Hindawi
Concepts in Magnetic Resonance Part B, Magnetic Resonance Engineering
Volume 2022, Article ID 1047616, 14 pages
https://doi.org/10.1155/2022/1047616

Research Article

Diagnosis of Alzheimer's Disease with Extreme Learning Machine on Whole-Brain Functional Connectivity

Jia Lu,[1] **Weiming Zeng**,[1] **Lu Zhang**,[2] **and Yuhu Shi**[3]

[1] *Laboratory of Digital Image and Intelligent Computation, Shanghai Maritime University, 1550 Harbor Avenue, Pudong, Shanghai 201306, China*
[2] *Basic Experiment and Training Center, Shanghai Maritime University, 1550 Harbor Avenue, Pudong, Shanghai 201306, China*
[3] *College of Information Engineering, Shanghai Maritime University, 1550 Harbor Avenue, Pudong, Shanghai 201306, China*

Correspondence should be addressed to Weiming Zeng; zengwm86@163.com

Received 13 December 2021; Revised 17 April 2022; Accepted 6 May 2022; Published 19 May 2022

Academic Editor: Jochen Leupold

The analysis of human brain fMRI subjects can research neuro-related diseases and explore the related rules of human brain activity. In this paper, we proposed an algorithm framework to analyze the functional connectivity network of the whole brain and to distinguish Alzheimer's disease (AD), mild cognitive impairment (MCI), and cognitively normal (CN). In other studies, they use algorithms to select features or extract abstract features, or even manually select features based on prior information. Then, a classifier is constructed to classify the selected features. We designed a concise algorithm framework that uses whole-brain functional connectivity for classification without feature selection. The algorithm framework is a two-hidden-layer neural network based on extreme learning machine (ELM), which overcomes the instability of classical ELM in high-dimensional data scenarios. We use this method to conduct experiments for AD, MCI, and CN data and perform 10-fold cross-validation. We found that it has several advantages: (1) the proposed method has excellent classification accuracy with high speed. The classification accuracy of AD vs. CN is 96.85%, and the accuracy of MCI vs. CN is 95.05%. Their AUC (area under curve) of ROC (receiver operating characteristic curve) reached 0.9891 and 0.9888, respectively. Their sensitivities are 97.1% and 94.7%, and specificities are 96.3% and 95.3%, respectively. (2) Compared with other studies, the proposed method is concise. Construction of a two-hidden-layer neural network is to learn features of the whole brain for the diagnosis of AD and MCI, without the feature screening. It avoids the negative effects of feature screening by algorithm or prior information. (3) The proposed method is suitable for small sample and high-dimensional data. It meets the requirements of medical image analysis. In other studies, its classifiers usually deal with several to dozens of feature dimensions. The proposed method deals with 4005 feature dimensions.

1. Introduction

Alzheimer's disease is a neurodegenerative disease with insidious onset and progressive development. Clinically, full-scale dementia is characterized by memory impairment, aphasia, apraxia, agnosia, impairment of visuospatial skills, executive dysfunction, and personality and behavior changes, which seriously affect patients' daily life. According to statistics [1], in most developed countries, about 50% of AD patients have been diagnosed and treated. In developing countries, less than 10% of AD patients are diagnosed and treated. Clinical diagnosis of AD is usually made after the onset of dementia symptoms, and most patients are already in the middle and late stages of AD at this stage, and treatment at this stage is often ineffective. Mild cognitive impairment (MCI) is a small, measurable change in the ability to think, but one who can still perform everyday activities. This is a transitional stage between healthy elderly people and AD [2]. About 15 to 20 percent of people over 65 years of age have MCI [3]. Compared with healthy elder adults, people with MCI, especially those with memory impairment, have a higher risk of developing AD or other forms of dementia. The review of Ward et al. [4] showed that about 32% of MCI patients would develop AD within 5 years, while the annual conversion rate of the elderly with normal cognitive function was only about 1%. Although

MCI has a high risk of developing AD, if early detection and timely intervention and treatment can be carried out, the condition of MCI patients does not necessarily develop to AD state. Therefore, early detection, diagnosis, and treatment of MCI can delay the occurrence of AD, which has important clinical and social significance.

In 2004, Huang et al. [5] proposed a simple and efficient single-hidden-layer feedforward neural network (SLFN) algorithm. It is called extreme learning machine (ELM). ELM randomly selects the input weights and hidden layer bias of the network, and obtains the output weights through analytical calculation, effectively overcoming the shortcomings of the traditional SLFN learning algorithm, and be widely used in many fields such as disease diagnosis, traffic sign recognition, image quality assessment, and so on [6–8]. ELM strives to solve the research problems in machine learning fields such as regression, classification, clustering, compression, and feature extraction under a single framework. From the perspective of learning efficiency, ELM is concise to implement, with extremely high learning speeds and less human intervention. From the perspective of theoretical studies, ELM can still maintain SLFN's interpolation ability [7], general approximation ability [9], and classification ability [10] even in the case of randomly generating hidden layer neuron parameters. From the perspective of structural risk minimization, the VC dimension (Vapnik–Chervonenkis dimension) of ELM depends on the number of neurons in the hidden layer [11]. The size of the VC dimension can be controlled by adjusting the number of neurons in the hidden layer of ELM, to make a compromise between training error and model complexity, and get the optimal generalization performance. ELM has also been extended to a deep learning model [12, 13] and made a lot of research results.

In recent years, machine learning techniques have also been gradually applied to the analysis of brain image data for the diagnosis of AD-like diseases. Wee et al. [14] proposed an approach to extract cortical morphological abnormality patterns from structural magnetic resonance imaging (MRI) data to predict AD and MCI. It has an accuracy of 92.35% for AD and 83.75% for MCI, with an area under the ROC curve (AUC) of 0.9744 and 0.9233, respectively. Jie et al. [15] proposed a connectivity-networks-based classification framework to identify accurately the MCI patients from cognitively normal (CN). It has an accuracy of 91.9%, with an AUC of 0.94. Khazaee et al. [16] combined graph theoretical approaches with advanced machine learning methods to study functional brain network alteration in patients with AD. Using support vector machines (SVM) to diagnose AD based on graph measure, with an accuracy of 97%. Nguyen et al. [17] proposed a voxel-wise discriminative framework applied to multimeasure resting-state fMRI that integrates hybrid MVPA and ELM for the automated discrimination of AD and MCI from CN. It has achieved an accuracy rate of 98.86% and 98.57% in the diagnosis of AD and MCI. Bi X et al. [18] proposed 2 deep learning methods of functional brain network classification. The convolutional learning method learns the deep regional connectivity features, while the recurrent learning method learns the deep adjacent positional features. The ELM-boosted structure is implemented to further improve the learning ability. Bi X et al. [19] proposed an aggregator based on ELM that boosts the aggregation ability and efficiency of graph convolution without iterative tuning and designed a graph neural network with ELM aggregator for the graph classification. Lama et al. [20] proposed a diagnosis approach using graph theory-based features from fMRI to discriminate AD, MCI, and CN. It includes linear SVM, and regularized ELM. It has achieved an accuracy rate of 90.93% and 98.91% in the diagnosis of AD and MCI.

In the field of AD diagnosis with fMRI/MRI, features are usually selected by manual or other methods and then classified using SVM, ELM, etc. [14, 15, 17–20]. It is generally considered that the accuracy of classification can be further improved through feature filtering or that classification methods such as SVM and ELM are not suitable for classification of high feature dimensional and small sample size data. In this paper, we designed classification experiments for AD and CN. It is confirmed that the ELM method is suitable for classification with high feature dimension and small sample size scenarios. And we found that ELM has the advantages of high accuracy, fast computation, and strong generalization ability in this scenario, and also found that it has the disadvantage of unstable accuracy. We propose the parallel ELM method, which inherits the advantages of the ELM method while improving the stability and the accuracy of the ELM method.

The main contributions of this study are summarized as follows:

(1) The proposed method is suitable for classification with high feature dimension and small sample size scenarios. And it avoids the instability of ELM methods in this scenario and improves the accuracy.

(2) The proposed method has excellent classification accuracy. The classification accuracy of AD vs. CN is 96.85%, and the accuracy of MCI vs. CN is 95.05%. Their AUC (area under curve) of ROC (receiver operating characteristic curve) reached 0.9891 and 0.9888, respectively.

(3) Compared with other studies, the proposed method is concise. Construction of a two-hidden-layer neural network is to learn features of the whole brain for the diagnosis of AD and MCI, without the feature screening. It avoids the negative effects of feature screening by algorithm or prior information.

2. Materials and Methods

2.1. Subjects. All fMRI data we used came from LONI's ADNI database, the ADNI2 project. The subjects were cognitively normal (CN), mild cognitive impairment (MCI), and Alzheimer's disease (AD). 100 CN subjects, 100 MCI subjects, and 100 AD subjects were obtained. It is important to note that ADNI 2 subdivides MCI into EMCI and LMCI. The MCI data obtained in this paper included 50 EMCI subjects and 50 LMCI subjects. The participants' data download address is https://adni.loni.usc.edu/. The age distribution of various subjects is shown in Figure 1. The mean age and standard deviation of CN, MCI, and AD subjects were 73.13 ± 6.49, 74.85 ± 5.94, and 75.07 ± 7.63, respectively. The ratio of males to females is 1 : 1 in all categories.

The fMRI scan parameters we selected are as follows: Field Strength = 3.0 tesla; Flip Angle = 80.0 degree; Matrix $X = 64.0$ pixels; Matrix $Y = 64.0$ pixels; Mfg Model = Intera; Pixel Spacing $X = 3.3125$ mm; Pixel Spacing $Y = 3.3125$ mm; Pulse Sequence = GR; Slices = 6720.0; Slice Thickness = 3.312999963760376 mm; TE = 30 ms; and TR = 3000 ms;

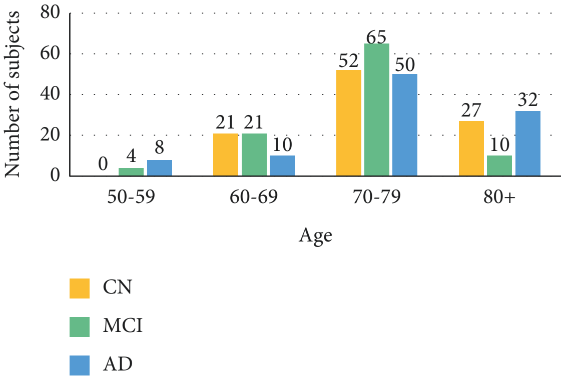


Figure 1: Age distribution of subjects.

2.2. Brain Functional Connectivity. Brain functional connectivity network is a mathematical representation defined by a set of nodes and edges [21]. These nodes represent brain regions on different scales. The temporal correlations (functional connectivity) between the fMRI time course of these nodes form the edge of the brain's functional network. The smaller the size of a node, the greater the number of nodes and edges. The more complex the described pattern of neural activity in the brain, the more difficult it is to calculate and analyze. Researchers often use templates to divide the brain into regions or nodes. Automatic anatomical labeling (AAL) [22] template is one of the most commonly used templates. AAL divides the brain into 116 regions, including 90 regions of the cerebrum and 26 regions of the cerebellum. In fMRI data, each region in the brain corresponds to a time series. The connectivity between each pair of brain regions can be represented by the correlation coefficients of their time series. The Pearson correlation coefficient is one of the most popular statistics for measuring the linear correlation between two normally distributed variables. The Pearson correlation coefficient of two brain regions x and y is calculated as follows:

$$r_p = \frac{\operatorname{cov}(x, y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n} (y_i - \overline{y})^2}}, \qquad (1)$$

where x and y are time series corresponding to brain regions, $x = (x_1, \ldots, x_n)$, $y = (y_1, \ldots, y_n)$. cov (x, y) is the covariation of x and y. σ_x and σ_y are the standard deviations of x and y. $\overline{x}$ and $\overline{y}$ are the mean values of x and y.

The feature measures adopted in this paper is the whole cerebrum functional connectivity network. In other words, the AAL template was used to extract the time course of 90 brain regions, and the Pearson correlation was calculated to form the functional connectivity network.

2.3. Extreme Learning Machine. Extreme learning machines (ELM) belong to single-hidden-layer feedforward neural networks (SLFNs) and have the characteristics of single-hidden-layer neural networks: 1. implement complex nonlinear mapping directly from the input layer and 2. can provide appropriate classification model for large category data sets. Compared with other single-hidden-layer neural network models, the speed of model training and classification is faster. Huang and Babri et al. pointed out in article [23] that the input layer weights and hidden layer bias values of other SLFNS networks need to be iteratively adjusted to fit the current training data, and in the case of a large number of hidden layer nodes, such calculation will bring greater calculation time consumption [24, 25]. At the same time, since gradient descent has become an effective method to solve SLFNs, this method not only limits the solving speed, but also can easily fall into the local minimum from the calculation principle of the calculation method. Aiming at the above problems, Huang et al. [5] proposed the algorithm of extreme learning machine, which transformed the iterative solution method into the solution method of linear equations by randomly specifying the weight and bias values of the input layer, and finally obtained the analytical solution of the network. The proposed algorithm can be quickly solved on the premise of ensuring the accuracy of calculation.

Extreme learning machine can be described as: given N arbitrary samples $\{X_i, t_i\}$, $X_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in R^n, t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \in R^m$. For a single-hidden-layer neural network with L hidden-layer nodes, it can be expressed as

$$\sum_{i=1}^{L} \beta_i g\left(W_i \cdot X_j + b_i\right) o_j, \quad j = 1, \ldots, N, \qquad (2)$$

where g (x) is the activation function. In our article, the activation function used is the sigmoid function. $W_i = [w_{i,1}, w_{i,2}, \ldots, w_{i,n}]^T$ is input weight , β_i is output weight, and b_i is the bias of the ith hidden-layer element. $W_i \cdot X_j$ is the scalar product of W_i and X_j. The goal of single-hidden-layer neural network learning is to minimize the output error, can be represented as

$$\sum_{i=1}^{L} o_j - t_j = 0, \quad j = 1, \ldots, N. \qquad (3)$$

There are β_i, W_i, and b_i, such that

$$\sum_{i=1}^{L} \beta_i g\left(W_i \cdot X_j + b_i\right) = t_j, \quad j = 1, \ldots, N. \qquad (4)$$

It can be expressed in matrix form

$$H\beta = T, \qquad (5)$$

where H is the output of the hidden-layer node, β is the output weight, and T is the expected output.

$$H = (W_1, \ldots, W_L, b_1, \ldots, b_L, X_1, \ldots, X_L)$$
$$= \begin{bmatrix} g(W_1 \cdot X_1 + b_1) & \cdots & g(W_L \cdot X_1 + b_L) \\ \vdots & \cdots & \vdots \\ g(W_1 \cdot X_N + b_1) & \cdots & g(W_L \cdot X_N + b_L) \end{bmatrix}_{N\times L},$$
$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L\times M}, \tag{6}$$
$$T = \begin{bmatrix} T_1^T \\ \vdots \\ T_L^T \end{bmatrix}_{N\times M}.$$

In the ELM algorithm, W_i and b_i are randomly determined, and the output matrix H of the hidden layer is uniquely determined. The training of the single-hidden-layer neural network can be transformed into adding a linear system, formula (5). And the output weight β can be determined by the following formula:

$$\widehat{\beta} = H^{\dagger}T, \tag{7}$$

where $H^{\dagger}$ is the Moore–Penrose inverse of H, and the solution norm of $\widehat{\beta}$ is minimal and unique. We solve for $\widehat{\beta}$ to construct ELM.

Proposed ELM algorithm framework based on whole-brain functional connectivity.

The hyper parameters involved in the ELM model include input weight, bias, activation function, number of input nodes, number of output nodes, and number of hidden-layer nodes. The input weight and bias are randomly generated. The number of input nodes is determined by the characteristic dimension of the data to be analyzed. This paper analyzes the features of the entire network of functional connections throughout the cerebrum. Calculated according to the functional connection network of 90 regions in the AAL template, the data characteristic dimension is 4005; that is, the number of input nodes is 4005. In hidden layer, select the most commonly used sigmoid function as the activation function. The number of output nodes is equal to the number of categories of sample data. For binary classification, the number of output nodes is 2; for ternary classification, the number of output nodes is 3, and so on.

Figure 2 depicts the entire process of the proposed algorithm framework.

(1) The fMRI data of AD, MCI, and CN obtained from ADNI were preprocessed to obtain the whole-brain functional connections of all subjects.

(2) The functional connection vectors of 90 brain regions of all subjects were extracted and constructed into a 4005 × N matrix, where N is the total number of subjects.

(3) Select a certain percentage subjects randomly from all the categories as the test set, and the other subjects serve as the training set. The proportion of subjects of each class in training set and test set is the same.

(4) Build ELM classifier with training set. Record training results, training accuracy, and other output information.

(5) The ELM classifier is used to classify and predict the test set. Record test results, test accuracy, sensitivity, specificity, etc.

(6) Perform the computation in a loop of step (3), (4), and (5) until the average accuracy converges. Finally, we get the average performance of the algorithm.

Since the input weights and bias in the ELM method are generated randomly, the division of the training set and the test set is also generated randomly. Therefore, the test accuracy varies randomly within a certain range. Formula (8) was designed to ensure a reliable average performance of the proposed method. The variable Loop is the number of cycles. The variable o is the fluctuation range of average accuracy. We need a large enough value of Loop and an appropriate value of o so that they satisfy formula (8). The precision of the average accuracy is controlled by adjusting the value of o. The logic of formula (8) is descripted Figure 3. Repeat Step (6) 2*Loop times to get a sequence with 2*Loop accuracies. In this sequence, the absolute value of the difference between the average accuracy of any continuous Loop accuracies should be less than or equal to o. The value of o in the experiment is 0.005. Formula (9) shows the average accuracy.

$$\left| \frac{\sum_{i=n}^{\text{Loop}+n-1}\text{Accuracy}_i}{\text{Loop}} - \frac{\sum_{i=n}^{\text{Loop}}\text{Accuracy}_i}{\text{Loop}} \right| \le o, n \in (2, 3, \ldots, \text{Loop} + 1), \tag{8}$$

$$\overline{\text{Accuracy}} = \frac{\sum_{i=1}^{\text{Loop}}\text{Accuracy}_i}{\text{Loop}}. \tag{9}$$

2.4. Parallel ELM Algorithm Framework. Because of the randomness of ELM method, the classification accuracy of the proposed ELM algorithm framework will also vary randomly. In order to improve the accuracy and stability of ELM classifier, we proposed a parallel ELM algorithm framework.

The parallel ELM algorithm framework is a 2-hidden-layer artificial neural network. It is constructed in the following ways. Firstly, a series of ELM classifiers are constructed using the training set and then validate all the ELM classifiers by validation set and picks out all the ELM classifiers with the highest validation accuracy. We define these classifiers as the optimal ELM classifiers. It should be noted that these classifiers only have the optimal classification accuracy on the verification set, not necessarily on the test set. Finally, we combine all the optimal ELM classifiers into a 2-hidden-layer artificial neural network. It is the target classifier to be constructed by the parallel ELM algorithm framework. It also inherits the advantages of ELM classifier and has the ability of binary classification and multiclassification.

Steps of parallel ELM algorithm framework (Figure 4) are as follows:

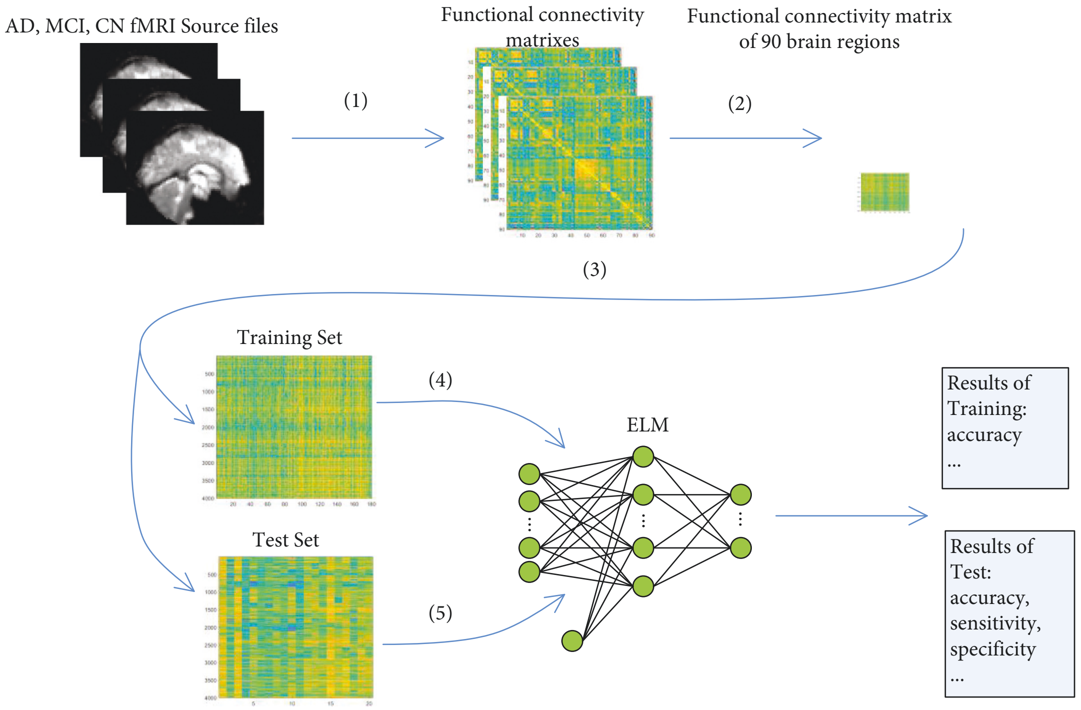

AD, MCI, CN fMRI Source files
(1)
Functional connectivity matrixes
(2)
Functional connectivity matrix of 90 brain regions
(3)
Training Set
(4)
Test Set
(5)
ELM
Results of Training: accuracy ...
Results of Test: accuracy, sensitivity, specificity ...

Figure 2: ELM algorithm framework.

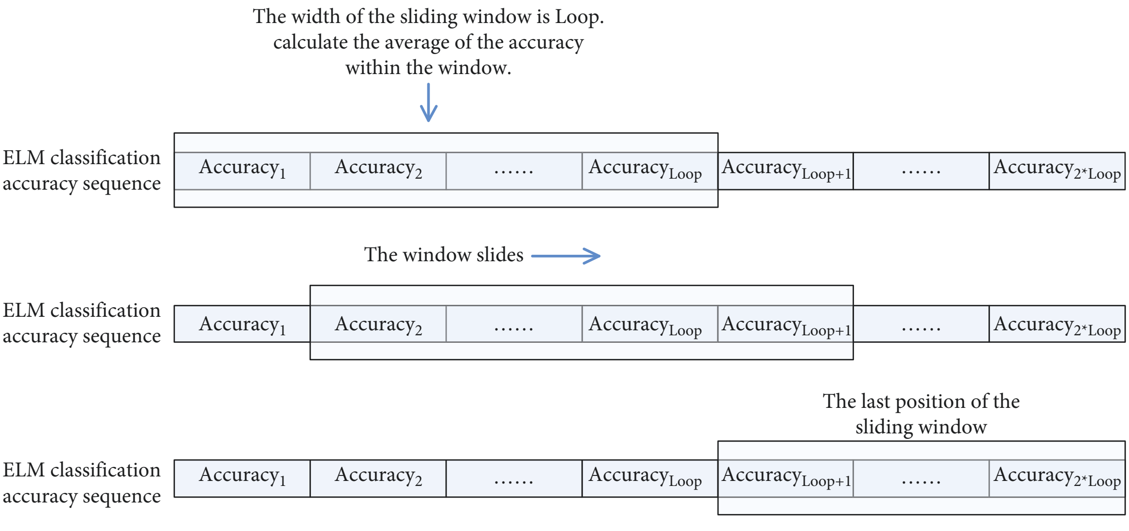

The width of the sliding window is Loop. calculate the average of the accuracy within the window.
ELM classification accuracy sequence
Accuracy$_1$
Accuracy$_2$
……
Accuracy$_{\text{Loop}}$
Accuracy$_{\text{Loop+1}}$
……
Accuracy$_{2*\text{Loop}}$
The window slides
ELM classification accuracy sequence
Accuracy$_1$
Accuracy$_2$
……
Accuracy$_{\text{Loop}}$
Accuracy$_{\text{Loop+1}}$
……
Accuracy$_{2*\text{Loop}}$
The last position of the sliding window
ELM classification accuracy sequence
Accuracy$_1$
Accuracy$_2$
……
Accuracy$_{\text{Loop}}$
Accuracy$_{\text{Loop+1}}$
……
Accuracy$_{2*\text{Loop}}$

Figure 3: Verify the convergence of the average accuracy by sliding window.

(1) Randomly divide the data set into a training set and a validation set in a specified proportion. The proportion of subjects in each category is the same in the training and test sets.

(2) Build ELM classifier with training set.

(3) Calculate the accuracy of ELM classifier with validation set.

(4) Repeat step (1) to (3) until the convergence of the average accuracy of ELM classification is verified. Record all the ELM classifiers with the highest verification accuracy, that is, the optimal ELM classifiers.

(5) Combine all the optimal ELM classifiers into a 2-hidden-layer neural network. All the output weights of the second hidden layer are 1. The formula in output node is formula (10), where T_i is the output of the optimal ELM classifier, n is the number of optimal ELM classifiers, and $\check{T}$ is the output of parallel ELM classifier.

$$\check{T} = \frac{\sum_{i=1}^{n} T_i}{n}. \tag{10}$$

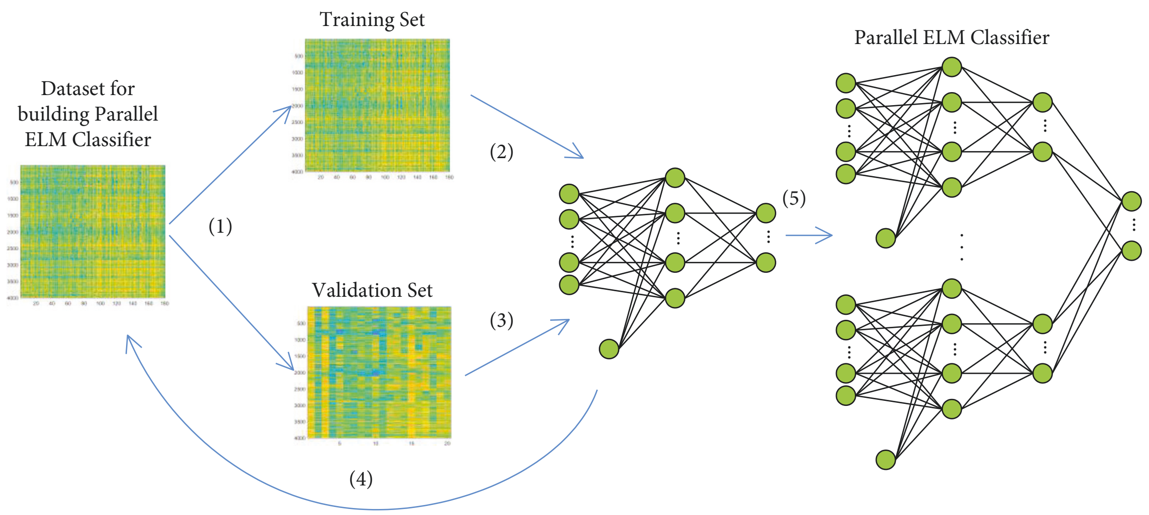


Figure 4: Steps of parallel ELM algorithm framework.

3. Experiments

All experiments in this paper are run on PC with Intel Core i7-8700 @ 3.20 GHz, NVIDIA GeForce RTX 2080 8 GB, 16 GB DDR4 3600 MHz, 250 GB SSD. Software environment is Windows 10 64 bit operating system. The tools used for development are MATLAB R2022b, DPARSF 5.0, and SPM 12.

We designed the experiments including data preprocessing, the performance of ELM, and the performance of parallel ELM methods. The workflow of the experiment is shown in Figure 5.

Data preprocessing: using tools such as DPARSF and SPM, the DICOM format files downloaded from ADNI are processed to brain functional connectivity matrixes.

The performance of ELM: it tests the ability of the ELM method to diagnose AD in high feature dimensions. This includes the effect of the number of hidden layer nodes on the classification accuracy, the generalization ability of the ELM method, the distribution of the accuracy of the ELM method, and the convergence speed of the average accuracy.

The performance of parallel ELM: it tests the classification ability of parallel ELM methods for AD, MCI, and CN.

3.1. Data Preprocessing. The data preprocessing tool selected in this experiment is Data Processing Assistant for RBF Advanced Edition (DPARSF 5.0 Advanced Edition; https://rfmri.org/DPARSF) and Statistical Parametric Mapping (SPM12; https://www.fil.ion.ucl.ac.uk/spm/software/spm12/). The fMRI image data obtained on ADNI were in the format of Digital Imaging and Communications in Medicine (DICOM). It converts DICOM format to Neuroimaging Informatics Technology Initiative (NIFTI) format. The first 10 time points of each subject were removed in order to remove the stage of unstable brain activity during familiarizing with the MRI scanner environment and noise at the beginning of data scanning. Slice Timing and Head Motion correction were performed for each subject, and EPI template was used for standardization. Band-pass filtering was used to obtain signals between 0.01 and 0.1 Hz. After processing, the bounding box of all subjects was [−90−126−72; 90 90 108], and the Voxel size was [3 3 3]. We obtain 300 functional connectivity matrices of 90*90 (100 AD, 100 MCI, and 100 CN). Due to the symmetry of these matrices, the full amount of features for each subject is 4005.

3.2. The Performance of ELM. We will test the ability of the ELM method to classify AD with CN in scenarios with high feature dimension and small sample size. We will also focus on the relevant characteristics of the method. The data set used for each of the following experiments consists of 100 AD subjects and 100 CN subjects. They are divided into a training set and a test at a certain ratio. The ratio of AD to CN was 1 : 1 in both the training and test sets.

(1) The relationship between the number of hidden-layer nodes and the accuracy of ELM classifiers. The number of hidden-layer nodes is an important hyper parameter in the ELM method. It has a very close relationship with the accuracy of the ELM classifier. In this experiment, the data set was randomly divided into a training set and a test set, with a ratio of 180 : 20. The number of hidden layer nodes is approximated according to formula (11) with reference to the feature dimensions of the subjects. The number of hidden layer nodes was 64, 125, 250, 500, 1000, 2000, 4000, 8000, 16000, 32000, and 64000. The experiment is repeated 1000 times for each number of hidden layer nodes. The ELM classifier is trained 10,000 times for the whole experiment. Each time the ELM is trained, the training and test sets are redivided randomly.

$$\text{Nodes} = 4000^{*}2^{n}, n \in [-6, 3], n \text{ is integer}. \quad (11)$$

(2) Generalization ability of ELM methods in high feature dimensionality scenarios. Generalization ability refers to the ability of machine learning

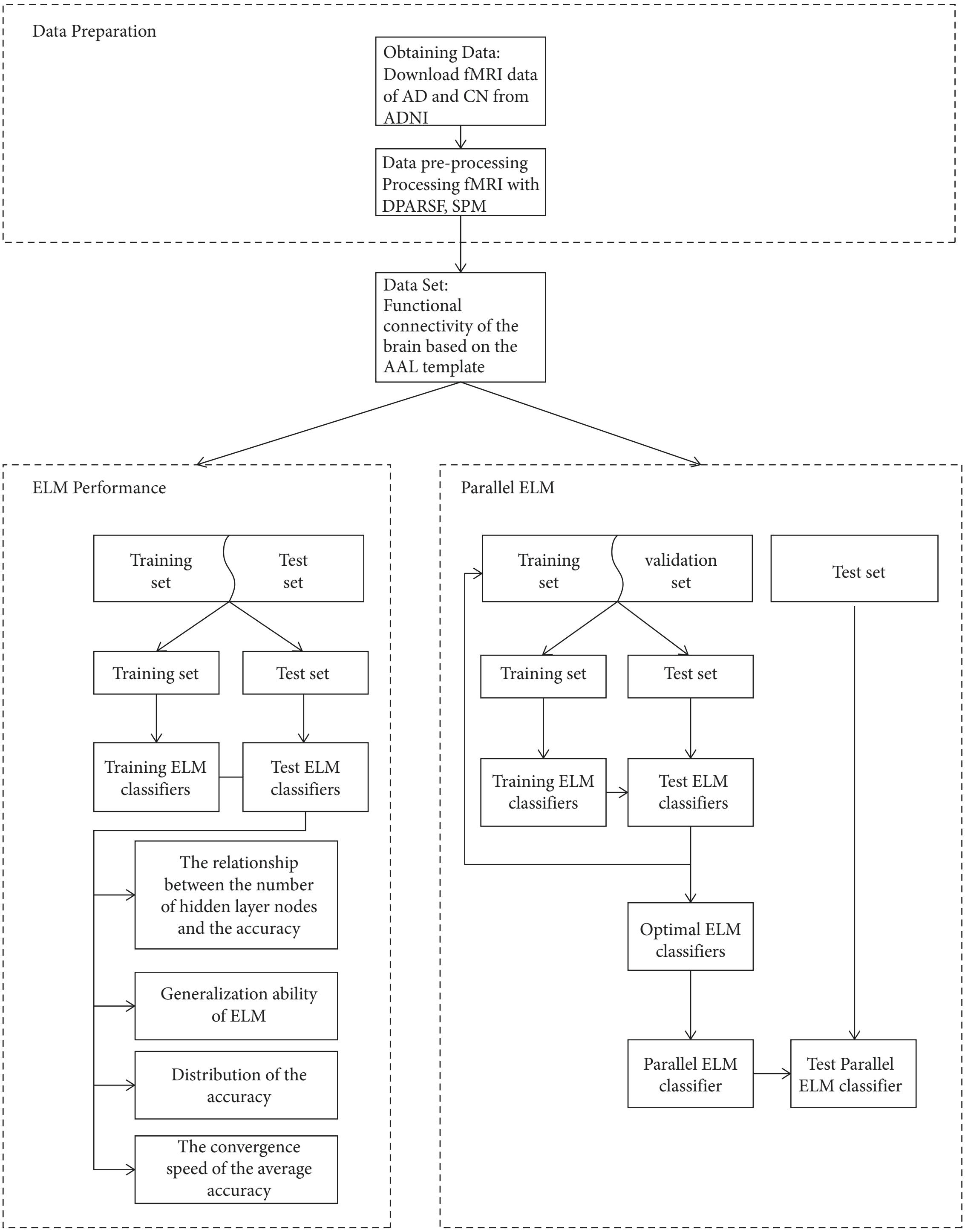

Data Preparation
Obtaining Data: Download fMRI data of AD and CN from ADNI
Data pre-processing Processing fMRI with DPARSF, SPM
Data Set: Functional connectivity of the brain based on the AAL template
ELM Performance
Training set
Test set
Training set
Test set
Training ELM classifiers
Test ELM classifiers
The relationship between the number of hidden layer nodes and the accuracy
Generalization ability of ELM
Distribution of the accuracy
The convergence speed of the average accuracy
Parallel ELM
Training set
validation set
Test set
Training set
Test set
Training ELM classifiers
Test ELM classifiers
Optimal ELM classifiers
Parallel ELM classifier
Test Parallel ELM classifier

Figure 5: Flow of the experiment.

algorithms to adapt to fresh samples. The number of samples used is limited. In this experiment, the training and test sets are divided in different ratios to describe approximately the generalization ability of the ELM method. In particular, the effect of using a small number of training sets is to construct ELM classifiers for prediction on a large number of test sets. In this experiment, the ratios of training and test sets are set as 1 : 9, 2 : 8, 3 : 7, 4 : 6, 5 : 5, 6 : 4, 7 : 3, 8 : 2, and 9 : 1. The number of ELM hidden layer nodes is set to 16000.1000 experiments are repeated for different training and test set ratios. The ELM classifier is trained 9000 times throughout the experiment. Each time the classifier is trained, the training and test sets are redivided.

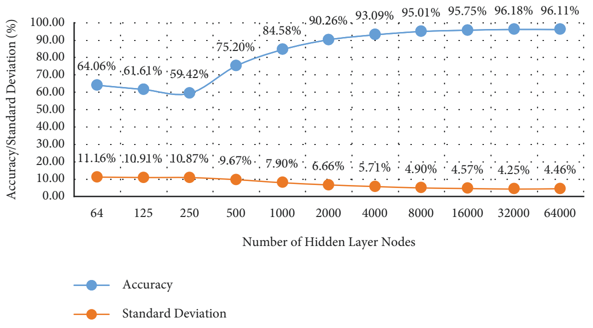


Figure 6: Relationship between the number of hidden layer nodes and the classification accuracy in AD and CN.

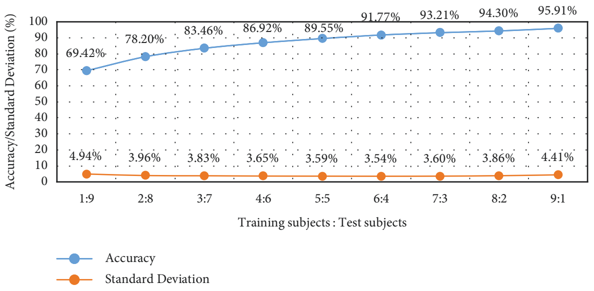


Figure 7: The relationship between the proportion of the data set and the classification accuracy of classification AD and CN.

(3) Distribution of the accuracy of the classifiers constructed by the ELM method. The input weights of the ELM method are randomly generated. In scenarios with high-dimensional features, there is also randomness in the accuracy of the ELM classifier. The number of hidden-layer nodes in the experiments is 16,000. The ratio of the training set to the test set is 9 : 1. There are two cases regarding the division of the training and test sets. Each time the ELM is trained, the training set and test set are divided randomly and the experiment is repeated 10,000 times to obtain the distribution of the ELM method accuracy. The training and test sets are divided randomly at one time, and the ELM classifier is trained and tested 10,000 times repeatedly to obtain the distribution of ELM method accuracy.

(4) The convergence speed of the average accuracy of ELM methods. The accuracy of ELM methods is random. Enough ELM classifiers are trained to describe the distribution of their accuracy rates. The convergence speed of the average accuracy rate is tested according to formula (8) and formula (9). The number of hidden-layer nodes is 16,000. The ratio of the number of subjects in the training set to the number of subjects in the test set is 9 : 1. Each time the ELM classifier is trained, the training and test sets are redivided randomly. The precision of the average accuracy o is set to 0.5%.

3.3. The Performance of Parallel ELM. We will test the classification ability of the parallel ELM method in AD vs. CN and MCI vs. CN. The data sets were divided into training set, validation set, and test set with a ratio of 8 : 1 : 1. The number of subjects in each pool is 1 : 1 for both categories. The test set is kept constant in one parallel ELM experiment. The training and validation sets are redivided randomly at each training of the ELM classifier. The number of hidden layer nodes is set to 16000. The parallel ELM method is validated according to 10-fold cross-validation for 10 times. The corresponding ROC curves are also plotted, and the AUC is calculated.

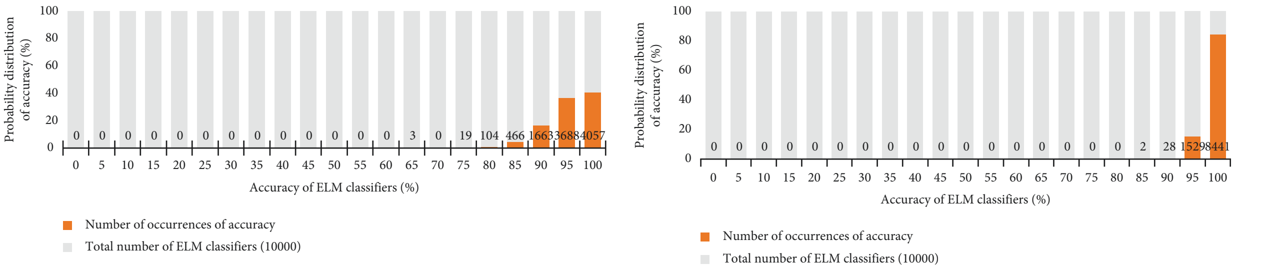
4. Results

In the experiment of AD and CN classification, the relationship between the number of ELM hidden-layer nodes and the accuracy is shown in Figure 6.

Each point in the figure corresponds to the number of hidden-layer nodes and to the results of training and testing the ELM classifier 1000 times. In total, 11,000 ELM classifiers are trained and tested. The vertical coordinate corresponds to the accuracy or standard deviation. The horizontal coordinate corresponds to the number of hidden layer nodes. The accuracy is decreasing as the number of hidden-layer nodes goes from 64 to 250. As the number of hidden-layer nodes becomes more than 250, the accuracy rate gradually increases, eventually reaching an accuracy rate of about 96.11%. It should be noted that when the number of hidden-layer nodes is 64 and 125, the average training accuracy is 83.94% and 96.14%, respectively. When the number of hidden layer nodes is 250 or more, the training accuracies are 100% in all cases. The variance of the mean accuracy decreases roughly with the number of hidden-layer nodes, from 11.16% to 4.46%. The rate of increase in accuracy is progressively slower, and the variance changes in a similar way. Since the number of hidden-layer nodes increases at an exponential rate, once the number of hidden-layer nodes reaches a certain level, more hidden-layer nodes have little meaningful effect on the improvement of the accuracy, and instead bring more computational power consumption. When the number of hidden-layer nodes is 16,000, the time to train and test an ELM classifier is about 0.2 seconds.

The generalization ability of the ELM method in the classification experiments of AD and CN is shown in Figure 7.

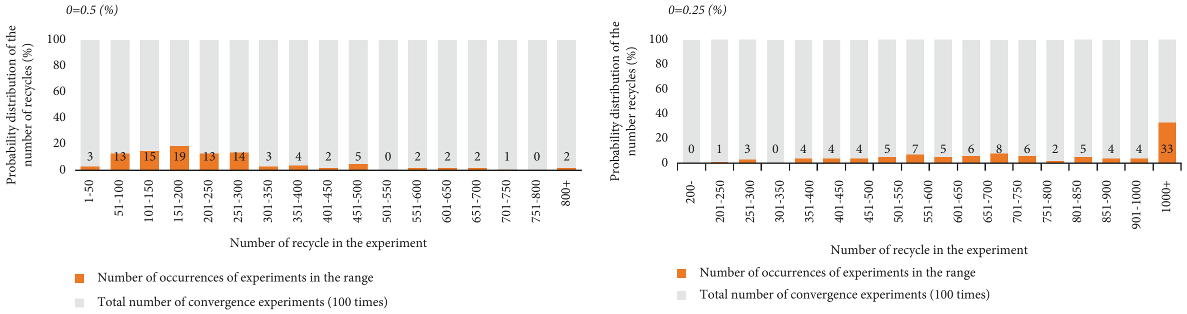
Generalization ability refers to the ability of machine learning methods to adapt to fresh samples. Due to the limitation of the number of subjects, we tried to build the ELM classifier with fewer training samples and predict a larger number of test samples. In this way, the generalization ability of the ELM method is described approximately. Each point in the figure corresponds to the training and testing of 1000 ELM classifiers. The corresponding accuracy is the average correct rate of the 1000 ELM classifiers as well as the standard deviation. A total of 9000 ELM classifiers are trained and tested. The training accuracy of all classifiers is 100%. When the ratio of the training set to the test set was 1 : 9, that is, the ELM classifier is trained with 20 samples to predict 180 samples, the accuracy was 69.42%. When the ratio of the training set to the test set was 1 : 9, that is, the ELM classifier is trained with 20 samples to predict 180 samples, the accuracy was 69.42%. When the ELM classifier



(a)

(b)

Figure 8: The distribution of accuracy. (a) Randomly divide the training and test sets when training each ELM. (b) The training set and test set are kept constant for training each ELM.



(a)

(b)

Figure 9: Distribution of the computation times. (a) $o = 0.5\%$; (b) $(o) = 0.25\%$.

is trained with 100 samples to predict another 100 samples, the accuracy reaches 89.55%. The accuracy of the ELM classifier increased monotonically as the proportion of the training set increased.

The distribution of accuracy in the AD and CN classification experiments is shown in Figure 8.

The horizontal coordinate corresponds to the accuracy of the ELM classifiers. The vertical coordinate corresponds to the number of classifiers with a certain accuracy as a proportion of the total number of classifiers (10,000). The number in the bar chart is the number of ELM classifiers with that accuracy. In the experiments corresponding to Figure 8(a), the training and test sets are randomly divided in each training ELM. The accuracies are distributed between 65% and 100%, and the accuracy of 9408 ELM classifiers are greater than or equal to 90%. In the experiments corresponding to Figure 8(b), the training and test sets are kept constant. Using the same methods, parameters, and data sets, we still found accuracies varying between 85% and 100%. Compared to the previous experiments, it has a more concentrated distribution.

In the classification experiments of AD and CN, the number of computations for the average accuracy of the ELM method to converge is shown in Figure (9). 100 experiments are performed with $o = 0.5\%$ and $o = 0.25\%$, respectively. o is the precision of the average accuracy and is used in formula (8). The horizontal coordinate is the number of times a convergence experiment needs to be computed. The vertical coordinate is the proportion that this result occurred to the total number of experiments (100). For example, there are 19 experiments where the number of computations is in the range "151–200." It accounted for 19% of the total number of experiments. Experiments with more than 800 computations occurred 2 times. It should be noted that, in order to avoid accidently ending the experiment early or for too long, we have set the number of computations in the program to be greater than 10 and less than 1000. When $o = 0.5\%$, the number of calculations is all less than 1000. The average number of calculations is 248.51. The number of computations resembles a Gaussian distribution. According to formula (8) and formula (9), the average number of computations required for the average accuracy to converge is 124.26. The number of computations increases substantially when $o = 0.25\%$. There are 33 experiments where the number of computations is greater than 1000.

The experiment results of the AD and CN classification using the parallel ELM method are shown in Table 1. The experiment performed a 10-fold cross-validation for 10 times. The training set, validation set, and test set are randomly generated for each 10-fold cross-validation. In a 10-fold cross-validation, 10 parallel ELM classifiers need to be trained. Test results are obtained on all subjects. Each row in Table 1 represents a 10-fold cross-validation. The first column is the serial number. The second column is the number of ELM classifiers trained in one 10-fold cross-validation. The third column shows the average test accuracy and standard deviation for all ELM classifiers. The fourth column is the worst test accuracy of these ELM classifiers. The fifth column is the number of optimal ELM classifiers selected in

Table 1: 10 times 10-fold cross-validation results for AD and CN classification.

No. of experiments	ELM classifiers			Optimal ELM classifiers			Parallel ELM classifiers		
	Amount	Mean Accuracy ± STD	Worst accuracy (%)	Amount	Mean Accuracy ± STD	Worst accuracy (%)	Amount	Mean accuracy ± STD	Worst accuracy (%)
1	2362	95.50% ± 4.91%	70.00	771	95.82% ± 4.88%	70.00	10	96.00% ± 6.15%	80.00
2	3024	95.60% ± 4.75%	75.00	1029	95.68% ± 4.62%	75.00	10	96.50% ± 3.37%	90.00
3	2090	94.97% ± 4.40%	75.00	662	95.38% ± 4.11%	80.00	10	97.00% ± 3.50%	90.00
4	3248	94.01% ± 5.15%	70.00	1052	94.14% ± 5.00%	80.00	10	96.50% ± 4.74%	85.00
5	2434	96.86% ± 4.36%	75.00	765	97.18% ± 3.96%	75.00	10	98.50% ± 2.42%	95.00
6	1748	95.05% ± 4.67%	80.00	545	95.41% ± 4.41%	80.00	10	97.50% ± 3.54%	90.00
7	2680	95.17% ± 4.31%	80.00	855	95.26% ± 4.30%	80.00	10	97.50% ± 3.54%	90.00
8	2110	94.45% ± 4.74%	75.00	702	94.60% ± 4.59%	75.00	10	96.00% ± 4.59%	90.00
9	1580	95.08% ± 4.45%	75.00	558	95.21% ± 4.17%	80.00	10	96.50% ± 2.42%	95.00
10	2558	94.42% ± 5.00%	75.00	846	94.77% ± 4.80%	80.00	10	96.50% ± 4.74%	90.00
Mean	**2383.4**	**95.11% ± 4.67%**	**75.00**	**778.5**	**95.35% ± 4.49%**	**77.50**	**10**	**96.85% ± 3.90%**	**89.50**

one 10-fold cross-validation. The sixth column is the average test accuracy and standard deviation of these optimal ELM classifiers. The seventh column is the worst test accuracy of the optimal ELM classifiers. The eighth column is the number of parallel ELM classifiers in one 10-fold cross-validation. Each fold yields one parallel ELM classifier. The ninth column is the average test accuracy and standard deviation of the parallel ELM classifiers. The tenth column is the worst accuracy of these classifications. For example, in the first 10-fold cross-validation, a total of 2362 ELM classifiers are trained. The average test accuracy of these ELM classifiers was 95.50% with a standard deviation of 4.91%, and the worst test accuracy is 70.00%. From the 2362 ELM classifiers, 771 optimal ELM classifiers are selected. The average test accuracy of these classifiers is 95.82%, with a standard deviation of 4.88% and a worst accuracy of 70%. 771 optimal classifiers are constructed for each of the 10 parallel ELM classifiers. The average accuracy of these classifiers was 96.00%, with a standard deviation of 6.15% and a worst test accuracy of 80%. It should be noted that the training accuracy of all classifiers was 100%.

In Table 1, it can be found that the optimal ELM classifier improved the accuracy by 0.25% on average and the worst accuracy by 2.5% compared to the ELM classifier. Parallel ELM classifiers improved accuracy by an average of 1.74% and worst accuracy by 14.50% compared to ELM classifiers, and the standard deviation of the average accuracy was reduced by 0.77%. The time to train and test an ELM classifier is about 0.2 seconds. The average time to train and test a parallel ELM classifier is less than 1 minute. The parallel ELM method inherits the speed of the ELM method, while offering higher accuracy and stability. Figure 10(a) shows the distribution of accuracy for the parallel ELM method. The data generated from the parallel ELM experiments allow to plot 10 ROC curves, as shown in Figure 11(a). The AUC of the 10 ROC curves ranged from 0.9799 to 0.9954. The mean value of the AUC is 0.9891. Still based on these data, we calculate a sensitivity of 97.1% and a specificity of 96.3% in the diagnosis of MCI.

The experiment results for MCI and CN classification are shown in Table 2. The format of Table 2 is the same as that of Table 1. We can find that 23515 ELM classifiers are trained and tested in the experiment. The average accuracy is 94.12%. The standard deviation is 5.35%. The average accuracy of the parallel ELM classifiers is 0.72% higher than that of ELM, the standard deviation is 0.28% lower, and the worst accuracy is 12% higher. It is worth noting that the average accuracy of the optimal ELM classifier is also lower than the average accuracy of the ELM classifier. The parallel classifiers constructed using these optimal ELM classifiers however have higher accuracy rates. Figure 10(b) shows the distribution of accuracy for parallel ELM methods.

The data generated from the parallel ELM experiments allowed 10 ROC curves to be plotted as shown in Figure 11(b). The AUC of the 10 ROC curves ranged from 0.9777 to 0.9960. The mean value of the AUC was 0.9888. Still based on these data, we calculate a sensitivity of 94.7% and a specificity of 95.3% in the diagnosis of MCI.

5. Discussion

From the results of the above experiments, we can find that the ELM method is effective. It can be used to classify AD with CN for high feature dimension and small sample size scenario. It has a high accuracy rate and better generalization ability. However, it also has some problems.

The classification accuracy of the ELM method is unstable. Figure 8(a) shows that in the experiments with AD and CN classification, the worst correct rate was 65% and the best was 100%. In Figure 8(b), even when using the same training set, test set, and ELM parameters, the accuracy of ELM still varied between 80% and 100%. In the process of ELM classifier training [26], a unique ELM classifier can be constructed as long as the input weights are randomly determined. This indicates that the input weights are good or bad in determining the accuracy of the classifier. The input weights determine the weights assigned to each feature. The higher the weight, the more useful it is, and the lower the weight, the less useful it is. We found in our experiments that classifiers with the same accuracy predicted different incorrect subjects. This indicates that different classifiers make use of

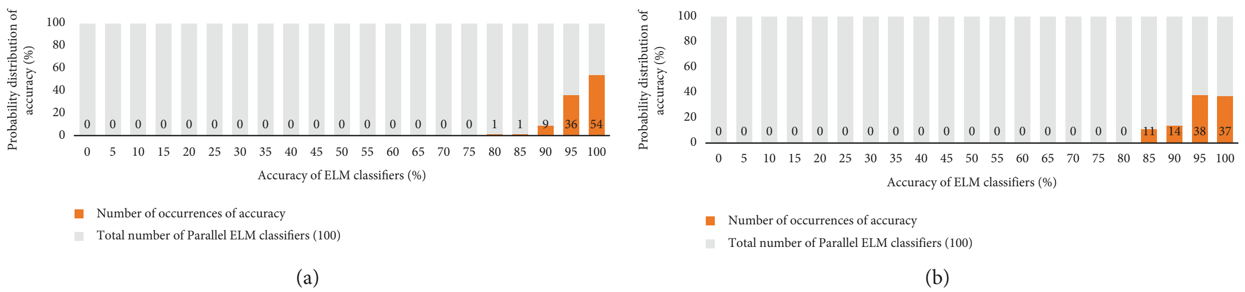


Figure 10: Distribution of accuracy of parallel ELM methods in the classification. (a) AD vs. CN; (b) MCI vs. CN.

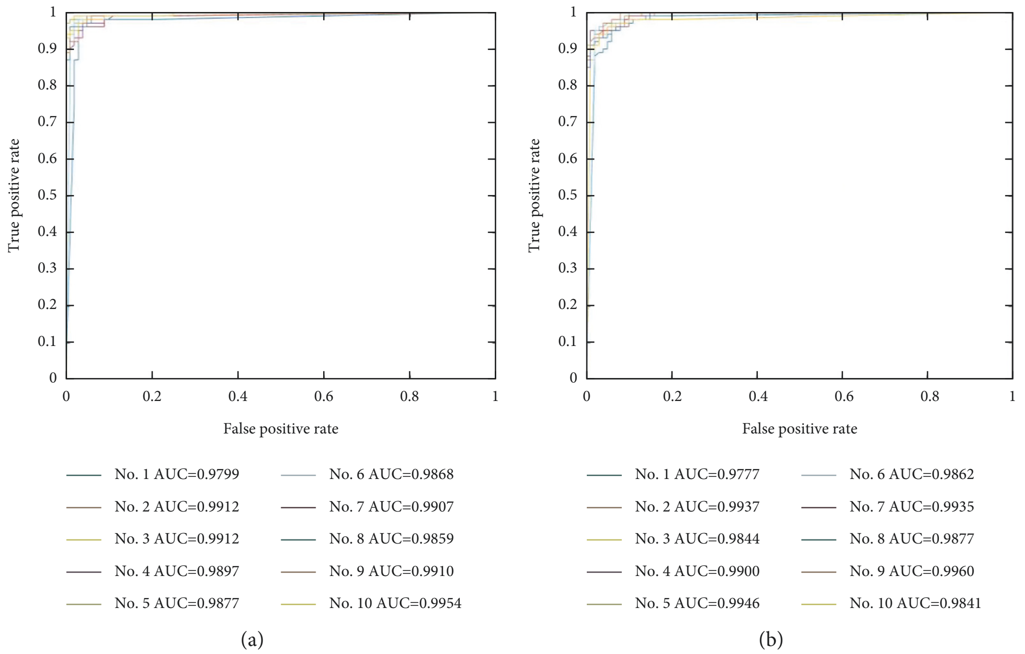


Figure 11: ROC and AUC for classification by the parallel ELM method. (a) AD vs. CN; (b) MCI vs. CN.

features differently. The ELM classifier's use of features is partial.

According to Table 1 and Table 2, it can be found that the accuracy of the optimal ELM classifier is lower than that of the parallel ELM classifier. The parallel ELM classifiers with higher accuracy are formed from these lower accuracy optimal ELM classifiers. This shows that the use of features by these optimal ELM classifiers is partial and varies from one to another. Therefore, the parallel ELM classifier can combine the learning ability of each ELM classifier to achieve higher accuracy.

Comparing the distribution of accuracy between ELM methods and parallel ELM methods is according to Figure 8(a) and Figure 10(a). 54% of the classifiers trained by the parallel ELM method achieved an accuracy of 100%, and the worst classification accuracy is 80%. 40.57% of the classifiers trained by the ELM method achieved an accuracy of 100%, and the worst classification accuracy is 65%. This indicates that the parallel classifier approach has higher accuracy and stability.

We found seven references on the diagnosis (classification) of AD and MCI in recent years for comparison, as shown in Table 3.

All of these references used subjects from ADNI. Five of the references used less than 100 subjects. 2 references used 354 subjects. We used 300 subjects. A sufficient number of subjects is more conducive to learning the patterns of the features, but may also negatively affect the accuracy. We used 300 subjects precisely to test the performance of the parallel ELM method better.

The references all use features of a network structure. Most used is the brain functional connectivity network (FC). Some references also use a combination of other features. We have adopted FC as features that are more comparable to these references.

The references all filter the features. The main purpose of this is usually to increase the speed and accuracy. The high dimensionality of the features has a significant impact on the classification. As far as we know, it is rare to find a scenario with high feature dimensionality where all features are used for classification. Feature filtering is usually based on a priori knowledge or presuppositions. This may result in some useful features being screened out. Our proposed method uses all features to avoid this problem.

Table 2: 10 times 10-fold cross-validation results for MCI and CN classification.

No. of experiments	ELM classifiers			Optimal ELM classifiers			Parallel ELM classifiers		
	Amount	Mean accuracy ± STD	Worst accuracy (%)	Amount	Mean Accuracy ± STD	Worst accuracy (%)	Amount	Mean accuracy ± STD	Worst accuracy (%)
1	2641	94.69% ± 4.97%	75.00	717	94.89% ± 5.01%	75.00	10	94.5% ± 4.97%	85.00
2	2164	91.23% ± 6.37%	70.00	631	91.09% ± 6.18%	75.00	10	95.00% ± 5.27%	85.00
3	1924	93.89% ± 5.92%	75.00	562	93.38% ± 5.97%	75.00	10	95.00% ± 5.27%	85.00
4	2096	94.60% ± 5.01%	70.00	634	94.58% ± 4.85%	75.00	10	94.50% ± 4.97%	85.00
5	1984	94.03% ± 5.31%	75.00	578	93.61% ± 5.62%	75.00	10	95.50% ± 4.97%	85.00
6	2016	95.17% ± 5.08%	75.00	556	94.62% ± 5.31%	75.00	10	96.00% ± 3.94%	90.00
7	3098	94.86% ± 4.78%	80.00	853	94.63% ± 4.89%	80.00	10	95.00% ± 4.74%	85.00
8	2710	92.05% ± 6.07%	70.00	855	91.50% ± 5.96%	80.00	10	94.50% ± 5.99%	85.00
9	2872	96.01% ± 4.80%	70.00	797	96.22% ± 4.90%	70.00	10	96.00% ± 4.59%	85.00
10	2010	94.64% ± 5.20%	75.00	595	94.44% ± 5.16%	75.00	10	94.50% ± 5.99%	85.00
Mean	**2351.5**	**94.12% ± 5.35%**	**73.50**	**677.8**	**93.90% ± 5.39%**	**75.50**	**10**	**95.05% ± 5.07%**	**85.50**

Table 3: Comparison of classification performances with references.

Reference	Data set AD : MCI: CN	Feature measures	Feature selection	Classifier	Accuracy (%)		AUC	
					AD vs. CN	MCI vs. CN	AD vs. CN	MCI vs. CN
[14]	-:25 : 25	Pearson correlation of regional cortical thickness	*t*-test; mRMR; SVM-RFE;	SVM	92.35	84	0.9744	0.9233
[15]	-:12 : 25	FC	*t*-test; SVM-RFE;	SVM	N/A	91.9	N/A	0.94
[27]	25:-:36	FC	Random neural network cluster	Elman neural network	92.31	N/A	N/A	N/A
[17]	34 : 31 : 31	ReHo; FC	SVM-RFE; LASSO; *t*-test	ELM	98.86	98.57	N/A	N/A
[18]	118 : 118 : 118	FC	Recurrent learning method; convolutional learning method;	ELM	N/A	N/A	0.913	0.824
[19]	118 : 118 : 118	FC	Select features by threshold	GNEA	N/A	N/A	0.813	0.703
[20]	31 : 31 : 31	FC, graph-embedding	SVM-RFE; LASSO; FSASL	LSVM RELM	90.63 93.86	97.8 98.91	N/A N/A	N/A N/A
Proposed method	**100:100: 100**	FC	None	Parallel ELM	**96.85**	**95.05**	**0.9891**	**0.9888**

SVM and ELM are two commonly used classifiers. Three references [14, 15, 20] used SVM as a classifier and four [17–20] used ELM as a classifier. One of them [19] used a deep neural network involve ELM. Our method is the most concise. This allows our method to inherit well the advantages of ELM, such as fast computation and high accuracy. Training a parallel ELM classifier takes less than 1 minute on average.

The data sets (including the number) used for these experiments differ. Their correct rates are just used as a reference for method comparison. Our method achieved a diagnostic accuracy of 96.85% for AD and 95.05% for MCI. The AUC values reached 0.9891 and 0.9888, respectively.

6. Conclusion

Based on the above work, we consider that (1) the proposed method framework is an effective AD/MCI classification method. (2) The proposed method framework is relatively concise and does not need other methods for feature screening. (3) The proposed method is suitable for small sample and high-dimensional data. It meets the requirements of medical image analysis. (4) The proposed method framework improves the accuracy and stability of ELM classifier. (5) The proposed method has a high speed.

6.1. Future Work.

ELM is suitable for scenarios with small sample amount and high feature dimensions. In this study, only brain functional connectivity was used as a feature measure. It is worth exploring the effects of other fMRI feature measures using the ELM approach. ELM has the ability to process the features of the whole brain, and it is worth using ELM method as a tool for fMRI feature analysis to search for biomarkers and explore the rules of human brain. The ELM method has the capability of multiclass classification and regressions, which makes it possible to design a method to fit the development process from CN to AD.

Data Availability

All fMRI data we used came from LONI's ADNI database, the ADNI2 project. The subjects were cognitively normal (CN), had mild cognitive impairment (MCI), and had Alzheimer's disease (AD). The participants' data download address is https://adni.loni.usc.edu/. The fMRI scan parameters we selected are as follows: Field Strength = 3.0 tesla; Flip Angle = 80.0 degree; Matrix X = 64.0 pixels; Matrix Y = 64.0 pixels; Mfg Model = Intera; Pixel Spacing X = 3.3125 mm; Pixel Spacing Y = 3.3125 mm; Pulse Sequence = GR; Slices = 6720.0; Slice Thickness = 3.312999963760376 mm; TE = 30 ms; and TR = 3000 ms;

Conflicts of Interest

The authors declared that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant nos. 31870979 and 61906117) and the Shanghai Sailing Program (Grant no. 19YF1419000). Thanks to the Alzheimer's Disease Neuroimaging Initiative (ADNI) for fMRI data sharing.

References

[1] M. J. Prince, "2018 Alzheimer's disease facts and figures," *Alzheimer's and Dementia*, vol. 14, no. 3, pp. 367–429, 2018.

[2] J. C. Morris, M. Storandt, J. P. Miller et al., "Mild cognitive impairment represents early-stage alzheimer disease," *Archives of Neurology*, vol. 58, no. 3, 2001.

[3] R. C. Petersen, R. O. Roberts, D. S. Knopman et al., "Mild cognitive impairment," *Archives of Neurology*, vol. 66, no. 12, 2009.

[4] A. Ward, S. Tardiff, C. Dye, and H. M. Arrighi, "Rate of conversion from prodromal Alzheimer's disease to Alzheimer's dementia: a systematic review of the literature," *Dementia and Geriatric Cognitive Disorders Extra*, vol. 3, no. 1, pp. 320–332, 2013.

[5] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, July 2004.

[6] D. Chyzhyk, A. Savio, and M. Graña, "Computer aided diagnosis of schizophrenia on resting state fMRI data by ensembles of ELM," *Neural Networks*, vol. 68, pp. 23–33, 2015.

[7] Z. Huang, Y. Yu, J. Gu, and H. Liu, "An efficient method for traffic sign recognition based on extreme learning machine," *IEEE Transactions on Cybernetics*, vol. 47, no. 4, pp. 920–933, 2017.

[8] S. Wang, C. Deng, W. Lin, G.-B. Huang, and B. Zhao, "NMF-based image quality assessment using extreme learning machine," *IEEE Transactions on Cybernetics*, vol. 47, no. 1, pp. 232–243, 2017.

[9] G.-B. Huang and L. Chen, "Convex incremental extreme learning machine," *Neurocomputing*, vol. 70, pp. 3056–3062, 2007.

[10] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2012.

[11] X. Liu, C. Gao, and P. Li, "A comparative analysis of support vector machines and extreme learning machines," *Neural Networks*, vol. 33, pp. 58–66, 2012.

[12] J. Kim, J. Kim, G.-J. Jang, and M. Lee, "Fast learning method for convolutional neural networks using extreme learning machine and its application to lane detection," *Neural Networks*, vol. 87, pp. 109–121, 2017.

[13] J. Tang, C. Deng, G.-B. Huang, and B. Zhao, "Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1174–1185, 2015.

[14] C.-Y. Wee, P.-T. Yap, and D. Shen, "Prediction of Alzheimer's disease and mild cognitive impairment using cortical morphological patterns," *Human Brain Mapping*, vol. 34, no. 12, pp. 3411–3425, 2013.

[15] B. Jie, D. Zhang, C.-Y. Wee, and D. Shen, "Topological graph kernel on multiple thresholded functional connectivity networks for mild cognitive impairment classification," *Human Brain Mapping*, vol. 35, no. 7, pp. 2876–2897, 2014.

[16] A. Khazaee, A. Ebrahimzadeh, and A. Babajani-Feremi, "Identifying patients with Alzheimer's disease using resting-state fMRI and graph theory," *Clinical Neurophysiology*, vol. 126, no. 11, pp. 2132–2141, 2015.

[17] D. T. Nguyen, S. Ryu, M. N. I. Qureshi et al., "Hybrid multivariate pattern analysis combined with extreme learning machine for Alzheimer's dementia diagnosis using multi-measure rs-fMRI spatial patterns," *PLoS One*, vol. 14, no. 2, Article ID e0212582, 2019.

[18] X. Bi, X. Zhao, H. Huang, D. Chen, and Y. Ma, "Functional brain network classification for Alzheimer's disease detection with deep features and extreme learning machine," *Cognitive Computation*, vol. 12, no. 3, pp. 513–527, 2019.

[19] X. Bi, Z. Liu, Y. He, X. Zhao, Y. Sun, and H. Liu, "GNEA: a graph neural network with ELM aggregator for brain network classification," *Complexity*, vol. 2020, Article ID 8813738, 11 pages, 2020.

[20] R. K. Lama and G.-R. Kwon, "Diagnosis of Alzheimer's disease using brain network," *Frontiers in Neuroscience*, vol. 15, Article ID 605115, 2021.

[21] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: uses and interpretations," *NeuroImage*, vol. 52, no. 3, pp. 1059–1069, 2010.

[22] N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou et al., "Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain," *NeuroImage*, vol. 15, no. 1, pp. 273–289, 2002.

[23] G. B. Huang and H. A. Babri, "Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions," *IEEE Transactions on Neural Networks*, vol. 9, no. 1, pp. 224–229, 1998.

[24] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning And Cybernetics*, vol. 2, no. 2, pp. 107–122, 2011.

[25] G.-B. Huang and L. Chen, “Enhanced random search based incremental extreme learning machine,” *Neurocomputing*, vol. 71, pp. 3460–3468, 2008.

[26] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, “Extreme learning machine: theory and applications,” *Neurocomputing*, vol. 70, pp. 489–501, 2006.

[27] X.-A. Bi, Q. Jiang, Q. Sun, Q. Shu, and Y. Liu, “Analysis of Alzheimer’s disease based on the random neural network cluster in fMRI,” *Frontiers in Neuroinformatics*, vol. 12, 2018.