



RFSEN-ELM: SELECTIVE ENSEMBLE OF EXTREME LEARNING MACHINES USING ROTATION FOREST FOR IMAGE CLASSIFICATION

Z. Zhou*, J. Chen*, Y. Song*, Z. Zhu[†], X. Liu[†]

Abstract: learning machine (ELM), as a new learning mechanism for single hidden layer feedforward neural networks (SLFNs), has shown its advantages, such as fast computation speed and good generalization performance. However, the weak robustness of ELM is an unavoidable defect for image classification. To address the problem, we propose a novel ensemble method which combines rotation forest and selective ensemble model in this paper. Firstly, ELM and rotation forest are integrated to construct an ensemble classifier (RF-ELM), which combines the advantages of both rotation forest and ELM. The purpose of rotation forest here is to enhance the diversity of each base classifier which can improve the performance generalization. Then several ELMs are removed from the ensemble pool by using genetic algorithm (GA) based selective ensemble model to further enhance the robustness. Finally, the remaining ELMs are grouped as a selected ensemble classifier (RFSEN-ELM) for image classification. The performance is analyzed and compared with several existing methods on benchmark datasets and the experimental results demonstrate that the proposed algorithm substantially improves the accuracy and robustness of classification at an acceptable level of training cost.

Key words: *neural network, extreme learning machine, rotation forest, genetic algorithm, selective ensemble model, image classification*

Received: June 29, 2016

DOI: 10.14311/NNW.2017.27.026

Revised and accepted: October 2, 2017

1. Introduction

Image classification is an important research area in computer vision and machine learning. Several image classification approaches have been proposed and shown good performance [1, 16, 17]. However, significant challenges still exist in current image classification tasks. For instance, due to the high complexity of some image

*Zhiyu Zhou; Ji Chen; Yacheng Song – Corresponding author; Zhejiang Sci-Tech University, Faculty of Information and Electronics, Hangzhou, Zhejiang, 310018, China, E-mail: 13065713897@163.com, 470494530@qq.com, syc906562@163.com

[†]Zefei Zhu; Xiangqi Liu; Hangzhou Dianzi University, Faculty of Mechanical Engineering, Hangzhou, Zhejiang, 310018, China, E-mail: zzf.3691@163.com, lxiangqi@hdu.edu.cn

datasets and the high dimensionality of the feature representation of images, support vector machine (SVM) and least-square support vector machine (LSSVM) [27], which are widely used for image classification [15], need to take a time-consuming and complex training step to find the optimal parameters. It is still a critical problem to enhance the classification performance while keeping the training time at an acceptable level.

In recent years, Huang et al. proposed a novel machine learning algorithm called extreme learning machine (ELM) [12,13], which has a fast computation speed and good generalization performance. Compared with other widely used machine learning algorithms, ELM has some outstanding advantages. The randomly selected input weights and biases for network training can substantially simplify the time-consuming training step and reduce the computational complexity. No extra parameters need to be tuned except the predefined network architecture, and all the parameters can be generated without training data. Moreover, it can handle the multi-class classification problem naturally without using the one-versus-all or one-versus-one strategy [14]. A number of classification applications based on ELM have been recently proposed, such as hyperspectral image classification [18], protein sequence classification [28], landmark recognition [2], gene expression data classification [21], brain detection [22,23,29], etc.

Nevertheless, there are still some drawbacks of ELM. For instance, the randomly assigned weights will widen the range of the classification accuracies in different trials with the same parameters. Some researchers suggested that the shortcomings of ELM could be overcome by using ensemble strategy [9]. For example, Cao et al. [3] proposed an algorithm called V-ELM, which trains multiple ELMs and combines their results by majority voting. To further improve the diversity of base classifiers in the ensemble, Samat et al. [25] combined the ELM with the commonly used ensemble strategy bagging approach, and proposed an algorithm called Bagging based ELM (B-ELM). Stosic et al. [26] introduced a voting based q -generalized extreme learning machine (V-QELM) which trains individual ELMs with q -Gaussian activation functions using different values of the parameter q . Han and He et al. [7,8] proposed two algorithms called RMSE-ELM and LARSEN-ELM. RMSE-ELM was a recursive model based ELM combined with selective ensemble for robustness improvement. While LARSEN-ELM model was a selective ensemble of ELM using least angle regression for blended data. Liu and He et al. [19] proposed a robust online sequential extreme learning machine (ROS-ELM). In [20], particle swarm optimization based selective ensemble of online sequential extreme learning machine was proposed. These models improved the robustness while keeping a relatively high speed for ELM.

Deep learning is a new approach in machine learning research. The motivation is to establish a neural network by simulating human brain for analysis and learning. It mimics the mechanism of human brain to explain the data. By combining low-level features obtained from the distributed characteristics of the data, more abstract high-level attribute classes or features are created. The concept of deep learning was presented by Hinton et al. [10] in 2006. The core idea of deep learning includes: (1) Unsupervised learning is used to pre-train each network layer. (2) Unsupervised learning only trains one layer each time, and the training results are used as the inputs of the higher layer. (3) Supervised learning is used to turn

all layers. The advantages of deep learning are as follows: (1) It outperforms all other machine learning methods when the datasets are large enough. (2) It has a good property of transfer learning, which means that a trained model can be used for another different but related problem with some simple refinement. The deep learning also has its disadvantages as follows: (1) Deep learning training is time consuming. And the model verification and testing of deep learning is difficult. (2) In some deep networks, GPU is required both in training and on-line deployment. Meanwhile, deep learning is extremely sensitive to the size of the network structure. In order to obtain high precision, it is necessary to manually adjust a large number of parameters, which requires massive computational resources. Hence the learning speed is extremely low, and it is easy to be overfitting. On the contrary, ELM is not sensitive to the size of the network structure. It has a fast learning speed, and emphasizes dynamic balance in learning accuracy and precision. It also can be used for small data set to solve complex applications.

Inspired by the previous trend of ELM, we propose a novel method for image classification which can achieve high accuracy with a cost efficient training step and improve classification robustness. So a new method called RFSEN-ELM is proposed in this paper. The method is formed by combination of rotation forest [24] and genetic algorithm (GA) [6] based selective ensemble model [30]. Rotation forest, as a powerful ensemble classifier generation method, is used here to train a set of base classifiers with high diversity. As suggested by the selective ensemble model, we utilize fewer but better base classifiers to assemble the strong ensemble classifier. Note that GA is used here as an optimization method to reduce the negative impact of un-optimal classifier in selective ensemble model. In our modification, the algorithm consists of two major procedures: Firstly, we use rotation forest to train a set of independent ELMs (RF-ELM) with high diversity. Then an optimal subset of the ensemble pool is selected by using GA based selective ensemble model. Finally, we use the remaining ELMs to make up the strong ensemble classifier (RFSEN-ELM).

The rest of this paper is organized as follows. In Section 2, we briefly introduce ELM and genetic algorithm. Our method is detailed in Section 3. In Section 4, experiments of empirical comparisons with ELM, B-ELM, V-ELM, SVM and LSSVM are exhibited. Section 5 draws a conclusion.

2. Preliminaries

2.1 Extreme learning machine

Extreme learning machine (ELM) is a learning mechanism for single hidden layer feedforward networks (SLFNs), which substantially increases the learning speed by randomly selecting the weights and biases of hidden nodes. Given a training set consisting of N samples $(\mathbf{x}_j, \mathbf{y}_j)$, where $\mathbf{x}_j = [x_{j1}, x_{j2}, \dots, x_{jn}] \in \mathbb{R}^n$, $\mathbf{y}_j = [y_{j1}, y_{j2}, \dots, y_{jm}] \in \mathbb{R}^m$, the mathematical expression of standard SLFNs with L hidden nodes is described as follows:

$$\sum_{i=1}^L \beta_i g_i(\mathbf{x}_j) = \sum_{i=1}^L \beta_i g(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \mathbf{o}_j, j = 1, 2, \dots, N \quad (1)$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ is a n -dimensional weight vector connecting i -th hidden node and input neurons, $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ is a m -dimensional weight vector connecting i -th hidden node and output neurons, b_i is the threshold of the i -th hidden node. Note that Eq. (1) can be rewritten as

$$\mathbf{H}\beta = \mathbf{O}, \tag{2}$$

where

$$\mathbf{H} = \begin{bmatrix} g_1(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g_L(\mathbf{w}_L \cdot \mathbf{x}_1 + b_L) \\ \vdots & \ddots & \vdots \\ g_N(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g_L(\mathbf{w}_L \cdot \mathbf{x}_N + b_L) \end{bmatrix}_{N \times L}, \tag{3}$$

$$\beta = [\beta_1, \beta_2, \dots, \beta_L]^T \in \mathbb{R}^{L \times m} \text{ and } \mathbf{O} = [o_1, o_2, o_3, \dots, o_N]^T \in \mathbb{R}^{N \times m}, \tag{4}$$

where \mathbf{H} is the hidden layer output matrix of the network, \mathbf{O} is the output matrix and β is the weight matrix. Subsequently, optimal weights could be found by minimizing the least-square error

$$\hat{\beta} = \min \|\mathbf{H}\beta - \mathbf{Y}\|, \tag{5}$$

where $\mathbf{Y} = [y_1, y_2, y_3, \dots, y_N]^T$ is the target matrix. In the case when the weights and biases of hidden layer can be randomly selected and \mathbf{H} is a square and invertible matrix, weights β of output layer can be solved by finding the minimum norm least-squares function of the linear model.

$$\hat{\beta} = \mathbf{H}^\dagger \mathbf{Y}, \tag{6}$$

where \mathbf{H}^\dagger is Moore–Penrose generalized inverse of matrix \mathbf{H} . The ELM algorithm for classification tasks can be summarized in Algorithm 1.

Algorithm 1 : ELM.

Given:

D : training set $D = \{(\mathbf{x}_i, \mathbf{y}_i) | \mathbf{x}_i \in \mathbb{R}^n, \mathbf{y}_i \in \mathbb{R}^m, i = 1, 2, 3, \dots, N\}$; $g(x)$: activation function; L : the number of hidden nodes

Training:

Random select the weights \mathbf{w}_i and biases b_i
 Calculate the hidden layer output matrix \mathbf{H}
 Calculate the output weights $\beta : \beta = \mathbf{H}^\dagger \mathbf{Y}$

Classification:

Given new unlabeled test instance $\mathbf{x}^{\text{test}} \in \mathbb{R}^n$
 Calculate the hidden layer output matrix $\mathbf{H}^T = g(\mathbf{w}_i \cdot \mathbf{x}^{\text{test}} + b_i), i = 1, 2, \dots, N$

Calculate the class label of the instance $\mathbf{Y}^T = \mathbf{H}^T \cdot \beta$

2.2 Genetic algorithm

Genetic algorithm (GA) is a heuristic search algorithm to solve the optimization problem. Given a specific problem, the input of GA should be an encoded potential

solution of the problem. Generally, we apply a metric called fitness function to quantitatively evaluate each individual. The fitness is the standard to judge the individuals, and we can use it to evolve the individuals in order to estimate them.

The implementation of GA is actually similar to the evolution in nature, and the algorithm iteratively updates a set of randomly generated individuals called population. All of the individuals in the population are evaluated according to the predefined fitness function. The fittest members are kept and allowed to reproduce in the current population, and then the genetic operators such as crossover and mutation are used for generating new offspring individuals. These new individuals then go on to the next generation forming a new pool of candidate solutions. The procedure of generating new individuals continues until the optimal solution of the problem can be found.

3. Selective ensemble of ELM using rotation forest

3.1 Rotation forest based extreme learning machine

Rotation forest is a powerful ensemble classifier generation method. The success of rotation forest relies on the randomized rotation matrix, which can highly improve the diversity of the base classifiers in the ensemble. Decision tree is always used as the base classifiers of the original rotation forest method due to the sensitivity to rotation of feature axes, but the complexity and overfitting problems of decision trees often disturb the generalization performance of classification. Thus, we introduce an ensemble method in this section that applies the advantages of both ELM and rotation forest (RF-ELM). In the training phase, we first compute the randomized rotation matrix of the training data, which is created by feature extraction and transformation of training data.

Given n -dimensional input vectors $\mathbf{x}_j = [x_1, x_2, x_3, \dots, x_n] \in \mathbb{R}^n$ and make up all the set of N samples into a matrix as follows:

$$\mathbf{X} = \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nn} \end{pmatrix}, \quad (7)$$

where x_{ij} is the value of the j -th dimension of the i -th input vector. The base classifiers in the ensemble are denoted by D_1, D_2, \dots, D_L and the feature set is denoted by F . To build up the training data for each base classifier D_i , the main steps are as follows:

Step 1: Randomly divide the feature set F into K subsets where each feature subset $F_i, i = 1, 2, \dots, K$ has $M = n/K$ features.

Step 2: For each feature subset $F_i, i = 1, 2, \dots, K$, extract the corresponding data S_i of training set. Run PCA using only the M features in and the corresponding data S_i . Obtain the coefficients of the principal components $a_{K1}, a_{K2}, \dots, a_{KM'}$ ($M' \leq M$).

Step 3: Organize the obtained K coefficients vectors in the matrix $\mathbf{R} = [R_1, R_2, \dots, R_K]^T$.

$$\mathbf{R} = \begin{bmatrix} a_{11}, \dots, a_{1M'} & 0 & \dots & 0 \\ 0 & a_{21}, \dots, a_{2M'} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{K1}, \dots, a_{KM'} \end{bmatrix}. \quad (8)$$

Step 4: To calculate the training set for classifier D_i , we need to rearrange the matrix \mathbf{R} in order to correspond to the original features. The new training data for D_i is $\mathbf{X}^{\text{new}} = \mathbf{X}\mathbf{R}^*$, where \mathbf{R}^* is the rearranged rotation matrix.

When the construction of randomized rotation matrix for each base classifier is completed, then the new training data of each base classifier could be calculated based on the randomized rotation matrix. Finally, multiple ELMs are trained on the rotated training data to make up the ensemble. The detailed procedure of RF-ELM is shown in Algorithm 2.

Algorithm 2 : RF-ELM.

Given:

\mathbf{X} : training data set ($N \times n$ matrix); \mathbf{Y} : the labels of the training data set ($N \times 1$ matrix); Q : the number of classifiers in the ensemble; K : the number of feature subsets

Training:

for $q = 1$ **to** Q **do**

 Divide F (the feature variable set) randomly into K subsets $F_{q,j}$, $j = 1, 2, \dots, K$

for each $F_{q,j}$, $j = 1, 2, \dots, K$ **do**

 Let $\mathbf{X}_{q,j}$ be the subset of \mathbf{X} corresponding to feature subset $F_{q,j}$

 Select a bootstrap sample from $\mathbf{X}_{q,j}$ of size 75%, denote the new set by $\mathbf{X}'_{q,j}$

 Apply PCA on $\mathbf{X}'_{q,j}$ to obtain the coefficients in a matrix $\mathbf{C}_{q,j}$

end for

 Arrange the $\mathbf{C}_{q,j}$ for $j = 1, 2, \dots, K$ in a rotation matrix \mathbf{R}

 Compose \mathbf{R}^* by rearranging the columns of \mathbf{R} so as to match the order of features

 Train ELM and compute β_q using the new training set $\mathbf{X}^{\text{new}} = \mathbf{X}\mathbf{R}^*$, according to the training step of Algorithm 1.

end for

Classification:

Given new unlabeled test instance $\mathbf{x}^{\text{test}} \in \mathbb{R}^n$

for $q = 1$ **to** Q **do**

 Obtain output matrix $\mathbf{Y}_{\text{test}}(q) = \mathbf{H}_{\text{test}}(q)\beta_q$

end for

Sum up all Q output matrices $\mathbf{Y}_{\text{test}} = \sum_{q=1}^Q \mathbf{Y}_{\text{test}}(q)$

Calculate the final class label by majority voting

3.2 Genetic algorithm based selective ensemble

Most conventional ensemble learning algorithms generally apply all of the base classifiers to build the final ensemble classifier. It is certain that these methods directly promote the classification performance, but compared with single classifier, the learning speed of ensemble algorithm has significantly decreased and the space complexity of ensemble classifier has substantially increased. Considering the relationship between the generalization performance of ensemble classifier and the correlation of each base classifier, it may be better to ensemble many instead of all of the base classifiers based on the selective ensemble model.

In selective ensemble, we need to assign a weight for each base classifier, which could represent their importance in the ensemble, and then we can select the optimal subset of the ensemble pool based on these weights. In other words, when we randomly assign the weight vector $\mathbf{w} = [w_1, w_2, \dots, w_n]^T$ to the ensemble, the main task in selective ensemble is to find an optimal weight vector $\mathbf{w}^* = [w_1, w_2, \dots, w_k]^T$ from a pool of different weight vectors, where k is the number of the selected individuals. To evolve the weights which reflect the importance of each individual in an ensemble to some extent, we consider the procedure of finding the optimal weight vector as an optimization problem for genetic algorithm, thus the weight vector $\mathbf{w} = [w_1, w_2, \dots, w_n]^T$ could be regarded as a member of the initial population, where w_i should satisfy Eq. (9) and Eq. (10).

$$0 \leq w_i \leq 1, \quad (9)$$

$$\sum_{i=1}^N w_i = 1. \quad (10)$$

When the initialization of the population of GA is completed, the predefined threshold λ is used to select the components of weight vector whose weight is bigger than λ . Because the weight vectors are randomly generated, the selected individuals are independent of each other in the population. But it is worth noting that we must normalize the weight vector for satisfying Eq. (10) after evolution due to the sum of \mathbf{w}_i . The normalization is described as

$$w'_i = \frac{w_i}{\sum_{i=1}^n w_i}. \quad (11)$$

In order to estimate the fitness of each individual of the population, we calculate the classification error E_w of each individual in the population on a validation set. The validation set is generated by a commonly used approach called bagging to avoid overfitting. The classification error E_w could reflect the quality of weight vector \mathbf{w} . It is clear that the smaller E_w is, the better \mathbf{w} is, so the fitness function is described as follows:

$$f(\mathbf{w}) = \frac{1}{E_w} = \frac{N}{\sum_{i=1}^N I(y_i \neq f(x_i))}, \quad (12)$$

$$I = \begin{cases} 1 & y \neq f(x), \\ 0 & y = f(x), \end{cases} \quad (13)$$

where N is the number of training samples in the validation set. I is the indicator function, $f(x)$ is the output of the classifier and y is the target output. When the evolution of population is completed, we can select the optimal subset based on the predefined threshold λ in the evolved optimal weight vector \mathbf{w}^* to make up the ensemble.

3.3 RFSEN-ELM

In order to overcome the drawbacks of ELM, we propose an ensemble based ELM algorithm to improve the classification performance. However, the conventional ensemble learning algorithms are not well suited for image classification, because the poor diversity of base classifiers undermines the generalization performance and several un-optimal classifiers in the ensemble disturb the robustness performance. Hence, we propose a novel method called RFSEN-ELM in this paper to improve the generalization performance as well as the performance robustness. The algorithm contains two major procedures. We first train multiple ELMs using rotation forest which can highly increase the diversity of the base classifiers, and then we apply the GA based selective ensemble model to make up an optimal ensemble classifier.

We show the specific algorithm procedure in Algorithm 3. In the training phase, for each base classifier, randomized rotation matrix is constructed. This step can highly improve the diversity, and the details are described in Section 3.1. The rotated training data will be used to train multiple ELMs with high diversity. To further improve the performance robustness, the genetic algorithm based selective ensemble model is utilized to select an optimal subset of the ensemble pool. Initially, we randomly assigned the weights vector $\mathbf{w} = [w_1, w_2, \dots, w_n]^T$ for each individual in the population, and then the fitness is computed to select the fittest individuals in the population, we can obtain an evolved best weight vector $\mathbf{w}^* = [w_1, w_2, \dots, w_k]^T$, where k is the number of the selected individuals based on the predefined threshold λ . Finally, the ensemble output comes from the selected base classifiers by majority voting.

4. Experiments and discussions

In this section, the classification performance of RFSEN-ELM is compared with several existing methods, and the criteria used here to evaluate the classifiers are accuracy, robustness, and computational complexity of classification. The comparison is implemented on two benchmark image datasets and the compared algorithms including the Voting based ELM (V-ELM) [3], Bagging based ELM (B-ELM) [25], and SVM [4]. V-ELM trains multiple ELMs with random parameters and combines their results by majority voting, B-ELM trains independent ELMs by using a standard bagging approach. All experiments were implemented in Matlab2012b running on a PC with 2.5 GHz CPU and 4 GB RAM.

4.1 Datasets

The 15 Scenes: This is a dataset of fifteen natural scene categories that expands on the thirteen category dataset released in [16]. The database contains 4485 scene

Algorithm 3 : RFSEN-ELM.**Given:**

\mathbf{X} : training data set ($N \times n$ matrix); \mathbf{Y} : the labels of the training data set ($N \times 1$ matrix); Q : the number of classifiers in the ensemble; K : the number of feature subsets; λ : the predefined threshold; $f(x)$: the fitness function

Training:

Train Q ELMs and remain the model parameters $\beta_q, q = 1, \dots, Q$ of each classifier according to the training step of Algorithm 2

Initialize population and randomly assign the weight vector w_q to each individual
for $k = 1$ **to** K **do**

 Compute the fitness of each individual on the validation set based on the fitness function $f(\mathbf{w}_q) = 1/E_{\mathbf{w}_q}, q = 1, \dots, Q$

 Select the population for the next generation based on the fitness of each individual

 Breed new individuals through crossover, mutation as new population

end for

Obtain the evolved best weight vector \mathbf{w}^*

Remain the individuals whose weight is larger than λ . Denote the remaining individuals by Q^*

Classification:

Given new unlabeled test instance $\mathbf{x}^{\text{test}} \in \mathbb{R}^n$

for $q = 1$ **to** Q^* **do**

 Obtain output matrix $\mathbf{Y}_{\text{test}}(q) = \mathbf{H}_{\text{test}}(q)\beta_q$

end for

Sum up all Q^* output matrices $\mathbf{Y}_{\text{test}} = \sum_{q=1}^{Q^*} \mathbf{Y}_{\text{test}}(q)$

Calculate the final class label by majority voting

images of 15 categories, including kitchen, bedroom, factories, streets, etc. Each category has 200 to 400 images, and the average size of the image is 300×250 pixels. Fig. 1(a) shows the selected sample images.

UIUC-sports: This dataset contains 8 sports event categories and a total number of 1579 images [17]. These images are taken from events belonging to rowing, badminton, polo, bocce, snowboarding, croquet, sailing and rock climbing. The number of images in each category varies from 137 to 250. Fig. 1(b) shows the selected sample images.

4.2 Experimental settings

In order to obtain the feature representation vector of images for the classifiers, we extract the global SIFT descriptor [5] of each image in the dataset, the SIFT descriptor is a local feature descriptor which is widely used in image classification, then we perform K -means clustering of the extracted features to form a visual vocabulary, the vocabulary size is set to 300 in this paper. Finally, we implement the Spatial Pyramid Matching (SPM) model [15] to form the representation vector of the image by aggregating the words of visual vocabulary into the histogram,

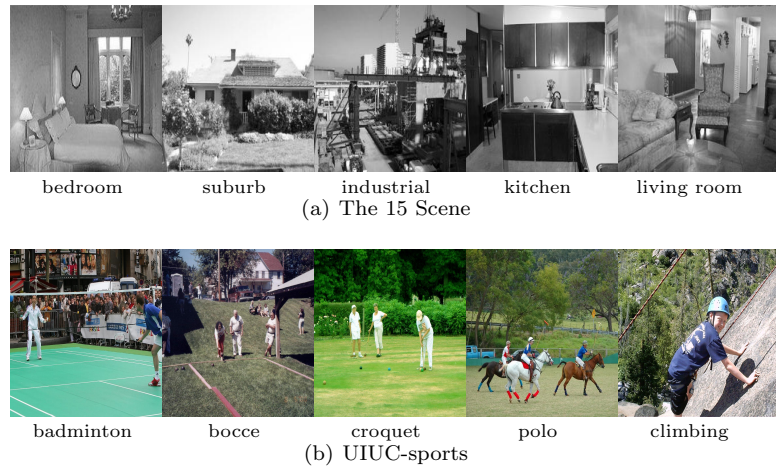


Fig. 1 Example images from *The 15 Scenes* and the *UIUC-sports* datasets.

and computing the histogram representation and concatenating all the histograms. The feature dimension of each image is calculated as

$$N = M \sum_{l=0}^L 4^l = M \frac{1}{3} (4^{L+1} - 1), \quad (14)$$

where M is the vocabulary size. l is the level of the scale space in SPM model. L is the max value of level, and is set to 2 in this paper.

The genetic algorithm in this paper is implemented by the GAOT toolbox developed by Houck et al [11]. The genetic operators, including select, crossover, and mutation, and the system parameters, are all set to the default values of GAOT. The predefined threshold λ is set to 0.05.

To make our tests more reasonable, we randomly partitioned the training images and the testing images. For the UIUC-sports dataset, we randomly select 80 training images from each category and the rest images are used for testing. For The 15 Scenes dataset, 100 images per class are randomly selected for training and the rest images are used for testing. The details are shown in Tab. I. All the experiments are repeated 50 times to ensure the reliability and the average results are reported in the paper.

Datasets	Classes	Training	Testing
UIUC-sports	8	640	939
The 15 Scenes	15	1500	2985

Tab. I Specifications of two types of datasets.

4.3 Experimental Results

The first set of experiments was set up to evaluate the impact of the number of the hidden nodes of ELM and the number of ELMs in an ensemble. To select the number of hidden nodes appropriately, we perform experiments with ELM algorithm on two different datasets. Because the selection of hidden nodes is relevant to the size of training data, we record the classification accuracy once every 25 and 50 hidden nodes on the UIUC-sports and The 15 Scenes datasets, respectively. The results are presented in Fig. 2. It can be observed that when the number of hidden nodes is set to 200 and 500 on two datasets respectively, the classification accuracy reaches the maximum. To make the comparison between the ensemble methods and ELM conveniently, the number of the hidden nodes of ensemble methods is set to 200 and 500 respectively when the experiments are performed on UIUC-sports and The 15 Scenes datasets.

In Fig. 3 we give comparison of the classification accuracy and the number of ensembles between the proposed RFSEN-ELM and other ensemble methods: RF-ELM, B-ELM and V-ELM. It can be observed that RFSEN-ELM has shown the best performance for all ensemble sizes on two datasets, while RF-ELM has slightly higher accuracy than V-ELM and B-ELM. RFSEN-ELM can achieve the robust performance only between 15 and 25 base classifiers, but the compared methods need more than 30 base classifiers to achieve the robust results.

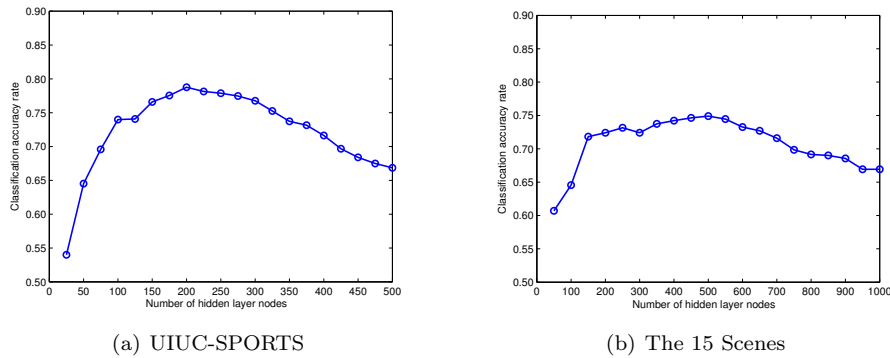


Fig. 2 The relationship between classification accuracy and the number of hidden nodes on two datasets.

In general, the combination of multiple ELMs in an ensemble substantially increases the classification accuracy, because the ensembles have more voting power with a larger number of classifiers. But when we further increase the number of ensembles, the marginal improvement of accuracy is negligible, and the training cost increases substantially, especially for the proposed RFSEN-ELM method. To illustrate this, the training costs of RFSEN-ELM and the compared methods are recorded on two datasets, adopting different number of ELMs in an ensemble but same number of hidden nodes. As shown in Fig. 4, the training cost of RFSEN-ELM with 50 ensembles is longer than 500 and 1800 seconds respectively, while V-ELM and B-ELM finished the training phase within several seconds. Therefore,

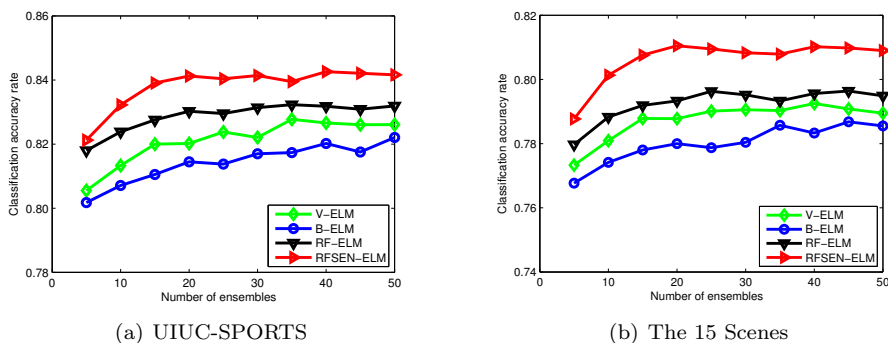


Fig. 3 The classification accuracy of proposed method and the compared methods with different numbers of ensembles.

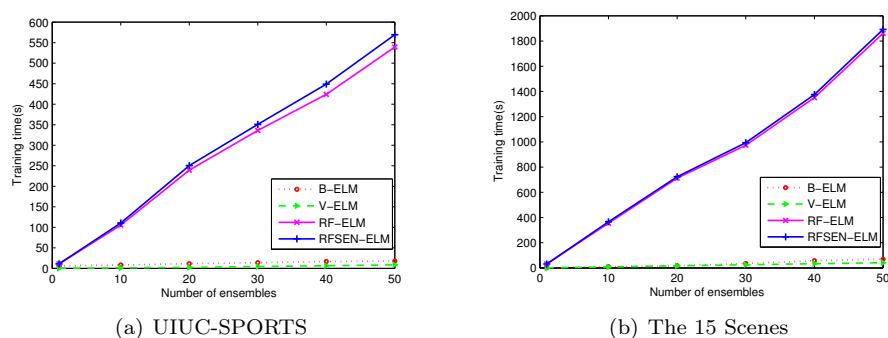


Fig. 4 The training time of proposed method and the compared methods with different numbers of ensembles.

it is inadvisable for applications on large image datasets when the ensemble size of RFSEN-ELM is too larger. When considering both the improvement of the classification accuracy and the increment of the training cost, we suggest the number of ensembles size for the proposed method should be less than 20.

The second set of experiments is conducted in order to compare the performances of the proposed method and several other recent classification methods. Simulations using V-ELM, B-ELM, RF-ELM, RFSEN-ELM, SVM and LSSVM are conducted on two datasets. The parameter settings of the ensemble methods are selected based on the analysis of the first set of experiment. The optimal choice of the number of ensembles of the ensemble algorithms should be set to 20 due to the balance of the classification accuracy and the training cost. For SVM and LSSVM, the Gaussian RBF is used as the kernel function, the optimal parameters C and γ are selected by a grid search over pairs of values $C = [2^{-4}, 2^{-2}, \dots, 2^{-10}]$, $\gamma = [2^{-10}, 2^{-8}, \dots, 2^4]$, using 5-fold cross validation.

The computational efficiencies of the proposed approaches are evaluated in comparison with other methods. Tab. II and Tab. III show the simulation of time

performance results of each method. It is obvious that the ELM algorithm is much faster than the other algorithms and that SVM is the most time consuming algorithm in both training and testing step. The experiment results is in consistent with the common acknowledge that the training cost of nonlinear SVM algorithm is sensitive to the process of finding the optimal combination of the parameters C and γ , and the searching process of parameters is generally time consuming. We could see from the results that RFSEN-ELM costs more time in training phase than the other ensemble based ELM methods due to the complex training step for improving diversity. When considering the testing phase, RFSEN-ELM shows the better results than the other ensemble methods, because the selective ensemble model utilizes fewer but better base classifiers to make up the ensemble, so the actual number of ensembles of RFSEN-ELM is less than 20.

Methods	Training(s)	Speedup	Testing(s)	Speedup
ELM	0.159	14448.4	0.081	87.28
B-ELM	3.58	641.7	1.57	4.5
V-ELM	2.98	770.9	1.48	4.77
RF-ELM	257.48	8.92	1.51	4.68
RFSEN-ELM	263.15	8.73	0.83	8.52
SVM	2297.3	1	7.07	1

Tab. II Computational costs comparison on UIUC-sports dataset (Speedup denotes training (or testing) time compared with SVM).

Methods	Training(s)	Speedup	Testing(s)	Speedup
ELM	0.953	4007.97	0.568	30.14
B-ELM	20.54	185.96	10.34	1.66
V-ELM	19.31	197.81	10.57	1.62
RF-ELM	712.36	5.36	10.29	1.66
RFSEN-ELM	723.45	5.28	6.36	2.69
SVM	3819.6	1	17.12	1

Tab. III Computational costs comparison on The 15 Scenes dataset (Speedup denotes training (or testing) time compared with SVM).

Compared with other machine learning algorithms, the random assignment of input weights and biases of ELM may result in unstable and non-optimal output. So we further evaluate the robustness of the proposed ensemble approach. Fig. 5 shows the simulations of each ensemble method in 50 trials on two datasets. It is obvious that RFSEN-ELM is more stable than V-ELM and B-ELM, and slightly better than RF-ELM. The reasons for the robustness of RFSEN-ELM algorithm are the high diversity of each individual classifier based on rotation forest and the selection of optimal subset of the ensemble based on selective ensemble model. Additionally, B-ELM is the most volatile one, because the step of bootstrapped replicas of the training data may be not applicable for these image datasets. The classification

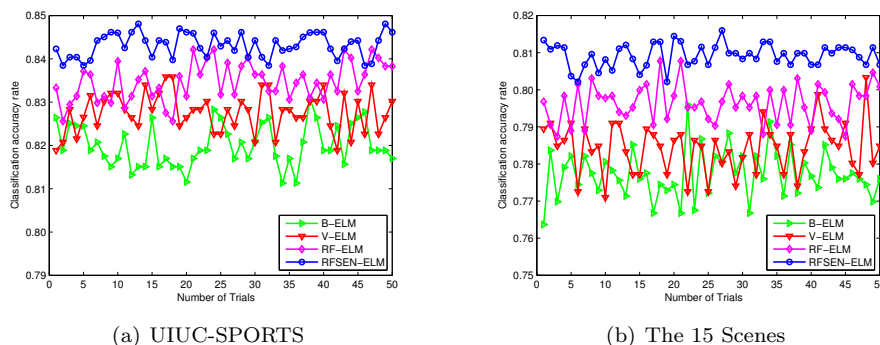


Fig. 5 The performance of proposed method and the compared methods in 50 trials on two datasets.

accuracy of proposed method and the compared methods with different numbers of ensembles.

Results in Tab. IV and Tab. V demonstrate the comparison of overall accuracies and class-specific accuracies for all of the classes on UIUC-sports. Tab. VI and Tab. VII show the same experimental results on The 15 Scenes. We can get the maximum of the overall classification accuracies on two datasets at 0.8459 and 0.8148. The proposed method RFSEN-ELM obtains higher overall accuracy than the other ensemble algorithms, and is slightly better than SVM and LSSVM. For most of the classes, we can get the maximum class-specific classification accuracies when using RFSEN-ELM. Notice that, SVM obtains the highest accuracies in some classes on two datasets. However, SVM has a time consuming training phase, and the cost will jump to thousands seconds for these datasets according to the analysis in computational efficiencies. Therefore, the proposed algorithm RFSEN-ELM could be used as an efficient and robust alternative to SVM and these experimental results illustrated that RFSEN-ELM has stronger generalization and robustness results for image classification.

class	ELM	B-ELM	V-ELM	RF-ELM	RFSEN-ELM	SVM	LSSVM
badminton	81.56	85.09	86.15	87.30	89.56	88.47	82.50
bocce	79.67	84.33	85.07	86.21	88.31	87.91	88.42
croquet	69.53	70.62	71.67	72.33	74.80	74.15	71.15
polo	78.33	79.54	81.23	82.38	83.49	83.08	78.43
climbing	81.67	85.48	87.29	88.27	90.71	90.34	90.23
rowing	78.42	80.38	80.86	81.24	82.03	81.72	77.06
sailing	80.23	83.82	85.63	85.91	86.23	86.31	84.91
boarding	76.17	79.05	79.33	80.72	81.56	81.44	78.36

Tab. IV Class-specific classification accuracy (%) on UIUC-sports dataset.

In the third set of experiments, we apply RFSEN-ELM and the other methods to the classification of UCI classification datasets [31], and the overall classifica-

Methods	Overall accuracy [%]	Dev [%]
ELM	78.20	1.38
B-ELM	81.04	0.71
V-ELM	82.15	0.55
RF-ELM	83.03	0.47
RFSEN-ELM	84.59	0.34
SVM	81.92	1.25
LSSVM	81.92	1.25

Tab. V Overall classification accuracy and standard derivation comparisons of different classifiers on UIUC-sports dataset.

class	ELM	B-ELM	V-ELM	RF-ELM	RFSEN-ELM	SVM	LSSVM
bedroom	61.25	65.38	66.34	66.51	68.91	68.43	66.38
suburb	86.61	88.89	91.44	92.83	94.51	95.02	93.62
industrial	63.67	66.58	67.01	67.92	70.05	68.87	67.35
kitchen	63.75	66.13	66.45	67.06	68.44	68.11	67.27
living room	61.86	65.43	65.97	66.55	68.13	67.20	69.02
coast	75.22	78.48	79.16	81.33	82.56	82.07	77.15
forest	89.58	91.75	93.83	94.05	94.57	94.84	93.86
highway	73.85	78.97	80.84	81.77	83.28	82.53	79.38
inside city	63.05	66.67	66.71	69.12	70.18	70.09	67.60
mountain	91.07	92.79	92.15	93.14	94.33	93.82	88.77
open country	70.97	75.81	77.21	77.84	79.76	78.03	80.90
street	79.55	82.14	82.65	84.01	86.80	88.22	87.54
building	81.13	85.26	86.11	88.43	90.94	90.33	87.12
office	90.63	92.88	92.51	93.22	93.68	93.56	94.78
store	63.42	69.05	70.66	73.69	76.15	75.72	75.81

Tab. VI Overall classification accuracy and standard derivation comparisons of different classifiers on The 15 Scenes dataset.

Methods	Overall accuracy [%]	Dev [%]
ELM	74.37	1.39
B-ELM	77.74	0.74
V-ELM	78.61	0.61
RF-ELM	79.83	0.49
RFSEN-ELM	81.48	0.38
SVM	81.12	0.56
LSSVM	80.16	1.38

Tab. VII Overall classification accuracy and standard derivation comparisons of different classifiers on The 15 Scenes dataset.

Datasets	Methods	Overall accuracy [%]	Dev [%]
derm	ELM	93.73	1.74
	B-ELM	97.25	0.97
	V-ELM	95.60	2.07
	RF-ELM	96.58	0.86
	RFSEN-ELM	97.37	0.77
	SVM	96.97	1.67
	LSSVM	96.97	1.30
optdigits	ELM	98.39	0.36
	B-ELM	98.00	1.95
	V-ELM	98.72	0.33
	RF-ELM	98.77	0.47
	RFSEN-ELM	98.83	0.35
	SVM	98.82	0.19
	LSSVM	98.79	0.31
segment	ELM	95.79	0.52
	B-ELM	95.84	0.67
	V-ELM	95.12	0.81
	RF-ELM	95.88	0.61
	RFSEN-ELM	96.03	0.53
	SVM	95.93	0.67
	LSSVM	95.82	0.92
zoo	ELM	93.55	4.30
	B-ELM	95.33	2.81
	V-ELM	96.33	2.22
	RF-ELM	96.67	1.67
	RFSEN-ELM	96.82	1.29
	SVM	95.00	3.24
	LSSVM	96.67	1.57

Tab. VIII Overall classification accuracy and standard derivation comparisons of different classifiers on UCI datasets.

tion accuracy and the standard deviation are used to measure the performance. The number of hidden nodes is set to 500 in ELM. The number of trails is set to 20 in B-ELM, V-ELM, RF-ELM and RFSEN-ELM. The method of parameters optimization for SVM and LSSVM is the same to that in the second set of experiments.

Results in Tab. VIII demonstrate the comparison of overall accuracies and standard deviations on different datasets. We can see that the method proposed in this paper has the maximum of the overall classification accuracies on four datasets at 0.9737, 0.9883, 0.9603 and 0.9682, which is higher than the accuracy of single ELM classification on four datasets. Compared to B-ELM, V-ELM and RF-ELM, our method combines the rotation forest with ELM to get the set of base classifiers, and uses selective ensemble model which based on genetic algorithm to optimal the ensemble pool. Seen from the experimental results, our method not only im-

proves the classification accuracies, but also further enhances the robustness on four datasets. Comparing our method with SVM and LSSVM, we can find that the accuracies of our method are higher than SVM and LSSVM. It can be seen that our method provides a new idea to solve multi-classification problem, avoiding the difficulties of using SVM and LSSVM to solve the multi-classification problem.

5. Conclusions

In this paper, we proposed a novel ensemble learning algorithm named RFSEN-ELM. It adopted rotation forest method to generate a set of candidate ELMs with high diversity, and then GA based selective ensemble method was implemented to evolve an optimal subset of the ensemble pool. The experiments have been conducted on two benchmark image datasets and four classification datasets, and the proposed algorithm consistently obtained better results than the compared algorithms. RFSEN-ELM achieved more robust and accurate results compared with the other ensemble based ELM learning algorithms. Moreover, compared with SVM and LSSVM, RFSEN-ELM yielded slightly better accuracy with a significantly lower cost of training phase. Therefore, we can draw the conclusion that RFSEN-ELM can generally achieve robustness of the classification and generate better generalization performance at an acceptable level of training cost for image classification.

Acknowledgement

This work supported by Zhejiang Provincial Natural Science Foundation of China (NO.LY18F030018) and Natural Science Foundation of China (51376055).

References

- [1] ANITHA J., HEMANTH D.J. An efficient Kohonen-fuzzy neural network based abnormal retinal image classification system, *Neural Network World*. 2013, 23(2), pp. 149–167, doi: [10.14311/NNW.2013.23.011](https://doi.org/10.14311/NNW.2013.23.011).
- [2] CAO J., CHEN T., FAN J. Landmark recognition with compact BOW histogram and ensemble ELM, *Multimedia Tools and Applications*. 2015, 75(2), pp. 1–19, doi: [10.1007/s11042-014-2424-1](https://doi.org/10.1007/s11042-014-2424-1).
- [3] CAO J., LIN Z., HUANG G.B., LIU N. Voting based extreme learning machine, *Information Sciences*. 2012, 185(1), pp. 66–77, doi: [10.1016/j.ins.2011.09.015](https://doi.org/10.1016/j.ins.2011.09.015).
- [4] CHANG C., LIN C. LIBSVM: a library for support vector machines, *ACM Transactions on Intelligent Systems and Technology*. 2011, 2(3), pp. 389–396, doi: [10.1145/1961189.1961199](https://doi.org/10.1145/1961189.1961199).
- [5] DAVID L.G. Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*. 2004, 60(2), pp. 91–110, doi: [10.1023/B:VISI.0000029664.99615.94](https://doi.org/10.1023/B:VISI.0000029664.99615.94).
- [6] GOLDBERG D.E. Genetic algorithms and machine learning, *Machine learning*. 1988 3(2), pp. 95–99, doi: [10.1145/168304.168305](https://doi.org/10.1145/168304.168305).
- [7] HAN B., HE B., MA M., SUN T., YAN T., LENDASSE A. RMSE-ELM: recursive model based selective ensemble of extreme learning machines for robustness improvement, *Mathematical Problems in Engineering*, 2014, 395686.

- [8] HAN B., HE B., RUI N., MA M., ZHANG S., LI M., LENDASSE A. LARSEN-ELM: Selective ensemble of extreme learning machines using LARS for blended data, *Neurocomputing*. 2015, 149, pp. 285–294, doi: [10.1016/j.neucom.2014.01.069](https://doi.org/10.1016/j.neucom.2014.01.069).
- [9] HANSEN L., SALAMON P. Neuralnetwork ensembles, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10)(1990) 993–1001, doi: [10.1109/34.58871](https://doi.org/10.1109/34.58871).
- [10] HINTON G.E., OSINDERO S., TEH Y. A fast learning algorithm for deep belief nets, *Neural Computation*. 2006, 18, pp. 1527–1554, doi: [10.1162/neco.2006.18.7.1527](https://doi.org/10.1162/neco.2006.18.7.1527).
- [11] HOUCK C.R., JOINES J., KAY M.G. A genetic algorithm for function optimization: a Matlab implementation, *NCSU-IE Technical Report*, 1995, pp. 95–109.
- [12] HUANG G.B., ZHU Q., SIEW C.K. Extreme learning machine: a new learning scheme of feedforward neural networks, In: *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2004, pp. 985–990.
- [13] HUANG G.B., ZHU Q., SIEW C.K. Extreme learning machine: theory and applications, *Neurocomputing*, 2006, 70(1), pp. 489–501, doi: [10.1016/j.neucom.2005.12.126](https://doi.org/10.1016/j.neucom.2005.12.126).
- [14] HUANG G.B., ZHOU H., DING X., ZHANG R. Extreme learning machine for regression and multiclass classification, *IEEE Transactions on Systems, Man, and Cybernetics Part B, Cybernetics*. 2012, 42(2), pp. 513–529, doi: [10.1109/TSMCB.2011.2168604](https://doi.org/10.1109/TSMCB.2011.2168604).
- [15] LAZEBNIK S., SCHMID C., PONCE J. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2169–2178.
- [16] LI F.F., PERONA P. A bayesian hierarchical model for learning natural scene categories, In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 524–531.
- [17] LI J., LI F.F. What, where and who? Classifying events by scene and object recognition, In: *Proceedings of the IEEE International Conference on Computer Vision*, 2007, pp. 1–8.
- [18] LI W., CHEN C., SU H.J., DU Q. Local binary patterns and extreme learning machine for hyperspectral imagery classification, *IEEE Transactions on Geoscience and Remote Sensing*. 2015, 53(7), pp. 3681–3693, doi: [10.1109/TGRS.2014.2381602](https://doi.org/10.1109/TGRS.2014.2381602).
- [19] LIU Y., HE B., DONG D., SHEN Y., YAN T., NIAN R., LENDASSE A. ROS-ELM: A Robust Online Sequential Extreme Learning Machine for Big Data Analytics, *Proceedings of ELM-2014*, Springer International Publishing, 2015, pp. 325–344.
- [20] LIU Y., HE B., DONG D., SHEN Y., YAN T., NIAN R., LENDASSE A. Particle swarm optimization based selective ensemble of online sequential extreme learning machine, *Mathematical Problems in Engineering*. 2015, 504120.
- [21] LU H., AN C., ZHENG E., LU Y. Dissimilarity based ensemble of extreme learning machine for gene expression data classification, *Neurocomputing*. 2014, 128(5), pp. 22–30, doi: [10.1016/j.neucom.2013.02.052](https://doi.org/10.1016/j.neucom.2013.02.052).
- [22] LU S., LU Z., YANG J., YANG M., WANG S. A pathological brain detection system based on kernel based ELM, *Multimedia Tools and Applications*. 2016, doi: [10.1007/s11042-016-3559-z](https://doi.org/10.1007/s11042-016-3559-z).
- [23] LU S., QIU X., SHI J., LI N., LU Z., CHEN P., YANG M.-M., LIU F.-Y., JIA W.-J., ZHANG Y. A pathological brain detection system based on extreme learning machine optimized by bat algorithm, *CNS & Neurological Disorders – Drug Targets*. 2017, 16(1), pp. 23–29.
- [24] RODRIGUEZ J.J., KUNCHEVA L.I. Rotation forest: A new classifier ensemble method, *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2006, 28(10), pp. 1619–1630, doi: [10.1109/TPAMI.2006.211](https://doi.org/10.1109/TPAMI.2006.211).
- [25] SAMAT A., DU P., LIU S., LI J. Ensemble extreme learning machines for hyperspectral image classification, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(4)(2014) 1060–1069, doi: [10.1109/JSTARS.2014.2301775](https://doi.org/10.1109/JSTARS.2014.2301775).
- [26] STOSIC D., LUDERMIR T. Voting based q-generalized extreme learning machine, *Neurocomputing*. 2016, 174, pp. 1021–1030, doi: [10.1016/j.neucom.2015.10.028](https://doi.org/10.1016/j.neucom.2015.10.028).
- [27] SUYKENS J.A.K., VANDEWALLE J. Least squares support vector machine classifiers, *Neural Processing Letters*. 1999, 9, pp. 293–300, doi: [10.1023/A:1018628609742](https://doi.org/10.1023/A:1018628609742).

- [28] WANG D., HUANG G.B. Protein sequence classification using extreme learning machine, In: Proceedings of the IEEE International Joint Conference on Neural Networks, 2005, pp. 1406–1411.
- [29] ZHANG Y., ZHAO G., SUN G.J., WU X., WANG Z.-H., LIU H.-M., GOVINDARAJ L.V., ZHAN T., LI J. Smart Pathological brain detection by synthetic minority oversampling technique, extreme learning machine, and jaya algorithm, *Multimedia Tools and Applications*. 2017, doi: [10.1007/s11042-017-5023-0](https://doi.org/10.1007/s11042-017-5023-0).
- [30] ZHOU Z., WU J., TANG W. Ensembling neural networks: many could be better than all, *Artificial intelligence*. 2002, 137(1), pp. 239–263, doi: [10.1016/j.artint.2010.10.001](https://doi.org/10.1016/j.artint.2010.10.001).
- [31] <http://archive.ics.uci.edu/ml/datasets.html>.