Neurocomputing 128 (2014) 166–174

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Aircraft recognition using modular extreme learning machine

Hai-Jun Rong [a,*], Ya-Xin Jia [b], Guang-She Zhao [a]

[a] State Key Laboratory for Strength and Vibration of Mechanical Structures, School of Aerospace, Xi'an Jiaotong University, Shaanxi 710049, PR China
[b] China Aerospace Construction Group Co., No. 83 South Segment of West 4th Ring Road, Beijing 100071, PR China

ARTICLE INFO

Article history:
Received 30 August 2012
Received in revised form
7 December 2012
Accepted 23 December 2012
Available online 31 October 2013

Keywords:
Single-hidden layer feedforward network
Extreme learning machine
Aircraft recognition
Hu moments
Zernike moments
Wavelet moments

ABSTRACT

In this paper, a novel recognition scheme is proposed for identifying the aircrafts of different types based on multiple modular neural network classifiers. Three moment invariants including Hu moments, Zernike moments and Wavelet moments are extracted from the characteristics exhibited by aircrafts and used as the input variables of each modular neural network respectively. Each modular neural network consists of multiple single-hidden layer feedforward networks which are trained using the extreme learning machine and different clustering data subsets. A clustering and selection method is used to get the classification rate of each modular neural network and then based on their weighted sum the final classification output is obtained. The proposed recognition scheme is finally evaluated by recognizing six different types of aircraft models and the simulation results show the superiority of the proposed method compared with the single ELM classifier and other classification algorithms.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Traditionally, an aircraft was recognized with the use of manual binoculars on the basis of their engine sound and shapes [1]. But the complex backgrounds where an aircraft flies limit the effectiveness of the manual techniques. A lot of research studies has been motivated by the need for automatically identifying aircraft types in air traffic control, as well as in military applications [2–5]. The general goal of automatic aircraft recognition is to analyze the images of a given scene and to identify the potential targets from the given scene. In the automatic aircraft recognition, the silhouette and boundary of an aircraft has been widely exploited by extracting certain features representing the image. The utilization of good features plays a key role in the aircraft recognition. The features should be independent of the object's position and orientation and should contain enough information to uniquely recognize one object from another. But in reality, the geometric distortion of the aircraft including shift, scale and rotation is often met and thus the image patterns have to be able to be extracted regardless of its geometric distortion.

Moments and functions of moments have been utilized as pattern features in the aircraft recognition. Such features capture global or local information about the image and do not require closed boundaries as boundary-based methods such as Fourier descriptors. Hu has derived a set of moment functions with the desired property of invariance under image translation and rotation, which have been applied by many researchers in automatic aircraft identification [2,3]. In [4], Wavelet moment has been used for feature extraction of the aircraft in the infrared image where the global and local features are extracted by using different scaling and shifting factors. The experiment results show that the recognition efficiency with Wavelet-moment invariants is better than that with Hu-moment. Although Hu moment and Wavelet moment are effective in the aircraft recognition, their individual discrimination abilities are limited. In [5], the different shape characteristics of an aircraft are extracted using the four methods including binary map, contours, Zernike moments, and Wavelet coefficients. For recognition, these different features are integrated together into a large feature vector by assigning a set of proper weights on features. Experimental results have shown that the recognition results with integrated features are better than those using the individual features extracted from one of the above four methods. Although the integrated features show the superior performance, they require a large amount of computation and storage capability due to the high dimensionality of the integrated feature vector.

After the features are extracted, they are input to a designed classifier to decide a label for the underlying image. In [6], on the basis of the Hu-moment features, two distinct classifiers including a Bayes decision rule and a distance-weighted *k*-nearest-neighbor rule are used in classification experiments. The nearest-neighbor distance algorithm is utilized for classifying the aircraft with the multiple features in [5]. Although the Bayes decision method is characterized by a well-defined sense of optimality, it requires

* Corresponding author. Tel.: +86 2982 664 543.
E-mail address: hjrong@mail.xjtu.edu.cn (H.-J. Rong).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2012.12.064

a priori information concerning the statistics of the observation. The nearest neighbor algorithm only considers partial measurement information. Neural networks can learn to classify from labeled training data without requiring the knowledge of statistical models and thus are attractive alternative in the aircraft recognition. In [7], the multiple-layer feedforward neural networks are used for the radar target classification and the results indicate that neural networks can achieve similar performance compared with decision-theoretic classification techniques which require either nearest neighbor prototypes or complete statistical models. A three-layer neural network is also presented to recognize the aircraft types in [8]. In the two methods, the learning abilities of neural networks are guaranteed based on the back-propagation algorithm that is widely used in many neural-network applications. However the back-propagation method faces some trivial issues such as learning parameters (learning epochs, learning rate, etc.), stopping criteria, and/or local minima.

Recently, a new fast neural learning algorithm referred to as Extreme Learning Machine (ELM) with any hidden nodes has been developed for Single-Hidden Layer Feedforward Networks (SLFNs) in [9,10]. In ELM, the hidden nodes need not be adjusted during training. All the parameters of hidden nodes could randomly be generated according to any given continuous probability distribution without any prior knowledge of the target function. The output weights of the network are analytically determined using simple generalized inverse operation of the hidden layer output matrices. The ELM algorithm not only possesses better computational efficiency in terms of the learning speed and generalization capability compared with the back-propagation algorithm but also avoids the difficulties faced by the back-propagation method [10,11]. ELM has been successfully applied in many real world applications [12–16].

The primary objective of the present investigation is to study the use of ELM for the aircraft recognition. In the paper a hierarchial modular ELM recognition scheme is proposed for recognizing the aircraft types. In order to improve the accuracy of aircraft recognition, three features derived from Hu moment, Wavelet moment and Zernike moment respectively are used in the proposed scheme. Different from [5], the three individual features need not be combined together to form a feature vector and are used to perform a module recognition task. In each module, the task is further decomposed by dividing the training dataset into several independent subsets using the clustering method. Then ELM is used to learn each subset and classify the aircraft types. The classification result of each module is first obtained according to a clustering and decision decision method and then the final classification result is achieved based on the weighted sum of each module's classification rate for the unknown patterns where the classification rate of each module got in the validation process is used as the weight factor. The proposed scheme is evaluated on a group of six aircraft models of different types and the simulation results verify its superior performance in terms of recognition accuracy, computation speed and robustness in recognizing aircrafts.

This paper is organized as follows. Section 2 reviews the feature extraction methods together with the feature selection process. In Section 3, the design procedure of the proposed recognition scheme is introduced together with the ELM algorithm and modular ELM classifier. Section 4 shows the simulation results from the recognition of six different types of aircrafts. Section 5 presents the conclusions from this study.

2. Feature extraction

Moments can describe the geometrical features of different objects and thus have been widely used in pattern recognition applications. In the paper, three features from Hu moments [17], Zernike moments [17] and Wavelet moments [18] are extracted to represent different characteristics of aircrafts so that a high recognition accuracy can be obtained based on the integration of their discrimination abilities. The following will give a simple description about each moment method.

2.1. Hu moment

For a 2D image with the density distribution function $f(x,y)$, the geometric moment of $(p+q)$ order is defined as

$$m_{pq} = \iint x^p y^q f(x,y)\,dx\,dy \tag{1}$$

where p, q are non-negative integers, $x^p y^q$ is a standard power basis, $f(x,y)$ is the gray value of the image at x and y location.

Invariance to translation can be achieved simply by shifting the polynomial basis into the object centroid. When the centroid of the image is (x_c, y_c), the central geometric moment is defined as

$$\mu_{pq} = \iint (x-x_c)^p (y-y_c)^q f(x,y)\,dx\,dy \tag{2}$$

where $x_c = m_{10}/m_{00}$ and $y_c = m_{01}/m_{00}$ are the gravity center of the image. m_{00} is an area of the object for binary images. m_{10} and m_{01} are one-order geometric moments. It is noted that $\mu_{10} = \mu_{01} = 0$ and $\mu_{00} = m_{00}$ always hold.

Scaling invariance is obtained by proper normalization of each moment. Since low-order moments are more stable to noise and easier to calculate, the moment is normalized most often by a proper power of μ_{00}, which is given by

$$\nu_{pq} = \frac{\mu_{pq}}{\mu_{00}^w} \tag{3}$$

where ν_{pq} is the normalized central geometric moment, $w = (p+q)/2+1$.

Hu has derived a set of seven moments invariant to translation, scale change and rotation using the low-order, second- and third-order normalized central moments. They are given as

$$\begin{aligned}
\phi_1 &= \nu_{20} + \nu_{02}\\
\phi_2 &= (\nu_{20}-\nu_{02})^2 + 4\nu_{11}^2\\
\phi_3 &= (\nu_{30}-3\nu_{12})^2 + (3\nu_{21}-\nu_{03})^2\\
\phi_4 &= (\nu_{30}+\nu_{12})^2 + (\nu_{21}+\nu_{03})^2\\
\phi_5 &= (\nu_{30}-3\nu_{12})(\nu_{30}+\nu_{12})((\nu_{30}+\nu_{12})^2 - 3(\nu_{21}+\nu_{03})^2)\\
&\quad + (3\nu_{21}-\nu_{03})(\nu_{21}+\nu_{03})(3(\nu_{30}+\nu_{12})^2 - (\nu_{21}+\nu_{03})^2)\\
\phi_6 &= (\nu_{20}-\nu_{02})((\nu_{30}+\nu_{12})^2 - (\nu_{21}+\nu_{03})^2) + 4\nu_{11}(\nu_{30}+\nu_{12})(\nu_{21}+\nu_{03})\\
\phi_7 &= (3\nu_{12}-\nu_{03})(\nu_{30}+\nu_{12})((\nu_{30}+\nu_{12})^2 - 3(\nu_{21}+\nu_{03})^2)\\
&\quad - (\nu_{30}-3\nu_{21})(\nu_{21}+\nu_{03})(3(\nu_{30}+\nu_{12})^2 - (\nu_{21}+\nu_{03})^2)
\end{aligned} \tag{4}$$

where ν_{20} and ν_{02} are the second-order normalized central moments. ν_{30}, ν_{03}, ν_{21} and ν_{12} are the third-order normalized central moments.

2.2. Zernike moment

Zernike moment is one of the orthogonal moments and constructed based on the Zernike polynomials orthogonal on a unit circle. A Zernike basis polynomial with p order and q repetition is given by

$$V_{pq}(x,y) = R_{pq}(r)e^{jp\theta} \tag{5}$$

where p is a non-negative integer, $|q| \le p$ and $p-|q|$ is even. (r,θ) are the polar coordinates of Cartesian coordinates satisfying $r = \sqrt{x^2+y^2}$ and $\theta = \arctan(y/x)$. $e^{jp\theta}$ is an angular part of the

polynomial and R_{pq} is a radial part defined as

$$R_{pq}(r)=\sum_{s=0}^{(p-|q|)/2}(-1)^s\frac{(p-s)!}{s!((p+|q|)/2-s)!((p-|q|)/2-s)!}r^{p-2s} \tag{6}$$

For a 2D image with size $M\times N$ and the density distribution function $f(x,y)$, the Zernike moment of the p-th order and repetition q is defined as

$$A_{pq}=\frac{p+1}{\pi}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}V_{pq}^{*}(r,\varphi)f(x,y) \tag{7}$$

where $p=0,1,2,\ldots$, $q=-p,-p+2,\ldots,p$ and V_{pq}^{*} is the conjugate complex of V_{pq}.

If the image is rotated by an angle θ, the value of Zernike moment changes as

$$A_{pq}^{'}=A_{pq}e^{-jq\theta} \tag{8}$$

Rotation invariant of the Zernike moment can be constructed by taking the magnitude $|A_{pq}|$. The equality between $|A_{pq}|$ and $|A_{pq}^{'}|$ guarantees this.

The Zernike moment is transition invariant by calculating A_{pq} according to the centroid (x_c,y_c) of the image. Supposing the center of rotation is in the centroid of the image, the conversion from Cartesian coordinates (x,y) to polar coordinates (r,φ) is expressed as

$$r=\frac{\sqrt{(x-x_c)^2+(y-y_c)^2}}{r_{\max}}$$
$$\varphi=\arctan\left(\frac{x-x_c}{y-y_c}\right) \tag{9}$$

where $r_{\max}=(\sqrt{m_{00}}/2)\sqrt{M/N+N/M}$.

The scaling invariance of the Zernike moment is obtained based on its division by the A_{00}, i.e., $\mathrm{ZM}_{pq}=A_{pq}/A_{00}$ where A_{00} is the density of sampling.

2.3. Wavelet moment

As Zernike moments, the Wavelet moments are orthogonal moments but differently they are defined based on an orthogonal wavelet basis function of the following form:

$$\psi_{ab}(r)=\frac{1}{\sqrt{a}}\psi\left(\frac{r-b}{a}\right) \tag{10}$$

where $a\in R^{+}$, $b\in R$ are the dilation and translation parameters. $\psi(\cdot)$ is the mother wavelet function that is used to generate the whole basis. The cubic B-spline is one of the widely used mother wavelet functions and by selecting the parameters a and b as follows:

$$a=0.5^{m},\quad m=0,1,\ldots,l$$
$$b=0.5n0.5^{m},\quad n=0,1,\ldots,2^{m+1} \tag{11}$$

Eq. (10) is rewritten as

$$\psi_{mn}(r)=2^{m/2}\psi(2^{m}r-0.5n) \tag{12}$$

For a $M\times N$ digital image $f(x,y)$, the Wavelet moments in the unit disc $(r\le 1)$ are defined as follows:

$$W_{mnq}=\sum_{x=1}^{M}\sum_{y=1}^{N}\psi_{mn}(r)e^{-jq\theta}f(r,\theta) \tag{13}$$

with $r=\sqrt{x^2+y^2}$, $\theta=\arctan(y/x)$.

The Wavelet moment invariants can be derived by applying the methodologies presented in the Zernike moments. For example the magnitude of the Wavelet moment, $|W_{mnq}|$, is a rotation invariant feature of the underlying image and can be used for invariant recognition. The translation invariant is achieved by moving the origin to the mass center (x_c,y_c) of the image, while scaling invariance is obtained by resizing the image with a scaling factor $\alpha=\sqrt{M_{00}/area}$ where M_{00} is the zeroth geometric moment order and *area* is a constant equal to the expected image size.

Hu moments describe the characterization of two-dimensional images using the geometric moments. Zernike and Wavelet moments are the orthogonal moments constructed based on the orthogonal basis functions. Compared with Hu moments, Zernike and Wavelet moments have minimum information redundancy where different moment orders describe different parts of the image. Besides, Hu and Zernike moments are global invariants calculated over the whole image space and capture the global information of the image, but they lack the abilities to distinguish similar but different objects. Wavelet moments are local invariants calculated from a certain neighborhood of dominant points. When the object undergoes any change, the local invariants are not affected. Thus Wavelet moments provide higher classification performance for similar objects with slight difference. Since the three moments have different pattern representation capabilities, they are separately used to exact the aircraft features for describing the different characteristics of an aircraft so that a higher classification accuracy can be achieved.

2.4. Feature selection

In the three moment invariants, the number of Hu moment invariants is fixed and equal to 7, whereas the number of Zernike and Wavelet moment invariants will be larger than 7. In our study, there are 792 wavelet moment invariant features for a single image considering $m=0,1,2,\ldots,5$, $n=0,1,\ldots,2^{m+1}$ and $q=0,1,2,\ldots,5$. To reduce computational cost in processing huge Wavelet and Zernike moment invariants, feature selection is necessary in which the irrelevant features will be discarded and the best features are selected for classifying.

Multiple feature selection algorithms can be used together to generate an optimal feature set by taking the interventions of domain experts into account [19]. Three different feature selection methods, viz, Correlation-based Feature Subset Selection (CFSE), chi-square and ReliefF are firstly used to generate some important feature candidates. CFSE measures the predictive ability of each feature, along with the degree of redundancy among them. Some feature candidates which are highly correlated with classes and lowly correlated with each other are chosen. Chi-square measures the association between the class labels and features. Based on the chi-square measure, the features are sorted in the descending order. The larger the chi-square value, the higher the relevance between the feature and the class is. ReliefF is an extension of Relief and an instance-based feature evaluation method. It performs random sampling on instances, and calculates the correlation between each feature and class. Then a specific number of features in the descending order are chosen as the candidate feature set.

Each of the feature selection methods is able to identify some useful features from the dataset and each feature selected by these methods implies its importance in representing the aircrafts. When the three feature candidate sets are obtained, the important features are subsequently grouped to form the optimal feature sets for recognition based on the voting technique that selects the candidates having the larger amount of selection frequency. The determination process of the optimal features is summarized below.

(1) Vote among the three feature candidate sets W_{CFSE}, W_{CS} and W_{RF}, obtained from CFSE, chi-square and ReliefF respectively.

(2) Calculate the frequency of selection for each feature f_i using the following formula:

$$W_{f_i} = \sum(W_{CFSE}(f_i) + W_{CS}(f_i) + W_{RF}(f_i)) \tag{14}$$

(3) Sort the features according to their W_{f_i} values, and select a list of features as the candidate feature set for classifying whose number is determined by the designer.

When the optimal features are determined, they will be input to the classifier for recognizing the aircraft types. In the following section, the proposed recognition scheme is presented.

3. Design procedure for aircraft recognition

The architecture of the proposed recognition scheme is demonstrated in Fig. 1 where three modular neural networks (MNNs) are utilized as the classifiers by separately receiving the three moment invariants as input features. A final classification result $\hat{c}$ is formed by combining the outputs of the MNNs based on a decision function. In each MNN, the classification task is further decomposed by choosing a decomposition of the dataset. Thus each MNN comprises multiple base classifiers and each base classifier implements the task in its own data subset. Generally, task decomposition can be obtained in three different ways [20]: explicit decomposition where the task is decomposed by the designer before training, class decomposition where the decomposition is made based on the classes of the problem and automatic decomposition where the decomposition is made using an unsupervised or clustering algorithm. Clustering is one of the suitable methods for partitioning the dataset and can find a set of subsets that will promote disagreement among the base classifiers. Here the *K*-means clustering technique is utilized for automatically performing the data decomposition and the number of clusters is equal to the number of class labels.

The networks used in the MNN are of the SLFNs which are one of the best known and commonly used NN models. It has been used successfully for pattern recognition, also recently the ELM has been developed for training a SLFN with fast learning speed and high generation performance. In the proposed design scheme, ELMs are employed as base classifiers considering their merits. In what follows, the ELM will be briefly reviewed to provide the necessary background for implementing the neural classifiers.

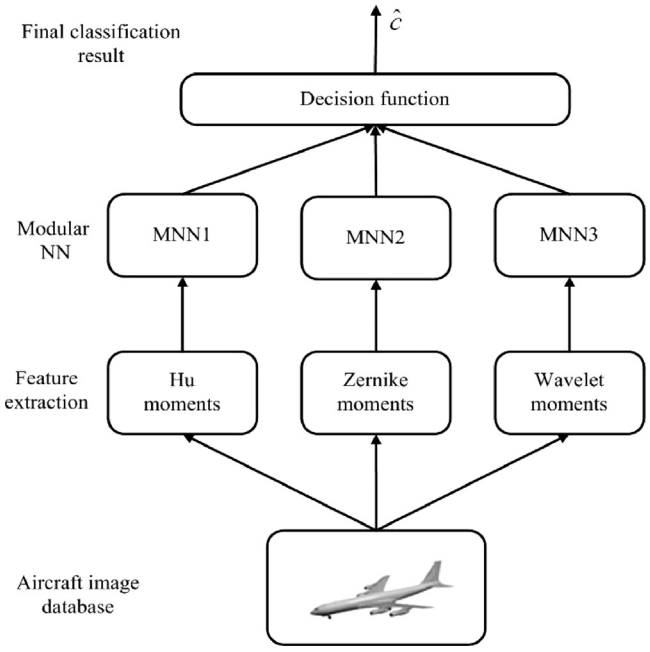

Fig. 1. Proposed aircraft recognition scheme.

3.1. Description of ELM

In this section a brief mathematical description of SLFN incorporating both additive and RBF hidden nodes in a unified way is given first.

3.1.1. Mathematical description of unified SLFN

The output of a SLFN with $\tilde{N}$ hidden nodes (additive or RBF nodes) can be represented by

$$f_{\tilde{N}}(\mathbf{x}) = \sum_{i=1}^{\tilde{N}} \boldsymbol{\beta}_i G(\mathbf{x}; \mathbf{c}_i, a_i), \quad \mathbf{x} \in \mathbf{R}^n, \ \mathbf{c}_i \in \mathbf{R}^n \tag{15}$$

where $\mathbf{c}_i$ and a_i are the learning parameters of hidden nodes, $\boldsymbol{\beta}_i$ is the output weight connecting the i-th hidden node to the output node, and $G(\mathbf{x}; \mathbf{c}_i, a_i)$ is the output of the i-th hidden node with respect to the input $\mathbf{x}$. For additive hidden nodes with the sigmoid or threshold activation function $g(x): R \to R$, $G(\mathbf{x}; \mathbf{c}_i, a_i)$ is given by

$$G(\mathbf{x}; \mathbf{c}_i, a_i) = g(\mathbf{c}_i \cdot \mathbf{x} + a_i), \quad a_i \in R \tag{16}$$

where $\mathbf{c}_i$ is the weight vector connecting the input layer to the i-th hidden node, a_i is the bias of the i-th hidden node, and $\mathbf{c}_i \cdot \mathbf{x}$ denotes the inner product of vectors $\mathbf{c}_i$ and $\mathbf{x}$ in $\mathbf{R}^n$.

For RBF hidden nodes with the Gaussian or triangular activation function $g(x): R \to R$, $G(\mathbf{x}; \mathbf{c}_i, a_i)$ is given by

$$G(\mathbf{x}; \mathbf{c}_i, a_i) = g(a_i \| \mathbf{x} - \mathbf{c}_i \|), \quad a_i \in R^+ \tag{17}$$

where $\mathbf{c}_i$ and a_i are the center and impact factor of the i-th RBF node. R^+ indicates the set of all positive real values. The RBF network is a special case of SLFN with RBF nodes in its hidden layer. Each RBF node has its own centroid and impact factor, and its output is given by a radially symmetric function of the distance between the input and the center.

3.1.2. ELM

For N arbitrary distinct samples $(\mathbf{x}_k, \mathbf{t}_k) \in \mathbf{R}^n \times \mathbf{R}^m$, if a SLFN with $\tilde{N}$ hidden nodes can approximate these N samples with zero error, it then implies that there exist $\boldsymbol{\beta}_i$, $\mathbf{c}_i$ and a_i such that

$$\sum_{i=1}^{\tilde{N}} \boldsymbol{\beta}_i G(\mathbf{x}_k; \mathbf{c}_i, a_i) = \mathbf{t}_k, \quad k = 1, \ldots, N. \tag{18}$$

Eq. (18) can be written compactly as

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{T} \tag{19}$$

where

$$\mathbf{H}(\mathbf{c}_1, \ldots, \mathbf{c}_{\tilde{N}}, a_1, \ldots, a_{\tilde{N}}, \mathbf{x}_1, \ldots, \mathbf{x}_N) = \begin{bmatrix} G(\mathbf{x}_1; \mathbf{c}_1, a_1) & \cdots & G(\mathbf{x}_1; \mathbf{c}_{\tilde{N}}, a_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ G(\mathbf{x}_N; \mathbf{c}_1, a_1) & \cdots & G(\mathbf{x}_N; \mathbf{c}_{\tilde{N}}, a_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \tag{20}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m} \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix}_{N \times m}^T \tag{21}$$

$\mathbf{H}$ is called the hidden layer output matrix of the network [10,21]; the i-th column of $\mathbf{H}$ is the i-th hidden node's output vector with respect to inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ and the k-th row of $\mathbf{H}$ is the output vector of the hidden layer with respect to input $\mathbf{x}_k$.

The ELM algorithm has the following property [16, p. 491]: For a SLFN with additive or RBF hidden nodes and an activation function $g(x): R \to R$ which is infinitely differentiable in any

interval, there exists $\tilde{N} \le N$ such that for N arbitrary distinct input vectors $\{\mathbf{x}_i | \mathbf{x}_i \in \mathbf{R}^n, i = 1, \ldots, \tilde{N}\}$, for any $\{(\mathbf{c}_i, a_i)\}_{i=1}^{\tilde{N}}$ randomly generated according to any continuous probability distribution $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\| < \varepsilon$ with probability one, where $\varepsilon > 0$ is a small positive value.

In this regard, for N arbitrary distinct samples $(\mathbf{x}_k, \mathbf{t}_k)$, in order to obtain arbitrarily small non-zero training error, one may randomly generate $\tilde{N}(\le N)$ hidden nodes (with random parameters $(\mathbf{c}_i, a_i)$). Eq. (19) then becomes a linear system and the output weights $\boldsymbol{\beta}$ are estimated as

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T} \tag{22}$$

where $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse [22] of the hidden layer output matrix $\mathbf{H}$. Calculation of the output weights is done in a single step here. Thus this avoids any lengthy training procedure where the network parameters are adjusted iteratively with appropriately chosen control parameters (learning rate, learning epochs, etc). The *three-step* ELM algorithm [9,10] can be summarized as follows:

ELM Algorithm: Given a training set $\aleph = \{(\mathbf{x}_k, \mathbf{t}_k) | \mathbf{x}_k \in \mathbf{R}^n, \mathbf{t}_k \in \mathbf{R}^m, k = 1, \ldots, N\}$, activation function g, and hidden nodes $\tilde{N}$

Step 1: Randomly assign hidden node parameters $(\mathbf{c}_i, a_i)$, $i = 1, \ldots, \tilde{N}$.
Step 2: Calculate the hidden layer output matrix $\mathbf{H}$.
Step 3: Calculate the output weight $\boldsymbol{\beta}$: $\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T}$.

3.2. Modular ELM

Supposing that the number of aircraft types is K and the cluster centroids of K clusters $C_1, \ldots, C_K$ are $\mathbf{v}_1, \ldots, \mathbf{v}_K$, K different base classifiers $(L_j, j = 1, 2, \ldots, K)$ are included in each MNN and trained with the same ELM learning algorithm using the different clustering data subsets $(\mathbf{x}^{(j)}, \mathbf{t}^{(j)})$. Fig. 2 illustrates the structure of multiple base ELM classifiers. The training data of all the base classifiers in each MNN are from the same kind of moment invariants as shown in Fig. 1 where the three moment invariants are used as the inputs of each MNN individually. It seems that it is troublesome to select multiple appropriate networks as the base classifiers. However in real implementation, the computation process can be simplified. Since in ELM the parameters $\{(c_{11}^{(1)}, a_1^{(1)}), \ldots, (c_{n\tilde{N}}^{(K)}, a_{\tilde{N}}^{(K)})\}$ of hidden nodes are assigned randomly and independent of training data, all the base classifiers can share a common hidden node set $(\mathbf{c}, \mathbf{a})$. As illustrated in the figure, the hidden nodes are the same for all the base classifiers and also K groups of independent training data $(\mathbf{x}^{(j)}, \mathbf{t}^{(j)}, j = 1, 2, \ldots, K)$ are obtained from the clustering to train the K independent base classifiers. It is noted that all of these base classifiers are designed to classify all the classes of targets, although some subsets may not consist of all classes of targets. $\boldsymbol{\beta}^{(j)}$ connecting the j-th base classifier outputneurons with the shared hidden nodes are learned separately without affecting other weights $(\boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(j-1)}, \boldsymbol{\beta}^{(j+1)}, \ldots, \boldsymbol{\beta}^{(K)})$.

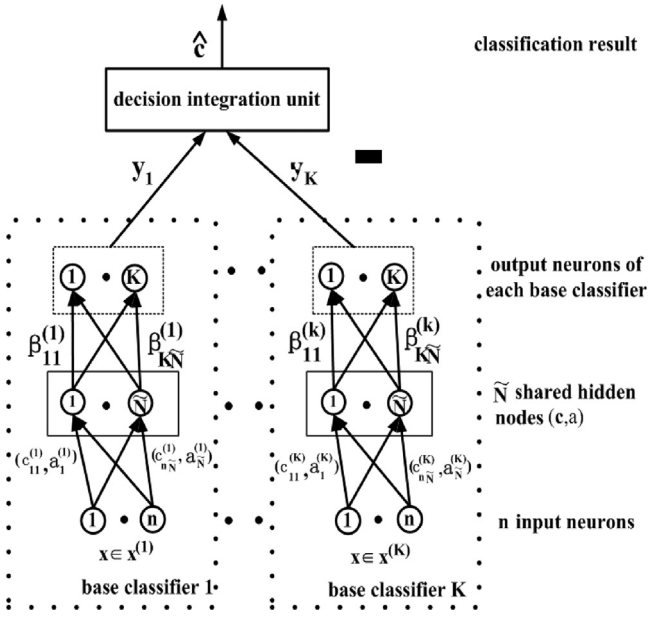

Fig. 2. Structure of multiple base classifiers.

Define weight vector $\boldsymbol{\beta}^{(j)}$ as $[\boldsymbol{\beta}_1^{(j)}, \ldots, \boldsymbol{\beta}_{\tilde{N}}^{(j)}]^T$ where $\boldsymbol{\beta}_i^{(j)} (i = 1, 2, \ldots, \tilde{N})$ denotes the weight connecting the i-th shared hidden node to the j-th base classifier output neurons. $\boldsymbol{\beta}^{(j)}$ can be determined as

$$\boldsymbol{\beta}^{(j)} = (\mathbf{H}^{(j)})^{\dagger}\mathbf{t}^{(j)} \tag{23}$$

where $\mathbf{t}^{(j)} (j = 1, 2, \ldots, K)$ is defined as

$$\mathbf{t}^{(j)} = [\mathbf{t}_1^{(j)} \ \cdots \ \mathbf{t}_{N^{(j)}}^{(j)}]_{K \times N^{(j)}}^T \tag{24}$$

Matrix $\mathbf{H}^{(j)} (j = 1, 2, \ldots, K)$ is defined as

$$\mathbf{H}^{(j)} = \mathbf{H}(\mathbf{c}_1, \ldots, \mathbf{c}_{\tilde{N}}, a_1, \ldots, a_{\tilde{N}}, \mathbf{x}_1^{(j)}, \ldots, \mathbf{x}_{N^{(j)}}^{(j)}) = \begin{bmatrix} G(\mathbf{x}_1^{(j)}; \mathbf{c}_1, a_1) & \cdots & G(\mathbf{x}_1^{(j)}; \mathbf{c}_{\tilde{N}}, a_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ G(\mathbf{x}_{N^{(j)}}^{(j)}; \mathbf{c}_1, a_1) & \cdots & G(\mathbf{x}_{N^{(j)}}^{(j)}; \mathbf{c}_{\tilde{N}}, a_{\tilde{N}}) \end{bmatrix}_{N^{(j)} \times \tilde{N}} \tag{25}$$

where $N^{(j)}$ is the training data size for the j-th base classifier.

Using the learning procedure described above and training data $(\mathbf{x}^{(j)}, \mathbf{t}^{(j)})$ achieved from the clustering method, each base classifier is trained based on the ten-fold cross-validation technique where the data is divided randomly into 10 parts. Each part is held out in turn and the learning scheme is trained on the remaining nine-tenths, then its confusion matrix CM is calculated on the holdout set. In the confusion matrix a datum under test is represented by the matrix row, and the confusion matrix column represents the classification of that particular datum. The element CM_{ik} gives the number of times that a class i object is assigned to class k. The diagonal elements indicate correct classification. Thus the classification rate for the k-th class is calculated as

$$A_k^j = \frac{CM_{kk}}{CM_{k1} + \cdots + CM_{kK}} \tag{26}$$

The learning procedure is executed a total of 10 times on different training sets. Finally, the 10 classification rates are averaged to yield an overall classification rate $\overline{A}_k^j$.

The learning procedure addressed above for each base classifier is suitable for three MNNs. For Hu moments, Zernike moments and Wavelet moments, the classification rate of the k-th base classifier on each clustering data $(\mathbf{x}^{(j)}, \mathbf{t}^{(j)})$ is expressed as $\overline{A}_k^{j,H}$, $\overline{A}_k^{j,Z}$ and $\overline{A}_k^{j,W}$ for simplicity.

For each MNN, when each unknown test pattern $\mathbf{x}$ is input to Fig. 2, it first compares with all cluster centroids for finding the nearest cluster $C_{nr} (1 \le nr \le K)$. Then the classifier L_{nr} that is trained on the nearest cluster is activated to get its output $\mathbf{y}_{nr}$. Since the classification accuracy rate from the training process represents a posteriori probabilities of each classification and is representative of its future behavior. Thus the classification rate of the nearest classifier $\overline{A}_k^{nr}$ from the training process is used as the weight factor of the output $\mathbf{y}_{nr}$. The final output $\hat{\mathbf{y}}$ for pattern $\mathbf{x}$ from the three MNNs are computed by normalizing the weight factors of the three MNNs, that is,

$$\hat{y}_k = \frac{\overline{A}_k^{nr,H}}{\overline{A}_k^{nr,H} + \overline{A}_k^{nr,Z} + \overline{A}_k^{nr,W}} y_{nr,k}^H + \frac{\overline{A}_k^{nr,Z}}{\overline{A}_k^{nr,H} + \overline{A}_k^{nr,Z} + \overline{A}_k^{j,W}} y_{nr,k}^Z$$

$$+\frac{\overline{A}_k^{nr,W}}{\overline{A}_k^{nr,H}+\overline{A}_k^{nr,Z}+\overline{A}_k^{nr,W}} y_{nr,k}^{W}, \quad k=1,2,\ldots,K \tag{27}$$

where $y_{nr,k}^{H}$, $y_{nr,k}^{Z}$ and $y_{nr,k}^{W}$ are the outputs of the nearest base classifier for the Hu moments, Zernike moments and Wavelet moments.

The final class result $\hat{c}$ is obtained as

$$\hat{c} = \arg\max_{k=1,2,\ldots,K} \hat{\mathbf{y}} \tag{28}$$

where $\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_K]$.

In summary, the whole learning procedure of the aircraft recognition scheme is given below.

Design procedure: Given the aircraft types K, the training image samples T and the testing image samples X

(1) *Training procedure*:
  (a) Extract the three moment invariants, viz., Hu moments, Zernike moments and Wavelet moments from the training image samples T and form three feature spaces F_H, F_Z and F_W.
  (b) For each feature space F_i $(i=\{H,Z,W\})$
    (i) Cluster the training data F_i into K clusters $C_1,\ldots,C_K$ using the K-means clustering method and calculate the clustering centroids $\mathbf{v}_1,\ldots,\mathbf{v}_K$.
    (ii) Design the base classifiers $L_1,\ldots,L_K$ using the elements of F_i which are in $C_1,\ldots,C_K$.
    (iii) Calculate the classification accuracies of $L_1,\ldots,L_K$ based on the ELM algorithm and ten-fold cross-validation technique, say $\overline{\mathbf{A}}^{i}$. For the j-th base classifier, its classification rate for each class is represented by $\overline{\mathbf{A}}^{j,i}=[\overline{A}_1^{j,i},\ldots,\overline{A}_K^{j,i}]$.
(2) *Testing procedure*:
  (a) Extract the three moment invariants, viz., Hu moments, Zernike moments and Wavelet moments from the testing image samples X and form three feature space F_H, F_Z and F_W.
  (b) For each testing sample $\mathbf{x}\in X$
    (i) For each feature spaces F_i $(i=\{H,Z,W\})$
      (A) Find the nearest cluster centroid from $\mathbf{v}_1,\ldots,\mathbf{v}_K$, say, $\mathbf{v}_{nr,i}$.
      (B) Use $\overline{\mathbf{A}}^{nr,i}$ to label $\mathbf{x}$ and calculate the output of the nearest classifier, i.e., $y_k^i=\overline{A}_k^{nr,i} * y_{nr,k}^i$, $k=1,2,\ldots,K$.
    (ii) Calculate the summation of the outputs from the three feature spaces, i.e., $\hat{y}_k=\sum_{i=1}^{\{H,Z,W\}} y_k^i / \sum_{i=1}^{\{H,Z,W\}} \overline{A}_k^{nr,i}$.
    (iii) Fuse the classification results from the three feature spaces and get the final classification result based on the decision function, i.e., $\hat{c}=\mathrm{argmax}_{j=1,2,\ldots,K}\hat{\mathbf{y}}$.
  (c) Return to step (b).

4. Performance evaluation of the proposed recognition scheme

In the section, the performance of the proposed recognition scheme is verified by identifying a group of six aircraft models of different types. As described in [23], no freely accessible database exists for aircraft recognition problem. Thus the dataset used to evaluate the proposed method is generated using a similar way in [5,23] where 3D digital models of six aircrafts are constructed and then they are rotated on an interval of 0–360°. The six aircrafts used are F-5, F-15, Boeing 707, IL-14, UH1 and AH-64D, which represent three kinds of typical aircrafts, viz., helicopters, civil aircrafts and fighters. Fig. 3 shows an example for the images of the six aircrafts in different orientations. To extract the features of the images well, the images need to be preprocessed based on the existing image processing techniques such as grayscale conversion, edge detection and image segmentation. The corresponding results are demonstrated in Fig. 4.

Fig. 3. Images of Boeing 707, IL-14, F-5, F-15, UH1 and AH-64D.

In the study, each aircraft model is rotated manually from 0° to 360° with an average interval of 0.6° to get different orientation of images. So the number of images for each target corresponds to 600 and the whole dataset includes $600\times 6=3600$ images. 2400 images of the aircraft models are generated for training and 1200 images of the aircraft models are produced for checking the proposed scheme. The parameter p of the Zernike moments is set as $p=0,1,\ldots,11$ while the parameters m and q of the Wavelet moments are set as $m=0,1,2$ and $q=0,1,2,3$. Also the sigmoid hidden nodes are utilized to evaluate the performance of the proposed scheme. The experimental results reported here are based on the average of 30 independent trials. All the simulations have been conducted in MATLAB 7.11 environment running in an ordinary PC with 1.7 GHz CPU.

4.1. Performance evaluation of modular ELM classifier

In this section, the performance of the proposed modular ELM (M-ELM) classifier is evaluated by comparing with three single ELM classifiers where the input variables of each single classifier are from the Hu moments, Zernike moments and Wavelet moments respectively. For simplicity, they are called as Hu-ELM, Zernike-ELM and Wavelet-ELM. Besides, various bad environmental conditions possible for the aircraft have been considered to evaluate the proposed algorithm. They include rain, snow and fog. The conditions of rain and snow are simulated by adding salt and pepper noise while the condition of fog is simulated by decreasing contrast of image with the background. Also the distorted patterns are introduced to evaluate the performance of the proposed scheme by extending or decreasing the length or width of the images. Fig. 5(a) depicts the aircraft image with a noise density of 0.2, Fig. 5(b) and (c) displays the images after decreasing contrast and distorting. Table 1 shows the classification results between the modular ELM and three single ELM classifiers under the ideal case and the bad weather cases. It is apparent to observe that the classification results of modular ELM are better than those using single classifiers under all the cases. Also it can be concluded that

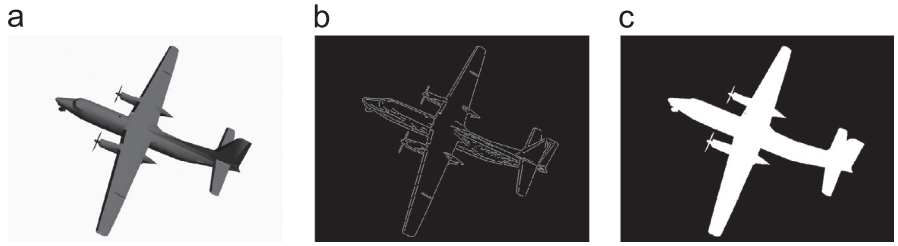

Fig. 4. Image preprocessing. (a) Grayscale, (b) edge detection and (c) image segmentation.

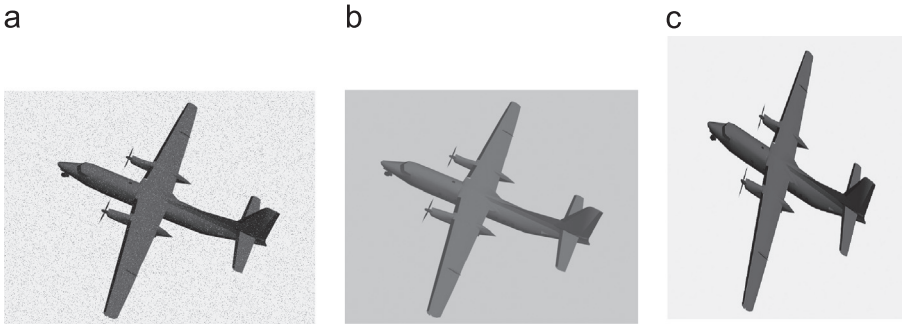

Fig. 5. Images with noise, decrease of contrast and distortion. (a) Noisy, (b) decreasing contrast and (c) distorted.

the proposed scheme can perform well in bad environmental conditions.

From the table, one can note that the results based on the wavelet moments are the lowest of all, but the Wavelet moments exhibit better performance in recognizing similar objects. We also perform simulations to evaluate the classification effect on different models of the same kind, e.g. Boeing 707 and IL-14. Fig. 6(a) shows the classification results on the two similar models when the number of features varies from 1 to 10. Fig. 6(b) shows the classification results when the number of hidden nodes varies from 10 to 200. From the two figures, it is clear that the results using Wavelet moments are better than those achieved using Hu and Zernike moments. The classification accuracies of modular ELM are the best of all. In addition, Fig. 6(a) and (b) demonstrates the effect of the number of features and hidden nodes on the classification accuracies. With the increase of features and hidden nodes, the accuracies become higher but decrease after certain values. Thus in our study, the number of features for Hu moment is equal to 7 while the number of features for Zernike and Wavelet moments is chosen as 10 and the number of hidden nodes is set as 100 during all the simulations. For illustration, Tables 2–4 list the Hu moments, Zernike moments and Wavelet moments for a typical view. From these tables, it is easy to find that all these moment values are similar and this reflects the similarity between the Boeing 707 and IL-14.

4.2. Performance comparison with other classification approaches

In this section, the ELM classifier is further evaluated by comparing with other classification algorithms which are included in the pattern recognition Matlab toolbox called Prtools [24], i.e., SVM, C4.5, KNN and Naive Bayes. When these methods are utilized to solve the classification task, their optimal parameters are selected using the optimization techniques in the toolbox. The three single classifiers using all the algorithms are firstly compared by receiving the Hu moments, Zernike moments and Wavelet moments as the input features and their comparison results are listed in Table 5. From the table, one can observe that although ELM classifiers require more computation time than C4.5 and Naive Bayes, it produces much higher classification rate than the two methods. Compared with SVM and KNN classifiers, the ELM classifiers have the highest classification rate along with the least training time. The proposed modular ELM classifier is then compared with other modular classifiers which respectively utilize the SVM, C4.5, KNN and Naive Bayes as the base classifiers for verifying its performance. These modular classifiers are designed based on the Prtools pattern recognition toolbox. For simplicity, they are named as M-SVM, M-C4.5, M-KNN and M-Naive Bayes. Table 6 displays the comparison results between these modular classifiers and from the table it can be found that the proposed modular ELM classifier produces the best classification accuracy with the least testing time.

Table 1
Comparison between modular and single ELM classifiers.

Classifier	Ideal (%)		Noise (%)		Decrease (%)		Distortion (%)	
	Rate	Dev	Rate	Dev	Rate	Dev	Rate	Dev
Hu-ELM	81.87	1.16	57.75	1.31	82.20	0.94	70.11	1.16
Zernike-ELM	94.14	0.62	86.25	0.97	94.08	0.62	65.63	0.74
Wavelet-ELM	77.70	1.05	77.57	1.21	78.50	1.26	67.88	1.41
M-ELM	94.25	0.90	89.41	0.88	94.58	0.62	83.65	1.04

5. Conclusions

The paper presents an intelligent recognition method to recognize types of aircraft. Firstly three features including Hu moments, Zernike moments and Wavelet moments are used to capture different shape characteristics of an aircraft. Then three modular neural classifiers are designed by using the SLFNs as the base classifiers and being trained using the ELM algorithm. The classification result of each modular NN for the unknown patterns are obtained using the clustering and selection method. The outputs of all the three modular NNs are fused to get the final class outputs based on the weighted sum of each modular NN's classification rate from the training process. Simulation results show that the proposed modular ELM classifier has a better classification rate than single ELM classifiers. Also the proposed neural recognition scheme using ELM as the base classifiers shows superior performance than other classification algorithms in terms

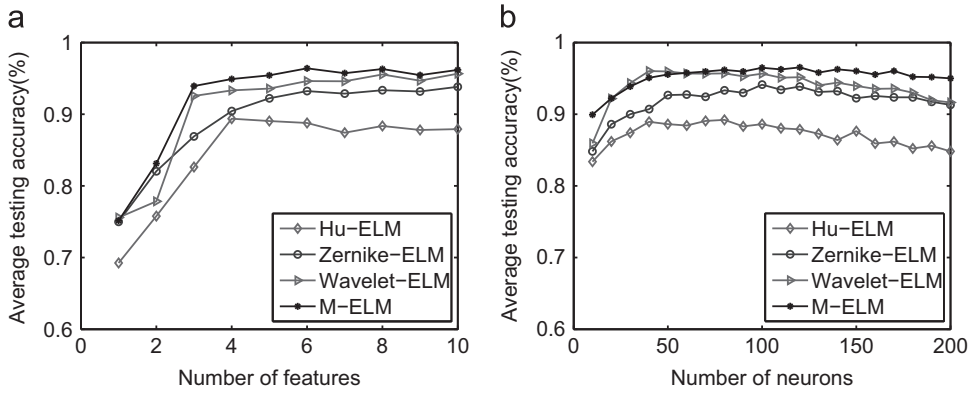

Fig. 6. Classification results between Boeing 707 and IL-14. (a) Features vary from 1 to 10 and (b) neurons vary from 10 to 200.

Table 2
Hu moments for Boeing 707 and IL-14.

Hu moments	Boeing 707	IL-14
ϕ_1	1.62666035	1.65188802
ϕ_2	5.81777508	5.87294481
ϕ_3	14.35183042	15.46450005
ϕ_4	12.38509077	13.06032908
ϕ_5	27.54279661	27.34761293
ϕ_6	17.03575036	16.10126949
ϕ_7	25.95851348	29.00624486

Table 3
Zernike moments for boeing 707 and IL-14.

Zernike moments	Boeing 707	IL-14
ZM_{64}	0.00564485	0.00692604
ZM_{44}	0.01048948	0.01381116
ZM_{22}	0.11650046	0.11768251
ZM_{33}	0.01043131	0.00925419
ZM_{31}	0.00772674	0.00612245
ZM_{00}	0.24279317	0.24230632
ZM_{11}	0.08590542	0.08806634
ZM_{75}	0.00380192	0.00195105
ZM_{20}	0.00500768	0.00331375
ZM_{51}	0.00401717	0.00254694

Table 4
Wavelet moments for Boeing 707 and IL-14.

Wavelet moments	Boeing 707	IL-14
W_{311}	140.0718544	140.0787337
W_{211}	466.0248519	478.0615038
W_{321}	429.2442656	439.5369077
W_{121}	458.453985	460.8712548
W_{221}	785.4016004	789.2186046
W_{111}	569.0802125	573.7609982
W_{241}	32.17394813	32.19306282
W_{131}	35.56922928	35.53978589
W_{331}	787.85863	793.2347445
W_{231}	433.3600556	433.0519878

of classification accuracy and the computation cost. The results from the bad flight environments indicate that the proposed method has a good robustness in recognizing aircrafts. The proposed recognition scheme can be extended easily for containing new feature set by adding a modular NN in parallel which receive the features as its inputs. This also will not affect the existing other modular NNs.

Table 5
Comparison of ELM and other algorithms.

Algorithm	Hu moment		Zernike moment		Wavelet moment	
	Rate (%)	Training time (s)	Rate (%)	Training time (s)	Rate (%)	Training time (s)
ELM	82.27	3.0285	94.34	3.1356	78.73	3.1793
SVM	47.14	2391.3	60.69	2258.8	35.68	2454.6
C4.5	77.07	0.5039	79.45	0.4836	62.11	0.7004
KNN	76.15	6.8063	92.01	6.5333	67.32	6.3430
Naive Bayes	55.76	0.0640	65.96	0.0702	49.98	0.0608

Table 6
Comparison of modular classifiers using different algorithms.

Classifier	Rate (%)	Training time (s)	Testing time (s)
M-ELM	92.26	74.30	0.2028
M-SVM	81.59	14832	15.6953
M-C4.5	81.29	4.96	9.3975
M-KNN	82.79	20.16	30.9225
M-Naive Bayes	73.30	0.31	0.7675

Acknowledgment

This work is funded in part by National Natural Science Foundation of China (Grant no. 61004055), the Fundamental Research Funds for the Central Universities and the State Key Lab of Astronautic Dynamics.

References

[1] S.F. Ali, J. Jaafar, A.S. Malik, Proposed technique for aircraft recognition in intelligent video automatic target recognition system (IVATRs), in: 2010 International Conference on Computer Applications and Industrial Electronics, Kuala Lumpur, Malaysia, 2010, pp. 174–179.
[2] S.A. Dudani, K.J. Breeding, R.B. McGhee, Aircraft identification by moment invariants, IEFE Trans. Comput. c-26 (1) (1977) 39–46.
[3] C.D. Ruberto, A. Morgera, Moment-based techniques for image retrieval, in: Proceedings of the 2008 19th International Conference on Database and Expert Systems Application, Turin, Italy, 2008, pp. 155–159.
[4] F. Zhang, S.-Q. Liu, D.-B. Wang, W. Guan, Aircraft recognition in infrared image using wavelet moment invariants, Image Vision Comput. 27 (2009) 313–318.
[5] J.-W. Hsieh, J.-M. Chen, C.-H. Chuang, K.-C. Fan, Aircraft type recognition in satellite images, IEE Proc.—Vision Image Signal Process. 152 (3) (2005) 307–315.
[6] I. Jouny, E.D. Garber, S.C. Ahalt, Classification of radar targets using synthetic neural networks, IEEE Trans. Aerospace Electron. Syst. 29 (2) (1993) 336–344.
[7] S. Chakrabarti, N. Bindal, K. Theagharajan, Robust radar target classifier using artificial neural networks, IEEE Trans. Neural Networks 6 (3) (1995) 760–766.

[8] A.A. Somaie, A. Badr, T. Salah, Aircraft image recognition using back-propagation, in: Proceedings of 2001 CIE International Conference on Radar, Beijing, China, 2001, pp. 498–501.
[9] G.-B. Huang, Q.-Y. Zhu, K.Z. Mao, C.-K. Siew, P. Saratchandran, N. Sundararajan, Can threshold networks be trained directly? IEEE Trans. Circuits Syst. II 53 (3) (2006) 187–191.
[10] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (2006) 489–501.
[11] H.-J. Rong, Y.-S. Ong, A.-H. Tan, Z. Zhu, A fast pruned-extreme learning machine for classification problem, Neurocomputing 72 (1–3) (2008) 359–366.
[12] N.-Y. Liang, P. Saratchandran, G.-B. Huang, N. Sundararajan, Classification of mental tasks from EEG signals using extreme learning machine, Int. J. Neural Syst. 16 (1) (2006) 29–38.
[13] S.D. Handoko, K.C. Keong, O.Y. Soon, G.L. Zhang, V. Brusic, Extreme learning machine for predicting HLA-peptide binding, in: Lecture Notes in Computer Science, vol. 3973, 2006, pp. 716–721.
[14] H.-J. Rong, G.-B. Huang, N. Sundararajan, P. Saratchandran, Online sequential fuzzy extreme learning machine for function approximation and classification problems, IEEE Trans. Syst. Man Cybern., Part B: Cybern. 39 (4) (2009) 1067–1072.
[15] C.-W.T. Yeu, M.-H. Lim, G.-B. Huang, A. Agarwal, Y.-S. Ong, A new machine learning paradigm for terrain reconstruction, IEEE Geosci. Remote Sensing Lett. 3 (3) (2006) 382–386.
[16] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Real-time learning capability of neural networks, IEEE Trans. Neural Networks 17 (4) (2006) 863–878.
[17] J. Flusser, T. Suk, B. Zitová, Moments and Moment Invariants in Pattern Recognition, John Wiley and Sons, 2009.
[18] M. Chessa, F. Solari, S.P. Sabatini, Human-Centric Machine Vision, InTech, 2012.
[19] K.-C. Khor, C.-Y. Ting, S.-P. Amnuaisuk, Forming an optimal feature set for classifying network intrusions involving multiple feature selection methods, in: 2010 International Conference on Information Retrieval and Knowledge Management, Shah Alam, Selangor, 2010, pp. 179–183.
[20] J.M. Santos, L.A. Alexandre, Modular neural network task decomposition via entropic clustering, in: Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications, Jinan, China, 2006, pp. 62–67.
[21] G.-B. Huang, H.A. Babri, Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions, IEEE Trans. Neural Networks 9 (1) (1998) 224–229.
[22] C. Rao, S.K. Mitra, Generalized Inverse of Matrices and Its Applications, John Wiley and Sons Inc, 1971.
[23] S.Z. Ali, M.A. Choudhry, A generalized higher order neural network for aircraft recognition in a video docking system, Neural Comput. Appl. 19 (2010) 21–32.
[24] R. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D. Tax, S. Verzakov, Prtools4.1, A Matlab Toolbox for Pattern Recognition, Delft University of Technology, 2007.

Hai-Jun Rong received the B.Eng. from Xi'an Technological University, PR China, in 2000, the M.Eng. from Xi'an JiaoTong University, PR China, in 2003, and the Ph. D. degree from Nanyang Technological University, Singapore, in 2008. From December 2006 to October 2008, she worked as a research associate and research fellow in Nanyang Technological University. Since then, she has been an associate professor in School of Aerospace, Xi'an Jiaotong University, PR China. Her research interest includes neural networks, fuzzy systems, pattern recognition and intelligent control.

Ya-Xin Jia received the B.Eng. and M.Eng. from Xi'an Jiaotong University, PR China, in 2009 and 2012. Currently she is working as an assistant engineer in China Aerospace Construction Group Co., Ltd. Her research interest includes neural networks and pattern recognition.

Guang-She Zhao received the B.Eng. and M.Eng. from Xi'an Jiaotong University, PR China, in 1986 and 1993. Currently he is working as a professor in School of Aerospace, Xi'an Jiaotong University, PR China. His research interest includes neural networks, fuzzy systems and intelligent control.