Neurocomputing 149 (2015) 340–353

Contents lists available at ScienceDirect

Neurocomputing

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

Discriminative graph regularized extreme learning machine and its application to face recognition

Yong Peng [a], Suhang Wang [b], Xianzhong Long [a], Bao-Liang Lu [a,c,*]

[a] Center for Brain-Like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[b] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, USA
[c] Key Laboratory of Shanghai Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 3 August 2013
Received in revised form
21 November 2013
Accepted 1 December 2013
Available online 8 September 2014

Keywords:
Extreme learning machine
Graph Laplacian
Manifold regularization
Face recognition

ABSTRACT

Extreme Learning Machine (ELM) has been proposed as a new algorithm for training single hidden layer feed forward neural networks. The main merit of ELM lies in the fact that the input weights as well as hidden layer bias are randomly generated and thus the output weights can be obtained analytically, which can overcome the drawbacks incurred by gradient-based training algorithms such as local optima, improper learning rate and low learning speed. Based on the consistency property of data, which enforces similar samples to share similar properties, we propose a discriminative graph regularized Extreme Learning Machine (GELM) for further enhancing its classification performance in this paper. In the proposed GELM model, the label information of training samples are used to construct an adjacent graph and correspondingly the graph regularization term is formulated to constrain the output weights to learn similar outputs for samples from the same class. The proposed GELM model also has a closed form solution as the standard ELM and thus the output weights can be obtained efficiently. Experiments on several widely used face databases show that our proposed GELM can achieve much performance gain over standard ELM and regularized ELM. Moreover, GELM also performs well when compared with the state-of-the-art classification methods for face recognition.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Extreme Learning Machine (ELM) is an emerging model proposed by Huang [1] as a least square based learning algorithm for single hidden layer neural networks (SLFNNs) [2–5]. In comparison with traditional neural networks which usually employ back propagation (BP) algorithm to train the connection weights, the tedious process of iterative parameter tuning is eliminated and the slow convergence and local minimum problems are avoided.

The consistency among ELM and SVM, least square support vector machine (LS-SVM) and proximal SVM was well studied and analyzed from the optimization point of view [5–7]. In [7], it was found that ELM can provide an unified solution for generalized SLFNNs, which include but not limit to neural network, support vector network and regularized network. That is to say, the feature mapping function can be any type of nonlinear piecewise function as in conventional ELM random nodes; or an unknown function to form a mercer's kernel as in SVMs and other kernel based algorithms.

Recently, much effort has been made on ELM from both theoretical and application aspects. Huang et al. showed that the universal approximation performance of SLFNNs can be implemented in an incremental method, which may simply choose hidden nodes at random and then adjust the output weights (I-ELM) [2]. An enhanced method for I-ELM (referred as EI-ELM) was proposed in [4]. I-ELM was proven to have the ability of approximating any target function in both the real and complex domains [8]. An error minimized ELM (EM-ELM) which can automatically determine the number of hidden nodes in generalized SLFNNs was proposed in [9]. Zong et al. applied ELM to relevance ranking and studied it as a learning-to-rank algorithm from the perspective of both pointwise and pairwise [10]. The impact of random weights between input and hidden layers was investigated in [11]. To alleviate the effect of outliers, robust ELM was proposed in [12]. Zhang et al. proposed a fuzzy ELM (FELM) in which the inputs with different fuzzy matrix can make different contributions to learn the output weights [13]. Shi et al. proposed the elastic net regularized ELM and put it to EEG based vigilance estimation [14]. Wang et al. proposed a parallelized ELM ensemble based on the Min-Max Modular network

* Corresponding author at: Key Laboratory of Shanghai Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.
E-mail address: bllu@sjtu.edu.cn (B.-L. Lu).

http://dx.doi.org/10.1016/j.neucom.2013.12.065
0925-2312/© 2014 Elsevier B.V. All rights reserved.

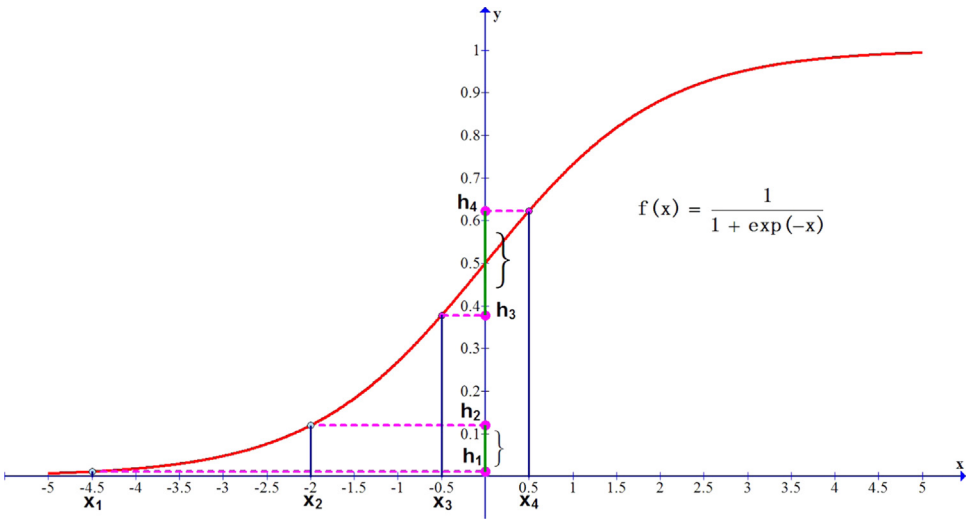


Fig. 1. An example to explain the distance information cannot be preserved by 'sigmoid' mapping.

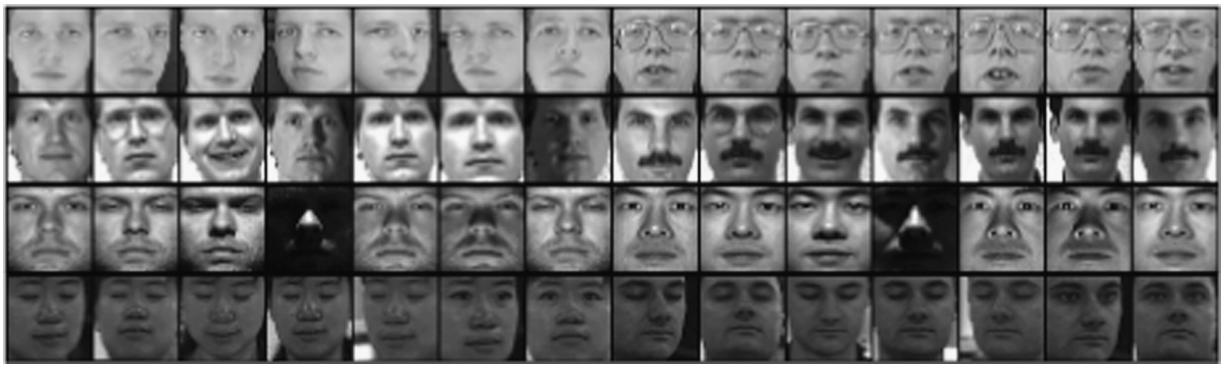


Fig. 2. The four rows show the fourteen image samples from the ORL, Yale, Extended Yale B and CMU PIE databases, respectively.

Table 1
Statistics of the four face databases.

Database	#samples	# classes	# samples/subject	dimension
ORL	400	40	10	32×32
Yale	165	15	11	32×32
Extended Yale B	2414	38	~64	32×32
CMU PIE	11,544	68	~170	32×32

(M^3-network) to meet the challenge of the so-called big data [15]. In addition, ELM has been put into diverse applications such as speaker recognition [16], neuroimage data classification [17], security assessment [18], data privacy [19], EEG and seizure detection [20], image quality assessment [21], image super-resolution [22], FPGA [23], face recognition [24], and human action recognition [25].

Recently, learning with local consistency of data has drawn much attention to improve the performance of existing machine learning models. In this paper, based on the idea that similar samples should share similar properties, we propose a discriminative graph regularized Extreme Learning Machine (GELM). In GELM, the constraint imposed on output weights enforces the output of samples from the same class to be similar. The constraint is formulated as a regularization term being added on the objective of basic ELM model, which also makes the output weights be solved analytically. We conduct experiments on four popular face databases to evaluate the performance of GELM. The experimental results demonstrate that GELM can obtain much better performance on most cases in comparison with basic ELM and state-of-the-art models.

The remainder of this paper is organized as follows. Section 2 describes the basic extreme learning machine model as well as its ℓ_2-norm regularized version. Section 3 introduces the proposed discriminative graph regularized ELM (GELM) including its model formulation and optimization method. Section 4 gives the detailed experiments to evaluate the efficiency of applying GELM to face recognition on several widely used data sets. Conclusion is given in Section 5.

2. Extreme Learning Machine

In this section, we review the Extreme Learning Machine algorithm in detail as the preliminary of our work.

Extreme Learning Machine proposed by Huang et al. [1] is an efficient and practical learning mechanism for the single layer feed forward neural networks.

Given a training data set, $L = \{(\mathbf{x}_i, \mathbf{t}_i) | \mathbf{x}_i \in \mathbb{R}^d, \mathbf{t}_i \in \mathbb{R}^m, i = 1, 2, \ldots, N\}$, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T$ and $\mathbf{t}_i = (t_{i1}, t_{i2}, \ldots, t_{im})^T$. An ELM with K

hidden nodes and activation function g is modeled as

$$\sum_{j=1}^{K}\beta_j g_j(\mathbf{x}_i)=\sum_{j=1}^{K}\beta_j g(\mathbf{w}_j\cdot\mathbf{x}_i+b_j)=\mathbf{o}_i,\quad i=1,\ldots,N, \tag{1}$$

where $\mathbf{w}_j=(w_{j1},w_{j2},\ldots,w_{jd})$ is the input weight vector connecting the jth hidden node with input nodes. $\beta_j=(\beta_{j1},\beta_{j2},\ldots,\beta_{jm})^T$ is the weight vector connecting the jth hidden node with the output nodes, b_j is the bias of the jth hidden node, and $\mathbf{o}_i=(o_{i1},o_{i2},\ldots,o_{im})^T$ is the network output corresponding to sample $\mathbf{x}_i$. Eq. (1) can be rewritten in matrix form as

$$\beta^T H=T, \tag{2}$$

where

$$H=\begin{bmatrix} g(\mathbf{w}_1\cdot\mathbf{x}_1+b_1) & \ldots & g(\mathbf{w}_1\cdot\mathbf{x}_N+b_1)\\ g(\mathbf{w}_2\cdot\mathbf{x}_1+b_2) & \ldots & g(\mathbf{w}_2\cdot\mathbf{x}_N+b_2)\\ \vdots & \vdots & \vdots\\ g(\mathbf{w}_K\cdot\mathbf{x}_1+b_K) & \ldots & g(\mathbf{w}_K\cdot\mathbf{x}_N+b_K)\end{bmatrix}_{K\times N} \tag{3}$$

$$\boldsymbol{\beta}=\begin{bmatrix}\beta_1^T\\ \beta_2^T\\ \vdots\\ \beta_K^T\end{bmatrix}_{K\times m}\quad\text{and}\quad T=[\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_N]_{m\times N}. \tag{4}$$

So the output weight of Eq. (2) can be estimated analytically by

$$\tilde{\beta}=\arg\min_{\beta}\|\beta^T H-T\|_2^2=H^{\dagger}T, \tag{5}$$

where $H^{\dagger}$ is the Moore-Penrose generalized inverse of H. If H is nonsingular, Eq. (5) can be written as

$$\tilde{\beta}=(HH^T)^{-1}HT^T. \tag{6}$$

In order to improve the stability and generalization ability of the traditional ELM, Huang et al. proposed the equality constrained optimization-based ELM [7]. In this method, the solution of ELM can be expressed as

$$\tilde{\beta}=\left(HH^T+\frac{I}{C}\right)^{-1}HT^T, \tag{7}$$

where C is a constant and I is the identity matrix. Let $\lambda=1/C$, Eq. (7) can be rewritten as

$$\tilde{\beta}=\left(HH^T+\lambda I\right)^{-1}HT^T. \tag{8}$$

The solution in Eq. (8) can be obtained by solving the following optimization problem:

$$\min_{\beta}\|\beta^T H-T\|_2^2+\lambda\|\beta\|_2^2, \tag{9}$$

where $\|\beta\|_2^2=\sum_{j=1}^{K}\|\beta_j\|_2^2$ is regarded as the regularization term and $\|\beta_j\|_2^2$ denotes the ℓ_2-norm of the vector β_j. Moreover, λ denotes the regularization parameter to balance the influence of error term and the model complexity. This is a general method to make the least square regression solution stable, which is called 'ridge regression' in statistics.

As a whole, training a single layer feed forward neural networks based on ELM algorithm can be summarized in Algorithm 1.

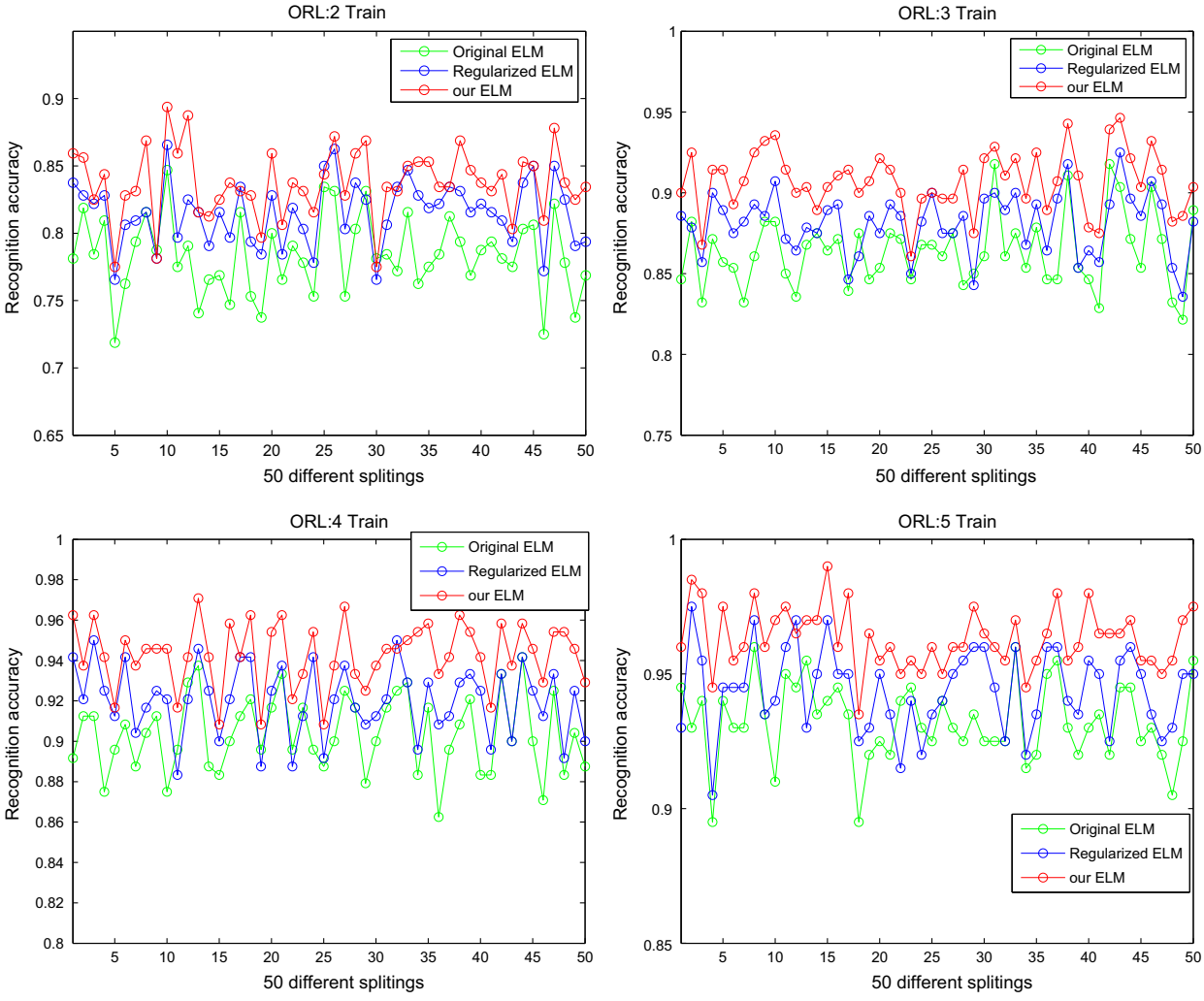


Fig. 3. Comparing GELM with conventional ELM and regularized ELM on ORL database.

Algorithm 1. Conventional Extreme Learning Machine.

Input: training set $\mathcal{N}=\{(\mathbf{x}_i,\mathbf{t}_i)|\mathbf{x}_i\in\mathbb{R}^d,\mathbf{t}_i\in\mathbb{R}^m,i=1,2,\ldots,N\}$, activation function g, number of hidden nodes K and regularization parameter λ;

Output: Output weight matrix β;

Randomly assign input weights $\mathbf{w}_j$ and biases $b_j, j=1,\ldots,K$;

Calculate the hidden layer output matrix H;

Calculate the output weight matrix $\tilde{\beta}$ according to Eq. (5) or Eq. (7).

3. Discriminative graph regularized ELM

3.1. Motivation of discriminative graph regularization

The motivation of discriminative graph regularized ELM lies in two folds:

• *The Local Consistency property can be used as side information for improving the performance of learning models.* Recently, various researchers have considered the case when data is drawn from sampling a probability distribution that has support on or near to a *submanifold* of the ambient space. The local consistency assumption usually means that nearby points (neighbors) should share similar properties, which emphasizes the importance of local geometrical structure in data set. Based on local consistency, many graph embedding (regularization) enhanced models were proposed by constructing a nearest neighbor graph based on some 'distance' measurement including local consistency Gaussian Mixture Model (LCGMM) [26,27], graph regularized Non-negative Matrix Factorization (GNMF) [28], and graph regularized Sparse Coding [29]. Thus our method can be viewed as one type of manifold learning, which aims at preserving the local structure during feature learning or classification. In other words, local consistency property can make the learned mapping function in ELM varies smoothly along the geodesics of the data manifold.

• *The distance information among samples are destroyed by nonlinear mapping in conventional ELM.* The activation functions used in neural networks are usually nonlinear like the 'sigmoid' function and 'gaussian' function. The nonlinear mapping enhances the feature extraction performance of neural network while destroying the local consistency contained in the data set. Fig. 1 shows an example to explain this phenomenon based on 'sigmoid' mapping. The distance between $\mathbf{x}_1$ and $\mathbf{x}_2$ is larger than the distance between $\mathbf{x}_3$ and $\mathbf{x}_4$ in the original data space. However, the distance between $\mathbf{h}_1$ and $\mathbf{h}_2$ is smaller than the distance between $\mathbf{h}_3$ and $\mathbf{h}_4$ in the hidden layer space. Therefore, the local consistency in unsupervised version based on distance information cannot be employed directly to construct the nearest neighbor

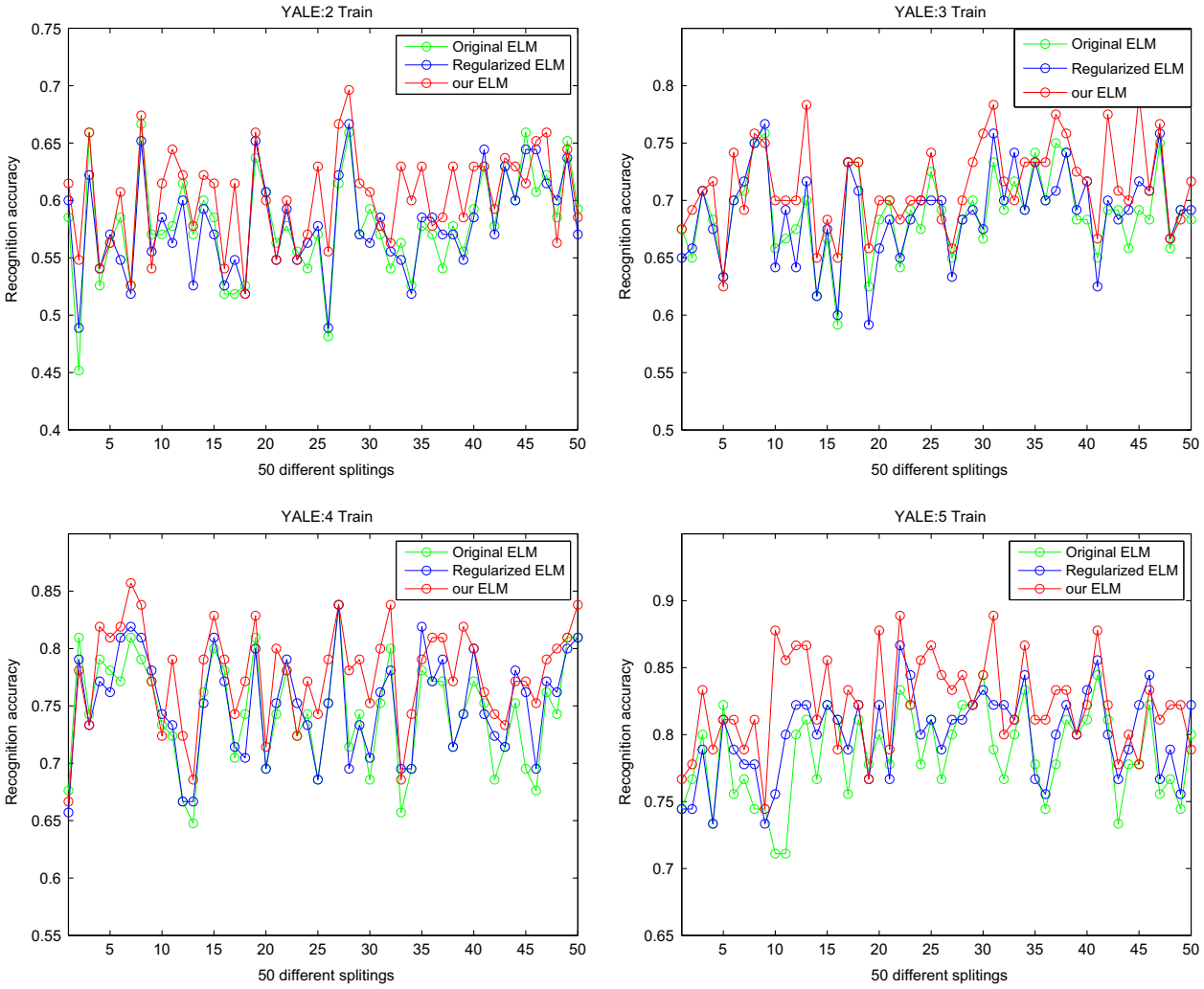


Fig. 4. Comparing GELM with conventional ELM and regularized ELM on Yale database.

graph here. As an alternate, we use the label information of the training samples to construct the adjacent matrix instead of the distance information.

3.2. The proposed GELM model

Suppose we have a data set with c classes and N samples in total. The tth class has N_t samples, $N_1+N_2+\cdots+N_c=N$. Similar to the discriminative analysis, we define the adjacent matrix W as follows:

$$W_{ij}=\begin{cases}1/N_t & \text{if both } \mathbf{h}_i \text{ and } \mathbf{h}_j \text{ belong to the } t\text{th class}\\ 0 & \text{otherwise}\end{cases} \tag{10}$$

where $\mathbf{h}_i=(g_1(\mathbf{x}_i),\ldots,g_K(\mathbf{x}_i))^T$ and $\mathbf{h}_j=(g_1(\mathbf{x}_j),\ldots,g_K(\mathbf{x}_j))^T$ are hidden layer representations for two input samples $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. Suppose a diagonal matrix D is defined, in which each of the entries is the column (or row, since W is symmetric) sums of W, $D_{ii}=\sum_j W_{ij}$. We can compute the graph Laplacian $L=D-W$ [30].

Let $\mathbf{y}_i$ and $\mathbf{y}_j$ be two vectors for $\mathbf{h}_i$ and $\mathbf{h}_j$ being mapped by output weight matrix β, respectively. Based on the idea that $\mathbf{y}_i$ and $\mathbf{y}_j$ should be similar to each other when $\mathbf{h}_i$ and $\mathbf{h}_j$ are from the same class, we need to minimize the following objective function:

$$\min \sum_{i,j}\|\mathbf{y}_i-\mathbf{y}_j\|_2^2 W_{ij}=\mathrm{Tr}(YLY^T), \tag{11}$$

where $Y=\beta^T H$ in Extreme Learning Machine setting.

By incorporating the graph regularization term into conventional ELM model, we can formulate the objective function of graph regularized Extreme Learning Machine as follows:

$$\min_{\beta}\|\beta^T H-T\|_2^2+\lambda_1\mathrm{Tr}(\beta^T HLH^T\beta)+\lambda_2\|\beta\|_2^2, \tag{12}$$

where $\mathrm{Tr}(\beta^T HLH^T\beta)$ is the graph regularization term, $\|\beta\|^2$ is the ℓ_2-norm regularization term, and λ_1 and λ_2 are regularization parameters to balance the impact of these two terms.

Set $F\triangleq\|\beta^T H-T\|_2^2+\lambda_1\mathrm{Tr}(\beta^T HLH^T\beta)+\lambda_2\|\beta\|_2^2$ and we cab obtain β by setting the differentiate of the objective function F with respect to β as zero as follows:

$$\begin{aligned}\frac{\partial F}{\partial\beta}&=\frac{\partial}{\partial\beta}\mathrm{Tr}[(\beta^T H-T)^T(\beta^T H-T)]+\lambda_1\mathrm{Tr}(\beta^T HLH^T\beta)+\lambda_2\|\beta\|_2^2\\&=\frac{\partial}{\partial\beta}\mathrm{Tr}(H^T\beta\beta^T H-H^T\beta T-T^T\beta^T H+T^T T)+\mathrm{Tr}(\beta^T HLH^T\beta)+\lambda_2\|\beta\|_2^2\\&=\frac{\partial}{\partial\beta}\mathrm{Tr}(H^T\beta\beta^T H-2H^T\beta T)+\mathrm{Tr}(\beta^T HLH^T\beta)+\lambda_2\|\beta\|_2^2\\&=(2HH^T\beta-2HT^T)+2\lambda_1 HLH^T\beta+2\lambda_2\beta\triangleq 0.\end{aligned} \tag{13}$$

As a result, we have

$$\beta=(HH^T+\lambda_1 HLH^T+\lambda_2 I)^{-1}HT^T. \tag{14}$$

The algorithm description of our proposed graph regularized Extreme Learning Machine is summarized in Algorithm 2.

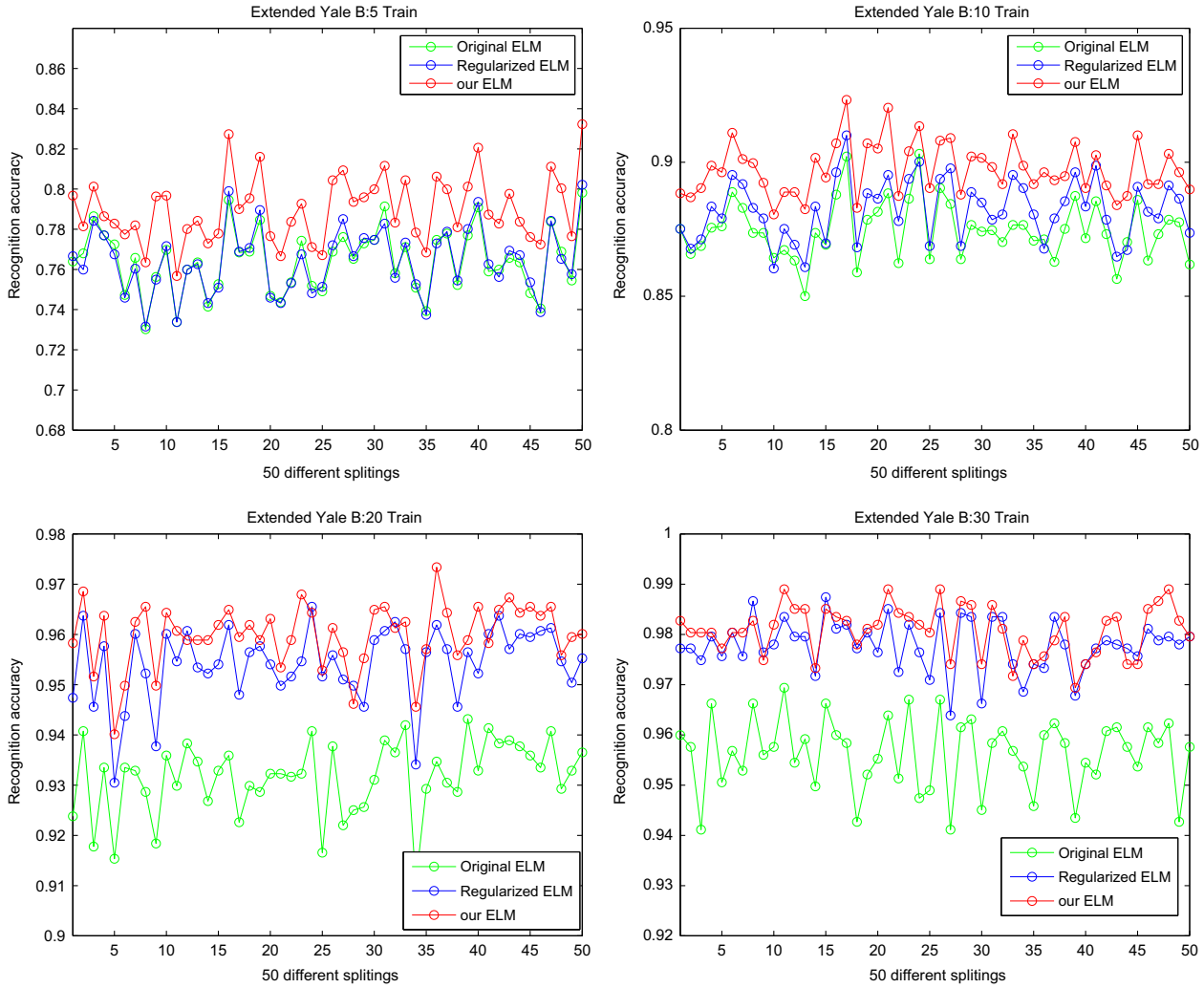


Fig. 5. Comparing GELM with conventional ELM and regularized ELM on Extended Yale B database.

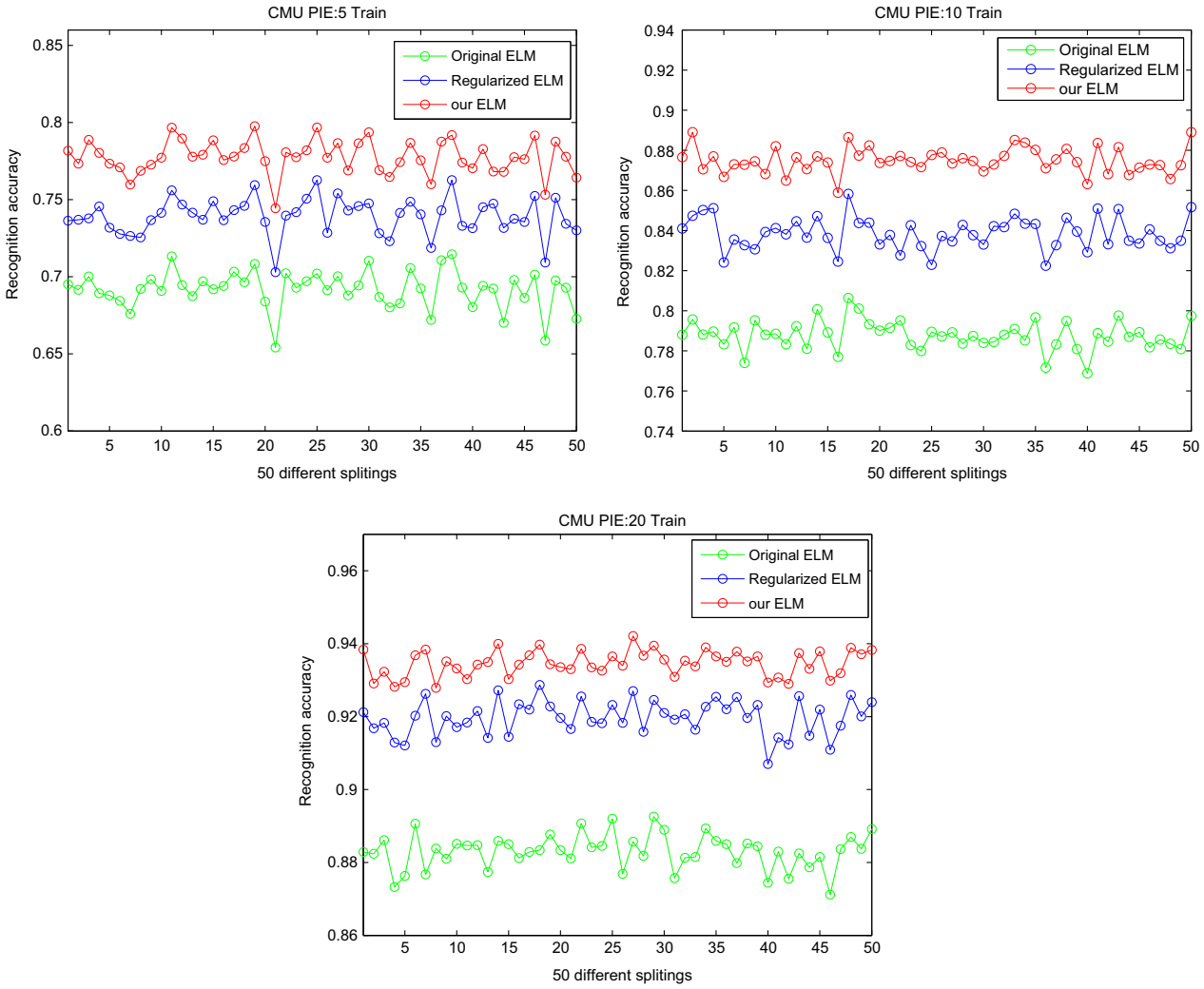


Fig. 6. Comparing GELM with conventional ELM and regularized ELM on CMU PIE database.

Algorithm 2. Graph Regularized Extreme Learning Machine.

Input: training set $\mathcal{N}=\{(\mathbf{x}_i,\mathbf{t}_i)|\mathbf{x}_i\in\mathbb{R}^d,\mathbf{t}_i\in\mathbb{R}^m,i=1,2,\ldots,N\}$, activation function g, number of hidden nodes K and regularization parameter λ_1 and λ_2;

Output: Output Weight matrix β;

Randomly assign input weights $\mathbf{w}_j$ and biases $b_j, j=1,\ldots,K$;
Calculate the hidden layer output matrix H;
Calculate the Laplacian matrix L;
Calculate the output weight matrix β according to Eq. (14).

4. Experimental studies

In this section, we evaluate the performance of our proposed graph regularized ELM model for face recognition. For all the experiments below, the activation function of the hidden layer is the '`sigmoid`' function. The face recognition task is handled as a multi-class classification problem. We experiment GELM on face recognition from two aspects: (1) comparing the proposed ELM model with the conventional ELM and regularized ELM; (2) comparing the proposed GELM model with the state-of-the-art classification algorithms for face recognition. For reproducing the experimental results described in this work, the source code will be available from http://bcmi.sjtu.edu.cn/ pengyong/papers/GELM.zip.

4.1. Experiment 1: Comparison with ELMs

Four publicly available face databases, ORL, Yale, extended Yale B and CMU PIE face databases, are used in this paper. The properties of these four data sets are briefly described below.

- *ORL Database.*[1] There are 40 subjects and each subject has 10 different face images in ORL database. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement).
- *Yale Database.*[2] It contains 165 gray scale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink.
- *Extended Yale Database*. It includes the Yale face database B [31] and the extended Yale face database [32]. The Yale face database B contains 5760 single light source images of 10 subjects each seen under 576 viewing conditions (9 poses × 64 illumination

[1] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.
[2] http://cvc.yale.edu/projects/yalefaces/yalefaces.html.

conditions). For every subject in a particular pose, an image with ambient (background) illumination was also captured. The extended Yale face database B contains 16,128 images of 28 human subjects under nine poses and 64 illumination conditions. The data format of this database is the same as the Yale face database B. For simplicity, a subset called Extended Yale

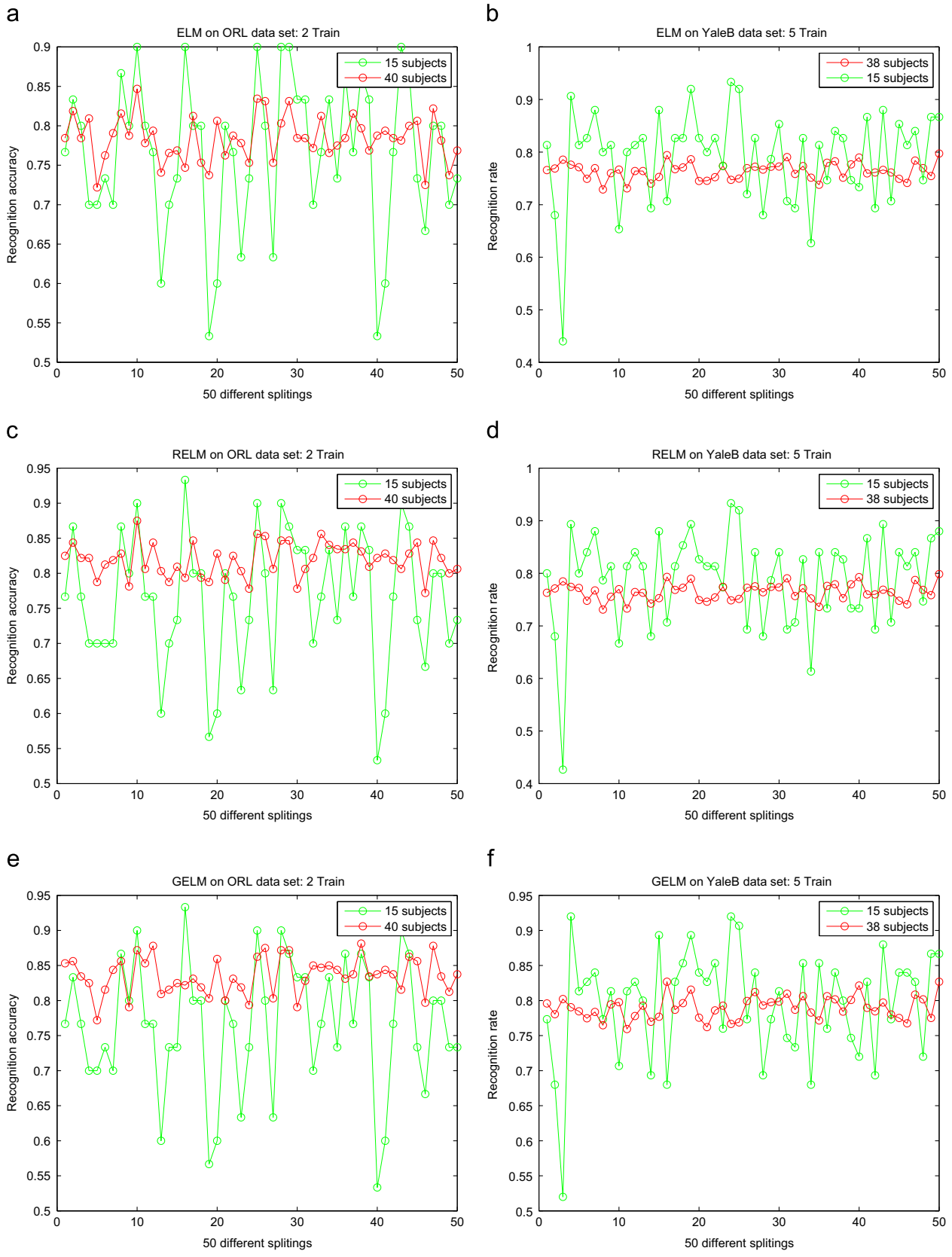


Fig. 7. Experimental results on ORL (1st column) and Extended Yale B (2nd column) data sets: first 15 subjects vs. all subjects. (a) ELM:ORL; (b) ELM:Extended Yale B; (c) RELM:ORL; (d) RELM:Extended Yale B; (e) GELM:ORL; (f) GELM:Extended Yale B.

Table 2
The face recognition results obtained by three different ELM models on ORL, Yale, Extended Yale B and CMU PIE databases (mean ± std-dev)%.

Algorithms	ORL			
	2 Train	3 Train	4 Train	5 Train
ELM	78.39 ± 2.87	86.36 ± 2.25	90.35 ± 1.89	93.22 ± 1.48
RELM	81.38 ± 2.41	88.11 ± 1.92	92.02 ± 1.76	94.41 ± 1.56
GELM	**84.17** ± 2.57	**90.74** ± 1.91	**94.29** ± 1.59	**96.34** ± 1.13
Algorithms	Yale			
	2 Train	3 Train	4 Train	5 Train
ELM	57.87 ± 4.51	68.85 ± 3.63	74.55 ± 4.64	78.56 ± 3.38
RELM	57.91 ± 4.23	68.90 ± 3.98	75.12 ± 4.53	79.96 ± 3.19
GELM	**60.31** ± 4.18	**71.33** ± 3.83	**77.79** ± 4.31	**82.36** ± 3.43
Algorithms	Extended Yale B			
	5 Train	10 Train	20 Train	30 Train
ELM	76.42 ± 1.59	87.47 ± 1.08	93.16 ± 0.74	95.62 ± 0.73
RELM	76.44 ± 1.65	88.23 ± 1.14	95.42 ± 0.74	97.79 ± 0.51
GELM	**79.00** ± 1.66	**89.75** ± 0.98	**95.99** ± 0.63	**98.09** ± 0.49
Algorithms	CMU PIE			
	5 Train	10 Train	20 Train	–
ELM	69.18 ± 1.24	78.78 ± 0.73	88.30 ± 0.48	–
RELM	73.92 ± 1.20	83.87 ± 0.81	91.98 ± 0.48	–
GELM	**77.77** ± 1.11	**87.50** ± 0.64	**93.47** ± 0.35	–

face database B was collected from these two databases, which contains 2414 face images of 38 subjects.

- *CMU PIE Database*. It contains 41,368 face images of 68 subjects, each subject under 13 different poses, 43 different illumination conditions and with 4 different expressions. We choose the five near frontal poses (C05, C07, C09, C27, C29) and use all 11,544 images under different illuminations and expressions where each person has 170 images except for a few bad images.

Fig. 2 shows some samples from these four face data sets, in which two subjects are randomly chosen from each database and each subject has 7 sample images. All the face images used in our experiments are manually aligned, cropped and resized to 32×32, with 256 gray levels per pixel. For the vector-based approaches, each face image is represented as a 1024-dimensional vector. The important statistics of these four databases are summarized in Table 1.

In this experiment, we compare our proposed GELM model with conventional ELM and ℓ_2-norm regularized ELM. For ORL and Yale databases, we randomly select $l = \{2, 3, 4, 5\}$ samples per subject for training and the rest for testing. For Extended Yale B and CMU PIE databases, we set $l = \{5, 10, 20, 30\}$ and $l = \{5, 10, 20\}$, respectively. This partition procedure is repeated 50 times to give a better estimation of recognition accuracy. Before classification, samples are projected to $N-1$ (N is the number of training samples) dimensional PCA subspace for these three models. The specific parameters setting for these three different ELM models will be described in Section 4.4.

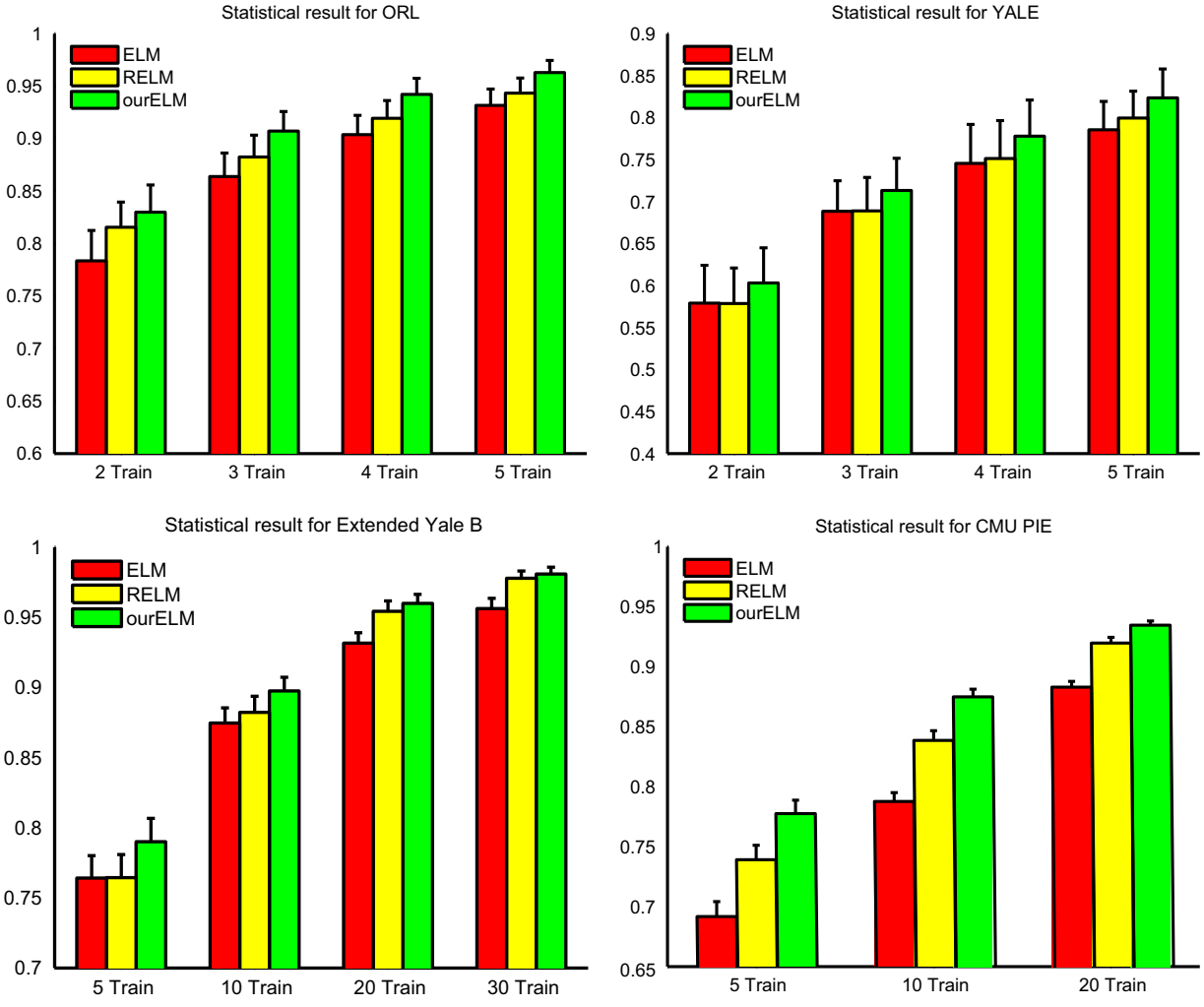


Fig. 8. Statistical results obtained by three different ELM models on ORL, Yale, Extended Yale B and CMU PIE databases.

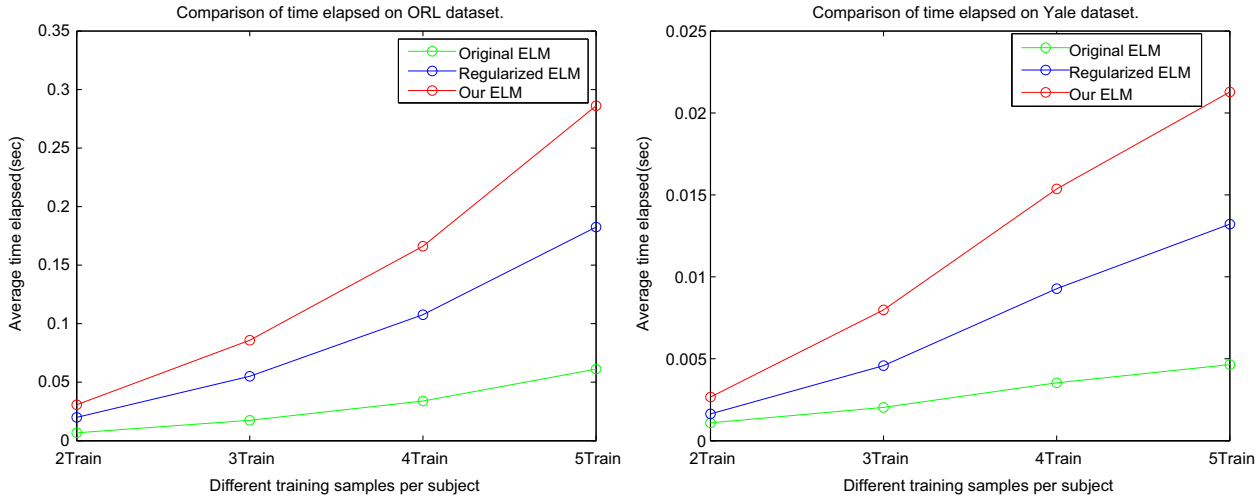


Fig. 9. Comparison of time elapsed of three ELM models on ORL and Yale data sets, respectively.

Table 3
The face recognition results of different classifiers on the Extended Yale B database.

Algorithms	#Dim=84 (%)	#Dim=150 (%)	#Dim=300 (%)
NN	85.8	90.0	91.6
LRC	94.5	95.1	95.9
SVM	94.9	96.4	97.0
SRC	95.5	96.8	97.9
CRC_RLS	95.0	96.3	97.9
GELM	**96.49**	**98.33**	**98.91**

*The accuracies of the first five algorithms are from [35].

Figs. 3–6 show the recognition results on the ORL, Yale, Extended Yale B and CMU PIE databases, respectively. It can be easily found that the graph regularized ELM achieves consistently better performance on all the databases than conventional ELM and regularized ELM. The main reason is that our proposed GELM simultaneously takes the classification error term and discriminative graph regularization term into consideration. Our experimental results demonstrate that the discriminative graph regularization term imposed on the output weights of ELM is effective for learning the consistency in the databases. The learned output weights can obtain the ability of mapping the samples in the same class to similar outputs. This smoothness property has been proved to be useful by manifold learning for discriminative tasks.

The conventional ELM is essentially a discriminative least square regression like classifier. The only difference between ELM classifier and least square regression classifier is that ELM nonlinearly maps the data from the original space to hidden layer space. Therefore, both of these two models may encounter the singular problem when calculating the inverse of normal equation. To deal with the singular problem, an ℓ_2-norm is usually incorporated. From our experimental results, we can observe that the regularized ELM outperforms the standard ELM because it is more stable in computing the inverse of normal equation after adding an ℓ_2-norm regularization.

From Figs. 3 to 6, we can observe that all ELM variants perform more stable on the other databases than Yale database. Here, the 'stability' means the variations of accuracies over 50 different splitings instead of the accuracy itself. For example, seen from these Figures, for ORL data set (5 training samples per subject), all ELM variants have accuracies mainly located in range 0.92–0.98; while for Yale data set (5 training samples per subject), the range is 0.74–0.88. For Extended Yale B and PIE data sets, the range intervals are much smaller. This variation is caused by two aspects: (1) the differences among these data sets, e.g., the nature of Yale is more complicated than that of ORL and there are more variations in Yale in terms of the facial expressions and the lighting illuminations; therefore, selecting different face images of each subject as training samples affect the accuracy a lot; (2) the number of classes in Yale is smaller than the remaining three data sets.

To support the second factor, we perform experiments on the ORL and Extended Yale B data sets by selecting the first 15 subjects and all the subject. The variations of ELM, RELM and GELM across 50 different splitings are shown in Fig. 7. We can find out that ELM variants get more stable accuracies for different splitings of training and test data when all the subjects are included.

Table 2 reports the mean accuracy and standard deviation over these 50 different partitions for each data set with different number of training samples per subject. It can be concluded that our GELM achieves the best performance as well as smallest variance.

For better visualizing the average performance of these three different ELM models, histograms including the accuracy as well as standard deviation are shown in Fig. 8.

4.2. Computing complexity analysis

Obviously, ELM obtains β based on Eq. (6) which computes the inverse of a $K\times K$ matrix HH^T (Here K is the number of hidden nodes). As in most cases, the number of hidden nodes K can be much smaller than the number of training samples N: $K \ll N$, and thus the computational cost reduces dramatically in comparison with LS-SVM and PSVM which needs to compute the inverse of a $N\times N$ matrix [7]. Similarly, the ℓ_2-norm regularized ELM and our proposed GELM have similar complexity as conventional ELM, which is based on computing the inverse of $K\times K$ matrices (HH^T+I/C and $HH^T+\lambda_1 HLH^T+\lambda_2 I$, respectively). Therefore, all of three ELMs have $O(K^3)$ complexity.

For quantitatively evaluating the time consuming for GELM, we compare GELM with conventional ELM and ℓ_2-norm regularized ELM on ORL and Yale data sets. The platform information is: Intel (R) Core(TM) i7-3770 CPU@3.40 GHz 16.0 GB, Windows 7 systems, Matlab 2013a. Fig. 9 shows the time elapsed of three ELM models on ORL and Yale data sets, respectively. For Extended Yale B and CMU PIE data sets, they have the similar results.

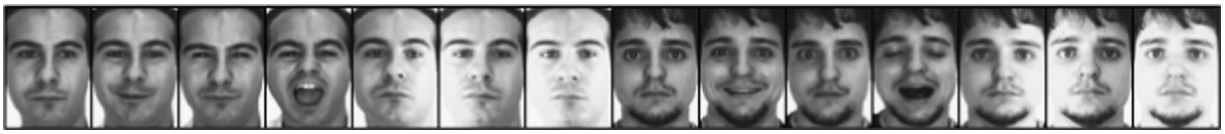


Fig. 10. The fourteen image samples from AR database.

Table 4
The face recognition results of different classifiers on the AR database.

Algorithms	#Dim=54 (%)	#Dim=120 (%)	#Dim=300 (%)
NN	68.0	70.1	71.3
LRC	71.0	75.4	76.0
SVM	69.4	74.5	75.4
SRC	83.3	89.5	93.3
CRC_RLS	80.5	90.0	93.7
GELM	**84.41**	**90.27**	**93.85**

*The accuracies of the first five algorithms are from [35].

4.3. Experiment 2: Comparison with the state-of-the-art classification algorithms

For further showing the efficiency of proposed graph regularized ELM, we compare GELM with the state-of-the-art classification methods used in face recognition, which are Nearest Neighbor Classifier (NN), Linear Regression Classifier (LRC) [33], Support Vector Machine (SVM), Sparse Representation based Classification (SRC) [34], Collaborative Representation based Regularized Least Square (CRC_RLS) [35]. The l1_*ls* [36] package is used to solve the ℓ_1-norm regularized minimization problem in SRC for its accuracy and efficiency. The respective parameters are set identically to the original papers. The number of hidden nodes are $10 \times numDim$ for GELM and the other related parameters setting is shown in Section 4.4.

For fair comparison, we use the same experimental paradigm as [35] on Extended Yale B and AR face databases.

- *Extended Yale B Database*: The Extended Yale B with 2414 frontal face images of 38 subjects are cropped and normalized to size 54×48 in this experiment. The database is randomly split into two halves. One half, which contains 32 images for each subject for training and the other half was used for testing. Table 3 shows the recognition rates versus feature dimension by NN, LRC, SVM, SRC, CRC_RLS and GELM. It can be seen that GELM performs better than all the other methods in three dimensions setting. When comparing with the very competitive SRC and CRC_RLS methods, GELM still can obtain 1%–2% accuracy improvement.
- *AR Database*: A subset (with only illumination and expression changes) that contains 50 male subjects and 50 female subjects was chosen from the AR data set [37] in this experiment. For each subject, the seven images from Session 1 were used for training, with the other seven images from Session 2 for testing. The images were cropped to 60×43. Fig. 10 shows the sample images from the AR database. The comparison of competing methods is given in Table 4. Generally, GELM can obtain the best accuracies in comparison with SRC and CRC_RLS.

These two experiments show that our proposed GELM can be used as an effective classifier in face recognition applications.

4.4. Parameters sensitivity analysis

There are two hyper-parameters in regularized ELM: the number of hidden nodes and the ℓ_2-norm regularization parameter C. And there are three hyper-parameters for our proposed GELM: the number of hidden nodes, the parameters λ_1 for graph regularization and λ_2 for ℓ_2-norm regularization. According to [7], the performance of ELM is not very sensitive to the number of hidden nodes (which is an open problem in ELM research). Therefore, we empirically set this parameter a near optimal value as 10 times the dimension of input data in all our experiments. For example, the number of hidden nodes will be 300 if the dimension of input data is 30.

We firstly give the parameter sensitivity analysis for regularized ELM. Fixing the number of hidden nodes as $10 \times numDim$, we experiment regularized ELM on these ORL, Yale, Extended Yale B and CMU PIE databases for analyzing the sensitivity to C. Fig. 11 shows the experimental results of regularized ELM to parameter C for each database with different number of training samples per subject. The vertical line shown in cyan in each subfigure of Fig. 10 gives the value for RELM used in previous experiments. Experimental results have shown that RELM performs well in both accuracy and stability using the given value though there is some randomness in ELM training (the input weights as well as the bias of hidden layer are randomly generated).

Similarly, we experiment our proposed GELM on CMU PIE database with different combinations of parameters λ_1 and λ_2 while fixing the number of hidden nodes as $10 \times numDim$. Fig. 12 shows the experimental results of GELM with different number of training samples per subject. As we can see, for each setting of training and testing data, there is a large flat area near the optimal value on the landscape. This means GELM is very stable with respect to the combination of parameters λ_1 and λ_2. For example, GELM achieves consistently good performance for $\lambda_1 = \{2^3, 2^4, \ldots, 2^{10}\}$ and $\lambda_2 = \{2^{-4}, 2^{-3}, \ldots, 2^0\}$ with $l=5$ and we can select parameter combination (λ_1, λ_2) from these candidate values. This sensitivity analysis means GELM encourages large λ_1 values on PIE data set, which further shows the importance of label consistency property reflected by the discriminative graph regularization term.

4.5. Decision boundary of GELM

The discriminative graph regularization term enforces the samples from the same class to have similar outputs. This will make the learned output weight matrix β have good clustering property. The similar techniques have been employed in many literatures including the manifold regularized discriminative NMF [38], spectral regression linear discriminative analysis [39] to make the same-class samples share similar properties.

In this subsection we give the decision boundary analysis of regularized ELM (RELM) and the discriminative graph regularized ELM (GELM) on banana data set [40], which was used in [7]. We randomly select 400 samples as training set (here we do not enforce each class to have equal number of training samples) and the rest 4900 samples as test set. We empirically set the number of hidden units as 1000 for both RELM and GELM. The parameter C in RELM is searched from the range $\{2^{-25}, 2^{-24}, \ldots, 2^{24}, 2^{25}\}$. The parameters λ_1 and λ_2 are searched from the range $\{2^{-25}, 2^{-24}, \ldots, 2^{24}, 2^{25}\}$ and $\{2^{-10}, 2^{-9}, \ldots, 2^{9}, 2^{10}\}$, respectively. Fifty trials are conducted for this problem, and we do not reshuffle the training and test data at each trial of simulation. The average testing accuracy and corresponding standard deviation for RELM and GELM are respectively $89.62 \pm 0.15\%$ and $89.78 \pm 0.05\%$ (using the sigmoid activation

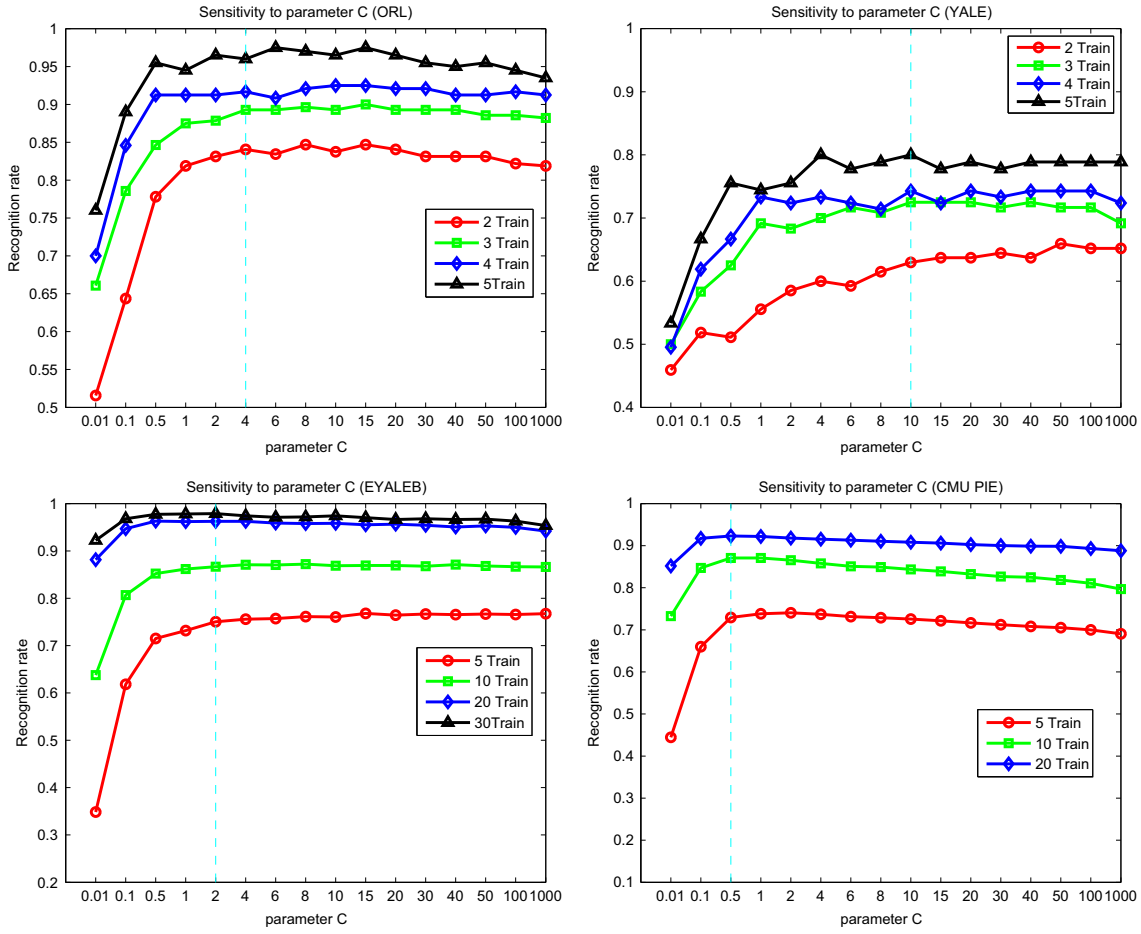


Fig. 11. Parameters sensitivity analysis of regularized ELM on ORL, Yale, Extended Yale B and CMU PIE databases.

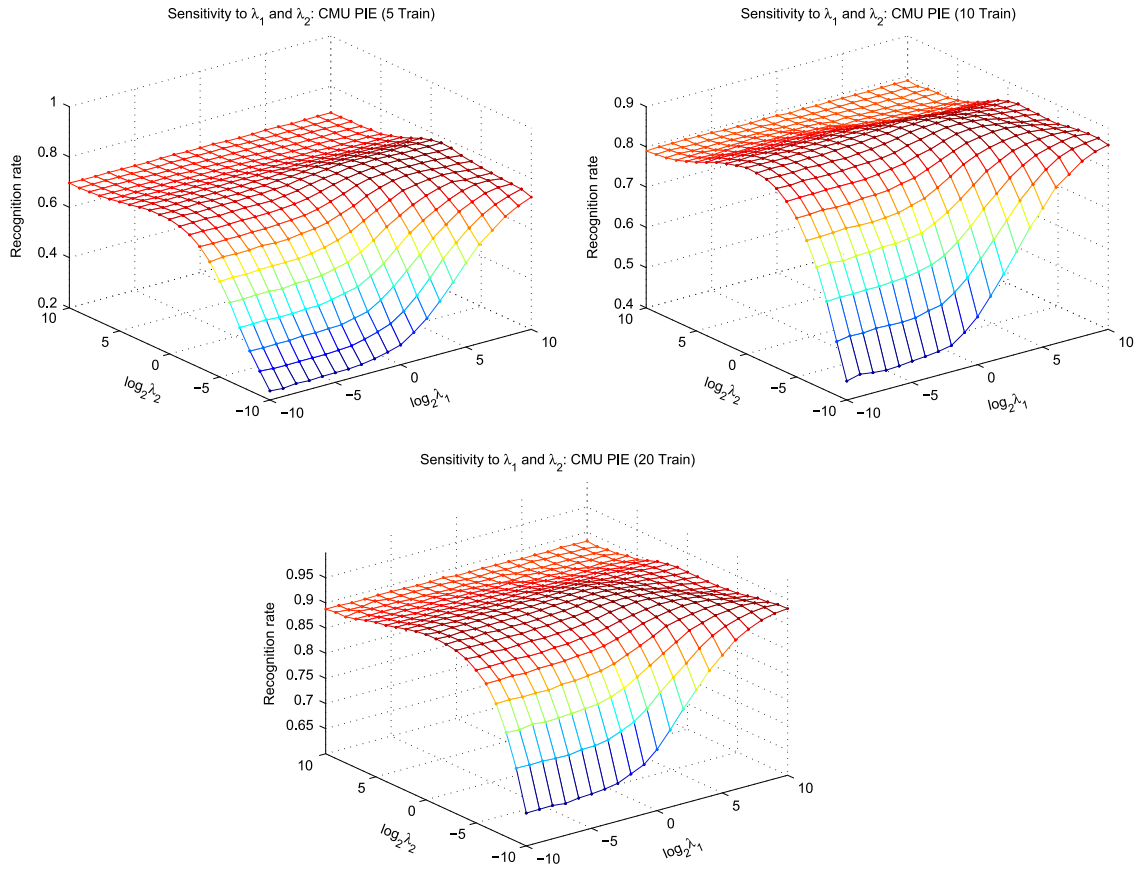


Fig. 12. Parameters sensitivity analysis of GELM on CMU PIE database.

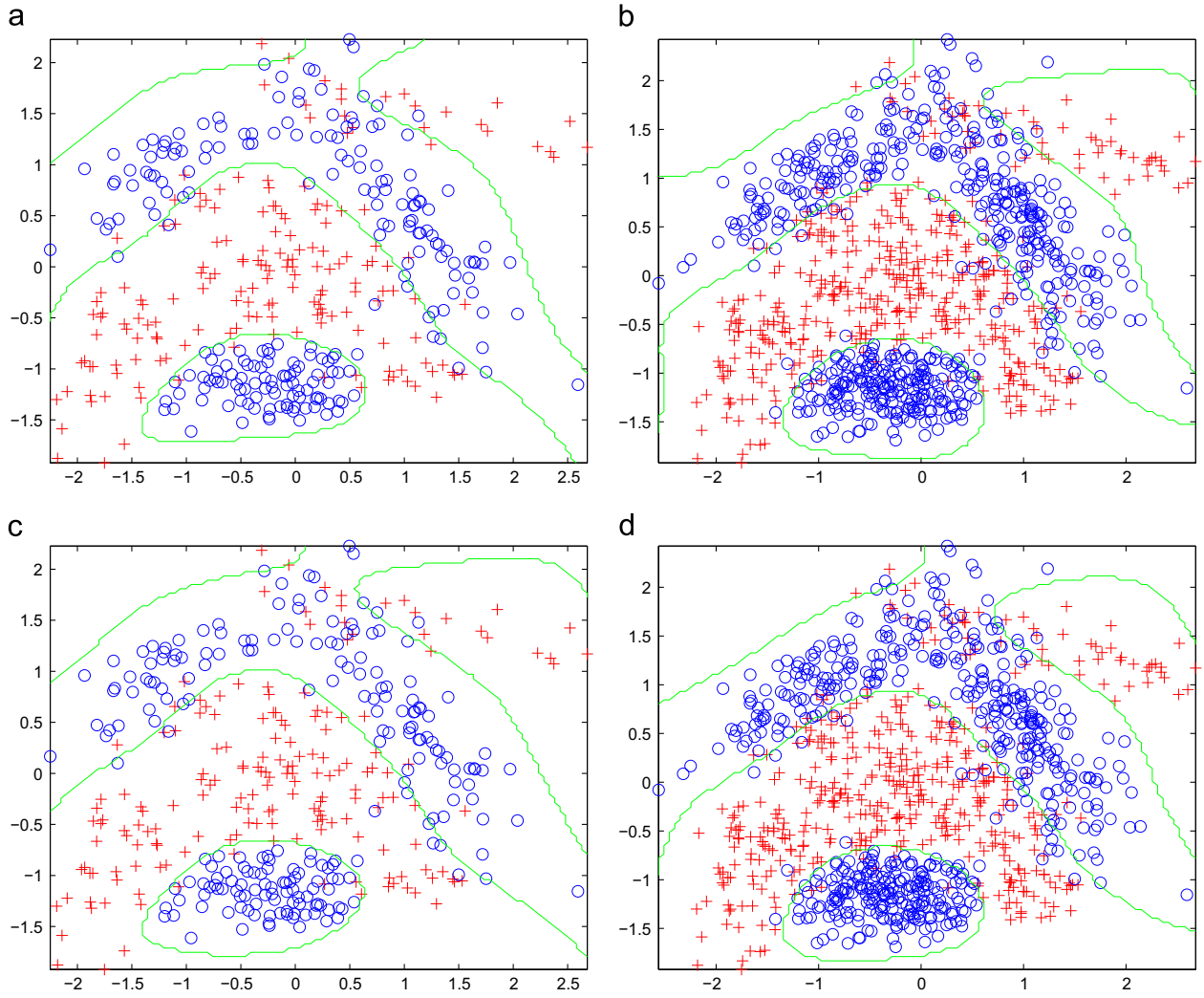


Fig. 13. Decision boundaries obtained by RELM and GELM respectively on banana data set. (a) RELM decision boundary (400); (b) RELM decision boundary (1000); (c) GELM decision boundary (400); (d) GELM decision boundary (1000).

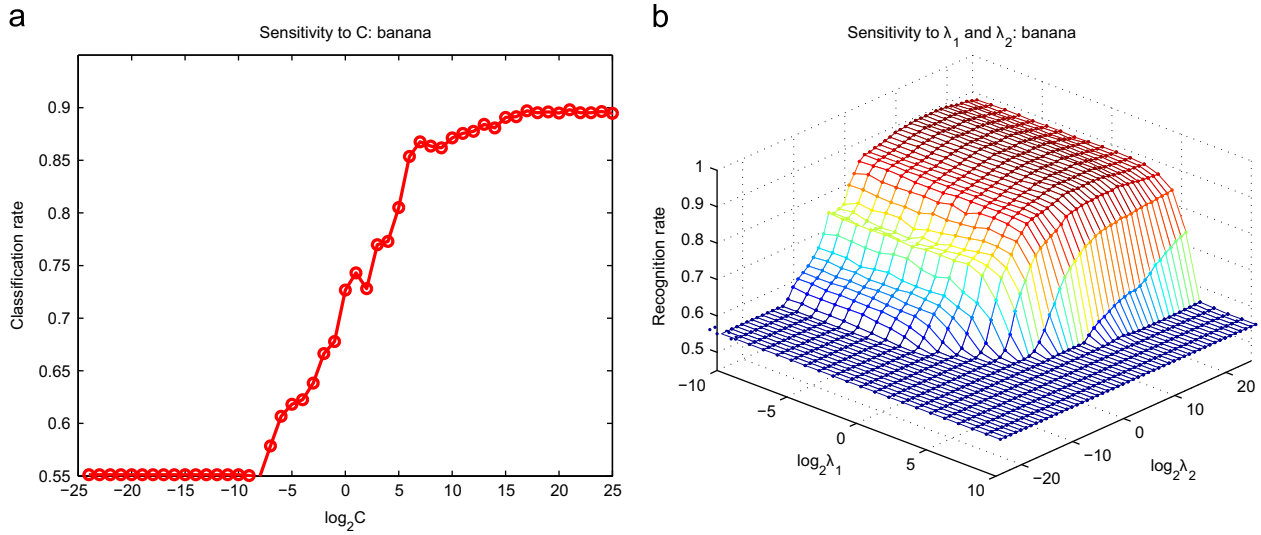


Fig. 14. Parameter sensitivity of RELM and GELM on banana data set. (a) Parameter sensitivity of RELM; (b) parameter sensitivity of GELM.

function). The corresponding decision boundaries obtained by RELM and GELM using the best tuned parameters are shown in Fig. 13(a) and (c). We can see that there exist some differences in the upper-right and bottom-right areas. However, it is not clear which is better than the other. Thus, we retrain RELM and GELM on 1000 randomly selected samples and the boundaries are shown in

Fig. 13(b) and (d) respectively. We can find that Fig. 13(c) is more similar to Fig. 13(b) and (d). Based on the idea that boundaries trained on 1000 samples are more faithful to the ideal one than that trained on 400 samples, we think the differences from Fig. 13(a)–(c) is beneficial.

The process for tuning parameters is shown in Fig. 14. The optimal C for RELM is 2^{18} and the parameter combination of GELM is $(\lambda_1, \lambda_2) = (2^{17}, 2^{-8})$. The boundaries above are obtained under such parameter settings.

5. Conclusions

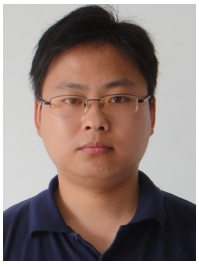
In this paper, we have proposed a discriminative graph regularized Extreme Learning Machine, termed as GELM, which takes the consistency property of data into consideration. We indicated that the widely used distance based consistency learning methods are no longer applicable in ELM, because the distance information among data points are not preserved after nonlinear mapping. Following the idea that the output weights of ELM should generate similar output for samples from the same class, we introduced a discriminative graph regularization term which encourages label consistency into ELM training. Our experimental results have demonstrated that our proposed GELM model possesses excellent performance in face recognition in comparison with conventional ELM, regularized ELM and several state-of-the-art classification methods.

Acknowledgment

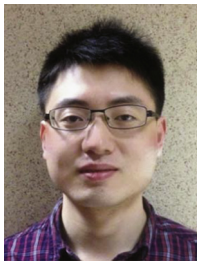
This work was supported partially by the National Basic Research Program of China (Grant No. 2013CB329401), the National Natural Science Foundation of China (Grant No. 61272248), the Science and Technology Commission of Shanghai Municipality (Grant No. 13511500200), and the European Union Seventh Framework Program (Grant No. 247619). The first author was supported by China Scholarship Council (Grant No. 201206230012).

References

[1] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.
[2] G.-B. Huang, L. Chen, C.-K. Siew, Universal approximation using incremental constructive feedforward networks with random hidden nodes, IEEE Trans. Neural Netw. 17 (4) (2006) 879–892.
[3] G.-B. Huang, L. Chen, Convex incremental extreme learning machine, Neurocomputing 70 (16) (2007) 3056–3062.
[4] G.-B. Huang, L. Chen, Enhanced random search based incremental extreme learning machine, Neurocomputing 71 (16) (2008) 3460–3468.
[5] G.-B. Huang, D.H. Wang, Y. Lan, Extreme learning machines: a survey, Int. J. Mach. Learn. Cybern. 2 (2) (2011) 107–122.
[6] G.-B. Huang, X. Ding, H. Zhou, Optimization method based extreme learning machine for classification, Neurocomputing 74 (1) (2010) 155–163.
[7] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Trans. Syst. Man Cybern. Part B: Cybern. 42 (2) (2012) 513–529.
[8] G.-B. Huang, M.-B. Li, L. Chen, C.-K. Siew, Incremental extreme learning machine with fully complex hidden nodes, Neurocomputing 71 (4) (2008) 576–583.
[9] G. Feng, G.-B. Huang, Q. Lin, R. Gay, Error minimized extreme learning machine with growth of hidden nodes and incremental learning, IEEE Trans. Neural Netw. 20 (8) (2009) 1352–1357.
[10] W. Zong, G.-B. Huang, Learning to rank with extreme learning machine, Neural Process. Lett. 39 (2) (2014) 155–166.
[11] R. Wang, S. Kwong, X. Wang, A study on random weights between input and hidden layers in extreme learning machine, Soft Comput. 16 (9) (2012) 1465–1475.
[12] P. Horata, S. Chiewchanwattana, K. Sunat, Robust extreme learning machine, Neurocomputing 102 (2012) 31–44.
[13] W. Zhang, H. Ji, Fuzzy extreme learning machine for classification, Electron. Lett. 49 (7) (2013) 448–450.
[14] L.-C. Shi, B.-L. Lu, EEG-based vigilance estimation using extreme learning machines, Neurocomputing 102 (2) (2013) 135–143.
[15] X.-L. Wang, Y.-Y. Chen, H. Zhao, B.-L. Lu, Parallelized extreme learning machine ensemble based on min-max modular network, Neurocomputing 128 (3) (2014) 31–41.
[16] Y. Lan, Z. Hu, Y.C. Soh, G.-B. Huang, An extreme learning machine approach for speaker recognition, Neural Comput. Appl. 22 (3–4) (2013) 417–425.
[17] M. Termenon, M. Gra na, A. Barrós-Loscertales, C. Ávila, Extreme learning machines for feature selection and classification of cocaine dependent patients on structural MRI data, Neural Process. Lett. 38 (3) (2013) 375–387.
[18] Y. Xu, Z.Y. Dong, J.H. Zhao, P. Zhang, K.P. Wong, A reliable intelligent system for real-time dynamic security assessment of power systems, IEEE Trans. Power Syst. 27 (3) (2012) 1253–1263.
[19] S. Samet, A. Miri, Privacy-preserving back-propagation and extreme learning machine algorithms, Data Knowl. Eng. 79–80 (2012) 40–61.
[20] Y. Song, J. Crowcroft, J. Zhang, Automatic epileptic seizure detection in eegs based on optimized sample entropy and extreme learning machine, J. Neurosci. Methods 210 (2) (2012) 132–146.
[21] S. Decherchi, P. Gastaldo, J. Redi, R. Zunino, E. Cambria, Circular-elm for the reduced-reference assessment of perceived image quality, Neurocomputing 102 (2012) 78–89.
[22] L. An, B. Bhanu, Image super-resolution by extreme learning machine, in: Proceeding of IEEE International Conference on Image Processing, 2012, pp. 2209–2212.
[23] S. Decherchi, P. Gastaldo, A. Leoncini, R. Zunino, Efficient digital implementation of extreme learning machines for classification, IEEE Trans. Circuits Syst. II: Express Briefs 59 (8) (2012) 496–500.
[24] K. Choi, K.-A. Toh, H. Byun, Incremental face recognition for large-scale social network services, Pattern Recognit. 45 (8) (2012) 2868–2883.
[25] R. Minhas, A.A. Mohammed, Q. Wu, Incremental learning in human action recognition based on snippets, IEEE Trans. Circuits Syst. Video Technol. 22 (11) (2012) 1529–1541.
[26] J. Liu, D. Cai, X. He, Gaussian mixture model with local consistency, in: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, vol. 10, 2010, pp. 512–517.
[27] X. He, D. Cai, Y. Shao, H. Bao, J. Han, Laplacian regularized gaussian mixture model for data clustering, IEEE Trans. Knowl. Data Eng. 23 (9) (2011) 1406–1418.
[28] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1548–1560.
[29] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Trans. Image Process. 20 (5) (2011) 1327–1336.
[30] F.R. Chung, Spectral graph theory, in: CBMS Regional Conference Series in Mathematics 92.
[31] A.S. Georghiades, P.N. Belhumeur, D.J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[32] K.-C. Lee, J. Ho, D.J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005) 684–698.
[33] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (11) (2010) 2106–2112.
[34] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[35] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: which helps face recognition? in: Proceeding of IEEE International Conference on Computer Vision, 2011, pp. 471–478.
[36] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, D. Gorinevsky, An interior-point method for large-scale ℓ_1-regularized least squares, IEEE J. Sel. Top. Signal Process. 1 (4) (2007) 606–617.
[37] A.M. Martinez, The AR Face Database, CVC Technical Report 24.
[38] N. Guan, D. Tao, Z. Luo, B. Yuan, Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent, IEEE Trans. Image Process. 20 (7) (2011) 2030–2048.
[39] D. Cai, X. He, J. Han, Spectral regression for efficient regularized subspace learning, in: Proceeding of IEEE International Conference on Computer Vision, 2007, pp. 1–8.
[40] C. Blake, C.J. Merz, UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California, Irvine.



Yong Peng received the B.S. degree from Hefei New Star Research Institute of Applied Technology, the M.S. degree from Graduate University of Chinese Academy of Sciences, both in computer science, in 2006 and 2010, respectively. Now he is working towards his Ph.D. degree in Shanghai Jiao Tong University. He was awarded by Presidential Scholarship, Chinese Academy of Sciences in 2009 and National Scholarship for Graduate Students, Ministry of Education in 2012. His research interests include machine learning, pattern recognition and evolutionary computation. He serves as the reviewer for Multimedia Tools and Applications (Springer) and Expert Systems with Applications (Elsevier).



Suhang Wang received his dual B.S. degree (one in electrical and computer engineering and the other in electrical engineering) from Shanghai Jiao Tong University and University of Michigan, Ann Arbor, respectively, in 2012. Now he is working towards his M.S. degree in digital signal processing in University of Michigan, Ann Arbor. His research interests include machine learning, data mining and computer vision.

Xianzhong Long received the B.S. degree from Henan Polytechnic University, the M.S. degree from Xihua University, both in computer science, in 2007 and 2009, respectively. Now he is a Ph.D. Candidate with Department of Computer Science and Engineering, Shanghai Jiao Tong University. His research interests include Pattern Recognition and machine learning.

Bao-Liang Lu received his B.S. degree from Qingdao University of Science and Technology, China in 1982, the M.S. degree from Northwestern Polytechnical University, China in 1989 and the Ph.D. degree from Kyoto University, Japan, in 1994. From 1982 to 1986, he was with the Qingdao University of Science and Technology. From April 1994 to March 1999, he was a Frontier Researcher at the Bio-Mimetic Control Research Center, the Institute of Physical and Chemical Research (RIKEN), Japan. From April 1999 to August 2002, he was a Research Scientist at the RIKEN Brain Science Institute. Since August 2002, he has been a full Professor at the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. His research interests include brain-like computing, neural networks, machine learning, pattern recognition, and brain–computer interface. He was the past president of the Asia Pacific Neural Network Assembly (APNNA) and the general Chair of ICONIP2011. He serves on the editorial board of Neural Networks Journal (Elsevier). He is a board member of APNNA and a senior member of the IEEE.