

# Partially Connected ELM for Fast and Effective Scene Classification

Dongzhe Wang, Rui Zhao and Kezhi Mao

**Abstract** Scene classification is often solved as a machine learning problem, where a classifier is first learned from training data, and class labels are then assigned to unlabelled testing data based on the outputs of the classifier. Generally, image descriptors are represented in high-dimensional space, where classifiers such as support vector machine (SVM) show good performance. However, SVM classifiers demand high computational power during model training. Extreme learning machine (ELM), whose synaptic weight matrix from the input layer to the hidden layer are randomly generated, has demonstrated superior computational efficiency. But the weights thus generated may not yield enough discriminative power for hidden layer nodes. Our recent study shows that the random mapping from the input layer to the hidden layer in ELM can be replaced by semi-random projection (SRP) to achieve a good balance between computational complexity and discriminative power of the hidden nodes. The application of SRP to ELM yields the so-called partially connected ELM (PC-ELM) algorithm. In this study, we apply PC-ELM to multi-class scene classification. Experimental results show that PC-ELM outperforms ELM in high-dimensional feature space at the cost of slightly higher computational complexity.

**Keywords** Scene classification · Extreme learning machine · Support vector machine · Partially connected extreme learning machine

---

D. Wang (✉) · R. Zhao · K. Mao  
School of Electrical and Electronic Engineering, Nanyang Technological University,  
50 Nanyang Avenue, Singapore, Singapore  
e-mail: DWANG015@e.ntu.edu.sg

R. Zhao  
e-mail: RZHAO001@e.ntu.edu.sg

K. Mao  
e-mail: EKZMAO@ntu.edu.sg

# 1 Introduction

In this paper, we consider the problem of scene classification, which is an important issue in many fields such as robotics and Unmanned Aerial Vehicle (UAV). Scene classification is often solved as a machine learning problem. A machine learning-based scene classification system consists of two main components, namely feature extraction and pattern classification. In computer vision studies, a number of feature extraction methods based on bag-of-features (BoF) [1] have been proposed (e.g., SIFT [2], GIST [3], etc.). In scene classification tasks, SVM [5, 6] classifiers are widely used.

One common but intractable problem in scene classification is that there are limited number of labeled image data for training while the dimensionality of image feature space is usually very high. Because of the so called “curse of dimensionality” [4], a low ratio of training sample size to feature dimension may lead to classifier over-fitting. To alleviate the over-fitting problem, support vector machine (SVM) attempts to find a discriminant function that maximizes the margin of separation, i.e., the shortest distance of training samples to the decision boundary. Although SVM has achieved high compatibility and robust performance in high-dimensional domains, it is often complained of its high computational complexity, especially for kernel SVMs.

Recently, a unified single-hidden-layer feedforward neural network namely extreme learning machine (ELM) [7] has been proposed. It has rapidly drawn attentions because of its much faster learning speed than SVMs, especially in high-dimensional space. This is because the synaptic weights from the input layer to the hidden layer in ELM are randomly generated, without involving any learning procedure. The synaptic weights from the input layer to the hidden layer plays the role of a linear mapping. Thus, the random weights in ELM can be interpreted as the matrix of a random projection (RP).

Inspired by the idea of random projection (RP), Zhao et al. [8] have proposed a Semi-Random Projection (SRP) framework, which takes the advantage of random feature sampling of RP, but employs learning mechanism in the determination of the mapping matrix. The SRP method was applied to ELM architecture to yield the so called “partially connected ELM” (PC-ELM). The supervised learning mechanism within SRP framework is able to find a latent space with large discriminative power and meanwhile to keep the computational complexity low because the semi-random projection is obtained through supervised learning in low dimensional space. Hence, PC-ELM can generate hidden layer nodes with more discriminative power than the original ELM. The main merit of PC-ELM is that it achieves a good balance between computational complexity and generalization performance.

In this work, we present a study of PC-ELM for the scene classification problem. Most image feature extraction methods generate high-dimensional image representations. Regardless of the consistent high learning speed of ELM and robust performance of SVM, the balance of classification performance and computational

simplicity is our major concern, especially when the feature space dimensionality is very high. In a sense, PC-ELM manages to achieve this balance by taking the merits of random feature sub-sampling of the SRP algorithm.

## 2 Semi-random Projection for Extreme Learning Machine

In this section, we first briefly review Extreme Learning Machine (ELM) and the connection between random projection (RP) and ELM. We then discuss the SRP method proposed in [8] and the SRP-based partially connected ELM (PC-ELM).

### 2.1 Extreme Learning Machine (ELM)

ELM is originally proposed as a generalized single-hidden-layer feedforward neural network whose hidden layer nodes need not be tuned [7]. Given a set of training data  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ , where  $\mathbf{x}_i \in \mathbb{R}^d$  with class label  $y_i \in \{1, 2, \dots, N_c\}$ , the output of ELM is given as follows:

$$f(\mathbf{x}) = \sum_{i=1}^M \beta_i h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\boldsymbol{\beta} \quad (1)$$

where  $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), \dots, h_M(\mathbf{x})]$  represents the output of the  $M$  hidden layer nodes with respect to the input sample  $\mathbf{x}$ , and  $\boldsymbol{\beta} = [\beta_1, \dots, \beta_M]^T$  denotes the weights connecting the hidden layer and the output layer. By minimizing the training error and the norm of vector  $\boldsymbol{\beta}$ , the output weights  $\boldsymbol{\beta}$  can be found as follows:

$$\boldsymbol{\beta} = \mathbf{H}^T \left( \frac{I}{C} + \mathbf{H}\mathbf{H}^T \right)^{-1} \mathbf{y} \quad (2)$$

where  $C$  is the cost parameter, and  $\mathbf{H}$  is the hidden-layer output matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) & \cdots & h_M(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ h_1(\mathbf{x}_N) & \cdots & h_M(\mathbf{x}_N) \end{bmatrix}. \quad (3)$$

Since the hidden nodes perform nonlinear piecewise continuous activation functions, the output of hidden layer is given as:

$$\mathbf{h}(\mathbf{x}) = [G(\mathbf{w}_1, b_1, \mathbf{x}), \dots, G(\mathbf{w}_M, b_M, \mathbf{x})] \quad (4)$$

where  $G$  denotes the activation function. One of the most popular activation functions is the sigmoid function:

$$G(\mathbf{w}_i, b_i, \mathbf{x}) = \frac{1}{1 + \exp(-z)} \quad (5)$$

$$z = \mathbf{w}_i \mathbf{x} + b_i \quad (6)$$

where  $\{(\mathbf{w}_i, b_i)\}_{i=1}^M$  are randomly sampled according to any continuous probability distribution.

ELM can be interpreted as the integration of three parts: linear random mapping; nonlinear activation; and linear model learning. The linear random mapping is a kind of random projection (RP). Similar discussions on the relationship between RP and ELM can be found in [12], which combines ELM with RP. Unlike the RP which aims to reduce dimensionality, the linear random mapping in ELM often maps data to even higher dimension. Usually the number of hidden nodes for RP in ELM is larger than the data dimension.

## 2.2 Random Projection (RP)

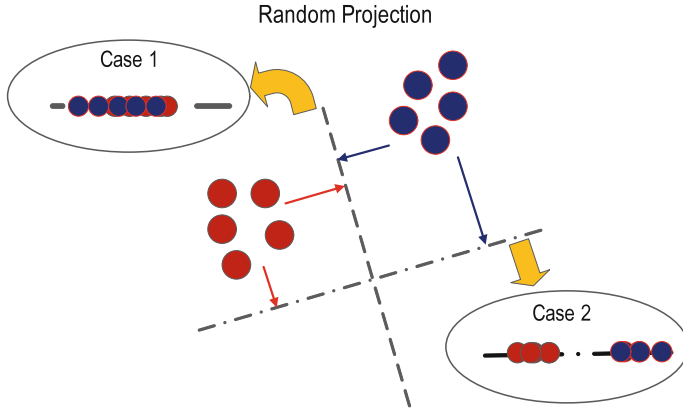
Given a set of training data denoted by matrix  $\mathbf{X} \in \mathbb{R}^{N \times d}$ , where  $N$  and  $d$  are the data dimension and the sample number of data. We can simply transform the data  $\mathbf{X}$  by mapping from the original space to a new space:

$$\mathbf{H} = \mathbf{XW} \quad (7)$$

where  $\mathbf{W} \in \mathbb{R}^{d \times r}$  is the linear transformation matrix, and  $\mathbf{H} \in \mathbb{R}^{N \times r}$  denotes the new data representation.

In [9, 10], RP has been proposed as a dimensionality reduction method. RP method is effective to address the computational complexity issue because the transformation matrix  $\mathbf{W}$  in RP is generated randomly. However, the randomly generated mapping matrix has some limitations. Without any supervised parameter learning or tuning, RP may not capture the information underlying the data.

The above mentioned limitation of RP is illustrated in Fig. 1. The figure shows samples from two classes in a two-dimensional feature space, which are depicted in red and blue, respectively. As the RP method project the data points onto randomly defined directions, two cases are demonstrated in Fig. 1. We provide an optimal case 2 (separable case), where the projection onto the dash dot line will result in good class separability. However, it is more likely to obtain a result as shown in case 1 (inseparable case) if the data points are projected on the dash line. After all, it is hard to yield ideal projections by solely relying on random projection without using any learning algorithms. The random operation shared by RP and linear random mapping in ELM inspires Zhao et al. [8] to introduce the semi-random projection (SRP) to ELM.



**Fig. 1** Illustrations of two possible cases of RP

### 2.3 Semi-random Projection (SRP)

To address the limitation of RP analysed above, Ref. [8] proposed a novel dimensionality transformation framework called Semi-Random Projection (SRP). In contrast to RP, the main idea of SRP is to learn a latent space where the task-related information can be preserved and meanwhile the learning speed would not drop much. The SRP consists of two parts: random sampling of features (random process) and the transformation matrix learning (non-random process).

Suppose we are given training data  $\mathbf{X} \in \mathbb{R}^{N \times d}$ , a subset of  $d_s$  ( $d_s$  is an integer close to  $\sqrt{d}$ ) features is randomly selected. Then the original data matrix is reduced to a sub-matrix  $\hat{\mathbf{X}}_i$  with a smaller size of  $N \times d_s$  in the  $i$ th iteration. The iterative transformation matrix  $\hat{\mathbf{w}}_i \in \mathbb{R}^{d_s \times r}$  maps data  $\hat{\mathbf{X}}_i$  into the  $r$ -dimensional space as follows:

$$\mathbf{h}_i = \hat{\mathbf{X}}_i \hat{\mathbf{w}}_i \quad (8)$$

where  $\mathbf{h}_i \in \mathbb{R}^{N \times r}$  is the data representation in  $i$ th iteration, in which column is the projection of a sample on the new dimension.  $\hat{\mathbf{w}}_i$  can be learned by using the linear discriminant analysis (LDA). The optimization solution for LDA can be derived as follows:

$$\mathbf{W}^* = \underset{\mathbf{W} \in \mathbb{R}^{d_s \times r}}{\operatorname{argmax}} \operatorname{Tr} \left( \frac{\mathbf{W}^T \mathbf{L} \mathbf{W}}{\mathbf{W}^T \mathbf{B} \mathbf{W}} \right) \quad (9)$$

where  $\operatorname{Tr}$  denotes the trace of a matrix. According to the fraction relation in Eq. (9),  $\mathbf{L}$  represents the quantity need to be enhanced and  $\mathbf{B}$  denotes that need to be suppress. In particular, matrices  $\mathbf{L}$  and  $\mathbf{B}$  in SRP can be calculated as:

$$\text{SRP} : \begin{cases} \mathbf{L} = \sum_{c=1}^{N_c} n_c (\hat{\mathbf{x}}^c - \hat{\mathbf{x}})(\hat{\mathbf{x}}^c - \hat{\mathbf{x}})^T \\ \mathbf{B} = \sum_{i=1}^N (\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^{c_i})(\hat{\mathbf{x}}^i - \hat{\mathbf{x}}^{c_i})^T + \eta \mathbf{I}_{d_s} \end{cases} \quad (10)$$

where  $c_i$  denotes the class label of the  $i$ th sample,  $\hat{\mathbf{x}}$  represents the mean vector of the all samples,  $\hat{\mathbf{x}}^c$  represents the mean vector of the  $c$ th class,  $N_c$  and  $n_c$  denote the number of classes and the number of samples in the  $c$ th class, respectively.  $\hat{\cdot}$  means the corresponding value calculated on the randomly selected feature subset from the original high dimension space. The second term in  $\mathbf{B}$  of Eq. (10) represents a regularization term.  $\mathbf{I}_{d_s}$  denotes an identity matrix with a size of  $d_s \times d_s$  and  $\eta$  represents the regularization weight.

In a  $N_c$ -class LDA model, the optimization problem in Eq. (9) can be solved by formulating the following generalized eigenvalue problem:

$$\mathbf{L}\boldsymbol{\varphi} = \lambda\mathbf{B}\boldsymbol{\varphi} \quad (11)$$

where  $\boldsymbol{\varphi} = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_r]$  and  $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_r]$  are the eigenvectors and their corresponding eigenvalues. Here,  $r$  can be an arbitrary integer from 1 to  $N_c - 1$ . The optimal transformation matrix can be calculated based on the eigenvectors  $\boldsymbol{\varphi}$  with the eigenvalues  $\lambda$  as  $\hat{\mathbf{w}}_i = [\sqrt{\lambda_1}\boldsymbol{\varphi}_1, \sqrt{\lambda_2}\boldsymbol{\varphi}_2, \dots, \sqrt{\lambda_{N_c}}\boldsymbol{\varphi}_r]$ .

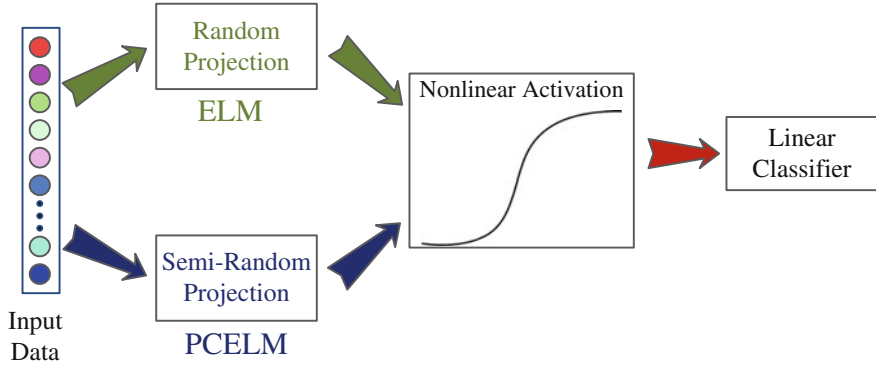
Next,  $\hat{\mathbf{w}}_i$  can be extended to the original data dimension  $\mathbf{w}_i \in \mathbb{R}^{d \times r}$  by interpolating zeros on the unselected positions in every column of  $\hat{\mathbf{w}}_i$  (recall that  $\hat{\mathbf{w}}_i$  is corresponding to the randomly selected feature subset of original  $d$  features). This process are repeated  $o$  times and a set of transformation sub-matrices will be obtained. Thus, the final transformation matrix  $\mathbf{W} \in \mathbb{R}^{d \times (r \cdot o)}$  can be denoted as:

$$\mathbf{W} = [\mathbf{w}_1 | \mathbf{w}_2 | \dots | \mathbf{w}_o] \quad (12)$$

The difference between RP and SRP is that learning is used in SRP for transformation matrix, while random assignment is used in RP.

## 2.4 Partially Connected ELM (PC-ELM)

Although ELM has achieved success in many applications, its performance in high dimensional space may not as great as in relatively low dimensional data. The limitation of randomised feature mapping from the input layer to the hidden layer has been demonstrated in Sect. 2.2. Inspired by the SRP as depicted in Sect. 2.3, Ref. [8] has implemented parameter-based model learning algorithm associated with linear random mapping in ELM. Specifically, the parameters  $\{(\mathbf{w}_i)\}_{i=1}^M$  are learned based



**Fig. 2** An overview of ELM and PC-ELM

on SRP instead of random assignment. The application of SRP to ELM yields the so-called partially connected ELM (PC-ELM) as every new dimension of SRP uses a subset of the original feature set. Figure 2 compares the PC-ELM with ELM.

The major difference between ELM and PC-ELM is that ELM assigns random values to the linear feature mapping, while PC-ELM learns  $\{(\mathbf{w}_i)\}_{i=1}^M$  using SRP. The degree of randomness is thereby suppressed in PC-ELM in contrast to ELM. According to the experimental results in [8], PC-ELM outperforms ELM at the cost of a slight higher computational complexity.

### 3 Experiments

In this section, we evaluate the application of PC-ELM for scene classification using the benchmark UIUC-Sport events dataset [13]. In order to verify the effectiveness of PC-ELM, we measure the experimental results and compare the classification performance with other state-of-the-art methods including ELM and rbf-SVM.

#### 3.1 Dataset and Experimental Setup

The UIUC-Sport dataset contains 8 sports event categories: rock climbing, badminton, bocce, croquet, polo, rowing, sailing, and snowboarding. The image number in each class ranges from 137 to 250, and there are 1579 images in total. It is noted that the difficulty levels of classification within a category are varying with the distance of the foreground objects. Figure 3 shows some example images in the dataset.

The event recognition task is an 8-class classification problem. Following the experiment setting of [13], 70 local images are randomly selected for training and we test on 60 images. In order to achieve statistically significant experimental results,



**Fig. 3** Example images for the UIUC-Sport dataset

we repeat 50 times of the training/ test data random split process and present the averaged results. For evaluation, we perform rbf-SVM, ELM and PC-ELM as the multi-class classifiers on the UIUC-Sports dataset. We measure the experimental results and compare the generalization performance of them in three state-of-the-art image representations including GIST, PHOW [15] (fast dense SIFT). We implement rbf-SVM using LIBSVM [14] with fixed parameter settings, where the hyperparameter  $C$  and  $\gamma$  is set to 2 and 0.1, respectively. In order to achieve good performance for ELM/ PC-ELM, we adjust their parameters for individual image representations. Experimental setup is as follows:

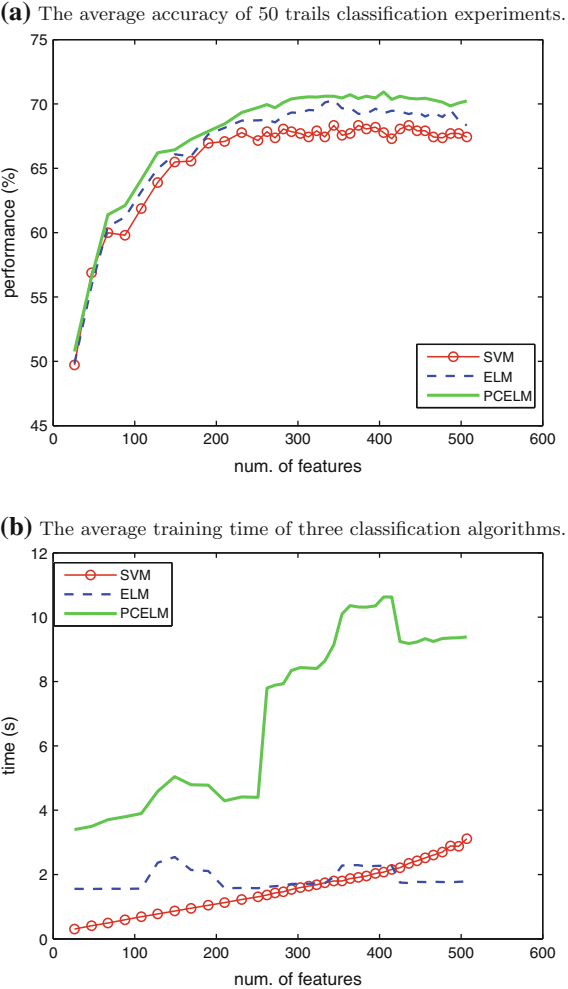
- **GIST descriptors.** We resize the images into  $256 \times 256$ ; the number of filters is 4; the number of orientations per scale is [8 8 8 8]. For ELM, the cost coefficient  $C$  and the number of hidden neurons are set as 1 and 2000, respectively. For PC-ELM, the penalty coefficient  $C$  is set to 1, and we define 4 sub-sections LDA (new projection dimension mentioned in Sect. 2.3) with 500 hidden neurons (iteration of feature random selection for each sub-section).
- **PHOW descriptors.** We perform the PHOW descriptors via VLFeat toolbox [16]. Image is resized to  $640 \times 480$  pixels; regular grids spacing [4 6 8 10]; SIFT descriptors of  $6 \times 6$  pixels; pyramid level  $L=3$ ; visual vocabulary size  $M=200$ . For ELM, the cost coefficient  $C$  and the number of hidden neurons are set as 0.01 and 4200, respectively. For PC-ELM, the penalty coefficient  $C$  is set to 0.01, and we define 7 sub-sections LDA with 600 hidden neurons for each sub-section.

### 3.2 Results

The first experiment uses the GIST descriptors. We obtain 512-dimension of features for each image samples. We show and compare the average classification performance of rbf-SVM, ELM, and PC-ELM in Fig. 4a, b. As shown in Fig. 4a, PC-ELM outperforms SVM and ELM. Note that the GIST image descriptors present relatively



**Fig. 4** Performance results of GIST. The  $x$ -axis indicates the number of engaged features applied for the experiments. The  $y$ -axis denotes the average test set classification accuracy results in (a). The  $y$ -axis indicates the training time results in (b)

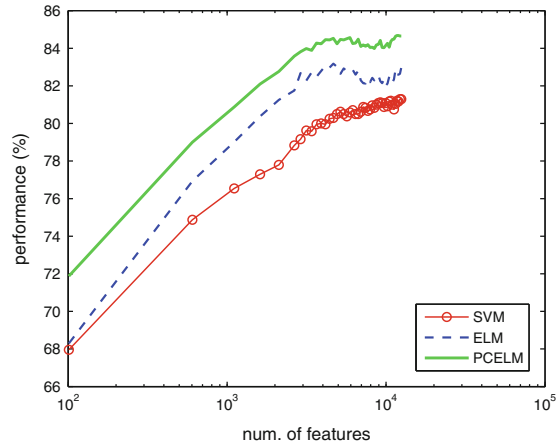


low-dimensional feature space, where the learning speed of PC-ELM is inferior to rbf-SVM and PC-ELM, as depicted in Fig. 4b.

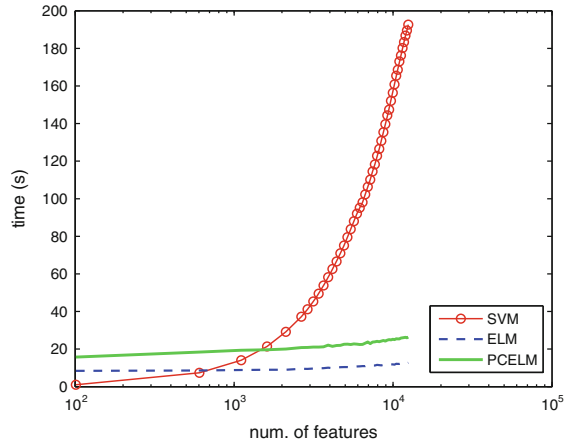
The second experiment uses the dense SIFT image descriptors (PHOW). We obtain 12,600-dimensional feature representation for each image. The second experiment is able to evaluate the performance of the three multi-class classifiers to high-dimensional domain. In Fig. 5a, b, we compare and analyze the average classification performance of rbf-SVM, ELM, and PC-ELM. We observe that PC-ELM produces robust performance and achieves remarkable improvements over ELM and rbf-SVM in Fig. 5a. Meanwhile, the difference of learning speed between the original ELM and PC-ELM is not much. Both ELM and PC-ELM run much faster than the SVM method as the feature dimensionality increases, as illustrated in Fig. 5b.

**Fig. 5** Performance results of PHOW. The  $x$ -axis indicates the number of engaged features applied for the experiments. The  $y$ -axis denotes the average test set classification accuracy results in (a). In b, the  $y$ -axis indicates the training time. Note that this figure is represented in semi-logarithmic coordinates

**(a)** The average accuracy of 50 trails classification experiments.



**(b)** The average training time of three classification algorithms.



## 4 Conclusion

In this paper, we have presented the SRP-based PC-ELM for scene classification. Experimental results on the benchmark dataset show that the PC-ELM network achieves a good balance between learning speed and generalization performance. Experimental results also show that PC-ELM has strong immunity to high-dimensional feature space, which often results in over-fitting to other classifiers. Exploration of the PC-ELM in other applications is undergoing, and results will be reported in our future publications.

## References

1. Csurka, Gabriella, et al. "Visual categorization with bags of keypoints." Workshop on statistical learning in computer vision, ECCV. Vol. 1. No. 1–22 (2004)
2. Lowe, David G. "Object recognition from local scale-invariant features." Computer vision, 1999. The proceedings of the seventh IEEE international conference on. Vol. 2, (1999)
3. Oliva, Aude, Torralba, Antonio: Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision* **42**(3), 145–175 (2001)
4. Trunk, G.V.: A problem of dimensionality: A simple example. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **3**, 306–307 (1979)
5. Boser, Bernhard E., Isabelle M. Guyon, and Vladimir N. Vapnik. "A training algorithm for optimal margin classifiers." *Proceedings of the fifth annual workshop on Computational learning theory* (1992)
6. Joachims, Thorsten: Text categorization with support vector machines: Learning with many relevant features. Springer, Berlin Heidelberg (1998)
7. Huang, Guang-Bin, Qin-Yu Zhu, and Chee-Kheong Siew. "Extreme learning machine: a new learning scheme of feedforward neural networks." *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on. Vol. 2* (2004)
8. Zhao, Rui, Mao, Kezhi: Semi-Random Projection for Dimensionality Reduction and Extreme Learning Machine in High-Dimensional Space. *Computational Intelligence Magazine, IEEE* **10**(3), 30–41 (2015)
9. Bingham E. and Mannila H., Random projection in dimensionality reduction: Applications to image and text data, in *Proc. 7th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, pp. 245250 (2001)
10. Li, P, Hastie, T. J., and Church, K. W. Very sparse random projections, in *Proc. 12th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, pp. 287296 (2006)
11. Huang, Guang-Bin, et al. "Extreme learning machine for regression and multiclass classification." *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* **42.2** pp. 513–529 (2012)
12. Cambria, E., Huang, G. B., Kasun, L. L. C., Zhou, H., Vong, C. M., Lin, J., ..., Liu, J. Extreme learning machines [trends & controversies]. *Intelligent Systems, IEEE* (2013)
13. Li, J-L. and Li, F-F. What, where and who? Classifying event by scene and object recognition. *IEEE Intern. Conf. in Computer Vision* (2007)
14. Chang, C-C., and Lin, C-J. LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (2011)
15. Bosch, A., Zisserman, A., Munoz, X.: Image classification using random forests and ferns. In *Proc. ICCV* (2007)
16. Vedaldi, A., Fulkerson, B. VLFeat: An open and portable library of computer vision algorithms, <http://www.vlfeat.org/> (2008)

Proceedings of ELM-2015 Volume 2

Theory, Algorithms and Applications (II)

Cao, J.; Mao, K.; Wu, J.; Lendasse, A. (Eds.)

2016, IX, 516 p. 146 illus., 52 illus. in color., Hardcover

ISBN: 978-3-319-28372-2