and Ph.D. degrees in mechanical engineering from the China University of Petroleum, Dongying, China, in 2010 and 2016, respectively. He is currently a Research Fellow with the School of Maritime Institute, Nanyang Technological University. His research interests include offshore engineering, structural health monitoring, experimental modal analysis, and intelligent algorithm.

FENGDE WANG received the M.S. and Ph.D. degrees in mechanical engineering from the China University of Petroleum, Qingdao, China, in 2018. He is currently a Lecturer with the College of Mechanical and Electronic Engineering, Shandong University of Science and Technology. His research interests are random vibration, structural health monitoring, and intelligent algorithm.

WENSHENG XIAO received the M.S. degree in mechanical engineering from the China University of Petroleum, Dongying, China, in 1996, and the Ph.D. degree in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2004. He is currently a Professor with the College of Mechanical and Electronic Engineering, China University of Petroleum, Qingdao. His research interests include optimization design of permanent magnet motor, offshore engineering, structural health monitoring, and intelligent algorithm.