Cogn Comput (2015) 7:103–110
DOI 10.1007/s12559-014-9268-x

A Cognitive Ensemble of Extreme Learning Machines for Steganalysis Based on Risk-Sensitive Hinge Loss Function

Vasily Sachnev · Savitha Ramasamy · Suresh Sundaram · Hyoung Joong Kim · Hee Joon Hwang

Received: 25 December 2013 / Accepted: 27 February 2014 / Published online: 7 May 2014

Abstract In this paper, we propose a risk-sensitive hinge loss function-based cognitive ensemble of extreme learning machine (ELM) classifiers for JPEG steganalysis. ELM is a single hidden-layer feed-forward network that chooses the input parameters randomly and estimates the output weights analytically. For steganalysis, we have extracted 548-dimensional merge features and trained ELM to approximate the functional relationship between the merge features and class label. Further, we use a cognitive ensemble of ELM classifier with risk-sensitive hinge loss function for accurate steganalysis. As the hinge loss error function is shown to be better than mean-squared error function for classification problems, here, the individual ELM classifiers are developed based on hinge loss error function. The cognition in the ensemble of ELM obtains the weighted sum of individual classifiers by enhancing the outputs of winning classifiers for a sample, while penalizing the other classifiers for the sample. Thus, the cognitive ensemble ELM classifier positively exploits the effect of initialization in each classifier to obtain the best results. The performance of the cognitive ensemble ELM in performing the steganalysis is compared to that of a single ELM, and the existing ensemble support vector machine classifier for steganalysis. Performance results show the superior classification ability of the cognitive ensemble ELM classifier.

Keywords Extreme learning machine · JPEG steganography · Undetectable data hiding · Steganalysis

Introduction

Recent advancements in communication technologies (i.e., Internet, mobile communication) permit easy intrusion to information communicated, and hence, important information has to be protected from threats and malicious actions. Steganography provides a high level of security for secret information transmitted over these communication channels. In steganography, secret messages are embedded in digital media files, such as images, video, or audio, by slightly modifying them. These digital media files are commonly referred to as cover objects, and the modified steganographic signal is statistically undetectable from the original cover signal.

Steganalysis is a powerful tool to detect the presence of secret hidden data in an object. The steganalysis problem can be formulated as a supervised classification problem and solved using machine learning approaches. The performance of machine learning approaches for steganalysis is highly influenced by the feature extraction and the nature of the classification algorithm. Earlier work uses higher order moments of coefficients obtained by transforming an image using quadratic mirror filters [8], 18 binary similarity metrics [1], 23 discrete cosine transform features [10], 27 higher order moments of wavelet coefficients [11], etc. However, for efficient steganalysis, using a higher dimensional feature vector becomes essential. The feature

V. Sachnev (✉)
Department of Information, Communication and Electronics Engineering, Catholic University of Korea, Bucheon, South Korea
e-mail: bassvasys@hotmail.com

S. Ramasamy · S. Sundaram
School of Computer Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798, Singapore

H. J. Kim · H. J. Hwang
Center of Information Security Technologies, Korea University, Seoul, South Korea

set designed for JPEG images described in [26] used 274 features and was later extended to twice its size [22] by Cartesian calibration, while 324- and 486-dimensional feature vectors were proposed in [42] and [4], respectively. Other higher dimensional feature extraction techniques for steganalysis can be found in [5, 23, 25, 43, 44].

The functional relationship between the features and class label can be approximated using machine learning approaches. The high dimensional features and growing image database require a machine learning approach which requires smaller training time with better generalization ability. The recently developed neural network learning algorithm, namely extreme learning machine (ELM) [14, 15], is computationally less intensive and has the better generalization ability. Hence, in this work, we use ELM for steganalysis. The ELM is a feed-forward neural network with a single hidden layer, where the input weights and bias are selected randomly, and the output weights are estimated analytically. ELM has been developed based on the earlier findings that a neural network with at most N hidden neurons is sufficient to learn a function with N samples [13, 16, 17, 20]. It has been shown that the learning mechanism in ELM is similar to brain learning without the need of knowing the actual activation function of living brain neurons [20].

Extreme learning machine [14, 15] is a single hidden-layer feed-forward neural network that assumes randomly generated input parameters, based on which the output weights are calculated analytically. Thus, ELM requires lesser training time and is hence computationally efficient. Several variants of the ELM algorithm are available in the literature, and an extensive review of the ELM methods is presented in [18]. The ELM classifiers available in the literature include the k-fold selection scheme ELM [31, 32] and the real-coded genetic algorithm ELM [31]. Similarly, a few fast-learning ELM classifiers have been developed in the complex domain, and they include the phase-encoded complex-valued ELM and the bilinear branch-cut complex-valued ELM [34], the fast-learning fully complex-valued neural classifier [35], the circular complex-valued ELM classifier [36], and meta-cognitive ELM [37]. Recently, it has been shown in [19] that ELM provides a unified learning platform with a widespread type of feature mappings and can be applied to solve both regression and multi-class classification applications directly. Regression and multi-class classification problems can be solved by using Fuzzy ELM [27] and incremental learning [9]. ELM is successfully used in intrusion detection [6], bioinformatics [29, 45], image/video processing [7, 31, 40], and control [28]. The generalization performance of the cognitive ensemble of ELM classifier is better than a single ELM classifier [38] for various applications.

In general, machine learning algorithm uses perception to extract the knowledge from the data. Proposed learning algorithm selects appropriate samples based on the current knowledge to learn efficiently. This process emulates the human metacognition. The proposed approach uses an ELM ensemble to learn the samples (knowledge representation using ELM classifier). The ensemble process uses appropriate sample/classifier selection based on the knowledge content in the current ELM classifier. The self-regulation in sample selection enables proposed classifier to perform better than other classifiers. Cognitive principals have been used for classification of social media affective information [12], of natural language concepts [2], of handwritten text [3], and offline–online personal photos [46].

However, as one would expect, owing to the random initialization of the input parameters, the performance of ELM algorithm is affected by the choice of the input parameters. Hence, there is a need for a cognitive machine learning algorithm with self-realization [21] that is capable of learning from observations of its own knowledge [39]. Moreover, it has been shown in the literature that the risk-sensitive hinge loss error function estimates the posterior probability more accurately than the mean-squared error function in classification problems. In this paper, we address the above requirements by developing a cognitive ensemble of ELM classifiers based on the risk-sensitive hinge loss error function and use the classifier to perform steganalysis. The input parameters of each individual ELM classifier in the cognitive ensemble are initialized in different regions of the domain considered. Thus, their performance varies widely. Samples that are repeatedly correctly classified are labeled as *winning* samples, and samples that are misclassified by the majority of the classifiers are labeled as *losing* samples. Based on the region of initialization, the losing samples are classified correctly only by a few classifiers. Hence, a cognitive ensemble of ELM classifiers combines the outputs of individual classifiers by obtaining the weighted sum of outputs of these classifiers for all the samples. In this process, the classifiers that classify only the *winning* samples correctly are discarded from the pool. Since the cognitive ensemble of ELM classifiers is capable of choosing the participating networks and their weights, it is referred to as, "cognitive ensemble of ELM classifier." This cognition enables improved performance of the cognitive ensemble of ELM classifier compared to each individual ELM.

We study the performance of the cognitive ensemble of ELM classifiers on a set of benchmark binary classification problems from the UCI machine learning repository [41], in comparison with support vector machine classifier and ELM classifier. Next, we use the cognitive ensemble ELM to solve steganalysis problems. Here, the performance of the cognitive ensemble of ELM classifier is compared to SVM, ELM and the ensemble of base classifier that is the best performing classifier available in the literature for this data set. To understand the statistical significance of the classifiers, we also perform a binomial test using the cognitive ensemble of ELM classifiers as the base classifier. Performance study results also show that the cognitive ensemble of ELM classifiers outperforms the single ELM and ensemble of support vector machine classifier [24].

The paper is organized as follows. In "Cognitive Ensemble of Extreme Learning Machine Classifier" section, the algorithm of ELM classifier is presented and the hinge loss error based cognitive ensemble of ELM classifier is introduced. "Performance Study of Cognitive Ensemble of ELM Classifier" section presents the performance study of the cognitive ensemble ELM with other classifiers available in the literature on benchmark classification problems and steganalysis data sets. Finally, fourth section summarizes the main "Conclusions" from the study.

Cognitive Ensemble of Extreme Learning Machine Classifier

In this section, we first present the learning algorithm of ELM classifier and next describe the risk-sensitive hinge loss function-based cognitive ensemble ELM.

Extreme Learning Machine

ELM is a single hidden-layer feed-forward neural network. The input weights and bias of the hidden neurons in ELM are assigned randomly, and the output weights are computed analytically. ELM provides a unified framework to solve both regression and classification problems [19]. As steganalysis is solved as a classification problem, in this paper, ELM has been used as a classifier. The classification problem can be defined as follows: Given a training data set with N samples, $\left\{(\mathbf{x}^1, c^1), \ldots, (\mathbf{x}^t, c^t), \ldots, (\mathbf{x}^N, c^N)\right\}$, where $\mathbf{x}^t \in \Re^m$ are the m-dimensional real-valued input features of t-th observation and $c^t \in \{1, 2, \ldots, C\}$ is its class label. The coded class label $\mathbf{y}^t$ are obtained using:

$$y_k^t = \begin{cases} 1, & \text{if } \ c_t = l, \\ -1, & \text{otherwise,} \end{cases} \quad k = 1, 2, \ldots, C \tag{1}$$

Now, the classification problem can be viewed as approximating the decision function ($\mathbb{F}$) that maps the input features to the coded class labels, i.e., $\mathbb{F} : \Re^m \rightarrow \mathbb{C}^C$ as close as possible, and then predicting the class labels of new, unseen samples with certain accuracy.

As there are m input features and C classes, the ELM used to solve the problem has m input neurons and C output neurons. Let the number of hidden neurons be L. The neurons in the input and output layer of the ELM are linear, while the neurons in the hidden layer of ELM employ Gaussian activation function. Thus, the response of the j-th hidden neuron for the t-th sample (h_j^t) is given by:

$$h_j^t = \exp\left(- \left(\frac{(\mathbf{x}^t - \mathbf{a}_j)^T (\mathbf{x}^t - \mathbf{a}_j)}{2b_j^2} \right) \right) \quad j = 1, \ldots, L \tag{2}$$

where $\mathbf{a}_j \in \Re^m$ and $b_j \in \Re$ are the center and width of the j-th hidden neuron, respectively.

Then, the predicted coded class label $\hat{y}_k^t$ of the k-th hidden neuron of the t-th sample is computed as:

$$\hat{y}_k^t = \sum_{j=1}^{L} \beta_{kj} \cdot h_j^t, \quad k = 1, \ldots, C \tag{3}$$

where β_{kj} is the weight connecting the j-th hidden neuron and the k-th output neuron.

The above equation can be written in the matrix form as

$$\hat{Y} = \boldsymbol{\beta} H, \tag{4}$$

where H is the hidden-layer output matrix as shown below:

$$H(A, B, X) = \begin{pmatrix} G(\mathbf{a}_1, b_1, \mathbf{x}_1^t) & \ldots & G(\mathbf{a}_1, b_1, \mathbf{x}_N^t) \\ \vdots & \ldots & \vdots \\ G(\mathbf{a}_L, b_L, \mathbf{x}_1^t) & \ldots & G(\mathbf{a}_L, b_L, \mathbf{x}_N^t) \end{pmatrix} \tag{5}$$

The optimum output weight W can be computed analytically as:

$$W^* = YH^{\dagger} \tag{6}$$

From the predicted coded class labels, the class label of each sample can be estimated as:

$$\widehat{c}^t = \arg \max_{k=1,\ldots,C} \widehat{y}_k^t \tag{7}$$

Thus, for a given training data set $\left\{(\mathbf{x}^1, \mathbf{y}^1), \ldots, (\mathbf{x}^N, \mathbf{y}^N)\right\}$, and the number of hidden neurons L, the learning algorithm of ELM is summarized below:

1. Randomly generate hidden neuron centers, $A \in \Re^{L \times m}$ and the Gaussian width of the hidden neurons, $\mathbf{b} \in \Re^{(L \times 1)}$
2. Compute the hidden-layer output matrix H.
3. Calculate the output weights:

$$\beta = YH^{\dagger} \tag{8}$$

where $H^{\dagger}$ is the Moore–Penrose generalization inverse of responses of the hidden neurons.

However, the generalization performance of the ELM is dependent on the randomly chosen centers and hidden neuron bias, especially in applications like steganalysis, where the dimension of the input feature is large. Hence, in this paper, we propose the cognitive ensemble of ELM classifiers using risk-sensitive hinge loss function.

Risk-Sensitive Hinge Loss Function-Based Cognitive Ensemble of ELM

In this section, we present a cognitive ensemble of ELM classifiers for steganalysis. The performance of the ensemble classifiers depends on the choice of classifiers and weightage of each classifier. First, we present the cognition for selection of classifiers. Next, we present gradient descent-based risk-sensitive hinge loss (RSHL) ensemble.

Cognition: Selection of Classifiers

As the performance of the ELM classifier is affected by the random choice of centers and width, a cognition exploits the effect of initialization in the performance of ELM classifiers to develop a cognitive ensemble of ELM classifier. We train a pool of n ELM classifiers, each with initializations in different regions of the input space to ensure that Gaussian centers is distributed in the entire range of the input space. The cognition of the ensemble computes the outputs of these n classifiers on the N training samples, and the samples that are correctly classified are listed. Samples that are repeatedly correctly classified are labeled as *winning* samples, and samples that are misclassified by the majority of the classifiers are labeled as *losing* samples. The classifiers that classify only the *winning* samples correctly are discarded from the pool. Finally, the ensemble output is obtained as the weighted sum of the outputs of the participating classifiers for all the samples as shown in Fig. 1. The cognition in the ensemble of ELM classifier computes the weightage of each classifier using RSHL.

Estimation of Weights for Risk-Sensitive Hinge Loss Error Function-Based Cognitive Ensemble of ELM

In this section, we present the cognition that performs gradient descent-based estimation of the weights (v_l) for the participating classifiers of the ensemble obtained as shown in section "Cognition: Selection of Classifiers". Let $\widehat{\mathbf{y}}_1^t, \ldots, \widehat{\mathbf{y}}_l^t, \ldots, \widehat{\mathbf{y}}_n^t$ be the outputs of the n ELM classifiers in the pool for the t-th sample. Then, the total output is obtained as follows:

$$\widehat{\mathbf{y}}^t = v_1 \widehat{\mathbf{y}}_1^t + \cdots + v_l \widehat{\mathbf{y}}_l^t + \cdots + v_n \widehat{\mathbf{y}}_n^t \tag{9}$$

Thus, the overall performance of the cognitive ensemble of ELM classifier is better than each individual ELM classifier in the pool. The weights $\mathbf{v}^s = [v_1^s \cdots v_l^s \cdots v_n^s]^T$ are obtained by a gradient descent-based procedure as

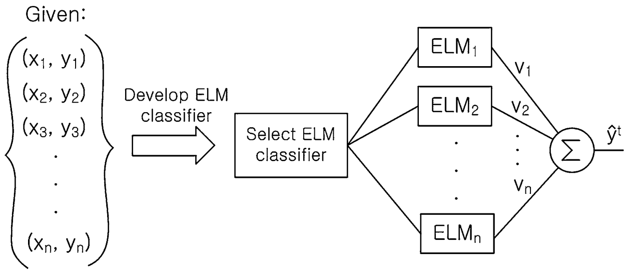

Fig. 1 Cognitive ensemble of ELM classifier

$$\mathbf{v}_l^s = \mathbf{v}_l^s + \eta \mathbf{e}_l^s \widehat{\mathbf{y}}_l^s + \beta \eta \delta_s; \; l = 1, \ldots, n \tag{10}$$

$$\text{where, } \delta_s = \eta \mathbf{e}_l^{s-1} \widehat{\mathbf{y}}_l^{s-1} + \beta \eta \delta_{s-1} \tag{11}$$

Here, η, β are the learning rate constants, δ_s is the adaptive factor for s-th iteration, which keeps the portion of updates $\eta \mathbf{e}_l^{s-1} \widehat{\mathbf{y}}_l^{s-1}$ from the previous iteration, v_l^s is the weight of the l-th classifier for the s-th iteration, and e_l^s is the error of the l-th classifier for the s-th iteration.

It has been shown in [30, 48] that hinge loss function predicts the posterior probability better than mean square error. Further, it has also been shown in [30] that a hinge loss error function with a risk factor plays a vital role in minimization of error for classification problems by penalizing the misclassification heavily and by minimizing the effect of imbalance in the training set. Therefore, in this paper, we adapt the risk-sensitive hinge loss error function [30] with a constant risk factor of 2, as shown below:

$$e_{l_k}^t = \begin{cases} 0 & \text{if } \widehat{y}_{l_k}^t \cdot y_{l_k}^t > 1 \\ y_{l_k}^t - \widehat{y}_{l_k}^t & \text{if } 0 < \widehat{y}_{l_k}^t \cdot y_{l_k}^t \leq 1 \\ 2 \cdot (y_{l_k}^t - \widehat{y}_{l_k}^t) & \text{if } -1 < \widehat{y}_{l_k}^t \cdot y_{l_k}^t \leq 0 \\ -2 \cdot (y_{l_k}^t \cdot \operatorname{sign}(\widehat{y}_{l_k}^t)) & \text{otherwise} \end{cases} \tag{12}$$

The learning algorithm of the cognitive ensemble of ELM classifiers can be summarized as shown below:

- **Train n individual ELM classifiers**:
  - Choose the number of hidden neurons (L), the center of Gaussian activation function ($\mathbf{a}_k$) and the Gaussian width (b_k) randomly.
  - Estimate the output weight W using Eq. (6).
  - Obtain the hinge output of the network.
- **Develop the cognitive ensemble of ELM classifiers:**
  - Define the samples that are correctly classified by most classifiers as winning samples.
  - If a network predicts only the winning samples correctly, delete the network.

- **Estimate using RSHL:**
  - Obtain the weighted sum of the participating classifiers using Eq. (9).
  - Update the weights of the participating classifiers using Eqs. (10) and (11).

From the algorithm, it can be seen that the cognitive ensemble of ELM classifier positively exploits the effect of initialization on individual ELM classifiers to obtain a better performing classifier. In the next section, we use the cognitive ensemble of ELM classifiers for steganalysis and compare its performance to the performance of an individual ELM classifier and the best results available in the literature for this problem.

The computational complexity of proposed cognitive ELM ensemble is $O(nL^3)$, where, n is the number of classifiers and L is the number of hidden neurons.

Performance Study of Cognitive Ensemble of ELM Classifier

In this section, we study the performance of the cognitive ensemble of ELM classifiers in comparison with other classifiers available in the literature. First, we evaluate the performance of the classifiers on benchmark binary classification problems and then use the classifier to solve a steganalysis problem. In our study, the number of hidden neurons in individual ELM classifier for all the problems considered in the study is chosen using the constructive–destructive procedure for selection of neurons, described in [33] and shown below.

Step 1 Select network with minimal configuration ($L = m + C$). Initialize the iteration count (count = 1).
Step 2 Train the network until $\eta_{\text{tr}} > \eta_{\text{desired}}$.
Step 3 Calculate the testing efficiency (η_{te}).
Step 4 If $\eta_{\text{tr}} < B$ (where B is a desired accuracy), then increase the number of hidden neurons and go to Step 2, else stop.

Performance Study on Benchmark Classification Problems

We first evaluate the performance of the cognitive ensemble ELM on benchmark binary classification problems from the UCI machine learning repository [41]. The details of the data sets including the number of features, the number of samples in training and testing, and the imbalance factor of these data sets are presented in Table 1. The imbalance factor (IF) is defined by:

Table 1 Description of benchmark data sets from UCI machine learning repository

Problem	No. of features	No. of samples		IF
		Training	Testing	
Liver disorders	6	200	145	0.17
PIMA	8	400	368	0.225
Heart	13	70	200	0.14
Breast cancer	9	300	383	0.26
Ionosphere	34	100	251	0.28

$$\text{IF} = 1 - \frac{C}{N} \min_{k=1,\ldots,C} N_k \tag{13}$$

where N_k is the number of samples in the class k. From the table, it can be observed that all the data sets used in the study are unbalanced data sets.

The results of the three classifiers, namely SVM, ELM, and a cognitive ensemble of ELM classifiers for the binary benchmark classification problems are presented in Table 2. From the table, it can be observed that the generalization ability of cognitive ensemble of ELM classifier is better than that of the SVM and ELM classifiers, although the training performance is better or almost similar to those of the SVM and ELM classifiers.

Hence, it is evident that the performance of a cognitive ensemble of ELM classifier is better than individual ELM classifier. Note that, for breast cancer and ionosphere problems, cognitive ensemble ELM does not show improvement (see Table 2). Due to testing and training efficiencies close to 100 %, presented method discards all ELM networks except the one with the best performance.

Table 2 Experiment results for benchmark data

Problem	SVM		ELM		Cognitive ensemble of ELM	
	η_{Tr}	η_{Te}	η_{Tr}	η_{Te}	η_{Tr}	η_{Te}
Liver disorders	79.50	71.03	88.50	72.41	**88.60**	**74.50**
PIMA	77.00	77.45	84.3	76.63	**84.6**	**80.40**
Heart	87.14	75.5	90.00	76.50	**90.11**	**80.51**
Breast cancer	98.67	96.60	100.00	96.35	100.00[a]	96.35[a]
Ionosphere	97.00	91.24	94.00	89.64	94.00[a]	89.64[a]

Bold values indicate cases where the proposed technique has improvement over the existing methods

[a] Note that, for breast cancer and ionosphere problems, cognitive ensemble ELM does not show improvement (see Table 2). Due to testing and training efficiencies close to 100 %, presented method discards all ELM networks except the one with the best performance. Thus, there is no room for further improvement using the cognitive ensemble of ELM. Next, we describe the steganalysis data set and present the performance study of the proposed classifier in solving steganalysis

Thus, there is no room for further improvement using the cognitive ensemble of ELM. Next, we describe the steganalysis data set and present the performance study of the proposed classifier in solving steganalysis.

Steganalysis Using Cognitive Ensemble of ELM Classifier

In this section, we employ the cognitive ensemble of ELM classifier to perform steganalysis. We first discuss the data set used in steganalysis, followed by the performance study of steganalysis using the cognitive ensemble of ELM classifier in comparison with individual ELM classifier and an ensemble of baseline classifier [24].

Merge Steganalysis Feature Data Set

In general, steganalysis method is based on analyzing features extracted from JPEG images. The quality of the extracted features predefines the performance of the binary classifier. Extracted features have to be sensitive to modifications due to data hiding, then steganalysis may correctly identify the cover and stego images.

In our experiments, we used BCH-based steganographic method proposed by Zhang et al. [47] for data hiding, and steganalysis method based on 548 features [26, 24]. In our experiments, steganalysis is performed by modifying the image data set with the hiding message with a length of 0.2-bits per nonzero DCT coefficients (bnpc). Such an experiment is capable of hiding reasonable amount of data in respect of the number of available nonzero DCT coefficients. Thus, JPEG image with small number of nonzero DCT coefficients always contain small number of hidden bits and vice versa.

In this paper, we used popular 548-dimension merged feature set proposed by Pevny et al. [26] and improved by Kodovsky [24]. Merged feature set contains 7 features subsets, i.e., $F_o = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5, \mathbf{f}_6, \mathbf{f}_7\}$ see Fig. 2:

$\mathbf{f}_1$: Global histogram H_l of all $64 \times n_b$ (n_b is the number of 8×8 blocks of JPEG image).

$\mathbf{f}_2$: 5 Local histograms of the AC coefficients $h^{ij} = \{h^{ij}_L, \ldots, h^{ij}_R\}$ located in the five fist positions according to the zigzag scan, i.e., $(i,j) \in \{(1,2), (2,1), (3,1), (2,2), (1,3)\}$.

$\mathbf{f}_3$: 11 Dual histograms.

$\mathbf{f}_4$: 6 Inter-block dependency coefficients.

$\mathbf{f}_5$: 2 Blockness coefficients.

$\mathbf{f}_6$: 3 Co-occurrence matrix of neighboring DCT coefficients.

$\mathbf{f}_7$: 81 Markov coefficients.

Feature set $F_o = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5, \mathbf{f}_6, \mathbf{f}_7\}$ has 274 features which is sufficient for efficient steganalysis. Later Kodovsky et al. [24] extended feature set by including another 274 features (i.e., $F_c = \{\mathbf{f}^c_1, \mathbf{f}^c_2, \mathbf{f}^c_3, \mathbf{f}^c_4, \mathbf{f}^c_5, \mathbf{f}^c_6, \mathbf{f}^c_7\}$) extracted from the cropped bitmap version of the examined JPEG image. Thus, resulted feature set $F = \{F_o, F_c\} = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, \mathbf{f}_4, \mathbf{f}_5, \mathbf{f}_6, \mathbf{f}_7, \mathbf{f}^c_1, \mathbf{f}^c_2, \mathbf{f}^c_3, \mathbf{f}^c_4, \mathbf{f}^c_5, \mathbf{f}^c_6, \mathbf{f}^c_7\}$ has 548 features. The image is cropped by 8 pixels in both vertical and horizontal directions. JPEG compression divides bitmap image into 8 by 8 blocks and processes each block separately. Thus, each JPEG block of the cropped image has four 4 by 4 sub-blocks from four neighboring noncropped JPEG blocks. Such design may amplify the artificial changes and may cause better detect-ability. For more details, refer to [26] and [24]. In the next section, we present a performance study of the proposed cognitive ensemble of ELM classifier for steganalysis.

Performance Study on Steganalysis

In this section, we study the steganalysis capability of the cognitive ensemble of ELM classifier, in comparison with the best performing classifier available in the literature for this data set. First, we perform a study based on the constructive–destructive procedure to fix the number of neurons in the hidden layer of the ELM. Table 3 presents the training and testing efficiencies of an ELM classifier for different number of hidden neurons. Although a wide range of neurons was chosen, we have presented the results only for six different numbers of hidden neurons. From the table, it can be seen that the best testing efficiency was obtained when $L = 1{,}000$. Hence, 1,000 hidden neurons employing Gaussian activation function are used at the hidden layer of each ELM of the cognitive ensemble of ELM classifier used in our experiments.

Next, the performances of the different classifiers, namely ensemble classifier proposed by Kodovsky et al. [24], single ELM, and the cognitive ensemble of ELM are studied in the steganalysis data set using 75 % of the total samples in the training and the remaining 25 % of the samples in testing. To enable fair comparison, 7 trials were conducted, in each of which 75 % of the samples are randomly chosen for training and the remaining 25 % is used to test the trained classifier. The performance results of the 7 random trials of all the three classifiers are presented in Table 4. From the table, it can be observed that the cognitive ensemble of ELM classifier outperforms the ensemble classifier [24] and the single ELM classifier in all the 7 trials. Thus, it can be concluded that the cognitive ensemble of ELM classifier is capable of performing steganalysis much better than the single ELM and ensemble classifiers [24].

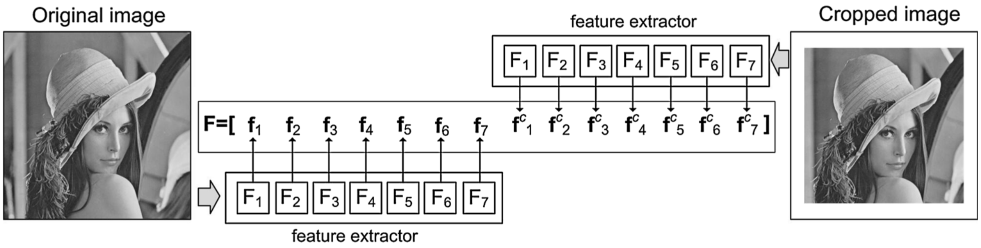

Fig. 2 Features set

Table 3 Training and testing efficiency for various number of hidden neurons *L*

Number of hidden neurons (*L*)	300	500	700	1,000	1,300	1,500
Training efficiency (%)	64.13	68.4	71.02	**71.1**	71.86	71.42
Testing efficiency (%)	60.95	63.25	63.29	**63.5**	62.85	62.77

Bold values indicate cases where the proposed technique has improvement over the existing methods

Table 4 Testing efficiency for tested classifiers and 7 different splits

Binary classifier	Testing efficiency (%)						
	Split's index						
	1	2	3	4	5	6	7
SVM	62.29	61.53	60.71	61.74	62.56	63.49	63.98
Ensemble classifier [24]	63.15	63.30	63.50	62.68	63.77	64.48	65.41
ELM	62.14	63.15	62.35	61.72	62.32	63.74	64.56
Cognitive ensemble of ELM	**64.53**	**65.63**	**64.15**	**64.97**	**65.46**	**65.63**	**66.19**

Bold values indicate cases where the proposed technique has improvement over the existing methods

In our experiments, we found 48 classifiers achieve better performance. Beyond 48, there is no much improvement in results. We have conducted sevenfold validation. We found that the average performance is 65.22 % in terms of testing accuracy; standard deviation is 0.71.

Conclusions

In this paper, we have presented a risk-sensitive cognitive ensemble of ELM classifier for JPEG steganalysis. We have trained several ELM classifiers using the 548-dimensional merged feature set, and the cognitive ensemble of ELM classifier is then constructed by obtaining the weighted sum of the outputs of the individual ELM classifiers. The cognition in the classifier estimates the weight of individual ELM classifier through the gradient descent update based on RHSL. Thus, the cognitive ensemble of ELM classifier positively exploits the effect of randomness in each classifier to obtain the best results. The performance of the cognitive ensemble ELM is evaluated on 5 benchmark binary classification data sets from the UCI machine learning repository and a practical steganalysis data set. In all these problems, the performance of the proposed classifier is compared to that of a single ELM and SVM. In addition, the performance of the classifier in solving steganalysis is also compared with the best performing classifier available in the literature for this data set. Performance results obtained from the study show the superior classification ability of the cognitive ensemble of ELM classifier.

Acknowledgments This work was supported by Catholic University of Korea, Research Funds 2013, National Research Foundation of Korea, Grant #2011-0013695.

References

1. Avcibus I, Kharrazi M, Memon ND, Sankur B. Image steganalysis with binary similarity measures. In: Proceedings of the 2002 international conference on image processing (ICIP 2002), vol 3, 2002. p. 645–648.
2. Cambria E, Hussain A. Sentic album: content-, concept-, and context-based online personal photo management system. Cogn Comput. 2012;4(4):477–96.
3. Cambria E, Hussain A. Sentic computing: techniques, tools, and applications, SpringerBriefs in cognitive computation. Dordrecht: Springer; 2012.
4. Chen C, Shi YQ. JPEG image steganalysis utilizing both intrablock and interblock correlations. In: Proceedings of the IEEE international symposium on circuits and systems; 2008. p. 3029–3032.
5. Chen B, Feng GR, Zhang XP, Li FY. Mixing high-dimensional features for JPEG steganalysis with ensemble classifier. Signal Image Video Process. doi:10.1007/s11760-012-0380-7
6. Cheng C, Tay WP, Huang GB. Extreme learning machines for intrusion detection. The 2012 International Joint Conference on Neural Networks (IJCNN) 2012;1(1):1–8.
7. Decherchi S, Gastaldo P, Zunino R, Cambria E, Redi J. Circular-elm for the reduced-reference assessment of perceived image quality. Neurocomputing. 2013;102:7889.
8. Farid H, Siwei L. Detecting hidden messages using higher-order statistics and support vector machines. In: Proceedings of the 5th International Workshop Information Hiding, Lecture Notes in Computer Science, vol 2578. New York: Springer; 2002. p. 340–354.

9. Feng GR, Huang GB, Lin QP, Gay R. Error minimized extreme learning machine with growth of hidden nodes and incremental learning. IEEE Trans Neural Netw. 2009;20(8):1352–7.
10. Fridrich J. Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes. Lect Notes Comput Sci. 2004;3200:67–81.
11. Goljan M, Fridrich J, Holotyak T. New blind steganalysis and its implications. In: Proceedings of the SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII, vol 6072, 2006. p. 1–13.
12. Grassi M, Cambria E, Hussain A, Piazza F. Sentic web: a new paradigm for managing social media affective information. In: Cognitive Computation, vol 3, no. 3; 2011. p. 480–489.
13. Huang G-B. Learning capability and storage capacity of two-hidden layer feedforward networks. IEEE Trans Neural Netw. 2003;14(2):274–81.
14. Huang G-B, Zhu QY, Siew CK. Extreme learning machine: a new learning scheme of feedforward neural network. In: Proceedings of International Joint Conference on Neural Networks (IJCNN2004), vol 2, 2004. p. 985–990.
15. Huang G-B, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing. 2006;70(1–3):985–90.
16. Huang G-B, Zhu QY, Siew CK. Real-time learning capability of neural networks. IEEE Trans Neural Netw. 2006;17(4):863–78.
17. Huang G-B, Wang DH, Lan Y. Universal approximation using incremental constructive feedforward network with random hidden nodes. IEEE Trans Neural Netw. 2011;17(4):879–92.
18. Huang G-B, Zhu QY, Siew CK. Extreme learning machines: a survey. Int J Mach Learn Cybern. 2011;2(2):107–22.
19. Huang G-B, Zhou H, Ding X, Zhang R. Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern Part B: Cybern. 2012;42(2):513–29.
20. Huang G.-B. , An Insight into Extreme Learning Machines: Random Neurons, Random Features and Kernels. Cogn Comput. (in press) 2014. doi:10.1007/s12559-014-9255-2.
21. Kazakov D. The self-cognisant robot. Cogn Comput. 2013;4(3): 347–53.
22. Kodovsky J., Fridrich J., Dittman J. , Craver S., Fridrich J., Calibration revisited. In: Proceedings of the 11th ACM Multimedia & Security Workshop, Princeton, NJ; 2009. p. 63–74.
23. Kodovsky J, Penvy T, Fririch J, Memon ND, Delp EJ, Wong PW, Dittmann J. Modern steganalysis can detect YASS. In: Proceedings of the SPIE, electronic imaging, security and forensics of multimedia XII, San Jose, CA, vol 7541; 2010. p. 02-01–02-11.
24. Kodovsky J, Fridrich J, Holub V. Ensemble classifiers for steganalysis of digital media. IEEE Trans Inf Secur Forensics. 2012;7(2):432–44.
25. Pereira F, Gordon G. The support vector decomposition machine. In: Proceedings of the 23rd international conference on machine learning (ICML 2006), Pittsburgh, PA; 2006. p. 689–696.
26. Pevny T, Fridrich J. Merging Markov and DCT features for multi-class JPEG steganalysis. In: Proceedings of SPIE, vol 6505, San Jose, CA; 2007. p. 311–314.
27. Rong HJ, Huang GB, Sundararaajan N, Saratchandran P. Online sequential fuzzy extreme learning machine for function approximation and classification problems. IEEE Trans Syst Man Cybern Part B: Cybern. 2009;39(4):1067–77.
28. Rong HJ, Suresh S, Zhao GS. Stable indirect adaptive neural controller for a class of nonlinear system. Neurocomputing. 2011;74(16):2582–90.
29. Saraswathi S, Sundaram S, Sundararajan N, Zimmermann M, Nilsen-Hamilton M. Icga-pso-elm approach for accurate multiclass cancer classification resulting in reduced gene sets in which genes encoding secreted proteins are highly represented. IEEE/ACM Trans Comput Biol Bioinform. 2011;8(2):452–63.
30. Suresh S, Sundararajan N, Saratchandran P. Risk-sensitive loss functions for sparse multi-category classification problems. Inf Sci. 2010;178(12):2621–38.
31. Suresh S, Venkatesh Babu R, Kim HJ. No-reference image quality assessment using modified extreme learning machine classifier. Appl Soft Comput. 2006;9(2):541–52.
32. Suresh S, Babu V, Sundararajan N. Image quality measurement using sparse extreme learning machine classifier. In: 9th International conference on control, automation, robotics and vision, 2006, ICARCV'06, art. no. 4150396, 2006.
33. Suresh S, Omkar SN, Mani V, Guru Prakash TN. Lift coefficient prediction at high angle of attack using recurrent neural network. Aerosp Sci Technol. 2003;7(8):595–602.
34. Savitha R, Suresh S, Sundararajan N. Fast learning fully complex-valued classifiers for real-valued classification problems. Lect Notes Comput Sci. 2011;6675(part 1):602–9.
35. Savitha R, Suresh S, Sundararajan N. A fast learning complex-valued neural classifier for real-valued classification problems. In: Proceedings of the international joint conference on neural networks, art. no. 6033508; 2011:2243–9.
36. Savitha R, Suresh S, Sundararajan N. Fast learning circular complex-valued extreme learning machine (cc-elm) for real-valued classification problems. Inf Sci. 2012;187(1):277–90.
37. Savitha R, Suresh S, Kim HJ. A meta-cognitive learning algorithm for an extreme learning machine classifier. Cogn Comput. (in press) 2013. doi:10.1007/s12559-013-9223-2
38. Sun YJ, Yuan Y, Wang G. An os-elm based distributed ensemble classification framework in p2p networks. Neurocomputing. 2011;74(16):2438–43.
39. Taylor JG, Cutsuridis V, Hartley M, Althoefer K, Nanayakkara T, observational learning: basis, experimental results and models, and implications for robotics. Cogn Comput. (in Press), 2013.
40. Babu RV, Suresh S, Makur A. Online adaptive radial basis function networks for robust object tracking. Comput Vis Image Understand. 2010;114(3):297–310.
41. Blake C, Merz C. UCI repository of machine learning databases. Department of Information and Computer Sciences, University of California, Irvine; 1998. URL: http://archive.ics.uci.edu/ml/.
42. Shi YQ, Chen C, Chen W. Markov process based approach to effective attacking jpeg steganography. Lect Notes Comput Sci. 2006;4437:249–64.
43. Solanki K, Jacobsen N, Madhow U, Manjunath BS, Chandrasekaran S. Robust image-adaptive data hiding based on erasure and error correction. IEEE Trans Image Process. 2004;13(12): 1627–39.
44. Solanki K, Sakar A, Manjunath BS. Yass: yet another steganographic scheme that resists blind steganalysis. Lect Notes Comput Sci. 2007;4567:16–31.
45. Wang GR, Zhao Y, Wang D. A protein secondary structure prediction framework based on the extreme learning machine. Neurocomputing. 2008;72(1–3):262–8.
46. Wang QF, Cambria E, Liu CL, Hussain A. Common sense knowledge for handwritten chinese recognition. Cogn Comput. 2013;5(2):234–42.
47. Zhang R, Sachnev V, Kim HJ. Fast bch syndrome coding for steganography. Lect Notes Comput Sci. 2009;5806:48–58.
48. Zhang T. Statistical behavior and consistency of classification methods based on convex risk minimization. Ann Stat. 2004;32(1):56–85.