

## Research Article

# Classification of Marine Mammals Using the Trained Multilayer Perceptron Neural Network with the Whale Algorithm Developed with the Fuzzy System

Ali Hosseini Nejad Takhti <sup>1</sup>, Abbas Saffari <sup>2</sup>, Diego Martín <sup>3</sup>, Mohammad Khishe <sup>4</sup>,  
and Mokhtar Mohammadi <sup>5</sup>

<sup>1</sup>Department of Information Technology, College of Engineering and Computer Science, Sari Branch, Islamic Azad University, Sari, Iran

<sup>2</sup>Department of Electrical Engineering, Imam Khomeini Marine Science University, Nowshahr, Iran

<sup>3</sup>ETSI Telecomunicación, Universidad Politécnica de Madrid, Av. Complutense 30, Madrid 28040, Spain

<sup>4</sup>Department of Electrical Engineering, Imam Khomeini Marine Science University, Nowshahr, Iran

<sup>5</sup>Department of Information Technology, College of Engineering and Computer Science, Lebanese French University, Kurdistan Region, Iraq

Correspondence should be addressed to Abbas Saffari; [abbas.saffari@birjand.ac.ir](mailto:abbas.saffari@birjand.ac.ir)

Received 4 February 2022; Revised 11 September 2022; Accepted 27 September 2022; Published 18 October 2022

Academic Editor: Jianli Liu

Copyright © 2022 Ali Hosseini Nejad Takhti et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The existence of various sounds from different natural and unnatural sources in the deep sea has caused the classification and identification of marine mammals intending to identify different endangered species to become one of the topics of interest for researchers and activist fields. In this paper, first, an experimental data set was created using a designed scenario. The whale optimization algorithm (WOA) is then used to train the multilayer perceptron neural network (MLP-NN). However, due to the large size of the data, the algorithm has not determined a clear boundary between the exploration and extraction phases. Next, to support this shortcoming, the fuzzy inference is used as a new approach to developing and upgrading WOA called FWOA. Fuzzy inference by setting FWOA control parameters can well define the boundary between the two phases of exploration and extraction. To measure the performance of the designed categorizer, in addition to using it to categorize benchmark datasets, five benchmarking algorithms CVOA, WOA, ChOA, BWO, and PGO were also used for MLPNN training. The measured criteria are concurrency speed, ability to avoid local optimization, and the classification rate. The simulation results on the obtained data set showed that, respectively, the classification rate in MLPFWOA, MLP-CVOA, MLP-WOA, MLP-ChOA, MLP-BWO, and MLP-PGO classifiers is equal to 94.98, 92.80, 91.34, 90.24, 89.04, and 88.10. As a result, MLP-FWOA performed better than other algorithms.

## 1. Introduction

The deep oceans make up 95% of the oceans' volume, which is the largest habitat on Earth[1]. Creatures are continually being explored in the depths of the ocean with new ways of life [2, 3]. Much research has been conducted in the depths of the ocean, but unfortunately, this research is not enough, and many hidden secrets in the ocean remain unknown [4].

Various species of marine mammals, including whales and dolphins, live in the ocean. Underwater audio signal processing is the newest way to measure the presence, abundance, and migratory marine mammal patterns [5–7]. The use of human-based methods and intelligent methods is one method of recognizing whales and dolphins [8]. Initially, human operator-based methods were used to identify whales and dolphins. Its advantages include simplicity and ease of work. However, the main

disadvantage is the dependence on the operator's psychological state and inefficiency in environments where the signal-to-noise ratio is low [9].

To eliminate these defects, automatic target recognition (ATR) based on soft calculations is used [10, 11]. Then, contour-based recognition was used for recognition of whales and dolphins due to its time complexity and low identification rate [12, 13, 14]. The next item from the subset of intelligent methods is the ATR method based on soft computing [15], which is wildly popular due to its versatility and parallel structure [16, 17].

The MLP-NN neural network, due to its simple structure, high performance, and low computational complexity, has become a useful tool for automatically recognizing targets [18–20]. In the past, MLP-NN training used gradient-based methods and error propagation, but these algorithms had a low speed of convergence and were stuck in local minima [21–23].

Therefore, this paper presents a new hybrid method of MLP-NN training using FWOA to classify marine mammals. The main contributions of this work are as follows:

- (i) Practical test design to obtain a real data set from the sound produced by dolphins and whales
- (ii) Classifier design using MLP-NN to classify dolphins and whales
- (iii) MLP-NN training using the proposed FWOA hybrid technique
- (iv) MLP-NN training using new metaheuristic algorithms (CVOA, ChOA, BWO, PGO) and WOA as benchmark algorithms

In the following paragraphs, the paper is structured in such a manner that Section 2 designs an experiment for data collection. Section 3 will cover how to extract a feature. Section 4 describes WOA and how to fuzzy. Section 5 will simulate and discuss it, and finally, Section 6 will conduct conclusions and recommendations.

*1.1. Background and Related Work.* MLP-NNs are a commonly used technology in the field of soft computing [11, 24]. These networks may be used to address nonlinear issues. Learning is a fundamental component of all neural networks and is classified as unsupervised and supervised learning. In most cases, back-propagation techniques or standard [25, 26] are also used as a supervised learning approach for MLP-NNs. Back-propagation is a gradient-based technique with limitations such as slower convergence, making it unsuitable for real world applications. The primary objective of the neural network learning mechanism is to discover the optimal weighted edge and bias combination that produces the fewest errors in network training and test samples [27, 28]. Nevertheless, the majority of MLP-NN faults will remain high for an extended period of time throughout the learning process, after which they will be reduced by the learning algorithm. This is particularly prevalent in mechanisms relying on gradients, such as back-propagation algorithms. In addition, the back-propagation algorithm's convergence is highly dependent on the initial values of the learning rate and the magnitude of the motion.

Incorrect values for these variables may potentially result in algorithm diverging. Numerous research works have been conducted to address this issue using the back-propagation algorithm [29], but there is not enough optimization that has occurred, and each solution has unintended consequences. We have seen an increase in the usage of meta-heuristic methods for neural network training in recent years. The following (Table 1) discusses many works on neural network training using different meta-heuristic techniques.

GA and SA are likely to minimize local optimization but have a slower convergence rate. This is inefficient in real-time processing applications. Although PSO is quicker than evolutionary algorithms, it often cannot compensate for poor solution quality by increasing the number of iterations. PSOGSA is a fairly sophisticated algorithm, and its performance is insufficient for solving problems with a high dimension. BBO requires lengthy computations. Despite their minimal complexity and rapid convergence speed, GWO, SCA, and IWT fall victim to local optimization and so are not appropriate for applications requiring global optimization. The primary cause for being stuck in local optimizations is a mismatch between the exploration and extraction stages. Various methods are provided to solve problems such as getting stuck in local optimizations and slow convergence speed in WOA, including parameterization  $\alpha$  by the linear control strategy (LCS) and arcsine-based nonlinear control strategy (NCS-arcsine) to establish the right balance between exploration and extraction [51, 52]. LCS and NCS-arcsine strategies usually do not provide appropriate solutions when used for high-dimensional problems.

On the other hand, the no free lunch (NFL) theorem logically states that meta-heuristic algorithms do not have the same answer in dealing with different problems [29]. Due to the problems mentioned and considering the NFL theory in this article, a fuzzy whale algorithm called Fuzzy-WOA is introduced for the MLP-NN training problem to identify whales and dolphins.

To investigate the performance of the FWOA, we design an underwater data accusation scenario, create an experimental dataset, and compare it to a well-known benchmark dataset (Watkins et al. 1992). To address the time-varying multipath and fluctuating channel effects, a unique two cepstrum liftering feature extraction technique is used.

## 2. The Experiment Design and Data Acquisition

As shown in Figure 1, to obtain a real data set of sound produced by dolphins and whales from a research ship called the Persian Gulf Explorer and a Sonobuoy, a UDAQ\_Lite data acquisition board and three hydrophones (Model B&k 8013) were obtained and were used with equal distance to increase the dynamic range. This test was performed in Bushehr port. The array's length is selected based on the water depth, and Figure 2 shows the hydrophones' location.

The raw data included 170 samples of pantropical spotted dolphin (8 sightings), 180 samples of spinner dolphin (five sightings), 180 cases of striped dolphin (eight

TABLE 1: Some related studies.

Paper	Type	Application	Training algorithms	Year
[30]	Feed-forward NN	Sonar image classification	Genetic algorithm (GA)	1989
[31]	MLP NN	Magnetic body detection in a magnetic field	Simulated annealing (SA)	1994
[32]	MLP NN	Predicting the solubility of gases in polymers	Particle swarm optimizer (PSO)	2013
[33]	MLP NN	Parkinson's disease diagnosis	Social-spider optimization (SSA)	2014
[34]	ANN	Artificial intelligence	Runge Kutta method	2015
[35]	MLP NN	XOR, heart, iris, balloon, and breast cancer dataset	Moth-flame optimization (MFO)	2016
[36]	MLP NN	Sonar dataset classification	Gray wolf optimization (GWO)	2016
[37]	Feed-forward NN	Sonar dataset classification	Particle swarm optimizer (PSO)	2017
[38]	Algorithm improvement	23 benchmark functions and solving infinite impulse response model identification	Lévy flight trajectory-based WOA (LWOA)	2017
[39]	MLP NN	Big data	Biogeography-based optimization (BBO)	2018
[40]	MLP NN	UCI dataset (benchmark)	Monarch butterfly optimization (MBO)	2018
[41]	Algorithm improvement	Five standard engineering optimization problems	LWOA	2018
[42]	MLP NN	Sonar target classification	Dragonfly optimization algorithm (DOA)	2019
[42]	MLP NN	Sonar dataset classification	Improved whale	2019
[43]	MLP NN	Classification of EEG signals	PSOGSA	2020
[44]	ANN	Prediction of urban stochastic water demand	Slime mould algorithm (SMA)	2020
[45]	Algorithm improvement	Optimize star sensor calibration	Hybrid WOA-LM algorithm	2020
[46]	Feed-forward NN	COVID19 detection	Colony predation algorithm (CPA)	2020
[47]	MLP NN	XOR, balloon, Iris, breast cancer, and heart	Harris hawks optimization (HHO)	2020
[48]	Deep convolutional NN	COVID19 detection	Sine-cosine (SCA)	2021
[49]	ANN	Predicting ground vibration	Hunger games search optimization (HGS)	2021
[50]	MLP NN	Sonar dataset classification	Fuzzy grasshopper optimization algorithm (FGOA)	2022



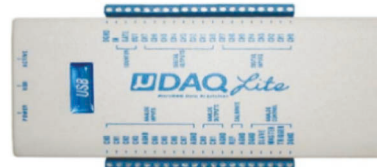
(a)



(b)



(c)



(d)

FIGURE 1: Items needed to collect data sets in: (a) Persian gulf explorer. (b) Research sonobuoys. (c) Hydrophone model 8103 of B&K company. (d) UDAQ\_Lite data collection board.

sightings), 105 cases of humpback whale (seven sightings), 95 samples of minke whale (five sightings), and 120 samples of the sperm whale (four sightings). The experiment was developed and performed in the manner shown in Figure 2.

2.1. *The Ambient Noise Reduction and Reverberation Suppression.* For example, the sounds emitted by marine mammals (dolphins and whales) recorded by the hydrophone array are considered  $x(t)$ ,  $y(t)$ ,  $z(t)$ , and the original

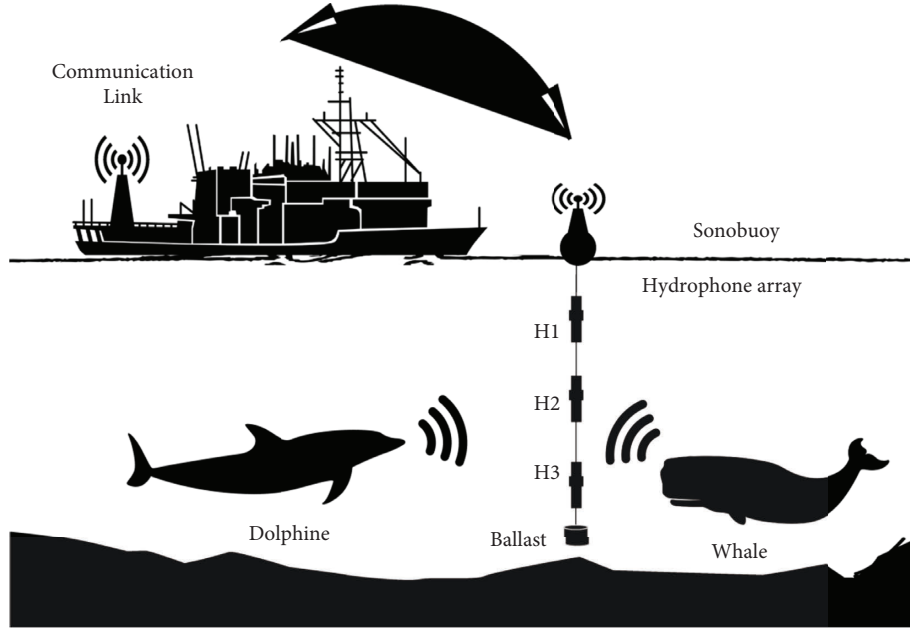


FIGURE 2: Test scenario and location of the hydrophones.

sound of dolphins and whales is considered  $s(t)$ . The mathematical model of the output of hydrophones is in

$$\begin{aligned} x(t) &= \int_{-\infty}^t h(t-\tau)s(\tau), \\ y(t) &= \int_{-\infty}^t g(t-\tau)s(\tau), \\ z(t) &= \int_{-\infty}^t q(t-\tau)s(\tau). \end{aligned} \quad (1)$$

In equation (1), the Environment Response Functions (ERF) are denoted by  $h(t)$ ,  $g(t)$ , and  $q(t)$ . ERFs are not known, and “tail” is considered uncorrelated [53], and naturally, the first frame of sound produced by marine mammals does not reach the hydrophone array at one time. Due to the sound pressure level (SPL) in the Hydrophone B&K 8103 and reference, which deals with the underwater audio standard, the recorded sounds must be preamplified by a factor of  $10^6$ .

The frequency domain SPL is then transformed using the Hamming window and fast Fourier transform (FFT). Following that, (2) reduces the frequency bandwidth to 1 Hz.

$$SPL_1 = SPL_m - 10 \log \Delta f. \quad (2)$$

$SPL_m$  is the obtained SPL at each fundamental frequency center in dB; re  $1 \mu\text{Pa}$ ,  $SPL_1$  is the SPL reduced to 1 Hz bandwidth in dB; re  $1 \mu\text{Pa}$ , and  $\Delta f$  represents the bandwidth for each 1/3 Octave band filter. To reduce the square mean error (MSE) between ambient noise and marine mammal noise, a Wiener filter was utilized [54]. Following that, the results were computed using (3) to identify sounds with a

low SNR, less than 3 dB, that should be eliminated from the database.

$$SPL_v = 10 \log \left[ 10 \left( \frac{SPL_T}{10} \right) - 10 \left( \frac{SPL_A}{10} \right) \right], \quad (3)$$

where  $T$ ,  $V$ , and  $A$  represent all the available signals, sound, and ambient sound, respectively. After that, the SPLs were recalculating at a standard measuring distance of 1 m as follows:

$$SPL = SPL_1 + 20 \log(r). \quad (4)$$

Figure 3 illustrates the block diagram for ambient noise reduction and reverberation suppression.

In the next part, the effect of reverberation must be removed. In this regard, the common phase is added to the band (reducing the phase change process using the delay between the cohesive parts or the initial sound is called the common phase) [55]. Therefore, a cross-correlation pass function by adjusting each frequency band’s gain eliminates noncorrelated signals and passes the correlated signals. Finally, the output signals from each frequency band are merged to form the estimated signal, i. e.,  $\hat{S}$ . The basic design for removing reverberation is shown in Figure 4. Figure 5 illustrates typical representations of dolphin and whale sounds and melodies, as well as their spectra.

### 3. Average Cepstral Features and Cepstral Liftering Features

The effect of ambient noise and reverberation decreases after detecting the audio signal frames obtained in the

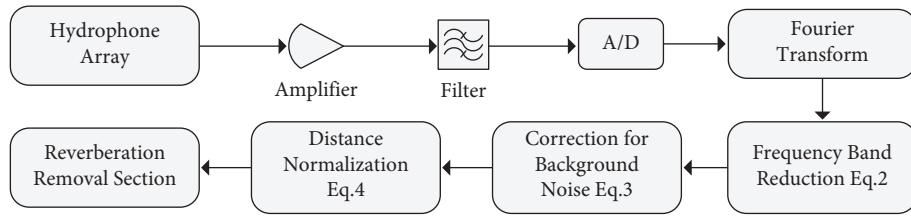


FIGURE 3: The block diagram of the ambient noise reduction and reverberation suppression system.

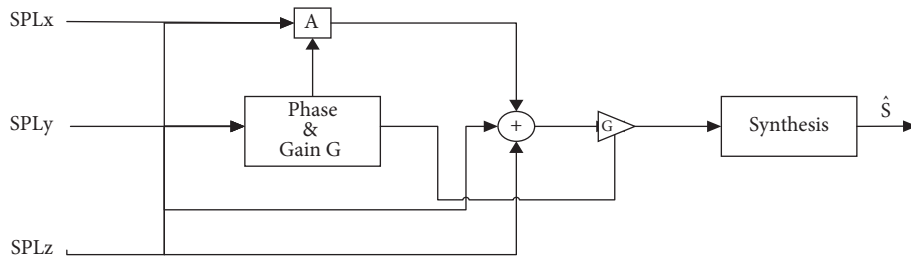


FIGURE 4: The reverberation removal section's block diagram.

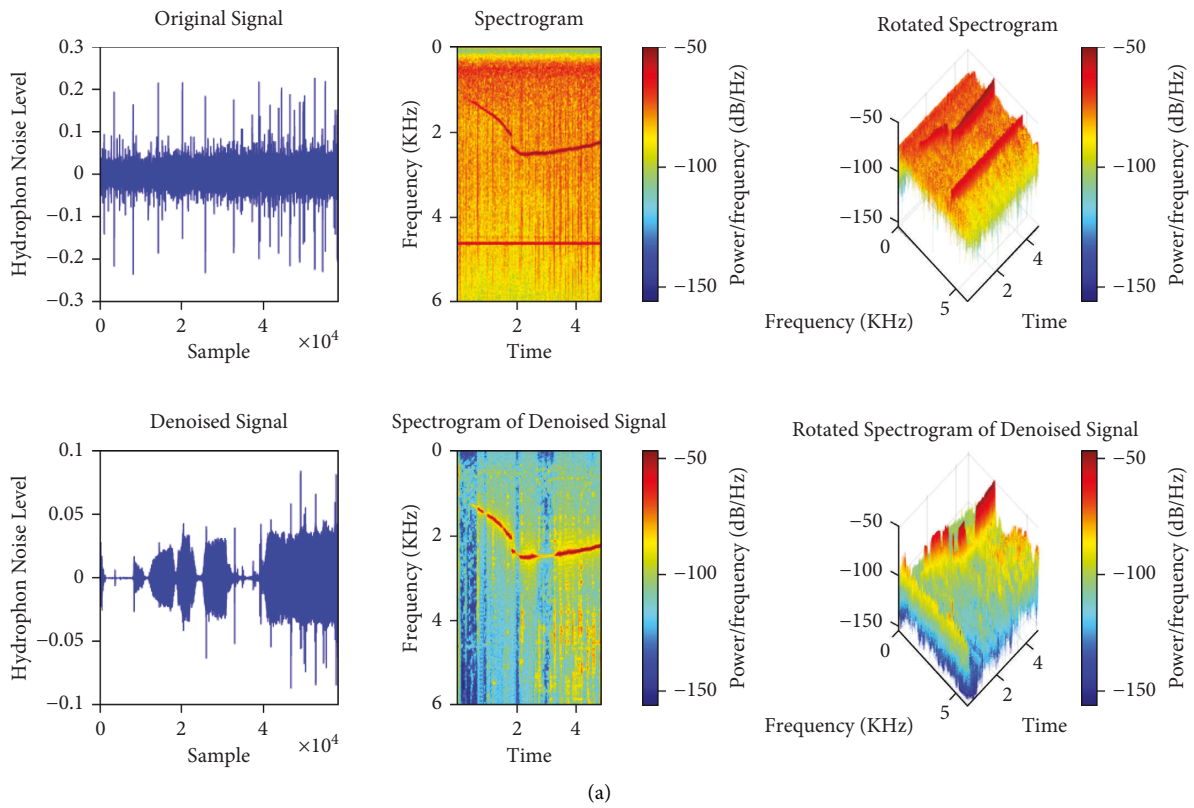


FIGURE 5: Continued.

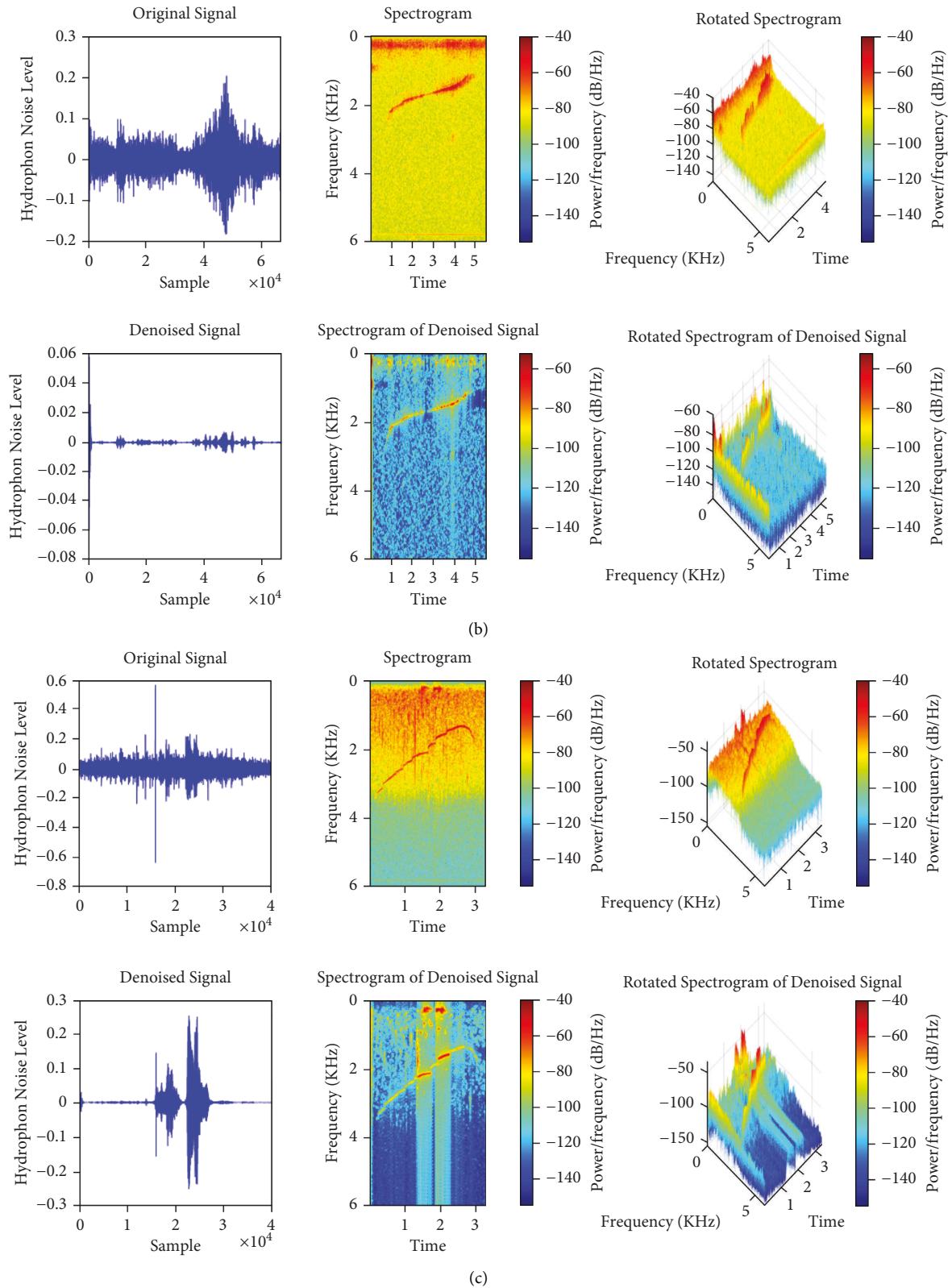
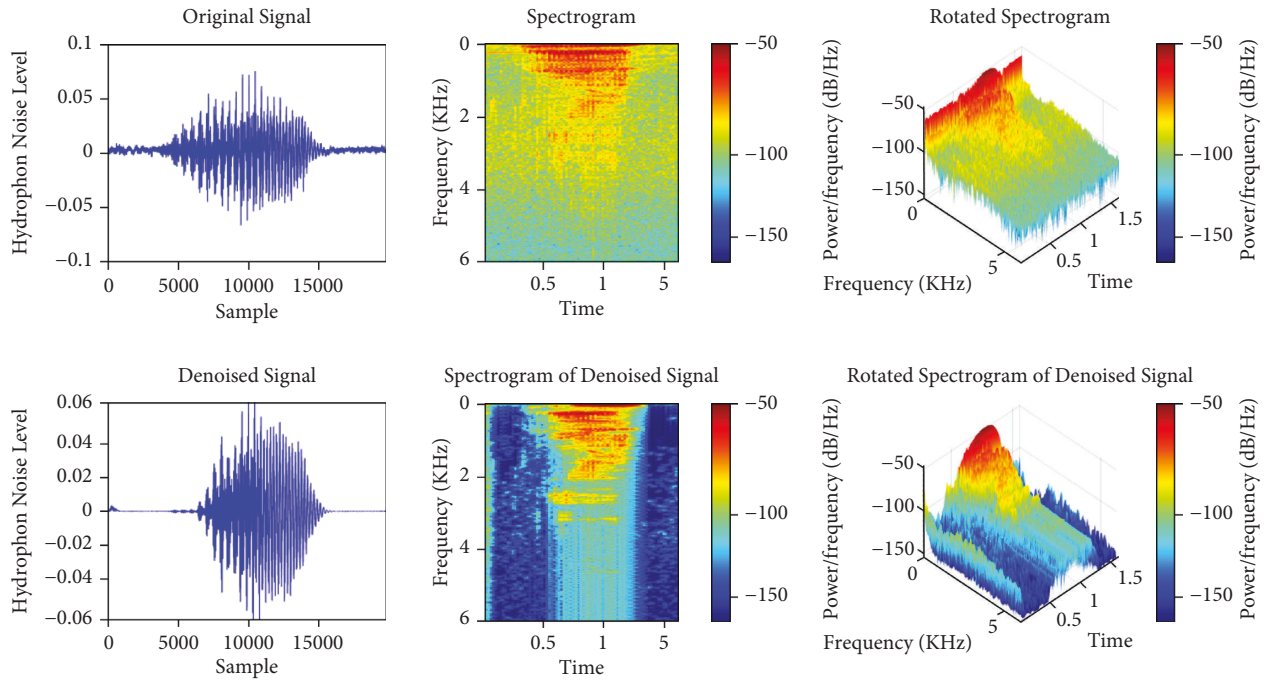
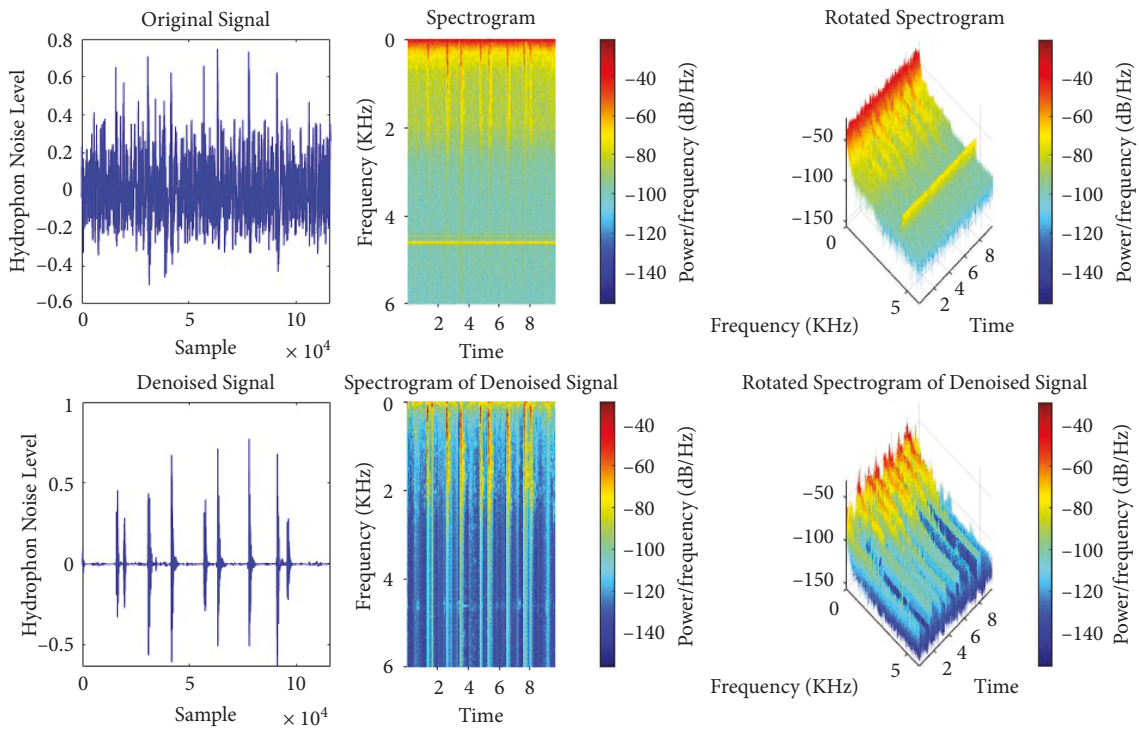


FIGURE 5: Continued.



(d)



(e)

FIGURE 5: Continued.

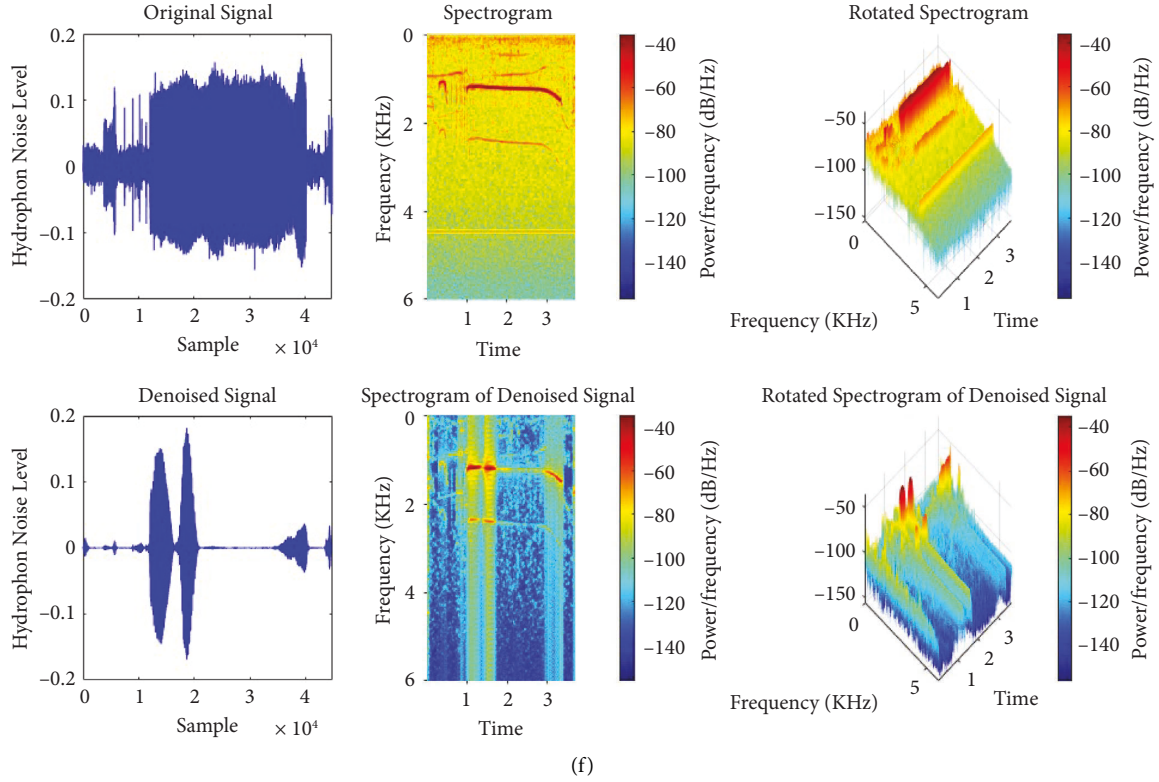


FIGURE 5: Typical sound presentations produced by dolphins and whales and their spectrogram. (a) Pantropical spotted dolphin. (b) Spinner dolphin. (c) Striped dolphin. (d) Humpback whale. (e) Minke whale. (f) Sperm whale.

preprocessing stage. In the next step, the detected signal frames enter the feature extraction stage. The sounds made by dolphins and whales emitted from a distance to the hydrophone experience changes in size, phase, and density. Due to the time-varying multipath phenomenon, fluctuating channels complicate the challenge of recognizing dolphins and whales. The cepstral factors combined with the cepstral liftering feature may considerably reduce the impacts of multipath, whilst the average cepstral coefficients can significantly minimize the time-varying effects of shallow underwater channels [56]. As a result, this section recommends the use of cepstral features, such as mean cepstral features and cepstral liftering features, in order to construct a suitable data set. Cepstrum indices of greater and lower values indicate that the channel response cepstrum and the original sound of dolphins and whale cepstrum are distinct. [57]. They are located in distinct regions of the liftering cepstrum. Therefore, by reducing time liftering, the quality of the features is increased. Following removing noise and reverberation, the frequency domain frames of SPLs ( $S(k)$ ) are passed to the portion extracting cepstrum features. The following equation determines the cepstrum characteristics of the sound produced by dolphin and whale signals.

$$c(n) = \sum_{l=1}^M \log \left( \sum_{k=0}^{N-1} |S(k)|^2 H_l(k) \right) \cos(nl / (1 - 0.5)\pi/M), \quad (5)$$

where  $S(k)$  indicates the frequency domain frames of sounds generated by dolphins and whales,  $N$  signifies the number of discrete frequencies employed in the FFT, and  $H_l(k)$  denotes the transfer function of the Mel-scaled triangular filter with  $l = 0, 1, \dots, M$ . Ultimately, using the discrete cosine transform (DCT), the cepstral coefficients are converted to the time domain as  $c(n)$ .

As previously stated, the sound generated by dolphins and whales is obtained via a method called low-time liftering. Consequently, (6) is recommended to separate the sound that originated from the whole sound.

$$w_e[n] = \begin{cases} 1, & 0 \leq n \leq L_c, \\ 0, & L_c \leq n \leq \frac{N}{2}. \end{cases} \quad (6)$$

$L_c$  indicates the liftering window's length, which is typically 15 or 20. The ultimate features may be computed by multiplying the cepstrum  $c(n)$  by and using the logarithm and DFT functions as described in the following equations:

$$c_e(n) = w_e[n]c(n), \quad (7)$$

$$c_L(n) = \text{Log}(DFT(c_e(n))). \quad (8)$$

Finally, the feature vector would be represented using

$$x_m = [c_L(0), c_L(1), \dots, c_L(P-1)]^T. \quad (9)$$



The first 512-cepstrum points (out of 8192 points in one frame for a sampling rate of 8192 Hz, expect for zeroth index  $\{c_y^m[0]\}$ ) are corresponded to 62.5 ms liftering coefficients and are windowed from the  $N$  indices, which is equivalent to one frame length to reduce the liftering coefficients to 32 features. Prior to averaging, the duration of subframes is five seconds. Ten prior frames compose 50 s average cepstral features throughout the averaging liftering technique, smoothing 10 frame results in the final average cepstral features. As a result, the average cepstral feature vector has 32 elements. The  $X_m$  vector would then be used as an input signal for an MLP-NN in the subsequent phase.

To summarize, the number of inputs to a neural network equals  $P$ . The whole feature extraction step is shown in Figure 6. To summarize, Figure 7 depicts the result of this step.

#### 4. Design of an FWOA-MLPNN for Automatic Detection of Sound Produced by Marine Mammals

MLP-NN is the simplest and most widely used neural network [58, 59]. Important applications of MLP-NN include automatic target recognition systems. For this reason, this article uses MLP-NN as an identifier [60]. MLP-NN is amongst the most durable neural networks available and is often used to model systems with a high degree of non-linearity. In addition, the MLP-NN is a feed-forward network able of doing more precise nonlinear fits. Despite what has been said, one of the challenges facing MLP-NN is always training and adjusting the edges' bias and weight [61].

The steps for using meta-heuristic algorithms to teach MLPNN are as follows: the first phase is to determine how to display the connection weights. The second phase involves evaluating the fitness function in order to determine the connection weights, which may be thought of as the mean square error (MSE) for recognition issues. The third step employs the evolutionary process to minimize the fitness function, that is represented by the MSE. Figure 8 and equation (10) illustrate the technical design of the evolutionary technique for connection weight training.

$$\vec{V} = \{\vec{W}, \vec{\theta}\} = \{W_{11}, W_{12}, \dots, W_{in}, b_1, b_2, \dots, b_n, M_{11}, M_{12}, \dots, M_{hm}\}, \quad (10)$$

where  $n$  represents the input nodes' amount,  $W_{ij}$  indicating the node's connection weight to the  $j^{th}$  node,  $\theta_j$  denotes the bias (threshold) of the  $j^{th}$  hidden neuron.

As noted before, the MSE is a frequently used criterion for assessing MLP-NNs, as the following equation demonstrates.

$$MSE = \sum_{i=1}^m (o_i^k - d_i^k)^2, \quad (11)$$

where  $m$  is the number of neurons in the MLP outputs,  $d_i^k$  is the optimal output of the  $i^{th}$  input unit in cases where  $k^{th}$  training sample is utilized, and  $o_i^k$  denotes the real output of the  $i^{th}$  input unit in cases where the  $k^{th}$  training sample is

observed in the input. To be successful, the MLP must be tuned to a collection of training samples. As a result, MLP performance is calculated as follows:

$$E = \sum_{k=1}^T \sum_{i=1}^m (o_i^k - d_i^k)^2 / T. \quad (12)$$

$T$  denotes the number of training samples,  $d_i^k$  denotes the optimal output related to  $i^{th}$  input when using  $k^{th}$  the training sample,  $m$  denotes the number of outputs, and  $O_i^k$  indicates the input's real output when using  $k^{th}$  the training sample. Finally, the recognition system requires a meta-heuristic method to fine-tune the parameters indicated above. The next subsection proposes an instructor based on an improved whale optimization algorithm (WOA) with fuzzy logic called FWOA.

**4.1. Fuzzy WOA.** This section upgrades WOA using fuzzy inference. In this regard, in the first subsection, it will review WOA, and in the second subsection, it will describe the fuzzy method for upgrading WOA.

**4.1.1. Whale Optimization Algorithm.** The WOA optimization algorithm was introduced in 2016, inspired by the way whales were hunted by Mirjalili and Lewis [62]. WOA starts with a set of random solutions. In each iteration, the search agents update their position by using three operators: encircling prey, bubble-net attack (extraction phase), and bait search (exploration phase). In encircling prey, whales detect prey and encircle it. The WOA assumes that the best solution right now is his prey. Once the optimal search agent is discovered, all other search agents will update their position to that of the optimal search agent. The following equations express this behavior:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}^*(t) - \vec{X}(t) \right|, \quad (13)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}, \quad (14)$$

where  $t$  is the current iteration,  $\vec{A}$  and  $\vec{C}$  are the coefficient vectors,  $(\vec{X}^*)$  is the place vector which is the best solution so far, and  $\vec{X}$  is the place vector. It should be noted that in each iteration of the algorithm, if there is a better answer,  $(\vec{X}^*)$  should be updated. The vectors  $\vec{A}$  and  $\vec{C}$  are obtained using the following equations:

$$\vec{A} = 2\vec{\alpha} \cdot \vec{r} - \vec{\alpha}, \quad (15)$$

$$\vec{C} = 2 \cdot \vec{r}, \quad (16)$$

where  $\vec{\alpha}$  decreases linearly from 2 to zero during repetitions and  $\vec{r}$  is a random vector in the distance  $[0, 1]$ . During the bubble-net assault, the whale swims simultaneously around its victim and along a contraction circle in a spiral pattern. To describe this concurrent behavior, it is anticipated that the whale would adjust its location during optimization using either the contractile siege mechanism or the spiral model. (17) is the mathematical model for this phase.

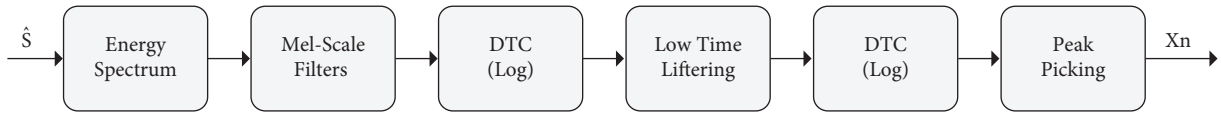


FIGURE 6: The procedure of extracting cepstrum liftering features.

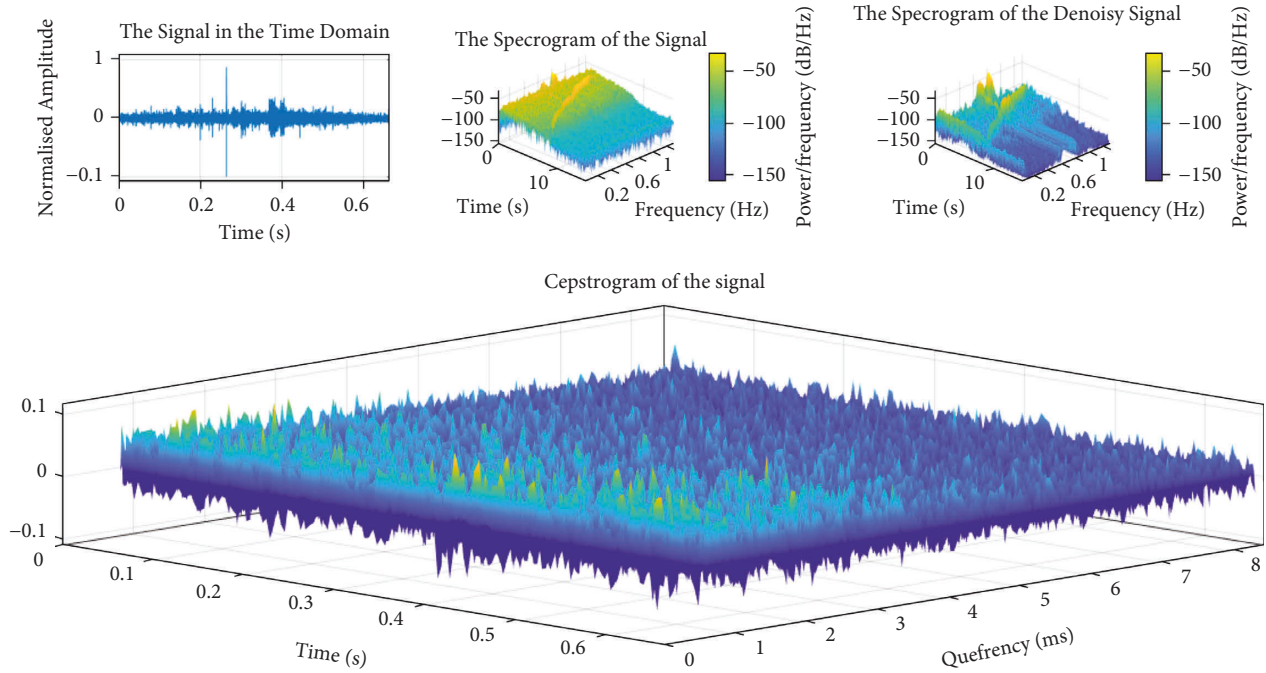


FIGURE 7: The typical visualization produces sound cepstrum features (striped dolphin).

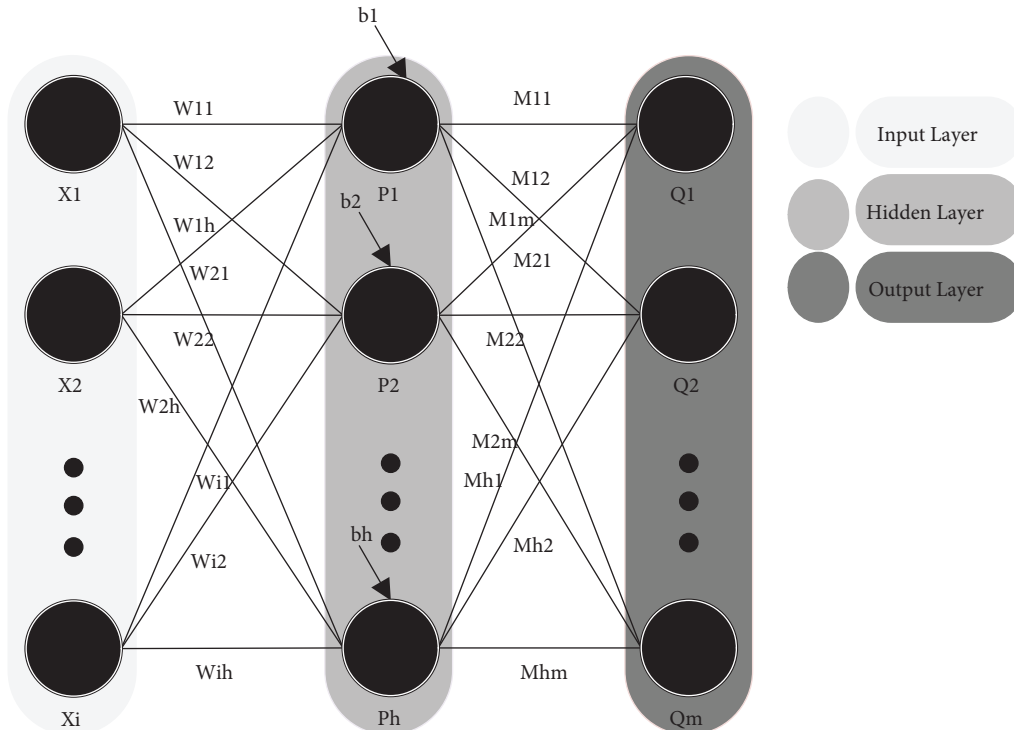


FIGURE 8: MLP-NN as a search agent for a meta-heuristic method.

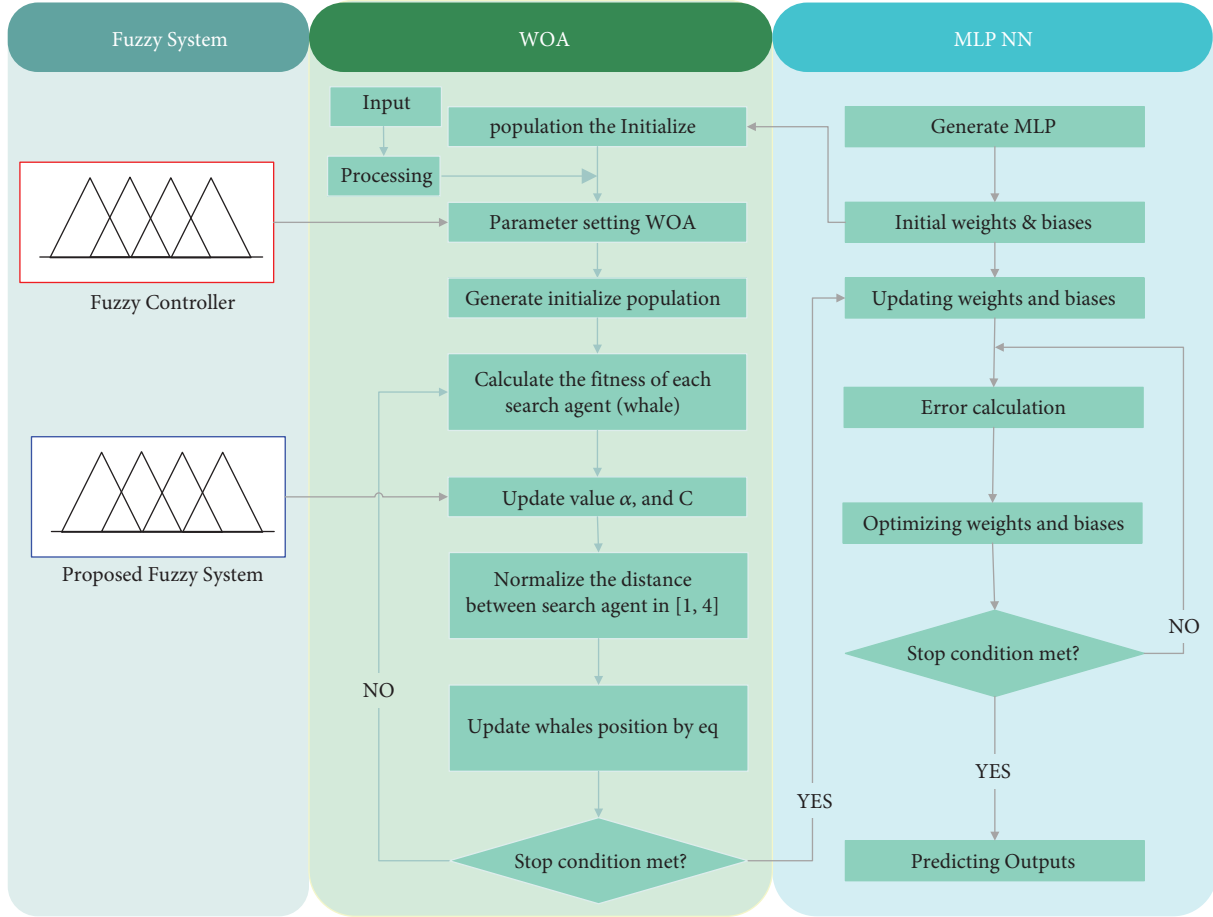


FIGURE 9: Flowchart of the WOA.

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D}, & \text{if } p < 0.5, \\ \vec{D} \cdot e^{bl} \cdot \cos(2\pi l), & \text{if } p \geq 0.5, \end{cases} \quad (17)$$

where  $\vec{D}$  is derived from (8) and denotes the distance  $i$  between the whale and its prey (the best solution ever obtained). A constant  $b$  is used to define the logarithmic helix shape;  $l$  is a random number between  $-1$  and  $+1$ .  $p$  is a random number between zero and one. Vector  $A$  is used with random values between  $-1$  and  $1$  to bring search agents closer to the reference whale. In the search for prey to update the search agent's position, random agent selection is used instead of using the best search agent's data. The mathematical model is in the form of the following equations:

$$\vec{D} = |\vec{C}| \cdot \vec{X}_{rand} - \vec{X}, \quad (18)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D}. \quad (19)$$

$\vec{X}_{rand}$  is the randomly chosen position vector (random whale) for the current population, and vector  $A$  is utilized with random values larger or smaller to one to drive the search agent away from the reference whale. Figure 9 shows the FWOA flowchart, and Figure 10 shows the pseudocode

of the FWOA. In the next section, we will describe the proposed fuzzy system.

**4.1.2. Proposed Fuzzy System for Tuning Control Parameters.** The proposed fuzzy model receives the normalized performance of each whale in the population (normalized fitness value) and the current values of the parameters  $\vec{\alpha}$  and  $\vec{C}$ . The output also shows the amount of change using the symbols  $\Delta\alpha$  and  $\Delta C$ . The NFV value for each whale is obtained as follows:

$$NFV = \frac{fitness - fitness_{min}}{fitness_{min} - fitness_{max}}. \quad (20)$$

The NFV value is in the range of  $[0, 1]$ . This paper's optimization problem is of the minimization type, in which the fitness of each whale is obtained directly by the optimal amount of these functions. (21) and (22) update the parameters  $\vec{\alpha}$  and  $\vec{C}$  for each whale which are as follows:

$$\vec{\alpha}^{t+1} = \vec{\alpha}^t + \Delta\alpha, \quad (21)$$

$$\vec{C}^{t+1} = \vec{C}^t + \Delta C. \quad (22)$$

```

Set the initial values of the whale's population size  $n$ , coefficient vectors  $A$ , and the maximum number of
iterations
Fuzzification parameters  $a$  and  $C$ 
Calculate the fitness of each search agent
Calculate the NFV by the Eq. (20)
Calculate the parameters  $a$  and  $C$  using fuzzy inference system Mamdani
Defuzzification parameters  $a$  and  $C$ 
 $X^*$  = the best search agent
while ( $t <$  maximum number of iterations)
  for each search agent
    Update  $a$ ,  $A$ ,  $C$ ,  $l$  and  $p$ 
    If1 ( $p < 0.5$ )
      If2 ( $|A| < 1$ )
        Update the position of the current search agent by the Eq. (13)
      else if2 ( $|A| \geq 1$ )
        Select a random search agent ( $X_{rand}$ )
        Update the position of the current search agent by the Eq. (19)
      end if2
    else if1 ( $p \geq 0.5$ )
      Update the position of the current search by the Eq. (17)
    end if1
  end for
  Check if any search agent goes beyond the search space and amend it to calculate the fitness of each
  search agent
  Update  $X^*$  if there is a better solution
   $t=t+1$ 
end while
return  $X^*$ 

```

FIGURE 10: Pseudocode of the FWOA.

The fuzzy system is responsible for updating the parameters  $\vec{\alpha}$  and  $\vec{C}$  of each member of the population (whale) and the three inputs of this system are the current value of parameters  $\vec{\alpha}$  and  $\vec{C}$ , NFV. Initially, these values are “fuzzification” by membership functions. Then, their membership value is obtained using  $\mu$ . These values apply to a set of rules and give the values  $\Delta\alpha$  and  $\Delta C$ . After determining these values, the “defuzzification” process is performed to estimate the numerical values  $\Delta\alpha$  and  $\Delta C$ . Finally, these values are applied in (12) and (13) to update the parameters  $\Delta\alpha$  and  $\Delta C$ . The fuzzy system used in this article is of the Mamdani type. Figure 11 shows the proposed fuzzy model and membership functions used to adjust the whale algorithm’s control parameters. The adjustment range for membership functions is obtained using the primary [62] of the WOA. Many experiments were performed for all types of membership functions, including trimf, trapmf, gbellmf, gaussmf, gauss2mf, sigmf, dsigmf, psigmf, pimf, smf, and zmf. Comparison of the results showed that trimf input and output membership functions are more suitable for using the data set obtained in Sections 2 and 3.

The semantic values used in the membership functions of the input variables  $\vec{\alpha}$ ,  $\vec{C}$ , and NFV are low, medium, and high. The semantic values used in the output variables  $\Delta\alpha$  and  $\Delta C$  are NE (negative), ZE (zero), and PO (positive). The fuzzy rules used are presented in Table 2, and how to train MLP-NN using FWOA is shown in Figure 12.

## 5. Simulation Results and Discussion

The evaluation of the IEEE CEC-2017 benchmark functions is presented in this section, followed by a discussion of the results achieved for the classification of marine mammals.

### 5.1. Evaluation of IEEE CEC-2017 Benchmark Functions.

The CEC-2017 benchmark functions and dimension size are shown in Table 3. Table 6 shows the parameters selected in the algorithms used for the benchmark functions. In all algorithms, the maximum number of iterations is 100, and the population size is 180.

As shown in Table 4, the FWOA algorithm has achieved more encouraging results compared to CVOA, WOA, ChOA, BWO, and PGO. From a more detailed comparison of WOA with its upgraded version with a fuzzy subsystem (FWOA), it can be seen that the improvement and upgrade of WOA have been successful.

5.2. Classification of Marine Mammals. In this section, to show the power and efficiency of MLP-FWOA, in addition to using the sounds obtained in Sections 2 and 3, the reference dataset (Watkins et al. 1992) is also used. As already mentioned, To obtain the data set, the  $X_m$  vector is assumed to be an input for the MLP-WOA. The  $X_m$  dimension is  $680 \times 42$ , which indicates that there are

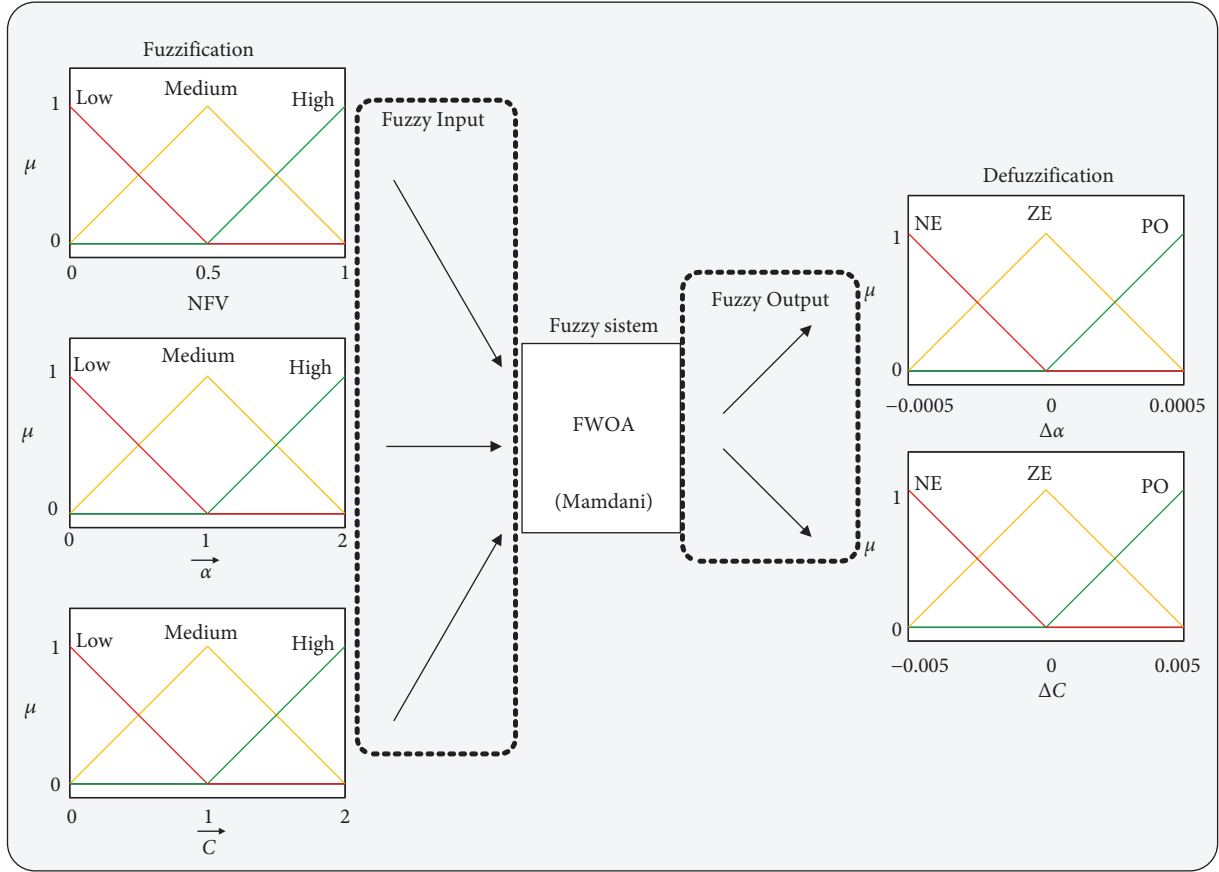

 FIGURE 11: A proposed fuzzy model for setting parameters  $\vec{\alpha}$  and  $\vec{C}$ .

TABLE 2: Applied fuzzy rules.

If (NFV is low) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is ZE)
If (NFV is low) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is NE)
If (NFV is low) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is medium) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is PO)
If (NFV is medium) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is ZE)
If (NFV is medium) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is high) and ( $\vec{\alpha}$ is low), then ( $\Delta\alpha$ is PO)
If (NFV is high) and ( $\vec{\alpha}$ is medium), then ( $\Delta\alpha$ is ZE)
If (NFV is high) and ( $\vec{\alpha}$ is high), then ( $\Delta\alpha$ is NE)
If (NFV is low) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is low) and ( $\vec{C}$ is medium), then ( $\Delta C$ is PO)
If (NFV is low) and ( $\vec{C}$ is high), then ( $\Delta C$ is ZE)
If (NFV is medium) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is medium) and ( $\vec{C}$ is medium), then ( $\Delta C$ is ZE)
If (NFV is medium) and ( $\vec{C}$ is high), then ( $\Delta C$ is NE)
If (NFV is high) and ( $\vec{C}$ is low), then ( $\Delta C$ is PO)
If (NFV is high) and ( $\vec{C}$ is medium), then ( $\Delta C$ is ZE)
If (NFV is high) and ( $\vec{C}$ is high), then ( $\Delta C$ is NE)

42 features and 680 samples in the data set. In addition, the benchmark dataset has a dimension of  $410 \times 42$ . In MLP-FWOA, the number of input nodes is equal to the number of features. The 10-fold cross-validation method is used to evaluate the models. Therefore, first, the data are divided into ten parts, and each time nine parts are used for training and another part for testing. Figure 13

shows the 10-fold cross-validation. The final classification rate for each classifier is calculated using the average of the ten classification rates obtained.

To have a fair comparison between the algorithms, the condition of stopping 300 iterations is considered. There is no specific equation for obtaining the number of hidden layer neurons, so (23) is used to obtain [63].

$$H = 2 \times N + 1, \quad (23)$$

where  $N$  indicates the total number of inputs and  $H$  indicates the total number of hidden nodes. Furthermore, the number of output neurons corresponds to the number of marine mammal classifications, namely six.

For a comprehensive assessment of FWOA performance, this algorithm is compared with WOA [62], ChOA [64], PGO [65], CVOA [66], and BWO [67] benchmark algorithms.

In all population base algorithms, population size is a hyper parameter that plays a direct role in the algorithm's performance in the search space. For this reason, many experiments were performed with different population numbers, some of which are shown in Table 5. The results showed that for the proposed model, a population of 180 is the most appropriate value. In other words, with an increasing population, in addition to no significant improvement in model performance, complexity increases, and processing time increases. Table 6 illustrates the

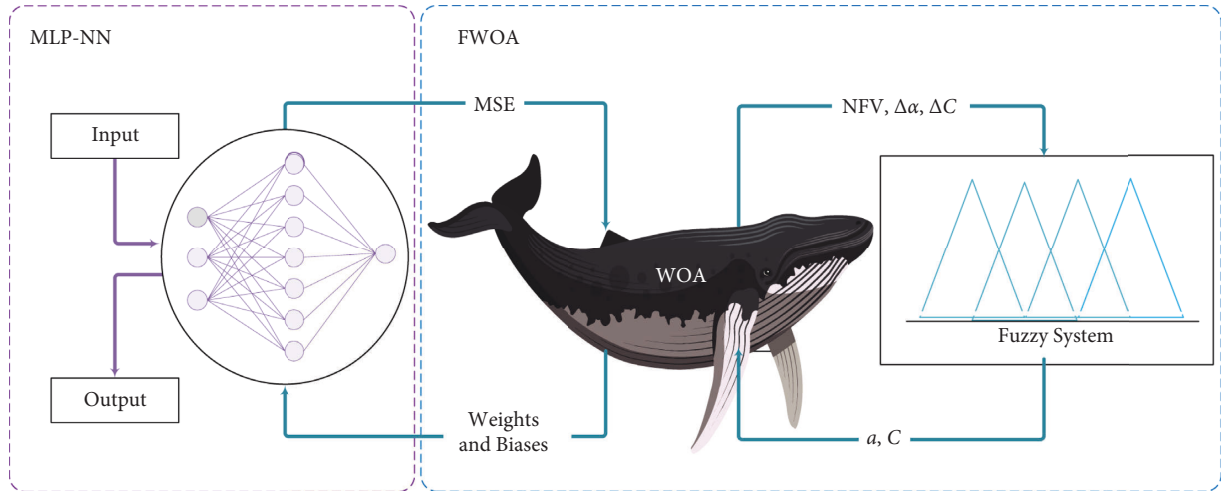


FIGURE 12: How to train MLP-NN using FWOA.

TABLE 3: IEEE CEC-2017 benchmark test functions.

No.	Functions	Dim	$f_{\min}$
$f_1$	Shifted and rotated bent cigar function	30	100
$f_2$	Shifted and rotated sum of different power function	30	200
$f_3$	Shifted and rotated Zakharov function	30	300
$f_4$	Shifted and rotated Rosenbrock's function	30	400
$f_5$	Shifted and rotated Rastrigin's function	30	500
$f_6$	Shifted and rotated expanded Scaffer's function	30	600
$f_7$	Shifted and rotated Lunacek Bi_Rastrigin function	30	700
$f_8$	Shifted and rotated noncontinuous Rastrigin's function	30	800
$f_9$	Shifted and rotated Lévy function	30	900
$f_{10}$	Shifted and rotated Schwefel's function	30	1000
$f_{11}$	Hybrid function 1 ( $N = 3$ )	30	1100
$f_{12}$	Hybrid function 2 ( $N = 3$ )	30	1200
$f_{13}$	Hybrid function 3 ( $N = 3$ )	30	1300
$f_{14}$	Hybrid function 4 ( $N = 4$ )	30	1400
$f_{15}$	Hybrid function 5 ( $N = 4$ )	30	1500
$f_{16}$	Hybrid function 6 ( $N = 4$ )	30	1600
$f_{17}$	Hybrid function 6 ( $N = 5$ )	30	1700
$f_{18}$	Hybrid function 6 ( $N = 5$ )	30	1800
$f_{19}$	Hybrid function 6 ( $N = 5$ )	30	1900
$f_{20}$	Hybrid function 6 ( $N = 6$ )	30	2000
$f_{21}$	Composition function 1 ( $N = 3$ )	30	2100
$f_{22}$	Composition function 2 ( $N = 3$ )	30	2200
$f_{23}$	Composition function 3 ( $N = 4$ )	30	2300

TABLE 4: AVG and STD deviation of the best optimal solution for 40 independent runs on IEEE CEC-2017 benchmark test functions.

FUNC		Algorithm					
		CVOA	WOA	FWOA	ChOA	PGO	BWO
$f_1$	AVG	$1.08E+03$	$3.41E+11$	$3.91E+01$	$4.04E+02$	$1.24E+11$	$1.72E+11$
	STD	$4.48E+04$	$6.06E+08$	$3.41E+02$	$4.34E+02$	$2.38E+10$	$2.83E+08$
$f_2$	AVG	$3.01E+03$	$6.51E+08$	$3.20E+02$	$9.81E+04$	$6.76E+08$	$6.15E+08$
	STD	$7.81E+04$	$9.04E+08$	$9.87E+01$	$7.77E+05$	$1.41E+08$	$2.82E+07$
$f_3$	AVG	$2.40E+03$	$7.07E+03$	$6.96E+03$	$2.46E+03$	$7.08E+03$	$7.49E+05$
	STD	$1.21E+03$	$8.96E+02$	$9.55E+04$	$7.71E+02$	$5.26E+04$	$1.32E+03$
$f_4$	AVG	$4.79E+01$	$1.47E+03$	$5.13E+03$	$5.43E+01$	$1.70E+04$	$1.62E+04$
	STD	$1.38E+02$	$1.96E+02$	$9.75E+01$	$3.38E+02$	$6.15E+03$	$2.48E+03$

TABLE 4: Continued.

FUNC		Algorithm					
		CVOA	WOA	FWOA	ChOA	PGO	BWO
$f_5$	AVG	6.68E+03	8.91E+01	5.20E+01	5.66E+01	8.12E+03	8.10E+03
	STD	2.68E+02	1.30E+02	1.80E+00	1.10E+02	2.83E+02	2.36E+02
$f_6$	AVG	6.38E+03	6.78E+03	6.01E+01	6.01E+01	6.78E+03	6.55E+03
	STD	5.21E+01	4.86E+01	5.32E-02	3.11E-02	6.23E+01	7.26E+01
$f_7$	AVG	1.12E+04	1.36E+04	7.53E+01	8.30E+03	1.26E+02	1.44E+04
	STD	7.55E+02	3.86E+02	9.37E+01	2.63E+00	7.17E+02	6.71E+02
$f_8$	AVG	9.31E+03	1.13E+04	8.80E+01	8.71E+01	1.05E+04	1.12E+04
	STD	2.27E+02	1.24E+02	2.52E+02	2.09E+02	2.53E+02	2.28E+02
$f_9$	AVG	4.03E+04	9.02E+04	1.04E+04	1.08E+04	8.87E+02	9.18E+04
	STD	8.67E+01	1.26E+02	3.21E+03	2.31E+01	1.02E+04	2.14E+04
$f_{10}$	AVG	4.66E+04	8.43E+02	2.13E+04	2.01E+02	7.48E+04	8.35E+02
	STD	5.35E+01	3.33E+01	5.94E+02	1.21E+04	7.92E+03	3.43E+01
$f_{11}$	AVG	1.23E+04	6.92E+04	1.23E+04	1.24E+04	3.16E+04	3.17E+04
	STD	2.87E+02	1.71E+01	1.55E+03	6.34E+02	1.12E+04	5.68E+03
$f_{12}$	AVG	1.31E+07	6.21E+08	6.26E+02	5.68E+03	1.34E+08	1.28E+05
	STD	7.82E+04	2.34E+08	4.55E+02	1.08E+06	9.70E+07	3.41E+07
$f_{13}$	AVG	1.61E+03	3.36E+08	9.85E+04	1.59E+05	2.58E+09	3.81E+07
	STD	1.13E+03	1.52E+08	9.45E+02	1.66E+06	2.04E+07	1.35E+07
$f_{14}$	AVG	4.64E+04	1.65E+05	2.73E+03	2.49E+06	1.25E+08	2.22E+04
	STD	3.56E+04	1.05E+07	2.35E+03	2.22E+05	6.77E+04	1.30E+04
$f_{15}$	AVG	5.32E+04	1.02E+07	4.97E+04	1.49E+05	4.41E+07	6.90E+09
	STD	4.33E+04	9.34E+06	4.60E+04	2.10E+03	4.11E+05	3.08E+08
$f_{16}$	AVG	2.85E+02	6.03E+04	2.44E+04	2.41E+02	3.71E+04	3.66E+04
	STD	3.03E+03	9.41E+03	2.94E+02	2.86E+01	4.47E+03	1.95E+03
$f_{17}$	AVG	2.43E+02	4.11E+04	1.95E+02	1.97E+02	2.55E+04	2.65E+04
	STD	2.76E+01	1.03E+02	9.28E+02	1.39E+03	2.53E+03	1.15E+03
$f_{18}$	AVG	1.35E+07	1.27E+06	1.01E+07	1.01E+07	6.55E+05	4.47E+05
	STD	1.35E+05	1.04E+06	5.08E+05	1.50E+07	3.44E+07	1.99E+05
$f_{19}$	AVG	8.86E+04	9.87E+06	1.81E+03	1.81E+03	6.41E+05	1.13E+07
	STD	6.38E+04	9.43E+06	6.26E+04	1.42E+05	3.01E+07	4.85E+06
$f_{20}$	AVG	2.63E+04	2.78E+04	2.26E+02	2.26E+02	2.75E+02	2.77E+04
	STD	2.01E+03	1.06E+03	2.22E+02	1.40E+01	1.99E+03	1.47E+03
$f_{21}$	AVG	2.43E+04	2.38E+04	2.36E+04	2.38E+04	2.62E+02	2.58E+04
	STD	2.52E+02	6.12E+02	5.14E+03	3.33E+02	3.87E+02	1.68E+02
$f_{22}$	AVG	5.53E+02	3.95E+04	4.31E+02	4.31E+02	8.76E+02	4.91E+05
	STD	2.23E+02	3.34E+01	1.26E+02	2.13E+01	1.32E+04	1.95E+04
$f_{23}$	AVG	3.14E+04	3.47E+04	2.73E+04	2.73E+02	3.22E+04	2.96E+02
	STD	7.76E+02	2.33E+01	1.44E+03	2.61E+02	8.91E+02	2.72E+02

	Dataset									
	10-fold cross-validation									
	Fold1	Fold2	Fold3	Fold4	Fold5	Fold6	Fold7	Fold8	Fold9	Fold10
Iteration 1	Test	Train	Train	Train	Train	Train	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train	Train	Train	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train	Train	Train	Train	Train	Train
Iteration 4	Train	Train	Train	Test	Train	Train	Train	Train	Train	Train
Iteration 5	Train	Train	Train	Train	Test	Train	Train	Train	Train	Train
Iteration 6	Train	Train	Train	Train	Train	Test	Train	Train	Train	Train
Iteration 7	Train	Train	Train	Train	Train	Train	Test	Train	Train	Train
Iteration 8	Train	Train	Train	Train	Train	Train	Train	Test	Train	Train
Iteration 9	Train	Train	Train	Train	Train	Train	Train	Train	Test	Train
Iteration 10	Train	Train	Train	Train	Train	Train	Train	Train	Train	Test

FIGURE 13: The 10-fold cross-validation.

TABLE 5: Classification rate (CR%) to adjust the population size of different algorithms for the data set is obtained from parts 2 and 3.

Algorithm	WOA		FWOA		ChOA		PGO		CVOA		BWO		
	Population size	CR%	Time (s)	CR%	Time (s)	CR%	Time (s)	CR%	Time (s)	CR%	Time (s)	CR%	Time (s)
60		31.28	0.314	35.67	0.246	31.00	0.254	22.14	0.377	33.16	0.733	27.42	0.411
80		37.98	0.377	42.11	0.299	35.11	3.008	29.06	0.408	38.93	986.4	34.73	1.819
100		46.85	0.461	59.61	3.113	45.31	0.355	35.02	0.483	48.70	1.075	43.04	2.167
120		59.98	0.522	64.78	3.785	56.21	0.477	48.60	0.567	61.58	1.306	53.89	2.483
140		71.25	0.608	75.00	0.407	67.18	0.519	63.75	0.658	73.77	1.668	66.82	3.150
160		82.65	0.820	88.25	0.485	81.99	0.686	80.09	0.633	84.35	1.761	78.50	3.445
180		91.42	0.791	94.97	0.531	90.66	0.794	88.62	0.715	92.73	1.809	89.15	4.173
190		92.02	1.087	94.99	1.010	91.10	0.819	89.22	1.764	92.98	2.060	89.83	4.890
200		92.64	1.964	95.00	1.973	92.00	0.921	91.45	3.234	93.07	2.364	90.13	5.789

TABLE 6: The initial parameters and primary values of the benchmark algorithms.

Algorithm	Parameter	Value
WOA	$a$	Linearly decreased from 2 to 0
	Population size	180
FWOA	$a$	Tuning by fuzzy system
	Population size	180
ChOA	$a$	[1, 1]
	$f$	Linearly decreased from 2 to 0
	Population size	180
PGO	DR	0.3
	DRS	0.15
	EDR	0.8
	Population size	180
CVOA	P_die	0.05
	P_isolation	0.8
	P_superspreader	0.1
	P_reinfection	0.02
	Social_distancing	8
	P_travel	0.1
	Pandemic_duration	30
BWO	Population size	180
	PP	0.6
	CR	0.44
	PM	0.4
	Population size	180

TABLE 7: Results obtained from different algorithms for dataset reference [42].

Algorithm	MSE (AVE $\pm$ STD)	$p$ values	A20 index	AUC (%)	Classification rate %
MLP-CVOA	0.1089 $\pm$ 0.1285	0.045	1	94.68	92.1283
MLP-WOA	0.1156 $\pm$ 0.1301	1.8534e-31	1	93.09	90.6422
MLP-FWOA	0.1002 $\pm$ 0.1154	0.027	1	97.71	93.0150
MLP-ChOA	0.1383 $\pm$ 0.1389	2.1464e-61	1	92.71	89.9476
MLP-PGO	0.1694 $\pm$ 0.1652	3.0081e-77	1	91.48	88.1115
MLP-BWO	0.1495 $\pm$ 0.1543	2.9813e-43	1	88.97	89.1014

fundamental parameters and major values of various benchmark methods. The classification rate to adjust the population size of different algorithms for the data set is obtained from parts 2 and 3.

The classifiers' performance is then tested for the classification rate, local minimum avoidance, and convergence speed. Each method is run 40 times, and the classification rate, mean, and standard deviation of the smallest error, the

A20-index [68], and the  $p$  value are listed in Tables 7 and 8. The mean and standard deviation of the smallest error, the A20-index, and the  $p$  value all reflect how well the method avoids local optimization.

Figures 14 and 16 show a comprehensive comparison of the convergence speed and syntax and the classifiers' final error rate. Figures 15 and 17 show a receiver operating characteristic for datasets.



TABLE 8: Results obtained from different algorithms for datasets obtained in parts 2 and 3.

Algorithm	MSE (AVE $\pm$ STD)	$p$ values	A20 index	AUC (%)	Classification rate %
MLP-CVOA	0.0857 $\pm$ 0.1021	0.009	1	94.32	92.8027
MLP-WOA	0.0908 $\pm$ 0.1622	0.042	1	93.05	91.3455
MLP-FWOA	0.0689 $\pm$ 0.1031	0.013	1	95.89	94.9801
MLP-ChOA	0.1090 $\pm$ 0.1076	1.0589e-34	1	92.76	90.2410
MLP-PGO	0.1596 $\pm$ 0.1167	3.0013e-11	1	90.66	88.1004
MLP-BWO	0.1194 $\pm$ 0.1622	2.9854e-64	1	84.25	89.0478

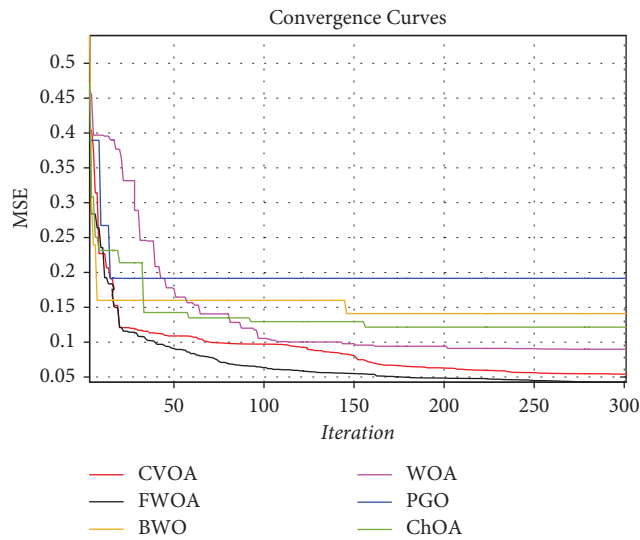


FIGURE 14: Convergence diagram of different training algorithms for dataset reference [42].

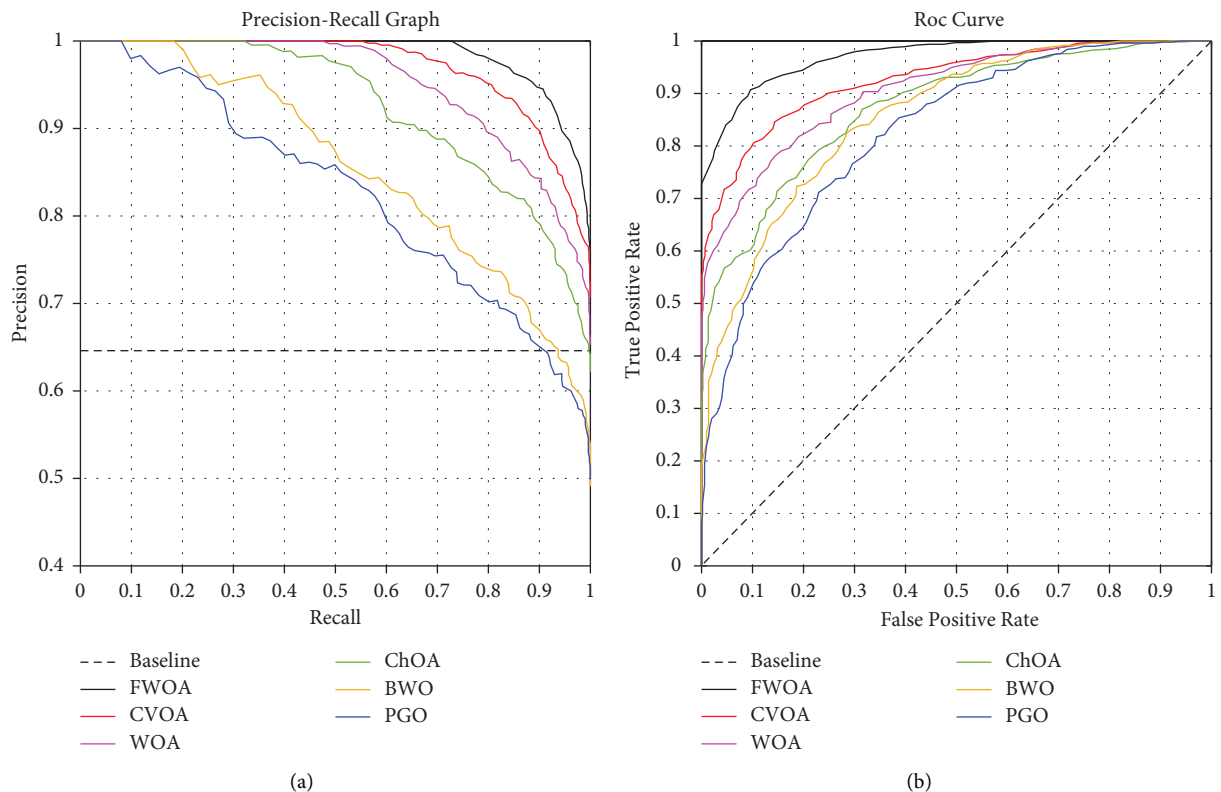


FIGURE 15: Receiver operating characteristic for dataset reference [42].

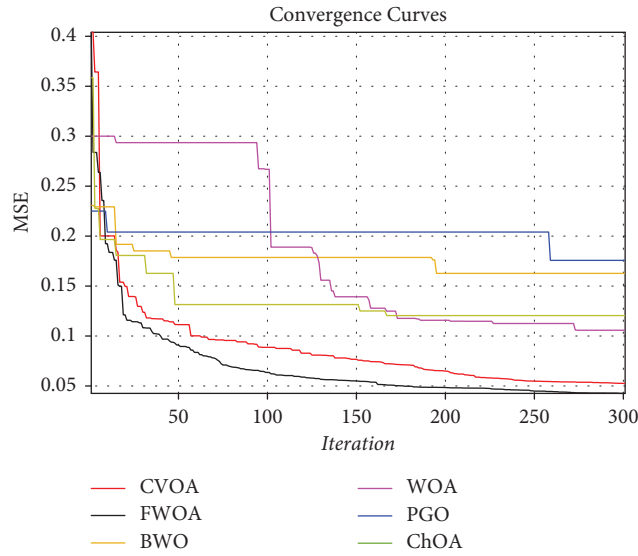


FIGURE 16: Convergence diagram of different training algorithms for datasets obtained in parts 2 and 3.

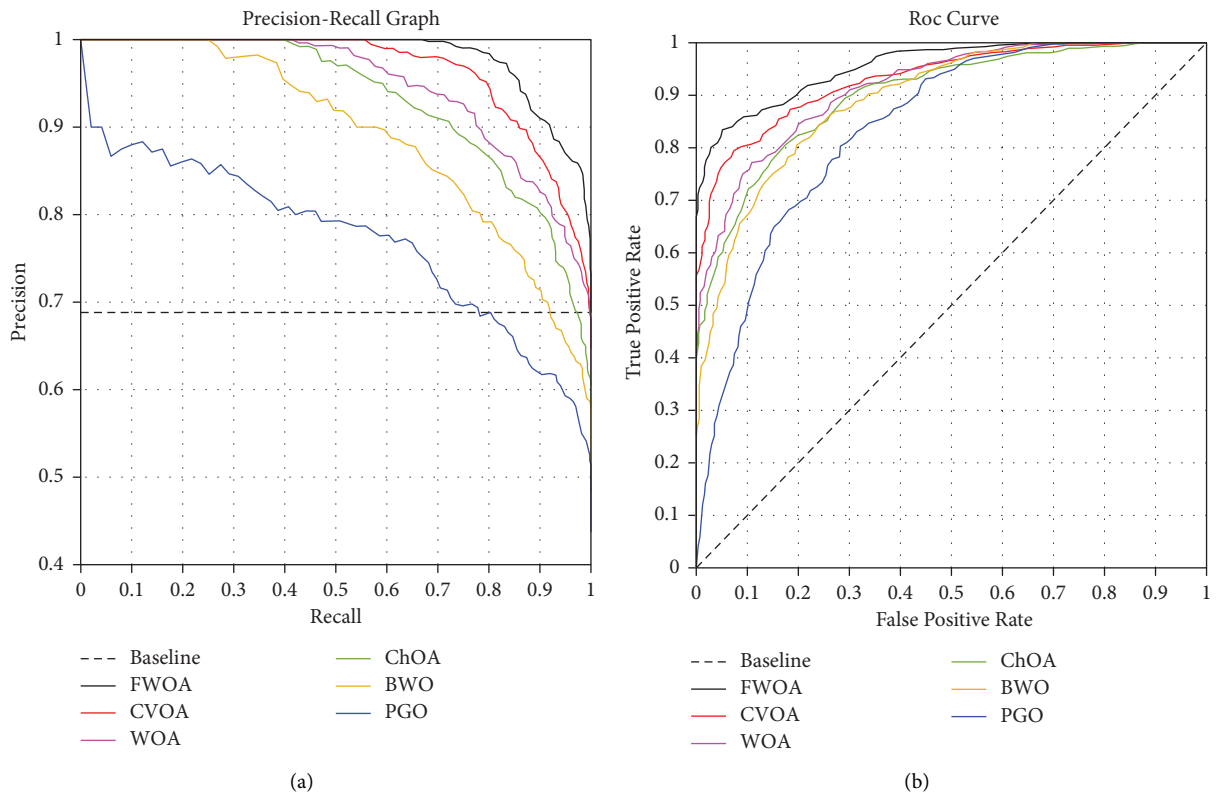


FIGURE 17: Receiver operating characteristic for datasets obtained in parts 2 and 3.

The simulation was conducted in MATLAB 2020a by using a personal computer with a 2.3 GHz processor and 5 GB RAM.

As shown in Figures 14 and 16., among the benchmark algorithms used for MLP training, FWOA has the highest convergence speed. PGO has the lowest convergence speed by adjusting control parameters by fuzzy inference, correctly

detecting the boundary between the exploration and extraction phases. As shown in Tables 7 and 8, MLP-FWOA has the highest classification rate, and MLP-PGO has the lowest classification rate among the classifiers. The STD values, shown in Tables 7 and 8, indicate that the MLP-FWOA results rank first in the two datasets, confirming that the FWOA performs better than other standard training

algorithms and demonstrates the FWOA's ability to avoid getting caught up in local optimism. A  $p$  value of less than 0.05 indicates a significant difference between FWOA and other algorithms. According to Tables 7 and 8, a20-index is 1 for all predictive classifiers. It confirms that all models can provide good results for similar data as well.

Adding a subsystem to a metaheuristic algorithm increases its complexity. However, a comparison of the convergence curves in Figures 14 and 16 shows that the FWOA achieved the global optimum faster than the other algorithms used. Other algorithms were stuck in the local optimum if they converged. In particular, by comparing the WOA and FWOA in Figures 14 and 16, it can be seen that using an auxiliary (fuzzy system) subsystem is necessary to avoid getting caught up in the local optimum in the WOA. In general, using a fuzzy system to improve WOA increases complexity. However, the convergence curves and better performance of FWOA than other algorithms used show a reduction in computational cost. The reduced MSE of this method compared to other algorithms employed is more indicative that despite increased complexity, FWOA performance is improving.

## 6. Conclusions and Recommendations

In this paper, to classify marine mammals, a fuzzy model of control parameters of the whale optimization algorithm was designed to train an MLP-NN. CVOA, WOA, FWOA, ChOA, PGO, and BWO algorithms have been used for the MLP-NN training stage. As the simulation results show, FWOA has a powerful performance in identifying the boundary between the exploration and extraction phases. For this reason, it can identify the global optimal and avoid local optimization. The results indicate that MLP-FWOA, MLP-CVOA, MLP-WOA, MLP-ChOA, MLPBWO, and MLP-PGO have better performance for classifying the sound produced by marine mammals. The convergence curve also shows that FWOA converges faster than the other five benchmark algorithms in convergence speed.

Due to the complex environment of the sea and various unwanted signals such as reverberation, clutter, and types of noise in the seabed, lack of access to data sets with a specific signal-to-noise ratio is one of the main limitations of the research.

For future research directions, we recommend the following list of topics:

- (i) MLP-NN training using other metaheuristic algorithms for the classification of marine mammals
- (ii) Using other artificial neural networks and using deep learning for the classification of marine mammals
- (iii) Direct use of metaheuristic algorithms as classifiers for classification of marine mammals.

## Data Availability

No data were used to support this study.

## Disclosure

An earlier version of the manuscript has been presented as Preprint in Research Square according to the following link [https://assets.researchsquare.com/files/rs-122787/v1\\_covered.pdf?c=1631848722](https://assets.researchsquare.com/files/rs-122787/v1_covered.pdf?c=1631848722).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] R. Danovaro, C. Corinaldesi, A. Dell'Anno, P. V. Snelgrove, and R. Snelgrove, "The deep-sea under global change," *Current Biology*, vol. 27, no. 11, pp. R461–R465, 2017.
- [2] H. Chen, Y. Xiong, S. Li, Z. Song, Z. Hu, and F. Liu, "Multi-sensor data driven with PARAFAC-IPSO-PNN for identification of mechanical nonstationary multi-fault mode," *Machines*, vol. 10, no. 2, 155 pages, 2022.
- [3] L. Thorne, "Deep-sea research II fronts , fi sh , and predators," *Deep-Sea Research Part II*, vol. 107, pp. 1-2, 2014.
- [4] W. Dreybrodt, "See also the following articles SPELEOTHEM DEPOSITION," *Encyclopedia of Caves*, vol. 1, no. 1, pp. 769–777, 2012.
- [5] I. G. Priede, *Metabolic Scope in Fishes*, pp. 33–64, Springer, Berlin, 1985.
- [6] Y. U. Shiu, K. J. Palmer, M. A. Roch et al., "Deep neural networks for automated detection of marine mammal species," *Scientific Reports*, vol. 10, no. 1, pp. 607–612, 2020.
- [7] M. Thomas, B. Martin, K. Kowarski, and B. Gaudet, *Marine Mammal Species Classification Using Convolutional Neural Networks and a Novel Acoustic Representation*, Springer, pp. 1–16, Berlin.
- [8] K. Jin, Y. Yan, M. Chen et al., "Multimodal deep learning with feature level fusion for identification of choroidal neovascularization activity in age-related macular degeneration," *Acta Ophthalmologica*, vol. 100, no. 2, 20 pages, 2022.
- [9] J. Wang, J. Tian, X. Zhang et al., "Control of time delay force feedback teleoperation system with finite time convergence," *Frontiers in Neurorobotics*, vol. 16, Article ID 877069, 9 pages, 2022.
- [10] M. S. H. AI-Furjan, M. Habibi, A. rahimi et al., "Chaotic simulation of the multi-phase reinforced thermo-elastic disk using GDQM," *Engineering with Computers*, vol. 38, no. 1, pp. 219–242, 2022.
- [11] A. Saffari, S. H. Zahiri, and M. Khishe, "Fuzzy whale optimisation algorithm: a new hybrid approach for automatic sonar target recognition," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 18, no. 1, pp. 1–17, 2022.
- [12] J. Hurtado, A. Montenegro, M. Gattass, F. Carvalho, and A. Raposo, "Enveloping CAD models for visualization and interaction in XR applications," *Engineering with Computers*, vol. 38, pp. 781–799, 2022.
- [13] D. Li, S. S. Ge, and T. H. Lee, "Fixed-time-synchronized consensus control of multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 89–98, 2021.
- [14] S. Lu, Y. Ban, X. Zhang et al., "Adaptive control of time delay teleoperation system with uncertain dynamics," *Frontiers in Neurorobotics*, vol. 16, Article ID 928863, 2022.
- [15] Y. Zhang, F. Liu, Z. Fang, B. Yuan, G. Zhang, and J. Lu, "Learning from a complementary-label source domain: theory and algorithms," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.

- [16] C. Pan, X. Deng, and Z. Huang, "Parallel computing - oriented method for long - time duration problem of force identification," *Engineering with Computers*, vol. 38, no. 1, pp. 919–937, 2022.
- [17] P. Sarir, "Developing GEP tree - based, neuro - swarm, and whale optimization models for evaluation of bearing capacity of concrete - filled steel tube columns," *Engineering with Computers*, vol. 37, no. 1, pp. 1–19, Article ID 0123456789, 2022.
- [18] H. Wang, H. Moayed, and L. Kok Foong, "Genetic algorithm hybridized with multilayer perceptron to have an economical slope stability design," *Engineering with Computers*, vol. 37, no. 1, 2020.
- [19] L. Zhou, Q. Fan, X. Huang, and Y. Liu, "Weak and strong convergence analysis of Elman neural networks via weight decay regularization," *Optimization*, pp. 1–23, 2022.
- [20] J. Yan, Y. Yao, S. Yan, R. Gao, W. Lu, and W. He, "Chiral protein supraparticles for tumor suppression and synergistic immunotherapy: an enabling strategy for bioactive supra-molecular chirality construction," *Nano Letters*, vol. 20, no. 8, pp. 5844–5852, 2020.
- [21] Q. Fan, Z. Zhang, and X. Huang, "Parameter conjugate gradient with secant equation based elman neural network and its convergence analysis," *Advanced Theory and Simulations*, vol. 5, no. 9, Article ID 2200047, 2022.
- [22] B. Keshtegar, "Limited conjugate gradient method for structural reliability analysis," *Engineering with Computers*, vol. 33, no. 3, pp. 621–629, 2016.
- [23] Z. Moradi, M. Mehrvar, and E. Nazifi, "Genetic diversity and biological characterization of sugarcane streak mosaic virus isolates from Iran," *VirusDisease*, vol. 29, no. 3, pp. 316–323, 2018.
- [24] M. Khishe and M. R. Mosavi, "Improved whale trainer for sonar datasets classification using neural network," *Applied Acoustics*, vol. 154, pp. 176–192, 2019.
- [25] S. C. Ng, C. C. Cheung, S. H. Leung, and A. Luk, "Fast convergence for backpropagation network with magnified gradient function," in *Proceedings of the International Joint Conference on Neural Networks*, Portland, OR, USA, July 2003.
- [26] W. Yang, W. Liu, X. Li, J. Yan, and W. He, "Turning chiral peptides into a racemic supraparticle to induce the self-degradation of MDM2," *Journal of Advanced Research*, In press, 2022.
- [27] Z. Li, M. Teng, R. Yang et al., "Sb-doped WO<sub>3</sub> based QCM humidity sensor with self-recovery ability for real-time monitoring of respiration and wound," *Sensors and Actuators B: Chemical*, vol. 361, Article ID 131691, 10 pages, 2022.
- [28] W. Zheng, Y. Zhou, S. Liu, J. Tian, B. Yang, and L. Yin, "A deep fusion matching network semantic reasoning model," *Applied Sciences*, vol. 12, no. 7, 3416 pages, 2022.
- [29] M. A. Simaan, "Simple explanation of the No-Free-Lunch theorem and its implications 1," *Journal of Optimization Theory and Applications*, vol. 115, no. 3, pp. 549–570, 2003.
- [30] D. J. Montana and L. Davis, "Training feedforward neural networks using genetic algorithms," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pp. 762–767, August 1989, <http://dl.acm.org/citation.cfm?id=1623755.1623876>.
- [31] C. S. Koh, O. Mohammed, and Song-Yop Hahn, "Detection of magnetic body using artificial neural network with modified simulated annealing," *IEEE Transactions on Magnetics*, vol. 30, no. 5, pp. 3644–3647, 1994.
- [32] M. Li, X. Huang, H. Liu et al., "Prediction of gas solubility in polymers by back propagation artificial neural network based on self-adaptive particle swarm optimization algorithm and chaos theory," *Fluid Phase Equilibria*, vol. 356, pp. 11–17, 2013.
- [33] L. A. M. Pereira, D. Rodrigues, P. B. Ribeiro, J. P. Papa, and S. A. T. Weber, "Social-spider optimization-based artificial neural networks training and its applications for Parkinson's disease identification," in *Proceedings of the-IEEE Symposium on Computer-Based Medical Systems*, pp. 14–17, New York, NY, USA, May 2014.
- [34] M. Dehghanpour, A. Rahati, and E. Dehghanian, "ANN-based modeling of third order Runge Kutta method," *Journal of Advanced Computer Science & Technology*, vol. 4, no. 1, p. 180, 2015.
- [35] W. Yamany, M. Fawzy, A. Tharwat, and Aboul Ella Hassanien, "Moth-flame optimization for training multi-layer perceptrons," in *Proceedings of the 2015 11th International Computer Engineering Conference: Today Information Society What's Next?, ICENCO 2015*, Dubai, November 2015.
- [36] M. R. Mosavi, M. Khishe, and A. Ghamgosar, "Classification of sonar data set using neural network trained by gray wolf optimization," *Neural Network World*, vol. 26, no. 4, pp. 393–415, 2016.
- [37] M. R. Mosavi and M. Khishe, "Training a feed-forward neural network using particle swarm optimizer with autonomous groups for sonar target classification," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 11, Article ID 1750185, 2017.
- [38] Y. Ling, Y. Zhou, and Q. Luo, "Lévy flight trajectory-based whale optimization algorithm for global optimization," *IEEE Access*, vol. 5, pp. 6168–6186, 2017.
- [39] X. Pu, S. Chen, X. Yu, and L. Zhang, "Developing a novel hybrid biogeography-based optimization algorithm for multilayer perceptron training under big data challenge," *Scientific Programming*, vol. 2018, Article ID 2943290, 2018.
- [40] H. Faris, I. Aljarah, and S. Mirjalili, "Improved monarch butterfly optimization for unconstrained global search and neural network training," *Applied Intelligence*, vol. 48, no. 2, pp. 445–464, 2018.
- [41] Y. Zhou, Y. Ling, and Q. Luo, "Lévy flight trajectory-based whale optimization algorithm for engineering optimization," *Engineering Computations*, vol. 35, no. 7, pp. 2406–2428, 2018.
- [42] M. Khishe and A. Safari, "Classification of sonar targets using an MLP neural network trained by dragonfly algorithm," *Wireless Personal Communications*, vol. 108, no. 4, pp. 2241–2260, 2019.
- [43] S. Afrakhteh, M. R. Mosavi, M. Khishe, and A. Ayatollahi, "Accurate classification of EEG signals using neural networks trained by hybrid population-physic-based algorithm," *International Journal of Automation and Computing*, vol. 17, no. 1, pp. 108–122, 2020.
- [44] S. L. Zubaidi, I. H. Abdulkareem, K. S. Hashim et al., "Hybridised artificial neural network model with slime mould algorithm: a novel methodology for prediction of urban stochastic water demand," *Water (Switzerland)*, vol. 12, no. 10, 2692 pages, 2020.
- [45] Y. Niu, Y. Zhou, and Q. Luo, "Optimize star sensor calibration based on integrated modeling with hybrid WOA-LM algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 38, no. 3, pp. 2683–2691, 2020.
- [46] H. Mohammad and T. Tayarani, "Applications of Artificial Intelligence in Battling Against Covid-19: A Literature Review," *Chaos, Solitons and Fractals*, vol. 142, pp. 1–31, 2021.

- [47] E. Eker, M. Kayri, S. Ekin, and D. Izci, *Training Multi-Layer Perceptron Using Harris Hawks Optimization*, International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 2020.
- [48] C. Wu, M. Khishe, M. Mohammadi, S. H. Taher Karim, and T. A. Rashid, "Evolving deep convolutional neural network by hybrid sine-cosine and extreme learning machine for real-time COVID19 diagnosis from X-ray images," *Soft Computing*, vol. 6, pp. 1–20, 2021.
- [49] H. Nguyen and X. N. Bui, "A novel hunger games search optimization-based artificial neural network for predicting ground vibration intensity induced by mine blasting," *Natural Resources Research*, vol. 30, no. 5, pp. 3865–3880, 2021.
- [50] A. Saffari, Seyed-hamid Zahiri, and M. Khishe, "Automatic recognition of sonar targets using feature selection in micro-Doppler signature," *Defence Technology*, In press, 2022.
- [51] X. Wu, S. E. N. Zhang, W. Xiao, and Y. Yin, "The exploration/exploitation tradeoff in whale optimization algorithm," *IEEE Access*, vol. 7, Article ID 125919, 28 pages, 2019.
- [52] Z. Zhuo, Y. Wan, D. Guan et al., "A loop-based and AGO-incorporated virtual screening model targeting AGO-mediated MiRNA-MRNA interactions for drug discovery to rescue bone phenotype in genetically modified mice," *Advanced Science*, vol. 7, no. 13, Article ID 1903451, 22 pages, 2020.
- [53] J. B. Allen, D. A. Berkley, and J. Blauert, "Multimicrophone signal-processing technique to remove room reverberation from speech signals," *The Journal of the Acoustical Society of America*, vol. 62, no. 4, 1977.
- [54] D. Gallardo, O. Sahni, and R. Bevilacqua, "Hammerstein - wiener based reduced - order model for vortex - induced non - linear fluid - structure interaction," *Engineering with Computers*, vol. 33, no. 2, pp. 219–237, 2016.
- [55] Report Final, *Specialist Committee on Hydrodynamic Noise*, no. 2, 2014.
- [56] A. Das, A. Kumar, and R. Bahl, "Marine Vessel Classification Based on Passive Sonar Data: The Cepstrum-Based Approach," *IET Radar, Sonar & Navigation*, vol. 7, no. 1, pp. 87–93, 2013.
- [57] A. Das, A. Kumar, and R. Bahl, "Radiated signal characteristics of marine vessels in the cepstral domain for shallow underwater channel," *The Journal of the Acoustical Society of America*, vol. 128, no. 4, 151 pages, 2012.
- [58] M. M. Mijwil and A. Alsaadi, *Overview of Neural Networks*, 2019.
- [59] C. Liu, Y. Wang, L. Li et al., "Engineered extracellular vesicles and their mimetics for cancer immunotherapy," *Journal of Controlled Release*, vol. 349, pp. 679–698, 2022.
- [60] M. Aitkin and R. O. B. Foxall, "Statistical modelling of artificial neural networks using the multi-layer perceptron," *Statistics and Computing*, vol. 13, no. 3, pp. 227–239, 2003.
- [61] D. Devikanniga, K. Vetrivel, and N. Badrinath, "Review of meta-heuristic optimization based artificial neural networks and its applications," *Journal of Physics: Conference Series*, vol. 1362, no. 1, Article ID 012074, 2019.
- [62] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [63] M. Kaveh, M. R. Mosavi, M. Kaveh, and M. Khishe, "Design and implementation of a neighborhood search biogeography-based optimization trainer for classifying sonar dataset using multi-layer perceptron neural network," *Analog Integrated Circuits and Signal Processing*, vol. 100, no. 2, pp. 405–428, 2018.
- [64] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Systems with Applications*, vol. 149, Article ID 113338, 2020.
- [65] A. Kaveh, H. Akbari, and Seyed Milad Hosseini, "Plasma Generation Optimization: A New Physically-Based Metaheuristic Algorithm for Solving Constrained Optimization Problems," *Engineering Computations*, vol. 38, no. 4, 2020.
- [66] J. Torres, "Coronavirus optimization algorithm: a bioinspired metaheuristic based on the COVID-19 propagation model," *Big Data*, vol. 8, pp. 1–15, 2020.
- [67] V. Hayyolalam and A. A. Pourhaji Kazem, "Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 87, Article ID 103249, 2020.
- [68] D. J. Armaghani, P. G. Asteris, S. A. Fatemi et al., "On the Use of Neuro-Swarm System to Forecast the Pile Settlement," *Applied Sciences*, vol. 10, no. 6, 2020.