

Article

A Moth–Flame Optimized Echo State Network and Triplet Feature Extractor for Epilepsy Electro-Encephalography Signals

Xue-song Tang , Luchao Jiang, Kuangrong Hao *, Tong Wang and Xiaoyan Liu

Faculty of Information Science, Donghua University, Shanghai 201620, China

* Correspondence: krhao@dhu.edu.cn

Abstract: The analysis of epilepsy electro-encephalography (EEG) signals is of great significance for the diagnosis of epilepsy, which is one of the common neurological diseases of all age groups. With the developments of machine learning, many data-driven models have achieved great performance in EEG signals classification. However, it is difficult to select appropriate hyperparameters for the models to file a specific task. In this paper, an evolutionary algorithm enhanced model is proposed, which optimizes the fixed weights of the reservoir layer of the echo state network (ESN) according to the specific task. As evaluating a feature extractor relies heavily on the classifiers, a new feature distribution evaluation function (FDEF) using the label information of EEG signals is defined as the fitness function, which is an objective way to evaluate the performance of a feature extractor that not only focuses on the degree of dispersion, but also considers the relation amongst triplets. The performance of the proposed method is verified on the Bonn University dataset with an accuracy of 98.16% and on the CHB-MIT dataset with the highest sensitivity of 96.14%. The proposed method outperforms the previous EEG methods, as it can automatically optimize the hyperparameters of ESN to adjust the structure and initial parameters for a specific classification task. Furthermore, the optimization direction by using FDEF as the fitness of MFO no longer relies on the performance of the classifier but on the relative separability amongst classes.

Keywords: epilepsy detection; moth–flame optimization; echo state network; feature extraction; EEG signals

MSC: 68W50



Citation: Tang, X.-s.; Jiang, L.; Hao, K.; Wang, T.; Liu, X. A Moth–Flame Optimized Echo State Network and Triplet Feature Extractor for Epilepsy Electro-Encephalography Signals. *Mathematics* **2023**, *11*, 1438. <https://doi.org/10.3390/math11061438>

Academic Editor: Ivan Lorencin

Received: 12 February 2023

Revised: 5 March 2023

Accepted: 10 March 2023

Published: 16 March 2023



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

1. Introduction

As a chronic non-communicable brain disease, epilepsy can occur in people of any age. It is one of the most common neurological diseases with about 50 million patients, and about 5 million people are diagnosed each year around the world [1]. Electro-encephalography (EEG) is a commonly used auxiliary diagnostic method in the discovery and treatment of brain diseases [2], but there are certain limitations in traditional EEG methods. On the one hand, traditional EEG diagnosis based on visual assessments requires experienced specialists for correct judgments, which is subject to their professional experience. Further, as the frequency of using EEG equipment in outpatients and inpatients increases, it usually takes several hours or days to record the EEG data, analyze and diagnose. On the other hand, patients with epilepsy can register as completely normal when undergoing an outpatient EEG examination, because the brain of an epilepsy patient usually does not consistently trigger seizures. Recording EEG for longer periods can capture abnormal EEG signals, but it is expensive and time-consuming for both patients and doctors.

EEG signals are nonlinear and nonstationary [3]. Figure 1 displays a group of normal, inter-ictal, and seizure EEG signals collected by Bonn University that are difficult to be interpreted visually. Moreover, the inter-ictal signals, which are important for the diagnosis and treatment of patients, cannot be easily distinguished from normal signals.

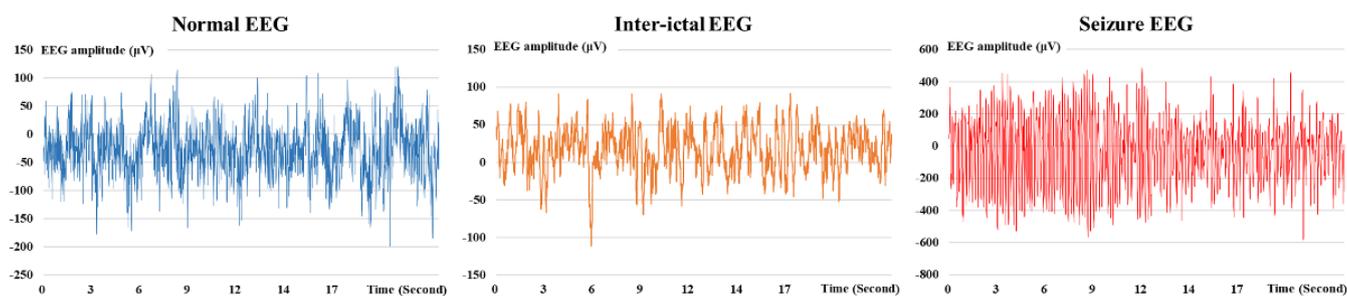


Figure 1. Samples of normal, interictal, and seizure EEG signals from the Bonn University data set.

With the development of computer technologies, a variety of algorithms have been designed with excellent results in the automatic classification of EEG signals [4]. EEG signal classification consists of two parts: feature extraction, which is the most important part, and classification. Methods based on time–frequency domain, such as the AR model [5], fast Fourier transform method [6], and Hilbert Huang method [7], have been applied to extract the features of EEG signals with trivial information loss. However, these methods are generally based on linear models, so the non-linear characteristics of EEG signals that are essential in EEG signals processing can be ignored.

In recent years, machine learning and deep learning models have been applied to capture the features of EEG signals. Zhou et al. [8] used CNN to extract features for EEG signals and applied the model to a seizure detection task. Although CNN can effectively focus on local features, it cannot directly capture the long-term relationships in the EEG signals. To solve this problem, Mishra et al. [9] combined CNN and RNN models to solve the sleep stage classification problem based on the EEG signals. RNN contains connections between nodes to form a directed graph sequence. This structure allows RNN to naturally handle the time dynamic behavior of time series. However, it also causes the problem of gradient vanishing [10], which makes the training of RNN very difficult and time-consuming. Sun et al. [11] proposed an ESN feature extractor model, which is an unsupervised self-encoding model based on an ESN. The model can extract EEG signal features and achieves good performance in the classification of the epileptic EEG signals. ESN is a special network model that provides a new structure of the recurrent neural network and a new criterion of supervised training [12]. Compared to RNN, most parameters of ESN are randomly generated, except for the readout layer that needs training, so the model is very fast and does not suffer from the problem of gradient vanishing. ESN can achieve excellent performance on a variety of chaotic time series prediction tasks and complex industrial time series problems. However, the performance of ESN largely depends on the choice of hyperparameters that requires massive experimental cost for achieving a good configuration [13].

Evolutionary algorithms, which are inspired by biological evolution, are effective in optimizing the hyperparameters of the model. Wang et al. [14] chose the genetic algorithm (GA) to optimize the hyperparameters of ESN and applied ESN to ECG signal prediction tasks, which achieves better results compared to the original ESN model. Moth–flame optimization (MFO) [15] is an extended version of the swarm intelligence algorithm, which simulates the special navigation method of moths flying around flames and provides a new heuristic search paradigm in the optimization field, called spiral search. The new search paradigm enables the MFO algorithm to search near the candidate optimal solutions. Mei et al. [16] applied MFO in ORPD problems to obtain the best combination of control variables. The MFO algorithm has good performances in various fields and the advantages of simplicity and rapid searching over other optimization algorithms [17]. In this paper, we propose a new model named MFO-ESN that applies the MFO algorithm to automatically search for better hyperparameters of an ESN feature extractor.

An essential problem of combining MFO and ESN is to define the fitness function, which can evaluate the performance of the feature extractors and determine the optimization direction. For classification tasks, the most intuitive way is to use the accuracy of a classifier. However, the method relies significantly on the choice of classifiers, which is not objective and may lead to overfitting problems. The basic idea behind a good feature extraction for classification tasks is to find a way to map the raw data onto a feature space, which draws the feature vectors from the same class closer than those from the different ones. We propose a new feature distribution evaluation function to fit the MFO-ESN, named FDEF. FDEF is based on the idea of triplet loss [18] and can evaluate the performance of a feature extractor without using a specific classifier.

Triplet loss is a metric that concerns the relation of the triplet. For classification tasks, a triplet consists of the observed sample: the corresponding positive sample whose class is the same as the observed sample, and the corresponding negative sample whose class is different from the observed sample. Triplet loss is a detailed and objective way to judge how easy the sample can be classified, which not only focuses on the degree of dispersion, but also considers the relation amongst triplets.

The main contributions of this work can be summarized as follows:

- (1) A novel feature extraction called moth–flame optimized echo state network (MFO-ESN) is developed, which uses MFO to optimize the hyperparameters of ESN for fitting the specific tasks.
- (2) A new function based on the triplet is introduced to evaluate the distribution of features extracted by MFO-ESN without relying on specific classifiers.
- (3) MFO-ESN is verified on the real-world single-channel EEG signals classification task with an accuracy of 98.16%. The results also show that MFO-ESN with FDEF can promote the performances of many classifiers.
- (4) We also conduct experiments on the multi-channel EEG signals classification task with the highest specificity of both the patient specific and the cross-patient task. The cross-patient task simulates the real diagnosis situation with high specificity, proving the strong generalization ability of MFO-ESN.

The remainder of this paper is organized as follows. In Section 2, a review of related works is given, including the ESN algorithm and the MFO algorithm. In Section 3, a new feature evaluation function (FDEF) fitting MFO-ESN is presented. A detailed description of the feature extractor combining ESN and MFO named MFO-ESN is also proposed. The experiment process and results of the single-channel and multi-channel epilepsy EEG signals classification task are described in Sections 4 and 5, respectively. In Section 6, we conclude this paper and propose a future study direction.

2. Related Works

2.1. Echo State Network (ESN)

The echo state network (ESN) is a particular RNN and a typical representative of reservoir computing (RC) [12]. The canonical ESN model is a neural network with a three-layer structure, including an input layer, a reservoir layer, and an output layer. ESN simulates the connection of neurons in the cerebral cortex and constructs a huge and sparse neuron reservoir layer. The neurons in the reservoir layer are randomly initialized and do not need training. The reservoir layer that imitates the sparsely connected structure of brain neurons replaces the hidden layer of the traditional feedforward neural networks for the function of information processing and storage [19]. Therefore, the reservoir layer is the most important part of a canonical echo state network, whose architecture is shown in Figure 2.

Focusing on the architecture details of the canonical ESN, we denote the number of neurons for the input layer, reservoir layer, and output layer as K , P , and L , respectively. The input unit, reservoir state, and output unit at time t can be represented as follows:

$$u(t) = (u_1(t), \dots, u_K(t))^T \quad (1)$$

$$x(t) = (x_1(t), \dots, x_M(t))^T \tag{2}$$

$$y(t) = (y_1(t), \dots, y_L(t))^T \tag{3}$$

The weights from the input layer to the reservoir layer amongst the internal neurons in the reservoir layer, and from the reservoir layer to the output layer, are denoted as W_{in} , W_{res} and W_{out} , respectively.

$$u(t) = (u_1(t), \dots, u_K(t))^T \quad y(t) = (y_1(t), \dots, y_L(t))^T$$

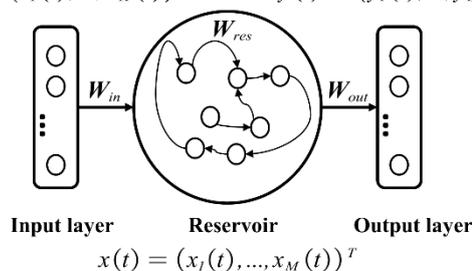


Figure 2. The ESN architecture.

The activation states of the reservoir neurons $x(t + 1)$ and the output layer $y(t + 1)$ at time $t + 1$ can be updated through the following equations [20]:

$$x(t + 1) = f(W_{in}u(t + 1) + W_{res}x(t)) \tag{4}$$

$$y(t + 1) = W_{out}x(t + 1) \tag{5}$$

where f presents the activation function of the neurons in the reservoir layer. We use \tanh function as the activation function.

In the canonical ESN model, W_{in} and W_{res} are initialized randomly and fixed, and the readout matrix W_{out} is trained by using the certain supervised learning algorithm [19]. This is called “free training”, and enables the ESN model to be a fast-training model. The purpose of the training process is to minimize the error between the desired output ($y_{desired}$) and the real output of the ESN model ($y(t)$) [21].

Furthermore, we apply the ridge regression algorithm to calculate the weights of the readout matrix W_{out} as follows [22]:

$$W_{out} = (X^T X + \lambda^2 I)^{-1} X^T y_{desired} \tag{6}$$

where $X = (x(t), u(t))^T$, I is the identity matrix, and $\lambda > 0$ is a regularization coefficient.

The hyperparameters of ESN include the number of neurons in the reserve pool (M), the spectral radius (SR), the connection density (CD), and the input scaling coefficient (IS). ESN has the advantage in training because most parameters of ESN are “free-training”. ESN is a simple and effective model but suffers from the problem that the effectiveness of ESN relies heavily on the selection of hyperparameters [23]. Therefore, hyperparameter configuration is an urgent problem to be solved. In the following, we will briefly introduce these hyperparameters and explain their importance.

- (1) *The number of neurons in the reservoir layer (P):* P is the most important hyperparameter, which determines the complexity of ESN and directly affects its performance in various tasks. Generally, increasing P can increase the memory capacity of ESN [24] that is of significance to handling complex tasks. However, excessively increasing P will lead to overfitting.
- (2) *Spectral radius (SR):* SR is the maximum achievable value of the characteristic value of the reservoir matrix W_{res} . To guarantee the echo state principle (ESP) of ESN, the value of SR should be less than 1 [25].

- (3) *Connection density (CD)*: CD indicates the number of neurons in the reservoir layer that participate in random connections. The selection range of CD is usually between 0.01 and 0.2 [26]. In addition, a range of interesting initial structures is proposed to select proper CD, such as distance-based [13], small-world [27], or scale-free [28,29].
- (4) *Input scaling coefficient (IS)*: It is important to choose an appropriate IS that can scale the input data to a proper range, so that the neurons of ESN can be well activated and fully take advantage of the non-linearity of the activation function [30].

2.2. Moth–Flame Optimization (MFO)

Moth–flame optimization (MFO) is an extension of the swarm intelligence algorithm [15]. It imitates the behavior of a moth flying spirally around the flame and, finally, reaching the flame.

MFO assumes that both moths and flames are candidate solutions to the problem. The sets of moths and flames are denoted as matrices $M = (N, \text{dim})$ and $F = (N, \text{dim})$, respectively, where N is the max numbers of moths and flames, and dim is the dimension of the solution. Moths represent real searching individuals in the solution space searching around the flame in a spiral motion. Flames denote the historical optimal solution reached by the moths spiraling around the closest flame. The positions of the individual moths and flames can be updated while the moths are flying [15] and have the following characteristics:

- (1) The moths can only fly within a limited search range.
- (2) The initialized positions of the moths are the starting points of the spiral flight.
- (3) The endpoints of the spiral flight are the positions of the flames.
- (4) The positions of the flames are adjusted during the process of searching.

The position of each moth can be updated by Equation (7), which simulates the flight mode of the moth.

$$M_i = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \quad (7)$$

where M_i denotes the i th moth, and F_j denotes the j th flame that is the closest flame to M_i . D_i is the Euclidean distance between M_i and F_j , which denotes the closest distance amongst M_i and flames. The shape of the spiral flight is controlled by the parameter b that is generally set to 1. The coefficient t is used to control the distance between the moth and the flame during the spiral flight and can be calculated by Equation (8).

$$t = \left(-\frac{l+L}{L} - 1 \right) \times \text{rand} + 1 \quad (8)$$

where l is the current iterations and L is the maximum iterations. rand is a uniformly distributed random number. For a certain iteration, the coefficient t calculated by Equation (8) is used to update the positions of all moths using Equation (7).

Equation (9) is used to decrease the number of flames (flame_{no}) during each iteration. L and l denote the maximum iterations and the current iterations, separately. N denotes the max number of flames.

With the number of flames decreasing in a linear way, the local search capacity of the moths is improved.

$$\text{flame}_{\text{no}} = \text{round} \left(N - l \times \frac{N-1}{L} \right) \quad (9)$$

3. Methodology

The previous introduction shows that the reservoir layer constructed by random initialization is the most important part of ESN. However, many researchers have proposed various methods to initialize ESN. For example, Strauss et al. [31] presented the design strategy of ESN, such as ensuring the echo state property (ESP) and reducing the influence of noise during the training process. Bianchi et al. [32] studied the dynamic characteristics of ESN through a recursive analysis, which contributes to further understanding and constructing the optimal reservoir layer. However, it is difficult to consider multiple param-

eters of a reservoir layer simultaneously, such as M , SR , CD , and IS . Therefore, tuning the hyperparameters is usually time-consuming and difficult, particularly for complex tasks.

To improve the above shortcomings, moth–flame optimization is used to select the hyperparameters of ESN, named MFO-ESN. Meanwhile, a novel evaluation function (FDEF), which can evaluate the performance of the feature extractor in a detailed and more objective way, is proposed as the fitness function of MFO-ESN.

3.1. Feature Distribution Evaluation Function (FDEF)

Features are the most important factor in machine learning projects where learning is easy if many independent features can be acquired and each correlate well with the class [33]. The quality of the features extracted from the original EEG signal has a great impact on the classification of the subsequent classifier. An excellent feature extractor should be able to produce obvious differences in the extracted features from different classes of EEG signals in distribution. For the extracted features of the same class, the similarities will be retained, while the differences that we are not concerned about will be ignored. Therefore, we propose a novel feature distribution evaluation function (FDEF), which is calculated as Equation (10).

$$\begin{aligned}
 \text{FDEF} &= \sum \frac{D(I_i^o, I_i^p)}{D(I_i^o, I_i^n)} \\
 \text{s.t.} \begin{cases} D(I_i^o, I_i^p) < D(I_i^o, I_i^n) \\ I_i^o \in I \end{cases} & \quad (10)
 \end{aligned}$$

where I is the feature set containing all features extracted from raw data, while I_i^o is the feature point we observed.

For every observed feature point I_i^o , we construct a triplet $I_i = \langle I_i^o, I_i^p, I_i^n \rangle$, including the positive point I_i^p and the negative point I_i^n . I_i^p is the center of the feature points whose class is the same as I_i^o , and I_i^n is the center of the feature points whose class is different from I_i^o . $D(I_i^o, I_i^p)$ and $D(I_i^o, I_i^n)$ denote the distance amongst I_i^o, I_i^p , and I_i^n , as measured by the L_2 -norm, respectively.

$D(I_i^o, I_i^p)$ is the positive distance that is the smaller the better, and $D(I_i^o, I_i^n)$ is the negative distance that should be larger than $D(I_i^o, I_i^p)$. FDEF can evaluate the ratio between positive distance and negative distance, which is different from another direct metric using subtraction, such as Equation (11):

$$\text{Loss}(I^o, I^p, I^n) = \sum \left(D(I_i^o, I_i^p) - D(I_i^o, I_i^n) + m \right)_+ \quad (11)$$

where m is a margin that is enforced between a positive pair and a negative pair. The purpose of minimizing Equation (11) is to make the distance between the positive pair smaller than the distance between the negative pairs by more than m .

Equation (11) is a direct way to describe the difference between positive distance and negative distance. It has been applied in FaceNet [18] as a loss function that achieved state-of-the-art performances in the person re-ID tasks. However, compared to Equation (11), Equation (10) has two advantages:

- (1) FDEF is robust to the mean value of features. The mean values of the features extracted through different ESN feature extractors are different. Further, features with a bigger mean value always obtain a better result using Equation (11) without considering the distribution of features.
- (2) FDEF is not sensitive to the dimension of the feature. The number of neurons in the reservoir layer (P) is a key parameter that needs to be adjusted. The feature dimension is the same as P , which means the feature dimension changes during the training

process of MFO-ESN. Equation (11) is sensitive to the varying feature dimensions that force the model to reduce P .

Therefore, compared to Equation (11), FDEF using the ratio description pays more attention to the distribution state of the aggregation and dispersion of the samples in the sample space, rather than the specific distance value and the difference in feature dimensions.

For Equation (10), the positive distance should be smaller than the negative distance, which promises that the observed feature point is closer to the center of the corresponding class—in other words, it is more centralized. Therefore, penalty items are added to those points that do not meet the constraint. The new form of FDEF is shown in the following:

$$\text{FDEF} = \sum \left(\frac{D(I_i^o, I_i^p)}{D(I_i^o, I_i^m)} + \alpha \cdot \text{Relu} \left(\frac{D(I_i^o, I_i^p)}{D(I_i^o, I_i^m)} - 1 \right) \right) \quad (12)$$

where $\alpha > 0$ is the punishment coefficient and $\text{Relu}(x)$ is the rectified linear unit. The parameter α is similar to the parameter m in Equation (11) that needs to be modified carefully to ensure the convergence of the model.

Regarding multiple classes, we follow the strictest method, which is to choose the smallest negative distance as the negative distance in the FDEF formula. This choice forces the spacing between classes to be more obvious. Of course, since all calculations are based on the L2 normalization, when the number of classes or points become larger, the issue of convergence should be considered.

3.2. Moth–Flame Optimized ESN

As mentioned above, most parameters of ESN are initialized randomly and fixed, and the initialization of ESN largely depends on the selection of the hyperparameters. The selection and adjustment of hyperparameters are crucial to the ESN model. Unfortunately, because the relations amongst these hyperparameters of ESN are not clear, choosing appropriate hyperparameters for the specific tasks is difficult and usually not sufficient. In response to this situation, the MFO-ESN model is proposed, and its structure is shown in Figure 3. In MFO-ESN, MFO is used to optimize the hyperparameters of ESN, so that the ESN feature extractor can better extract the features of the input EEG signal fitting EEG classification tasks.

The number of neurons in the reservoir layer (P), spectral radius (SR), connection density (CD), and input scaling coefficient (IS) are selected as the hyperparameters to be adjusted. Therefore, we set the dimension of moth and flame as 4 to represent these hyperparameters.

The features of the EEG signals are extracted using an ESN feature extractor [9] whose architecture is the same as ESN. The ESN feature extractor is an unsupervised model that applies the idea of the autoencoder to extract features from the EEG signals. It utilizes a readout matrix W_{out} as the hidden layer, as well as the extracted feature. MFO-ESN uses the label information to optimize the effectiveness of ESN by selecting appropriate hyperparameters. Different from the usual optimization ways, MFO-ESN does not use the classification accuracy of the classification task that may be influenced by classifiers as a fitness function; rather, it uses FDEF to evaluate the distribution of features extracted by ESN.

To evaluate the fitness of the moth, the corresponding hyperparameters are used to initialize ESN for extracting features from the raw EEG signals. Then, the fitness is calculated according to Equation (12). With the positions of moths updated according to Equation (7), their fitness changes. Flames denote the historical optimal solution reached by the moths that spiral around the closest flame.

Considering the calculation cost and the suggestion of initializing the ESN model in [31], P is set within the range of [5, 100], SR is within the range of [0.1, 1], CD is within the range of [0.1, 1], and IS is within the range of [0.1, 5].

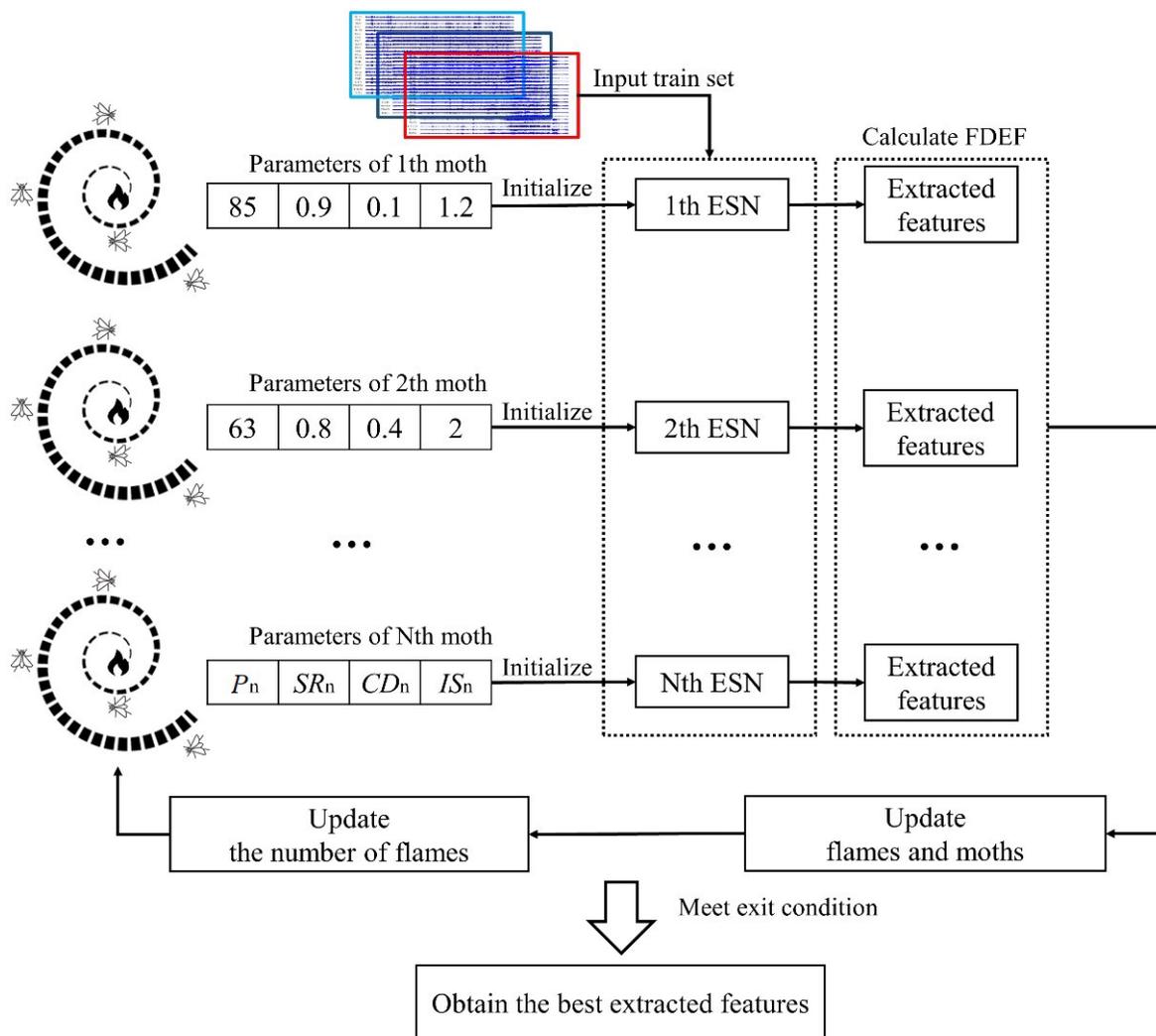


Figure 3. The structure of MFO-ESN.

MFO requires multiple moths to search around multiple flames. Since the search path of the moth is spiral, it converges slowly in the latter part of the iteration, while the local search capability decreases. Therefore, the number of moths cannot be too small. We set 20 as the population of moths as well as flames and set the maximum iterations as 100. The running process of the MFO-ESN algorithm is shown in Algorithm 1.

Algorithm 1 MFO-ESN

Input: the population number (N), the maximum times of iteration (T)

Output: the best hyperparameters of ESN

Steps:

- (a) Set the population number and maximum number of iterations.
- (b) Initialize the moth population.
- (c) Initialize the ESN using hyperparameters represented by moths.
- (d) Extract features using initialized ESN.
- (e) Calculate fitness values of moths according to Equation (12) and sort fitness values.
- (f) Update the moth position based on Equation (7).
- (g) Update the flame position to determine the current optimal solution.
- (h) Determine the number of moths and flames based on Equation (9).
- (i) Repeat step (c) to step (h) until the constraint is met.
- (j) The process ends.

4. Experiments on the Bonn University EEG Data Set

4.1. A Brief Description of the Data Set

We used the epileptic seizure events data from the Bonn University EEG data set that were provided by Andrzejak et al. in 2001 [34]. The complete data set consists of five subsets denoted from A to E, and each subset contains 100 single-channel EEG signals of 23.6-s duration (4097 time steps). These EEG signals are cut and selected from continuous multi-channel EEG records and after artificial artifact removal that mainly removes myoelectric artifacts and ocular artifacts. The electrode position adopts the international 10–20 system. The sampling frequency is 173.61 Hz and the filter bandwidth is 0.53–40 Hz (12 db/oct). Subsets A and B are taken from five healthy volunteers with eyes open and closed separately. Subsets C, D, and E are recorded from patients with epilepsy. Subsets C and D store the EEG signals during seizure-free intervals. The difference in C and D lies in the zone in which they are recorded. Set D is recorded within the epileptogenic zone, while set C is from the hippocampal formation of the opposite part of the brain. Each EEG segment in set E contains data related to at least one seizure activity.

4.2. Feature Extraction of Epileptic Seizure EEG Signals

To evaluate the performance of the MFO-ESN feature extraction model and compare this study with other similar studies, this paper divides the five groups of EEG signals into three classes (AB/CD/E), representing normal EEG signals, EEG signals from patients without seizures, and EEG signals from patients with epilepsy seizures.

We separated the training set and testing set using a random division whose division ratio is 8:2, and the MFO-ESN was trained with the training set. During each iteration of MFO-ESN, we recorded the fitness function FDEF and hyperparameters of the best extractor in the training set. Figure 4 shows the change of FDEF as the number of iterations increases.

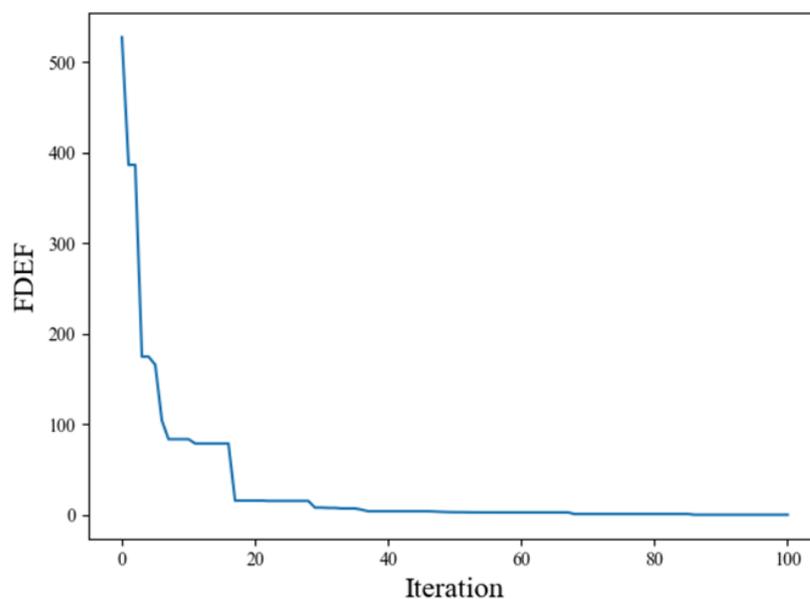


Figure 4. FDEF after 100 iterations by MFO-ESN.

To visualize the distribution of the extracted features, two dimensions of the extracted features are shown in Figure 5. It can be observed that the distribution of the three classes of the extracted features is regional. The first dimension of the class of A/B and the class of C/D is close because the EEG of two classes is not in the epileptic period. The second dimension is obviously different, which reflects the difference of two classes, i.e., the class of A/B is from healthy people, while the class of C/D is the inter-ictal EEG.

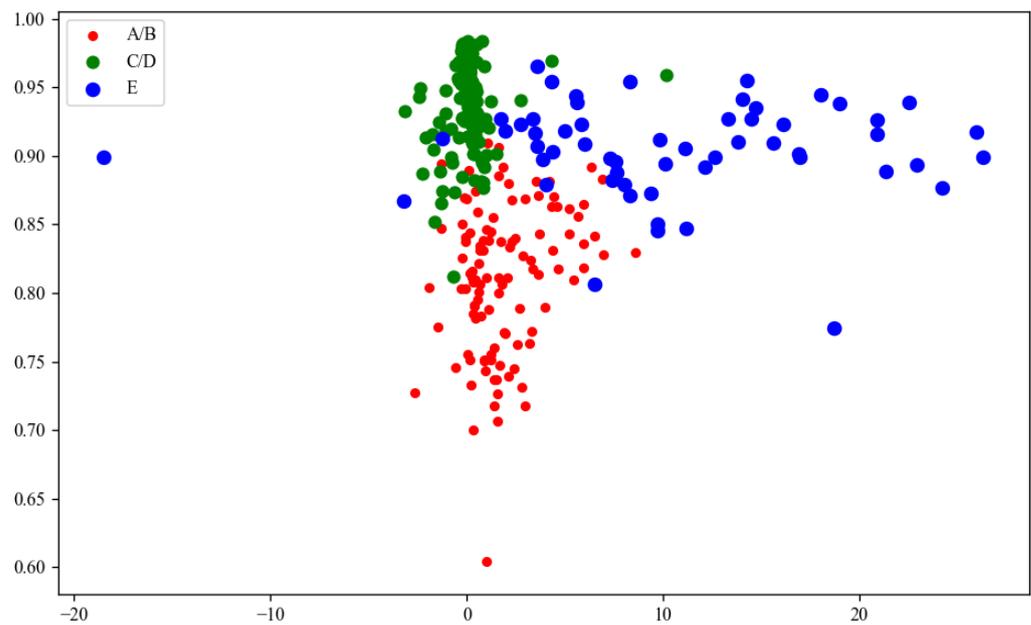


Figure 5. The distribution of extracted features in two dimensions.

4.3. Epileptic EEG Signal Classification

To compare the performance of MFO-ESN with other works, we applied the MFO-ESN combined with SVM to solve the epileptic classification task. SVM is a type of supervised learning method whose target is to find the maximum-margin hyperplane of the learning samples. The performance of SVM is widely verified by various applications [35,36]. The division of the data set is the same as in Section 4.2. In detail, we repeated the experiment 20 times and took the average value as the results for the MFO-ESN combining SVM.

Table 1 shows the results achieved by the MFO-ESN combining SVM and that of other methods. It is reasonable that the performance of MFO-ESN is better than the ESN feature extractor because the MFO-ESN can optimize the hyperparameters of the ESN feature extractor. Therefore, MFO-ESN combined with SVM obtained the highest accuracy of 98.16%. Moreover, to prove the effectiveness of FDEF, MFO-ESN without FDEF is proposed as the ablation experiment that uses the accuracy of SVM in the data set as the fitness function of MFO. However, MFO-ESN without FDEF achieves an accuracy of 93.19%, which is lower than the ESN feature extractor because of the problem of overfitting. As the direction of optimization is not to promote the accuracy for a specific classifier in the training set but to maximize the relative separability between classes, the MFO-ESN can achieve a more robust extractor. The ablation study shows that the FDEF plays an essential role in the MFO-ESN, which is also the main contribution of this research.

Table 1. Comparison of MFO-ESN + SVM with other methods.

Methods	Feature Extractor	Classifier	Accuracy
Guler et al. (2005) [37]	Lyapunov Exponents + RNN	Logistic regression	0.9697
Guo et al. (2011) [38]	Genetic Programming	KNN	0.9350
Shoeibi et al. (2018) [39]	CNN	Logistic regression	0.8870
Tuncer et al. (2019) [40]	LSP + NCA	SVM	0.9650
Raghu et al. (2019) [41]	Matrix Determinant	MLP	0.9652
Sema et al. (2020) [42]	5 BPFs + TFEBE	SVM	0.9488
this paper	ESN feature extractor	SVM	0.9592
this paper	MFO-ESN without FDEF	SVM	0.9319
this paper	MFO-ESN	SVM	0.9816

We aimed to evaluate the effectiveness of the features extracted by the MFO-ESN with other classifiers in the epilepsy classification task. We trained a variety of different classifiers using the training set data and evaluated them in the testing set. We compared the results with the original models and display the results in Table 2. It can be seen from the results that the proposed method can also outperform most of the previous research. It can be observed that compared with the original feature extraction models, the MFO-ESN feature extraction improves the classification accuracy of the classifier to varying degrees, which proves the effectiveness of MFO-ESN.

Table 2. The results of different classifiers using MFO-ESN.

Methods	Feature Extractors	Classifier	Accuracy
Guler et al. (2005) [37] this paper	Lyapunov Exponents + RNN MFO-ESN	Logistic regression	0.9697 0.9713
Guo et al. (2011) [38] this paper	Genetic Programming MFO-ESN	KNN	0.9350 0.9760
Raghu et al. (2019) [41] this paper	Matrix Determinant MFO-ESN	MLP	0.9652 0.9793
Shoeibi et al. (2018) [39] this paper	CNN MFO-ESN	Logistic regression	0.8870 0.9426
this paper this paper	ESN feature extractor MFO-ESN	SVM	0.9392 0.9816

5. Experiments on the CHB-MIT EEG Data Set

ESN can be effectively used to process multi-channel time series data, while all the EEG signals in the Bonn University data set are in single-channel form. To demonstrate that MFO-ESN is also suitable for multi-channel EEG tasks and to evaluate the performance of MFO-ESN with SVM in a further step, we conducted experiments on epilepsy classification by using the CHB-MIT EEG dataset.

5.1. A Brief Description of the Data Set

This dataset collected by Boston Children’s Hospital (CHB) and the Massachusetts Institute of Technology (MIT) consists of 916 h of continuous scalp EEG recordings grouped into 24 cases. This dataset is available to download online [43] and is described in detail in [44]. The EEG signals of each case recorded in the international 10–20 electrode system, with a sampling rate of 256 Hz, are saved in several EDF format files, and most of them contain 23 channels (24 or 26 channels in a few cases). Most files contain exactly one hour of EEG signals, while some files belonging to case 10 are two hours long, and those belonging to cases 4, case 6, case 7, case 9, and case 23 are four hours long.

A total of 198 seizures are manually annotated by medical specialists (pointing out the start time and end time) in 916 h. There are 45 s of seizure activity recordings on average that are too small compared to the normal EEG signals. To increase the sample number of seizure class and balance the data set, the data used in the experiments are converted into a series of segments using a sliding window whose size is 5 s, and every segment contains 23 channels as case 1. Segments belonging to a normal class do not include seizure EEG signals annotated by the experts; meanwhile, segments belonging to a seizure class do not include normal EEG signals. Figure 6 shows a period of EEG signals of case 5 and, according to the doctor’s annotations, the seizure begins in 2348 seconds and ends in 2465 seconds. The five-seconds-long window slides on the EEG signals without overlapping. In more detail, for a 117-seconds-long seizure process, we can obtain 23 segments of 5 seconds in length.

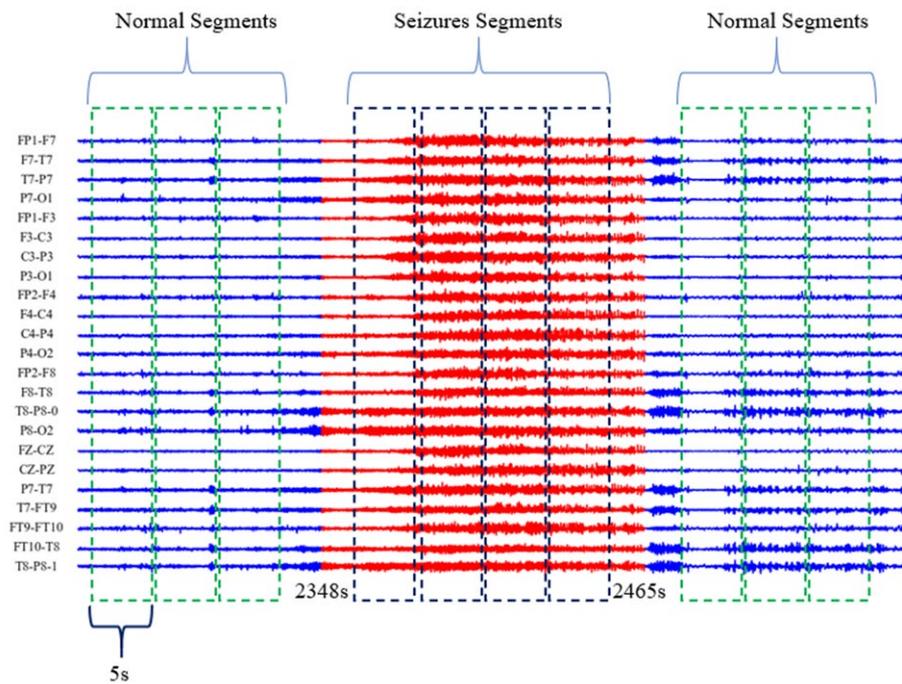


Figure 6. The division of the CHB-MIT data set.

This experiment consists of two parts: patient-specific and cross-patient. Patient specific denotes that the training set and testing set are both from the same case. We randomly chose a 20% segment of the seizure class; we chose the same number of segments from the seizure class for training and the rest of the segments for testing. Cross-patient denotes that the data from different cases are used to train and test. In more detail, the data from case 1, whose EEG signals are collected from an 11-year-old female, and case 2 from an 11-year-old male, are not involved in training but are only used for testing. The data from the other 22 cases are divided into the training set and the testing set. This setting, where case 1 and case 2 do not participate in training, can better simulate epilepsy diagnosis where the data from new patients cannot be trained.

5.2. Results and Discussion of the Multi-Channel Classification

To better evaluate the performance of the model and compare it to other works, three statistical indicators are used for evaluation:

$$Sensitivity = \frac{TP}{TP + FN} \tag{13}$$

$$Specificity = \frac{TN}{TN + FP} \tag{14}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

where TP , TN , FN , and FP are shown in Table 3.

Table 3. Confusion matrix.

		Predict Class	
		Seizure Class	Normal Class
Actual class	Seizure class	TP	FN
	Normal class	FP	TN

The quantity of the seizure class is far lower than the normal class, which heavily affects the accuracy. Therefore, of greater concern to us is the sensitivity, which evaluates how many seizure segments are classified correctly, and the specificity, which represents the ability of the model to accurately classify normal segments.

Table 4 shows the results of the patient-specific and cross-patient experiments. The patient-specific experiment can be considered as a private custom system where the training set and testing set are from the same case. The overall performance achieves a specificity of 92.56%, a sensitivity of 96.79%, and an accuracy of 92.75%. Out of all cases, case 7 obtains the highest accuracy of 99.97%, with a specificity of 100% and a sensitivity of 99.96%. As we can see, the accuracy is much closer to the specificity because the number of normal segments in the testing set is far greater than seizure segments. For an epilepsy diagnosis, sensitivity is more important than either accuracy or specificity, as high sensitivity can ensure that any suspicious patients will be further judged by the doctor in time.

Table 4. Results of patient-specific and cross-patient experiments.

Case	Patient-Specific				Cross-Patient			
	Sen. (%)	Spec. (%)	Acc. (%)	F1 (%)	Sen. (%)	Spec. (%)	Acc. (%)	F1 (%)
1	95.76	97.05	97.09	97.07	94.26	93.64	95.36	94.49
2	99.46	100.00	99.51	99.75	99.03	97.54	98.65	98.09
3	100.00	84.28	84.63	84.45	100.00	81.22	82.39	81.80
4	94.52	97.51	98.06	97.78	94.74	95.73	97.19	96.45
5	98.36	87.65	88.48	88.06	97.12	87.99	88.46	88.22
6	100.00	83.26	83.64	83.45	97.81	82.91	83.01	82.96
7	99.96	100.00	99.97	99.98	98.73	99.09	99.52	99.30
8	96.25	83.44	83.52	83.48	93.59	81.98	82.86	82.42
9	94.71	75.53	75.59	75.56	94.56	73.04	74.21	73.62
10	93.45	100.00	99.32	99.66	94.36	99.43	99.52	99.47
11	96.25	100.00	99.45	99.72	94.69	98.88	99.01	98.94
12	99.46	81.53	81.77	81.65	96.68	78.74	79.33	79.03
13	97.72	99.10	98.16	98.63	95.30	98.23	98.29	98.26
14	99.75	89.49	89.64	89.56	100.00	87.19	88.03	87.61
15	97.54	88.13	88.43	88.28	93.69	85.22	86.83	86.02
16	96.35	98.69	99.52	99.10	96.35	97.54	99.46	98.49
17	90.13	93.24	93.63	93.43	90.13	93.91	95.42	94.66
18	98.96	86.36	86.37	86.36	95.29	86.51	87.07	86.79
19	100.00	99.46	99.61	99.53	96.24	98.54	98.16	98.35
20	95.80	98.36	99.10	98.73	96.72	96.76	98.15	97.45
21	98.05	88.83	88.86	88.84	96.81	86.69	87.25	86.97
22	96.38	99.58	99.84	99.71	100.00	98.46	99.41	98.93
23	90.25	94.66	94.67	94.66	97.16	91.59	92.21	91.90
24	93.83	95.30	95.98	95.64	94.00	93.67	95.06	94.36
Ave.	96.79	92.56	92.75	92.63	96.14	91.02	91.92	91.44

Sen. = sensitivity, Spec. = specificity, Acc. = accuracy.

Compared to the patient-specific experiment, the cross-patient experiment achieves an average specificity of 91.02%, a sensitivity of 96.14%, and an accuracy of 91.92%. With the increase in the training segments and test samples, the performance of the model is not significantly different from the patient-specific experiment. More importantly, the performance of case 1 and case 2, which have not been trained before, is still good enough. The results prove that our model not only performs well on existing patient data, but also on other unseen patients.

As shown in Table 5, we compared the studies that use the CHB-MIT data set. Most of these studies use complex deep models such as CNNs, RNNs, and their variants. Li et al. [45] obtained the highest accuracy of 95.96% and highest sensitivity of 96.05% for the patient-specific experiment. Chen et al. [46] obtained the highest accuracy of 92.30% and

highest sensitivity of 92.89% for the cross-patient experiment. This is probably because ESN is a simple fast-training model whose ability to handle complex tasks is inferior to complex deep learning models. However, we achieved the highest sensitivity—which is more significant for diagnosis—of 96.79% in the patient-specific and 96.14% in cross-patient experiments. This may be one reason why minimizing FDEF can be regarded as a new way to maximize the difference between classes, so that our model can more effectively detect seizure segments.

Table 5. Comparisons of results between MFO-ESN + SVM and other methods.

Methods	Detector Type	Sen. (%)	Spec. (%)	Acc. (%)	F1 (%)
MIDS + CNN [47]	Patient-specific	74.08	92.46	83.27	87.62
Data argument + CNN [47]	Patient-specific	72.11	95.89	84.00	89.55
LMD + Bi-LSTM [48]	Patient-specific	93.61	91.85	92.66	92.25
CE-stSENet [45]	Patient-specific	92.41	96.05	95.96	96.00
KNN [49]	Cross-patient	88.00	88.00	88.00	88.00
Dyadic WT + SVM [46]	Cross-patient	91.71	92.89	92.30	92.59
MFO-ESN + SVM	Patient-specific	96.79	92.56	92.75	92.65
MFO-ESN + SVM	Cross-patient	96.14	91.02	91.92	91.47

6. Conclusions

In this paper, the proposed MFO-ESN model uses MFO to optimize the hyperparameters of ESN. Further, a new feature distribution evaluation function, FDEF, is proposed as the fitness function of MFO-ESN by using the label information of the EEG signals. Without using specific classifiers, MFO-ESN can extract more suitable features for the classification tasks. Combined with the SVM model, the effectiveness of the extracted features is verified on the Bonn EEG multi-classification data set and obtains an average accuracy of 98.16%. It is also found that the features extracted by the MFO-ESN model improve the performance of multiple classifiers, which means that MFO-ESN is an effective preprocessing method for EEG signal classification tasks. Furthermore, we apply MFO-ESN with SVM in the multi-channel EEG signals classification task of CHB-MIT and obtain the highest sensitivity of 96.79% in patient-specific and 96.14% in cross-patient experiments with good specificity and accuracy.

Apart from its higher performance, MFO-ESN with FDEF has two additional advantages compared with the previous EEG feature extraction methods: (1) it can automatically optimize the hyperparameters of ESN to adjust the architecture and initial parameters for the specific classification task; (2) using FDEF as the fitness of MFO decouples the optimization of the feature extractor from the classifier, which means the optimization direction no longer relies on the performance of the classifier but on the relative separability amongst classes. These advantages mean that the proposed MFO-ESN model can achieve efficient feature extraction and demonstrate that it has a better generalization ability.

In the future, this method can be further studied with respect to the following considerations. From the perspective of practical applications, the efficiency of MFO-ESN can be further evaluated on other EEG signal classification tasks. Given that the hyperparameters of ESN are searched in the set parameter space, how to improve the efficiency of the search and stop functions in the proper time are also key problems. In the theoretical direction, we can further improve the current ESN method. For example, the relationships amongst channels can be well described by the positioned EEG electrodes where epileptic seizures in different brain regions have different manifestations and treatments. By designing the structure of ESN to combine both the spatial information represented by the channels and the temporal information existing in the EEG time series, a better feature extraction result is promising.

Author Contributions: Methodology, X.-s.T.; Investigation, L.J.; Data curation, X.L.; Supervision, K.H. and T.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Fundamental Research Funds for the Central Universities (2232021D-37), the National Natural Science Foundation of China (nos. 61806051, 62176052), and the Natural Science Foundation of Shanghai (21ZR1401700, 20ZR1400400).

Data Availability Statement: The data can be acquired via emails of the first author and corresponding author.

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

References

1. Megiddo, I.; Colson, A.; Chisholm, D.; Dua, T.; Nandi, A.; Laxminarayan, R. Health and economic benefits of public financing of epilepsy treatment in India: An agent-based simulation model. *Epilepsia* **2016**, *57*, 464–474. [[CrossRef](#)] [[PubMed](#)]
2. Liu, Q.; Ganzetti, M.; Wenderoth, N.; Mantini, D. Detecting Large-Scale Brain Networks Using EEG: Impact of Electrode Density, Head Modeling and Source Localization. *Front. Neuroinform.* **2018**, *12*, 4. [[CrossRef](#)] [[PubMed](#)]
3. Lehnertz, K. Epilepsy and Nonlinear Dynamics. *J. Biol. Phys.* **2008**, *34*, 253–266. [[CrossRef](#)] [[PubMed](#)]
4. Chan, H.-L.; Kuo, P.-C.; Cheng, C.-Y.; Chen, Y.-S. Challenges and Future Perspectives on Electroencephalogram-Based Biometrics in Person Recognition. *Front. Neuroinform.* **2018**, *12*, 66. [[CrossRef](#)] [[PubMed](#)]
5. Vijayan, A.E.; Sen, D.; Sudheer, A. EEG-Based Emotion Recognition Using Statistical Measures and Auto-Regressive Modeling. In Proceedings of the 2015 IEEE International Conference on Computational Intelligence & Communication Technology, IEEE, Ghaziabad, India, 13–14 February 2015; pp. 587–591. [[CrossRef](#)]
6. Farihah, S.N.; Lee, K.Y.; Mansor, W.; Mohamad, N.B.; Mahmoodin, Z.; Saidi, S.A. EEG average FFT index for dyslexic children with writing disorder. In Proceedings of the 2015 IEEE Student Symposium in Biomedical Engineering & Sciences (ISSBES), IEEE, Shah Alam, Malaysia, 4 November 2015; pp. 118–121. [[CrossRef](#)]
7. Yuyi, Z.; Surui, L.; Lijuan, S.; Zhenxin, L.; Bingchao, D. Motor imagery eeg discrimination using hilbert-huang entropy. *Biomed. Res.* **2017**, *28*, 727–733.
8. Zhou, M.; Tian, C.; Cao, R.; Wang, B.; Niu, Y.; Hu, T.; Guo, H.; Xiang, J. Epileptic Seizure Detection Based on EEG Signals and CNN. *Front. Neuroinform.* **2018**, *12*, 95. [[CrossRef](#)]
9. Mishra, S.; Birok, R. Sleep Classification using CNN and RNN on Raw EEG Single-Channel. In Proceedings of the 2020 International Conference on Computational Performance Evaluation (ComPE), IEEE, Shillong, India, 2–4 July 2020; pp. 232–237. [[CrossRef](#)]
10. Hochreiter, S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **1998**, *6*, 107–116. [[CrossRef](#)]
11. Sun, L.; Jin, B.; Yang, H.; Tong, J.; Liu, C.; Xiong, H. Unsupervised EEG feature extraction based on echo state network. *Inf. Sci.* **2019**, *475*, 1–17. [[CrossRef](#)]
12. Jaeger, H. Echo state network. *Scholarpedia* **2007**, *2*, 2330. [[CrossRef](#)]
13. Jarvis, S.; Rotter, S.; Egert, U. Extending stability through hierarchical clusters in Echo State Networks. *Front. Neuroinform.* **2010**, *4*, 11. [[CrossRef](#)]
14. Wang, L.; Hu, H.; Ai, X.-Y.; Liu, H. Effective electricity energy consumption forecasting using echo state network improved by differential evolution algorithm. *Energy* **2018**, *153*, 801–815. [[CrossRef](#)]
15. Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl. Based Syst.* **2015**, *89*, 228–249. [[CrossRef](#)]
16. Mei, R.N.S.; Sulaiman, M.H.; Mustafa, Z.; Daniyal, H. Optimal reactive power dispatch solution by loss minimization using moth-flame optimization technique. *Appl. Soft Comput.* **2017**, *59*, 210–222. [[CrossRef](#)]
17. Shehab, M.; Abualigah, L.; Al Hamad, H.; Alabool, H.; Alshinwan, M.; Khasawneh, A.M. Moth-flame optimization algorithm: Variants and applications. *Neural Comput. Appl.* **2020**, *32*, 9859–9884. [[CrossRef](#)]
18. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.
19. Jaeger, H. *Tutorial on Training Recurrent Neural Networks, Covering BPPT, RTRL, EKF and the “Echo State Network” Approach*; GMD-Forschungszentrum Informationstechnik: Bonn, Germany, 2002.
20. Bo, Y.-C.; Zhang, X. Online adaptive dynamic programming based on echo state networks for dissolved oxygen control. *Appl. Soft Comput.* **2018**, *62*, 830–839. [[CrossRef](#)]
21. Wang, X.; Jin, Y.; Hao, K. Evolving Local Plasticity Rules for Synergistic Learning in Echo State Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 1363–1374. [[CrossRef](#)]
22. Wyffels, F.; Schrauwen, B.; Stroobandt, D. Stable Output Feedback in Reservoir Computing Using Ridge Regression. In *International Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 808–817. [[CrossRef](#)]
23. Chouikhi, N.; Ammar, B.; Rokbani, N.; Alimi, A.M. PSO-based analysis of Echo State Network parameters for time series forecasting. *Appl. Soft Comput.* **2017**, *55*, 211–225. [[CrossRef](#)]

24. Lukoševičius, M.; Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Comput. Sci. Rev.* **2009**, *3*, 127–149. [CrossRef]
25. Jaeger, H.; Maass, W.; Principe, J. Special issue on echo state networks and liquid state machines. *Neural Netw.* **2007**, *20*, 287–289. [CrossRef]
26. Song, Q.; Feng, Z. Effects of connectivity structure of complex echo state network on its prediction performance for nonlinear time series. *Neurocomputing* **2010**, *73*, 2177–2185. [CrossRef]
27. Watts, D.J.; Strogatz, S.H. Collective dynamics of ‘small-world’ network. *Nature* **1998**, *393*, 440–442. [CrossRef] [PubMed]
28. Deng, Z.; Zhang, Y. Collective Behavior of a Small-World Recurrent Neural System With Scale-Free Distribution. *IEEE Trans. Neural Netw.* **2007**, *18*, 1364–1375. [CrossRef] [PubMed]
29. Barabási, A.-L.; Albert, R. Emergence of scaling in random networks. *Science* **1999**, *286*, 509–512. [CrossRef] [PubMed]
30. Jaeger, H.; Haas, H. Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. *Science* **2004**, *304*, 78–80. [CrossRef]
31. Strauss, T.; Wustlich, W.; Labahn, R. Design Strategies for Weight Matrices of Echo State Networks. *Neural Comput.* **2012**, *24*, 3246–3276. [CrossRef]
32. Bianchi, F.M.; Livi, L.; Alippi, C. Investigating Echo-State Networks Dynamics by Means of Recurrence Analysis. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *29*, 427–439. [CrossRef]
33. Domingos, P. A Few Useful Things to Know about Machine Learning. *Commun. ACM* **2012**, *55*, 78–87. [CrossRef]
34. Andrzejak, R.G.; Lehnertz, K.; Mormann, F.; Rieke, C.; David, P.; Elger, C.E. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Phys. Rev. E* **2001**, *64*, 061907. [CrossRef]
35. Chauhan, V.K.; Dahiya, K.; Sharma, A. Problem formulations and solvers in linear SVM: A review. *Artif. Intell. Rev.* **2019**, *52*, 803–855. [CrossRef]
36. Raj, S.; Ray, K.C. ECG Signal Analysis Using DCT-Based DOST and PSO Optimized SVM. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 470–478. [CrossRef]
37. Güler, N.F.; Übeyli, E.D.; Güler, I. Recurrent neural networks employing Lyapunov exponents for EEG signals classification. *Expert Syst. Appl.* **2005**, *29*, 506–514. [CrossRef]
38. Guo, L.; Rivero, D.; Dorado, J.; Munteanu, C.R.; Pazos, A. Automatic feature extraction using genetic programming: An application to epileptic EEG classification. *Expert Syst. Appl.* **2011**, *38*, 10425–10436. [CrossRef]
39. Shoeibi, A.; Ghassemi, N.; Alizadehsani, R.; Rouhani, M.; Hosseini-Nejad, H.; Khosravi, A.; Panahiazar, M.; Nahavandi, S. A comprehensive comparison of handcrafted features and convolutional autoencoders for epileptic seizures detection in EEG signals. *Expert Syst. Appl.* **2020**, *163*, 113788. [CrossRef]
40. Tuncer, T.; Dogan, S.; Akbal, E. A novel local senary pattern based epilepsy diagnosis system using EEG signals. *Australas. Phys. Eng. Sci. Med.* **2019**, *42*, 939–948. [CrossRef]
41. Raghu, S.; Sriraam, N.; Hegde, A.S.; Kubben, P.L. A novel approach for classification of epileptic seizures using matrix determinant. *Expert Syst. Appl.* **2019**, *127*, 323–341. [CrossRef]
42. de la O Serna, J.A.; Paternina, M.R.A.; Zamora-Mendez, A.; Tripathy, R.K.; Pachori, R.B. EEG-Rhythm Specific Taylor–Fourier Filter Bank Implemented With O-Splines for the Detection of Epilepsy Using EEG Signals. *IEEE Sens. J.* **2020**, *20*, 6542–6551. [CrossRef]
43. Goldberger, A.; Amaral, L.; Glass, L.; Hausdorff, J.; Ivanov, P.C.; Mark, R.; Mietus, J.; Moody, G.; Peng, C.; Stanley, H. Components of a new research resource for complex physiologic signals, PhysioBank, PhysioToolkit, and Physionet. *Circulation* **2000**, *101*, e215–e220. [CrossRef]
44. Shoeb, A.H. Application of Machine Learning to Epileptic Seizure Onset Detection and Treatment. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2009. Available online: <https://dspace.mit.edu/handle/1721.1/7582> (accessed on 3 March 2023).
45. Li, Y.; Liu, Y.; Cui, W.-G.; Guo, Y.-Z.; Huang, H.; Hu, Z.-Y. Epileptic Seizure Detection in EEG Signals Using a Unified Temporal-Spectral Squeeze-and-Excitation Network. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 782–794. [CrossRef]
46. Chen, D.; Wan, S.; Xiang, J.; Bao, F.S. A high-performance seizure detection algorithm based on Discrete Wavelet Transform (DWT) and EEG. *PLoS ONE* **2017**, *12*, e0173138. [CrossRef]
47. Wei, Z.; Zou, J.; Zhang, J.; Xu, J. Automatic epileptic EEG detection using convolutional neural network with improvements in time-domain. *Biomed. Signal Process. Control.* **2019**, *53*, 101551. [CrossRef]
48. Hu, X.; Yuan, S.; Xu, F.; Leng, Y.; Yuan, K.; Yuan, Q. Scalp EEG classification using deep Bi-LSTM network for seizure detection. *Comput. Biol. Med.* **2020**, *124*, 103919. [CrossRef] [PubMed]
49. Fergus, P.; Hussain, A.; Hignett, D.; Al-Jumeily, D.; Abdel-Aziz, K.; Hamdan, H. A machine learning system for automated whole-brain seizure detection. *Appl. Comput. Inform.* **2016**, *12*, 70–89. [CrossRef]

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