

Received July 9, 2018, accepted August 21, 2018, date of publication September 10, 2018, date of current version September 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2868713

# Safe Semi-Supervised Extreme Learning Machine for EEG Signal Classification

**QINGSHAN SHE<sup>1</sup>, BO HU<sup>1</sup>, HAITAO GAN<sup>1</sup>, YINGLE FAN<sup>1</sup>, THINH NGUYEN<sup>2</sup>,  
THOMAS POTTER<sup>2</sup>, AND YINGCHUN ZHANG<sup>1,2</sup>**

<sup>1</sup>Institute of Intelligent Control and Robotics, Hangzhou Dianzi University, Hangzhou 310018, China

<sup>2</sup>Department of Biomedical Engineering, University of Houston, Houston, TX 77204, USA

Corresponding authors: Qingshan She (qsshe@hdu.edu.cn) and Yingchun Zhang (yzhang94@uh.edu)

This work was supported in part by the National Nature Science Foundation under Grant 61201302, Grant 61671197, and Grant 61601162, and in part by the University of Houston.

**ABSTRACT** One major challenge in the current brain–computer interface research is the accurate classification of time-varying electroencephalographic (EEG) signals. The labeled EEG samples are usually scarce, while the unlabeled samples are available in large quantities and easy to collect in real applications. Semi-supervised learning (SSL) methods can utilize both labeled and unlabeled data to improve performance over supervised approaches. However, it has been reported that the unlabeled data may undermine the performance of SSL in some cases. To improve the safety of SSL, we proposed a new safety-control mechanism by analyzing the differences between unlabeled data analysis in supervised and semi-supervised learning. We then develop and implement a safe classification method based on the semi-supervised extreme learning machine (SS-ELM). Following this approach, the Wasserstein distance is used to measure the similarities between the predictions obtained from ELM and SS-ELM algorithms, and a different risk degree is thereby calculated for each unlabeled data instance. A risk-based regularization term is then constructed and embedded into the objective function of the SS-ELM. Extensive experiments were conducted using benchmark and EEG datasets to evaluate the effectiveness of the proposed method. Experimental results show that the performance of the new algorithm is comparable to SS-ELM and superior to ELM on average. It is thereby shown that the proposed method is safe and efficient for the classification of EEG signals.

**INDEX TERMS** Brain–computer interface, electroencephalogram, semi-supervised learning, extreme learning machine, risk degree.

## I. INTRODUCTION

Brain-computer interface (BCI) research has focused primarily on providing communication and control capabilities that do not depend on the brain's normal output channels of peripheral nerves and muscles [1]. Over the past few years, BCI has been extended to help individuals with disorders of consciousness and stroke [2]. At present, scalp electroencephalography (EEG) is the most practical choice for BCI [3], as it is non-invasive, safe, inexpensive, can offer favorable temporal resolution (on the order of milliseconds) [4], and functions in most environments. As a crucial component embodied in general EEG-based BCIs, the classification problem of EEG signals is a key focus. Unfortunately, due to the high variability of EEG recordings, the real-time, accurate classification of EEG signals remains a challenging task.

To effectively parse the various patterns of EEG signals, a number of different classification algorithms can be

employed, including linear discriminant analysis (LDA), neural networks (NN), support vector machine (SVM), nonlinear Bayesian classifiers, nearest neighbor classifiers, and combinations thereof [5]–[7]. Recently, a new method, known as the extreme learning machine (ELM), has been proposed by Huang *et al.* [8]–[10] and demonstrated an improved learning speed and with better generalizability than well-known back propagation neural networks and SVMs. As a result, ELMs have been applied to pattern recognition tasks in images, speech, and EEG signals [11]–[14]. Unfortunately, labeled EEG samples are scarce, expensive, and time-consuming to obtain (especially when labelling requires manual effort and expertise), while unlabeled EEG data are abundant, but not fully applicable to real clinical applications. Over the past few decades, semi-supervised learning has attracted significant attention and widely applied in speech recognition [15], object tracking [16], and face recognition [17].

Semi-supervised learning (SSL) attempts to exploit the information from both labeled and unlabeled data to improve learning performance, relying on the intrinsic data distribution information embedded in the unlabeled samples. Generally speaking, SSL utilizes the following assumptions on the data space: smoothness, cluster, manifold and disagreement. Of these, the manifold regularization assumption has become a particular topic of debate. It assumes that data points are distributed on some low dimensional manifold represented by a Laplacian graph and that similar instances will share similar classification outputs. Up to now, a number of algorithms have been proposed based on the manifold assumption, and these have achieved encouraging performance [18]–[20]. By using a Laplacian regularization term to learn from labeled and unlabeled data, Belkin *et al.* [18] proposed the Laplacian regularized least squares (LapRLS) and support vector machines (LapSVM) algorithms, and the results demonstrate that the manifold regularization technique can effectively exploit the information of unlabeled data. Gan *et al.* [19] presented a Laplacian regularized kernel minimum squared error (LapKMSE) method by incorporating the manifold structures of labeled and unlabeled data into the objective function of KMSE. Recently, Huang *et al.* [20] developed a semi-supervised extreme learning machine (SS-ELM) algorithm which introduced the manifold regularization framework into a classical ELM, and yielded better clustering results than LapSVM when applied to University of California Irvine (UCI) datasets.

The aforementioned SSL methods usually assume that all unlabeled data are safe to be exploited. This may not be the case, however, as SSL methods may yield worse performance than their supervised counterparts in some cases [21]–[23]; when the used assumption is not consistent with data distribution, the unlabeled data may reduce the performance of semi-supervised learning. Therefore, it is necessary to design a SSL method that can safely employ unlabeled data, regardless of the distribution. Up to now, there have been few attempts to research safe SSL methods for exploiting unlabeled instances. Li and Zhou [22] proposed the S3VM-us algorithm to reduce the chance of performance degradation for S3VMs [24]. This method used a hierarchical clustering algorithm to select the unlabeled instances, which were predicted by transductive SVM (TSVM) [25], while the remaining data were predicted by SVM. Additionally, they also developed safe semi-supervised SVMs (S4VMs) [23]. While the S3VMs sought an optimal low-density separator, S4VMs utilized the candidate low-density separators simultaneously to reduce the risk of identifying a poor separator from unlabeled instances. This led to an overall performance that was comparable to S3VMs and rarely performed worse than inductive SVMs. Wang and Chen [26] extended the semi-supervised classification method based on class membership (SSCCM) to create the safety-aware SSCCM (SA-SSCCM) by designing a safety mechanism that served as a trade-off between least-square SVM (LS-SVM) and SSCCM. Experimental results then showed promising performance for the

SA-SSCCM when compared to the original LS-SVM and SSCCM methods. Gan *et al.* [27] similarly developed a risk-based safe Laplacian regularized least squares (RsLapRLS) by analyzing the different risk degrees for unlabeled data. These methods focused on safe learning procedures for semi-supervised SVM, SSCCM, and LapRLS and demonstrated the value of estimating the risk when incorporating unlabeled instances.

In the family of neural network models, ELMs have recently gained popularity due to their efficiency; they learn quickly and accurately, and require minimal human intervention [10]. To help utilize both labeled and unlabeled data, ELM has been naturally extended to allow for SSL, leading to the creation of the SS-ELM algorithm that is based on manifold regularization [20]. Unfortunately, the SS-ELM still lacks proper safety controls when attempting to exploit unlabeled data; in certain cases, the inclusion of unlabeled data instances may degrade classification accuracy of the SS-ELM instead of improving it. In this paper, we introduce the safe semi-supervised learning for SS-ELM (Safe-SSELML) algorithm for multi-classification situations, which considers how unlabeled data points are treated by supervised and semi-supervised ELMs. The basic concept is to assign a risk degree to each unlabeled data instance. This risk degree would be small if the unlabeled data is helpful in training a semi-supervised classifier and large if it is beneficial. Firstly, ELM and SS-ELM are used to separately classify the unlabeled instances. Secondly, the similarities between the obtained predictions are measured using the Wasserstein distance and risk degrees are calculated for each of the unlabeled instances. Finally, the risk degrees are utilized to construct a risk-based regularization term that is embedded into the SS-ELM algorithm. Hence, our method can be considered as a trade-off between the supervised and semi-supervised ELM approaches. The Safe-SSELML algorithm was tested on several benchmark datasets and BCI Competition IV Dataset 2a and the results demonstrate that the performance of our algorithm is highly competitive with SS-ELM. Moreover, it rarely performs significantly inferior to the ELM that only uses labeled data.

The rest of the paper is organized as follows: Section 2 presents a brief introduction to basic ELM and SS-ELM. Section 3 provides a detailed description of the proposed Safe-SSELML algorithm. Section 4 evaluates the performance of our method through a series of experiments on several benchmark datasets and motor imagery EEG datasets. Finally, the paper is concluded and future work is presented in Section 5.

## II. BACKGROUND KNOWLEDGE

In this section, we introduce two predecessors – ELM [20] and SS-ELM [10] – in detail before introducing our risk-based Safe-SSELML method, which uses a novel safety-control mechanism that measures the risk degree by analyzing the behavior of unlabeled data in the ELM and SS-ELM algorithms.

### A. BASIC ELM

Suppose we have a training set with  $N$  samples,  $\{\mathbf{X}, \mathbf{Y}\} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$ , where each sample is denoted by  $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{ip}]^T \in \mathbb{R}^p$  with a corresponding network target vector of  $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{iq}]^T \in \mathbb{R}^q$ , where  $p$  and  $q$  represent the respective dimensions, and  $T$  denotes a transpose operation. Assuming that  $m$  is the number of neurons in hidden layer, the output function of ELM [8] is given by

$$\mathbf{y}_i = \sum_{j=1}^m \boldsymbol{\beta}_{jg} \left( \mathbf{a}_j^T \mathbf{x}_i + b_j \right), \quad i = 1, 2, \dots, N \quad (1)$$

where  $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \dots, \boldsymbol{\beta}_m]^T \in \mathbb{R}^{m \times q}$  is the output weight vector between the hidden layer to the output nodes and  $g(\bullet)$  is the activation function, which can be any nonlinear piecewise continuous function, such as the Sigmoid function and Gaussian function.  $\mathbf{a}_j = [a_{j1}, a_{j2}, \dots, a_{jp}]^T \in \mathbb{R}^p$  then represents the weights between the  $j$ -th hidden node and the input layer, with a corresponding weight vector denoted by  $\mathbf{a} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m]^T \in \mathbb{R}^{m \times q}$ , and  $b_j$  is the bias of the  $j$ -th hidden node, with a corresponding bias vector represented by  $\mathbf{b} = [b_1, b_2, \dots, b_m]^T \in \mathbb{R}^m$ . Note that both  $\mathbf{a}$  and  $\mathbf{b}$  can be randomly generated according to a continuous probability distribution.

For convenience, formulation (1) can be written in matrix form:

$$\mathbf{Y} = \mathbf{H}\boldsymbol{\beta} \quad (2)$$

where  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]^T \in \mathbb{R}^{N \times q}$  is the target matrix of the training data and  $\mathbf{H}$  is the hidden layer output matrix defined as

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{a}_1^T \mathbf{x}_1 + b_1) & \dots & g(\mathbf{a}_m^T \mathbf{x}_1 + b_m) \\ \dots & \dots & \dots \\ g(\mathbf{a}_1^T \mathbf{x}_N + b_1) & \dots & g(\mathbf{a}_m^T \mathbf{x}_N + b_m) \end{bmatrix} \quad (3)$$

where  $g(\cdot)$  is the activation function. By adopting a regularization technique [9], basic ELM aims to minimize the following objective function:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{m \times q}} \|\boldsymbol{\beta}\|_2^2 + C \|\mathbf{H}\boldsymbol{\beta} - \mathbf{Y}\|_2^2 \quad (4)$$

where  $C$  is a penalty coefficient for the training errors and  $\|\cdot\|_2$  denotes the  $l_2$ -norm of a matrix or a vector. The output weight vector  $\boldsymbol{\beta}$  is then obtained according to the Moore-Penrose principle. If the number of training data  $N$  is more than  $m$ , the solution of Eq. (4) is:

$$\boldsymbol{\beta}^* = \left( \frac{\mathbf{I}_m}{C} + \mathbf{H}^T \mathbf{H} \right)^{-1} \mathbf{H}^T \mathbf{Y} \quad (5)$$

where  $\mathbf{I}_m$  is an identity matrix of dimension  $m$ . If  $N$  is less than  $m$ , the solution of Eq. (4) is:

$$\boldsymbol{\beta}^* = \mathbf{H}^T \left( \frac{\mathbf{I}_N}{C} + \mathbf{H} \mathbf{H}^T \right)^{-1} \mathbf{Y} \quad (6)$$

where  $\mathbf{I}_N$  is an identity matrix of dimension  $N$ .

### B. SEMI-SUPERVISED ELM

Given a training set  $\{\mathbf{X}, \mathbf{Y}\} = \{\{\mathbf{X}_l, \mathbf{Y}_l\}, \mathbf{X}_u\}$ , where  $\{\mathbf{X}_l, \mathbf{Y}_l\} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \dots, (\mathbf{x}_l, \mathbf{y}_l)\}$  is the first  $l$  labeled samples and  $\mathbf{X}_u = \{\mathbf{x}_{l+1}, \dots, \mathbf{x}_{l+u}\}$  is the last  $u$  unlabeled samples in  $\mathbf{X}$ , the objective function of SS-ELM can be written as:

$$\begin{aligned} \min_{\boldsymbol{\beta} \in \mathbb{R}^{m \times q}} & \|\boldsymbol{\beta}\|_2^2 + \left\| \mathbf{C}^{1/2} (\mathbf{H}_l \boldsymbol{\beta} - \mathbf{Y}) \right\|_2^2 \\ & + \lambda_1 \text{Tr}(\boldsymbol{\beta}^T \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} \boldsymbol{\beta}) \end{aligned} \quad (7)$$

where  $\|\boldsymbol{\beta}\|_2^2$  denotes the regularization term,  $\|\mathbf{H}_l \boldsymbol{\beta} - \mathbf{Y}\|_2^2$  contributes the empirical risk minimization,  $\boldsymbol{\beta}^T \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} \boldsymbol{\beta}$  is the manifold regularization framework, and  $\lambda_1$  is a trade-off parameter.  $\mathbf{C}$  is a diagonal matrix whose entries are  $C_{jj} = C/N_{t_j}$ , where  $N_{t_j}$  is the number of training samples belonging to class  $t_j$ ,  $j = 1, 2, \dots, l$ .  $\text{Tr}(\cdot)$  denotes the trace of a matrix and  $\mathbf{L} = \mathbf{D} - \mathbf{W}$  is known as the graph Laplacian built from both labeled and unlabeled samples, wherein  $\mathbf{D}$  is a diagonal matrix and  $\mathbf{W}$  denotes the similarity matrix. It is convenient to calculate the hidden layer output matrices  $\mathbf{H}_{l+u} \in \mathbb{R}^{(l+u) \times q}$  for  $\mathbf{X}$  as  $\mathbf{H}_{l+u} = h_\theta(\mathbf{X})$ , and  $\mathbf{H}_l \in \mathbb{R}^{l \times q}$  for  $\mathbf{X}_l$  as  $\mathbf{H}_l = h_\theta(\mathbf{X}_l)$ .

According to [20], when the number of data is greater than or equal to the number of hidden neurons ( $l + u > m$ ), the solution to the SS-ELM algorithm is given as

$$\boldsymbol{\beta}^* = (\mathbf{I}_m + \mathbf{H}_l^T \mathbf{C} \mathbf{H}_l + \lambda \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u})^{-1} \mathbf{H}_l^T \mathbf{C} \mathbf{Y} \quad (8)$$

where  $\mathbf{I}_m$  is an identity matrix of dimension  $m$ . When the number of labeled data is less than the number of hidden neurons ( $l + u < m$ ), which is common in semi-supervised learning, an alternative solution is given by:

$$\boldsymbol{\beta}^* = \mathbf{H}_l^T (\mathbf{I}_{l+u} + \mathbf{C} \mathbf{H}_l \mathbf{H}_l^T + \lambda \mathbf{L} \mathbf{H}_{l+u} \mathbf{H}_{l+u}^T)^{-1} \mathbf{C} \mathbf{Y} \quad (9)$$

where  $\mathbf{I}_{l+u}$  is an identity matrix of dimension  $l + u$ .

## III. OUR METHOD

In this section, we present our proposed safe semi-supervised extreme learning (Safe-SSELM) method, which designs an efficient safety-control mechanism to exploit unlabeled data.

### A. RISK DEGREE GENERATING ALGORITHM

The degree of risk for unlabeled data is defined by analyzing their different behaviors in ELM and SS-ELM. When the SS-ELM prediction results for unlabeled samples are significantly different from those obtained by supervised ELM, the unlabeled sample may be assigned a large degree of risk and should be classified by ELM. Otherwise, the unlabeled sample may be considered safe and assigned a small risk degree, in which case samples should be classified by SS-ELM.

Given a training set  $\{\mathbf{X}, \mathbf{Y}\} = \{\{\mathbf{X}_l, \mathbf{Y}_l\}, \mathbf{X}_u\}$ , where  $\{\mathbf{X}_l, \mathbf{Y}_l\} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \dots, (\mathbf{x}_l, \mathbf{y}_l)\}$  is the first  $l$  labeled samples and  $\mathbf{X}_u = \{\mathbf{x}_{l+1}, \dots, \mathbf{x}_{l+u}\}$  is the last  $u$  unlabeled samples of  $\mathbf{X}$ , the detailed steps of the risk degree generation algorithm are given as follows:

*Step (1):* The supervised ELM and semi-supervised SS-ELM methods are used to separately predict the unlabeled samples.

The ELM classifier is trained with  $\{\mathbf{X}_l, \mathbf{Y}_l\}$  while the SS-ELM classifier is trained with  $\{\mathbf{X}, \mathbf{Y}\}$ , and both algorithms are subsequently used to obtain predictions of  $\mathbf{x}_j$  that belongs to the set of unlabeled samples,  $\mathbf{X}_u$ , yielding the respective probability distribution vectors of  $\bar{\mathbf{y}}_j$  and  $\tilde{\mathbf{y}}_j$ . The columns of the max values in  $\bar{\mathbf{y}}_j$  and  $\tilde{\mathbf{y}}_j$  denote the predicted category labels, which are written as  $\bar{\mathbf{y}}_{j\_label}$  and  $\tilde{\mathbf{y}}_{j\_label}$  respectively.

*Step (2):* The Wasserstein distance is used to measure the similarity between the predictions made in Step (1) and the degrees of risk for unlabeled instances are calculated.

The risk degree  $s_j$  of unlabeled data  $\mathbf{x}_j$  is determined by a difference function  $df(\mathbf{x}_j)$ . The Wasserstein distance is a general distance measure between any probability distributions. As a metric, it respects fundamental metric properties such as non-negativity, symmetry and the triangle inequality [28]. It has recently found some applications in multi-label classification [29]. In this paper, it is employed to calculate  $df(\mathbf{x}_j)$  and its formula is given as:

$$df(\mathbf{x}_j) = \text{sum} \frac{|\bar{\mathbf{y}}_j - \tilde{\mathbf{y}}_j|}{Len} \quad (10)$$

where  $Len$  is the length of vector  $\bar{\mathbf{y}}_j$ . Next, the risky degree  $s_j$  is defined as:

$$s_j = \begin{cases} e^{df(\mathbf{x}_j)} & \text{if } \bar{\mathbf{y}}_{j\_label} \neq \tilde{\mathbf{y}}_{j\_label} \\ 1 - e^{-df(\mathbf{x}_j)} & \text{if } \bar{\mathbf{y}}_{j\_label} = \tilde{\mathbf{y}}_{j\_label} \end{cases} \quad (11)$$

If the predicted labels are different (i.e.,  $\bar{\mathbf{y}}_{j\_label} \neq \tilde{\mathbf{y}}_{j\_label}$ ),  $s_j$  will be large, indicating that the unlabeled sample  $\mathbf{x}_j$  maybe harmful to the classifier. Otherwise, the  $s_j$  value should be small, indicating that the sample is benign to classifier performance. The Wasserstein distance between the probability distribution vectors  $\bar{\mathbf{y}}_j$  and  $\tilde{\mathbf{y}}_j$  should match the calculated  $s_j$  value; a large distance should accompany a large  $s_j$  value while a small Wasserstein distance should pair with a similarly small  $s_j$  value. In special cases,  $s_j$  will be zero if  $\bar{\mathbf{y}}_j = \tilde{\mathbf{y}}_j$ , which will indicate that  $\mathbf{x}_j$  is safe for semi-supervised learning.

## B. FORMULATION

In the following, the risk degrees for unlabeled instances are quantified by the new “risky” item, which is then used to modify the objective function of the basic SS-ELM algorithm.

By introducing a “risky” item, the new objective function of the proposed Safe-SSELM method is differentiated from that of the SS-ELM in Eq. (7), which is written as:

$$\min_{\beta} Q(\beta) = \|\beta\|_2^2 + \left\| \mathbf{C}^{1/2} (\mathbf{H}_l \beta - \mathbf{Y}) \right\|_2^2 + \lambda_1 \text{Tr}(\beta^T \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} \beta) + \lambda_2 \sum_{j=1}^u s_j \|\bar{\mathbf{y}}_j - \tilde{\mathbf{y}}_j\|_2^2 \quad (12)$$

where the first three terms are used to define the semi-supervised classifier and the last term controls the trade-off between supervised and semi-supervised learning. As a

result, the new algorithm will function the same as a traditional SS-ELM in cases where  $\lambda_2 = 0$ .

By substituting  $\bar{\mathbf{y}}_j = \sum_{k=1}^m \bar{\beta}_k g(\mathbf{a}_k^T \mathbf{x}_j + b_k)$  and  $\tilde{\mathbf{y}}_j = \sum_{k=1}^m \beta_k g(\mathbf{a}_k^T \mathbf{x}_j + b_k)$  into Eq. (12), the modified objective function can be rewritten as:

$$\begin{aligned} \min_{\beta} Q(\beta) = & \|\beta\|_2^2 + \left\| \mathbf{C}^{1/2} (\mathbf{H}_l \beta - \mathbf{Y}) \right\|_2^2 \\ & + \lambda_1 \text{Tr}(\beta^T \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} \beta) \\ & + \lambda_2 (\mathbf{H}_u \beta - \mathbf{H}_u \beta_{elm})^T \mathbf{S} (\mathbf{H}_u \beta - \mathbf{H}_u \beta_{elm}) \end{aligned} \quad (13)$$

where  $\beta_{elm} = [\bar{\beta}_1, \bar{\beta}_2, \dots, \bar{\beta}_m]^T \in \mathbf{R}^{m \times q}$  is the optimal output weights for a basic ELM,  $\mathbf{H}_u$  is the hidden layer output matrix for  $\mathbf{X}_u$ , and  $\mathbf{S}$  is a diagonal matrix expressed as  $\text{diag}(s_1, s_2, \dots, s_u)$ . The derivative of Eq. (13) with respect to  $\beta$  is then:

$$\begin{aligned} \frac{\partial Q}{\partial \beta} = & \beta + \mathbf{H}_l^T \mathbf{C} (\mathbf{H}_l \beta - \mathbf{Y}) + \lambda_1 \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} \beta \\ & + \lambda_2 \mathbf{H}_u^T \mathbf{S} (\mathbf{H}_u \beta - \mathbf{H}_u \beta_{elm}) \end{aligned} \quad (14)$$

By setting Eq. (14) to zero, we can get:

$$\begin{aligned} \beta^* = & (\mathbf{H}_l^T \mathbf{C} \mathbf{H}_l + \mathbf{I} + \lambda_1 \mathbf{H}_{l+u}^T \mathbf{L} \mathbf{H}_{l+u} + \lambda_2 \mathbf{H}_u^T \mathbf{S} \mathbf{H}_u)^{-1} \\ & \times (\mathbf{H}_l^T \mathbf{C} \mathbf{Y} + \lambda_2 \mathbf{H}_u^T \mathbf{S} \mathbf{H}_u \beta_{elm}) \end{aligned} \quad (15)$$

where  $\mathbf{I}$  is an identity matrix.

For the test data  $\mathbf{X}_{test}$ , the ELM algorithm is first used to calculate its corresponding hidden layer output matrix  $\mathbf{H}_{test}$ , and the prediction result is given by:

$$\mathbf{Y}_{predict} = \mathbf{H}_{test} \beta^* \quad (16)$$

The implementation of our Safe-SSELM method is described in Algorithm 1. It is important to highlight the key differences between the proposed Safe-SSELM method and its predecessors, including both ELM and SS-ELM. Firstly, considering the uncertainty of unlabeled data, we have designed a new safety-controlled strategy by introducing a measure of risk into the objective function of basic SS-ELM, which can lead to a safer exploitation of unlabeled data. Secondly, by building upon the traditional SS-ELM classifier, the new method inherits its advantages and is suitable for multi-classification.

## IV. EXPERIMENTAL RESULTS AND DISCUSSION

Several experiments were performed on benchmark and EEG datasets, and their results are presented here to demonstrate the effectiveness of the proposed Safe-SSELM method. The new algorithm was constructed based on both the ELM and SS-ELM codes ([http://www.ntu.edu.sg/home/egbhuang/elm\\_codes.html](http://www.ntu.edu.sg/home/egbhuang/elm_codes.html)), and performance was compared to the conventional ELM and SS-ELM. All the methods were implemented in the MATLAB 2014a environment on a PC with a 3.4 GHz processor and 8.0 GB RAM.

**Algorithm 1** The Safe-SSELML

**Input:** A training set  $\{\mathbf{X}, \mathbf{Y}\} = \{\{\mathbf{X}_l, \mathbf{Y}_l\}, \mathbf{X}_u\}$ , the parameters  $C$ ,  $\lambda_1$ , and  $\lambda_2$ .

**Output:** the output weights  $\beta^*$  between the hidden layer and output layer.

- 1: Perform supervised ELM classification using Eq. (5) or Eq. (6) and the semi-supervised SS-ELM classification using Eq. (8) or Eq. (9);
- 2: Compute the risk degrees for unlabeled samples using Eq. (11);
- 3: Utilize the calculated risk degrees for unlabeled samples to construct the new “risky” item and obtain the new objective function following Eq. (12);
- 4: Compute the output weights  $\beta^*$  using Eq. (15);
- 5: Return  $\beta^*$ .

**TABLE 1.** Description of the benchmark datasets.

Datasets	Dimension	Numbers of samples	Number of categories
Waveform1	21	5000	3
Waveform2	40	5000	3
Har1	561	2947	6
Har2	561	7352	6
USPST	256	2007	10
COIL-20	1024	1440	20
Oral	10304	400	40
Wine	13	178	3
Yale	10000	165	15
IRIS	4	150	3

**A. EXPERIMENTS ON BENCHMARK DATASETS****1) DATASETS AND SETTINGS**

In order to evaluate its performance, the proposed safe-SS-ELM method was first applied to eight popular benchmark datasets from the UCI repository [30] and two face recognition datasets [31]. These datasets are widely used to evaluate the performance of machine learning algorithms [32], and the details for each of these multiclass datasets are shown in Table 1.

The experimental setup is described in detail in [33]. Briefly, the analysis for each of the 4 datasets followed a 3-steps procedure: (i) The data pool was randomly divided into two subsets, such that 40% of the trials served as training data while the remaining 60% comprised the testing data. Training data was then divided into two equal subsets: a labeled training set and an unlabeled training set. (ii) The training set was first normalized, and the testing set data was then normalized according to the mean and standard deviation calculated from the training set using the z-score approach. (iii) The classification process was repeated ten times, and the averaged accuracy and training time were recorded for further analysis. In the experiment, three hyperparameters  $C$ ,

**TABLE 2.** The classification results for benchmark datasets using different methods.

Datasets	ELM	SS-ELM	Our method
Waveform1	84.43±1.17	<b>84.67±1.55</b>	84.50±1.52
Waveform2	81.88±1.23	82.88±1.60	<b>85.76±1.47</b>
Har1	96.40±2.64	94.90±2.48	<b>97.15±2.01</b>
Har2	97.77±2.87	95.72±2.57	<b>98.06±2.48</b>
USPST	89.22±1.88	89.89±1.96	<b>90.45±1.57</b>
COIL-20	94.94±1.51	95.19±1.44	<b>95.75±1.47</b>
Oral	83.60±5.26	<b>84.10±6.15</b>	83.70±5.12
Wine	94.83±0.74	<b>96.85±0.68</b>	95.06±1.01
Yale	88.05±9.55	87.80±9.24	<b>88.29±8.49</b>
IRIS	95.31±0.87	94.88±1.48	<b>95.42±0.93</b>

$\lambda_1$ , and  $\lambda_2$  were defined, and optimal model parameters were determined from  $C \in \{10^{-5}, 10^{-4}, \dots, 10^4, 10^5\}$ ,  $\lambda_1 \in \{10^{-9}, 10^{-8}, \dots, 10^{-1}, 10^0\}$ , and  $\lambda_2 \in \{10^{-5}, 10^{-4}, \dots, 10^4, 10^5\}$ . To ensure a valid comparison between the different methods, the same ranges of possible user-specified parameters were used for cross-validation. The best parameters for each dataset were found to be:  $C = 10^{-2}$ ,  $\lambda_1 = 10^{-6}$ , and  $\lambda_2 = 10^{-1}$ . ELM-based algorithms also made use of a fourth parameter,  $m$ . According to [13], however, the performance of ELM-based methods is not very sensitive to the number of hidden neurons, so  $m = 100$  was chosen for all datasets.

**2) EXPERIMENTAL RESULTS**

In this experiment, we compared the proposed Safe-SSELML to the state-of-the-art methods ELM and SS-ELM. The classification performance was evaluated in terms of average accuracy. Table 2 summarizes the performance results for each classification method when applied to the benchmark datasets.

From the results shown in Table 2, it is evident that our algorithm overall achieved better performance than ELM and SS-ELM. Specifically, SS-ELM performed worse than ELM in the Har1, Har2, Yale, and IRIS datasets, while the new algorithm outperformed the ELM in each of these cases. This indicates that our algorithm can be effectively used for safe semi-supervised learning. Additionally, in cases where the SS-ELM outperformed the ELM, our algorithm obtained results that were comparable to the SS-ELM. This illustrates that the safety mechanism employed by our algorithm is effective, with minimal risk for reduced performance.

**3) PERFORMANCE WITH DIFFERENT RATIOS OF LABELED TRAINING DATA**

In accordance with the experimental setup described in [20], we sought to evaluate the performance of the algorithms when different proportions of labeled and unlabeled data were used for training. 60% of all the samples were first

designated as the testing data set, and the training set was then selected from the remaining samples in which the ratio of labeled-to-unlabeled data varied systematically from 10%:90% to 90%:10%. Experimental results from benchmark datasets with different sample numbers are shown, wherein Waveform1 featured a large number of samples (5000), USPST and COIL20 presented a moderate sample numbers (2007 and 1440, respectively), and Wine had relatively few samples (178). Figure 1 shows the performance of the three methods when applied to the four benchmark datasets with different proportions of labeled data.

#### 4) THE IMPACT OF THE PARAMETERS

Four regularization parameters are used in our algorithm:  $C$ ,  $\lambda_1$ ,  $m$  and  $\lambda_2$ . Since classification performance is not very sensitive to  $m$  in ELM-based methods [33], this section will be used to investigate the impact of the remaining three parameters ( $C$ ,  $\lambda_1$  and  $\lambda_2$ ) on classifier performance when 30% of the data is labeled. These parameters are respectively changed within the ranges of  $C \in \{10^{-5}, 10^{-4}, \dots, 10^4, 10^5\}$ ,  $\lambda_1 \in \{10^{-9}, 10^{-8}, \dots, 10^{-1}, 10^0\}$ , and  $\lambda_2 \in \{10^{-5}, 10^{-4}, \dots, 10^4, 10^5\}$ . Figure 2 shows the testing results from applying our algorithm to benchmark datasets as the parameter  $\lambda_2$  changed. As can be seen, the new algorithm obtained optimal performance in almost all datasets when  $\lambda_2 = 10^{-1}$  and results degraded when the value was too large or too small.

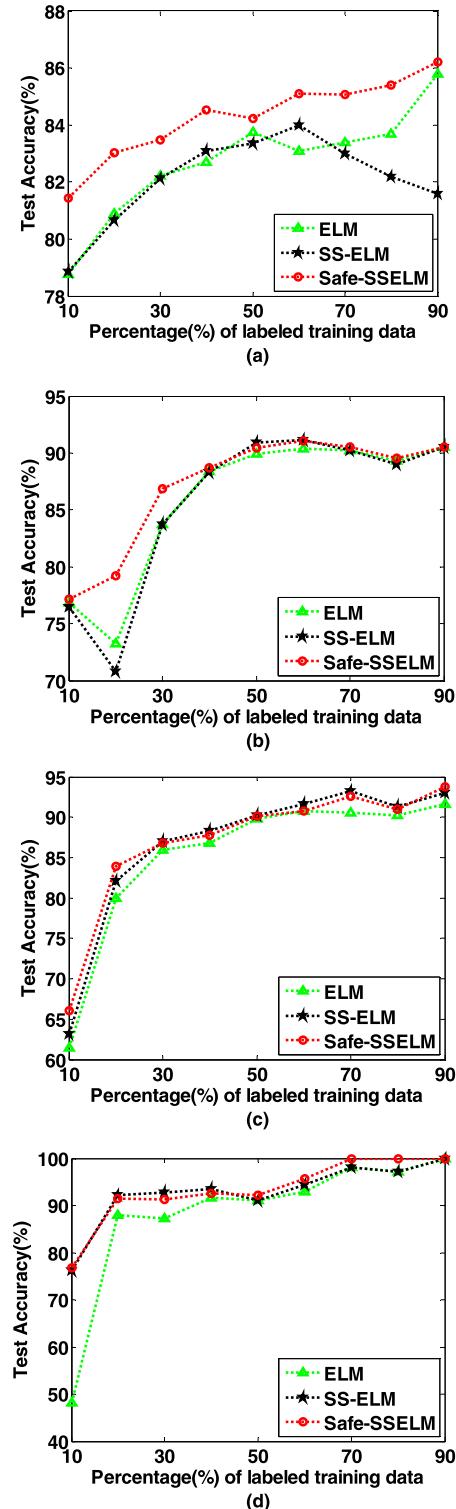
Figure 3 shows the testing accuracies of the benchmark datasets as  $C$  and  $\lambda_1$  values were varied. From these plots, we can see that the best performance was generally obtained with smaller values for both  $C$  and  $\lambda_1$ .

#### B. EXPERIMENTS ON BCI DATASETS

##### 1) DATASETS AND SETTINGS

This section evaluates the performance of the proposed method on MI EEG datasets from the BCI Competition IV Dataset 2a [34]. Data was comprised of MI EEG measurements from 9 subjects with 4 classes of movement; left hand, right hand, feet, and tongue. Training and evaluation sessions were recorded for each subject, where each session consisted of 288 trials of data recorded from 22 EEG channels and 3 monopolar electrooculogram (EOG) channels (with left mastoid serving as reference) [35].

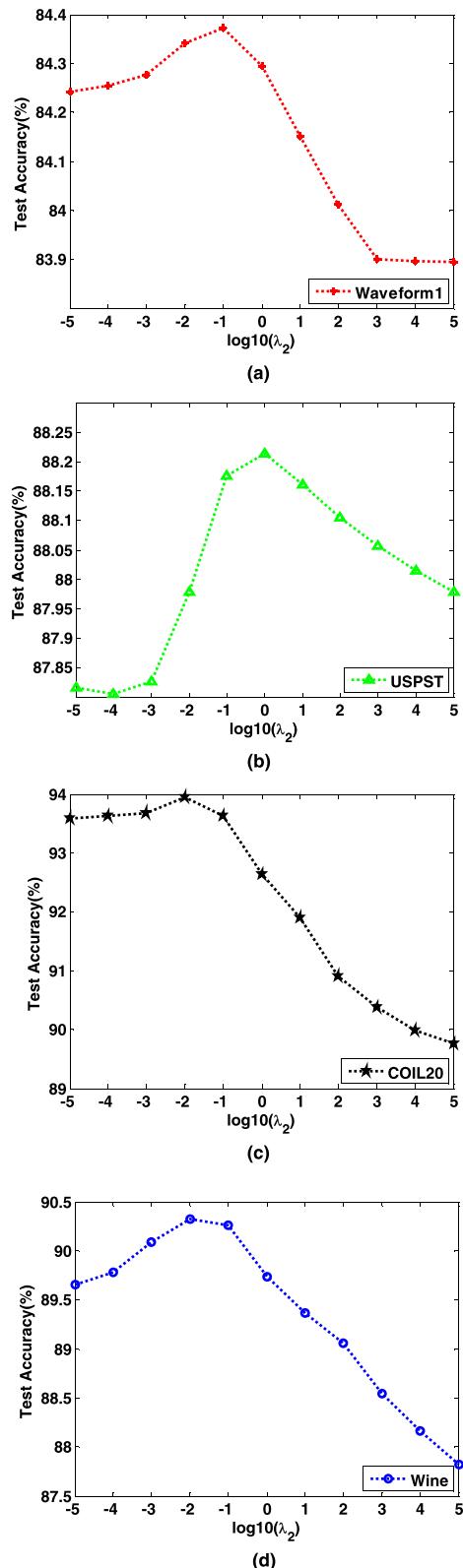
In this experiment, all 288 trials from the training session were used as the labeled training set. The remaining 288 trials from the evaluation session served as the unlabeled training set, so that both sessions were used for testing. For each trial in the BCI Competition IV Dataset 2a, EEG data from 0–2 s after the initiation of MI were segmented and processed with an 8–30 Hz band-pass filter. Preliminary features were extracted from the filtered data using the common spatial pattern (CSP) approach, as performed in [36], yielding 24 feature dimensions. Finally, the proposed Safe-SSELML method was used to learn and classify the compact features. In this experiment, the SS-ELM parameters were set to be:  $C = 10^{-2}$ ,  $\lambda_1 = 10^{-6}$ ,  $\lambda_2 = 10^{-1}$  and  $m = 100$ .



**FIGURE 1.** Testing accuracy with respect to different proportions of labeled data when applying three methods to four benchmark datasets. (a) Waveform1. (b) USPST. (c) COIL20. (d) Wine.

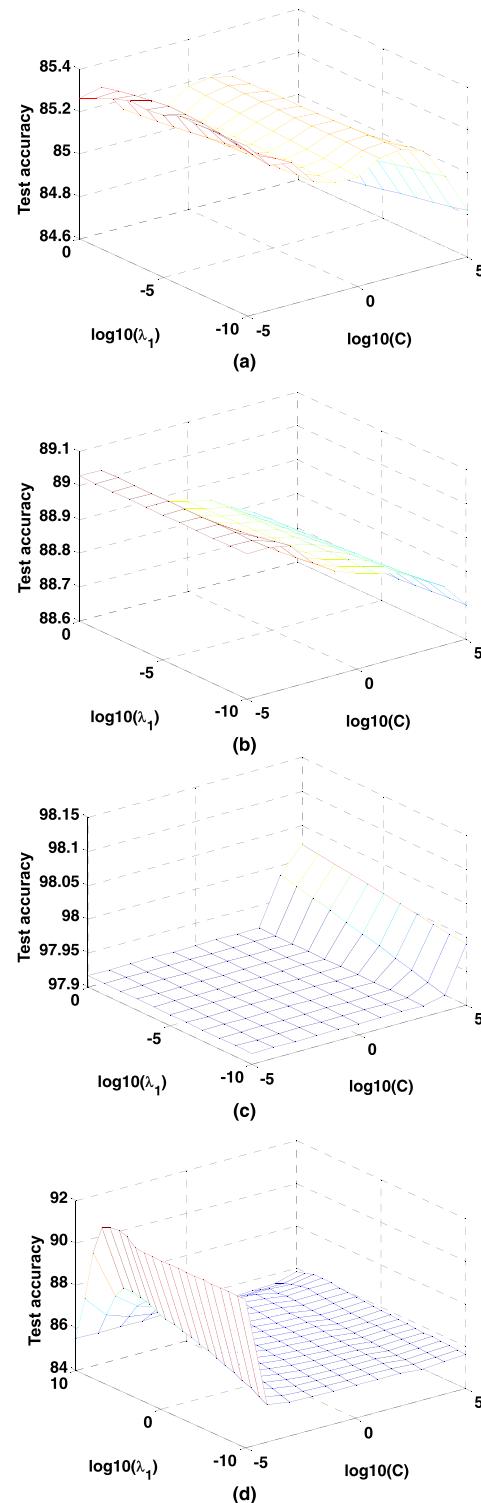
#### 2) EXPERIMENTAL RESULTS

Results comparing the new method to the traditional ELM and SS-ELM classifiers can be seen in Table 3. The proposed



**FIGURE 2.** Testing accuracy of our algorithm as the parameter  $\lambda_2$  changes on benchmark datasets. (a) Waveform1. (b) USPST. (c) COIL20. (d) Wine.

Safe-SSELM achieved results that were comparable to the semi-supervised learning algorithm SS-ELM, and better than the ELM for all tested datasets. This demonstrates that the



**FIGURE 3.** Performance of the Safe-SSELM on benchmark datasets as the parameters  $C$  and  $\lambda_1$  were varied. (a) Waveform1. (b) USPST. (c) COIL20. (d) Wine.

utilized safety-control strategy generally improved classification accuracy over traditional methods. Specifically, it was observed that the Safe-SSELM outperformed ELM in all the datasets, and surpassed the SS-ELM in 7 of the 9 subjects

**TABLE 3.** Classification accuracy on BCI Competition IV Dataset 2A.

Datasets	ELM	SS-ELM	Our method
Suject1	77.60±1.26	79.27±1.08	<b>80.45±1.10</b>
Suject2	45.56±1.17	48.68±1.24	<b>50.02±1.87</b>
Suject3	77.15±1.72	77.70±1.57	<b>78.32±1.65</b>
Suject4	63.22±1.31	62.99±1.01	<b>64.64±1.22</b>
Suject5	43.82±1.81	<b>44.86±1.45</b>	44.45±1.43
Suject6	46.84±1.05	<b>49.44±1.22</b>	48.83±0.98
Suject7	81.57±1.43	80.83±1.39	<b>82.29±1.75</b>
Suject8	81.49±1.22	81.21±1.47	<b>82.17±1.28</b>
Suject9	79.02±1.38	80.69±1.34	<b>81.87±1.16</b>
Mean	66.25±1.37	67.30±1.31	<b>68.12±1.38</b>

(1, 2, 3, 4, 7, 8, 9). Notably, the SS-ELM performed worse than the ELM on 3 subjects (4, 7, 8), while the new algorithm outperformed both. These results demonstrate the effectiveness of the proposed safe strategy for unlabeled data, even in the cases where the semi-supervised classification approach (SS-ELM) yields worse performance than its supervised counterpart (ELM).

## V. CONCLUSION

In this paper, we have proposed a novel safe semi-supervised ELM method based on ELM and SS-ELM for multi-classification. To develop the safety-control mechanism, the characteristics of unlabeled data classification were analyzed for both the ELM and SS-ELM, and degrees of risk were assigned to each unlabeled instance using the Wasserstein distance between the respective predictions of each algorithm. This risk degree reflects how beneficial unlabeled data instances are when training a semi-supervised classifier; a small risk degree indicates the unlabeled data is helpful while a large risk degree indicates that unlabeled data is detrimental to include. The experimental results from several benchmark datasets and the BCI EEG datasets show that the performance of the Safe-SSELM algorithm is never significantly inferior to that of the ELM or SS-ELM; the new algorithm consistently matched or exceeded the best results from traditional classifiers. In future experiments, we will investigate other measures to analyze the risk degree of unlabeled data instances, such as the Kullback-Leibler divergence and Jensen-Shannon distance [37], [38]. Additionally, it would be worthwhile to study the safe version of the other semi-supervised learning algorithms, such as broad learning [39].

## CONFLICT OF INTERESTS

The authors declare no conflict of interests regarding the publication of this article.

## ACKNOWLEDGMENT

The authors would like to acknowledge the BCI Competition IV Dataset 2a which was used to test the algorithms proposed in this study.

## REFERENCES

- [1] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, and G. Curio, “The non-invasive Berlin brain–computer interface: Fast acquisition of effective performance in untrained subjects,” *Neuroimage*, vol. 37, no. 2, pp. 539–550, Aug. 2007.
- [2] C. Brunner, N. Birbaumer, B. Blankertz, C. Guger, A. Kübler, and D. Mattia, “BCNI Horizon 2020: Towards a roadmap for the BCI community,” *Brain-Comput. Interfaces*, vol. 2, no. 1, pp. 1–10, Feb. 2015.
- [3] K. K. Ang and C. Guan, “EEG-based strategies to detect motor imagery for control and rehabilitation,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 4, pp. 392–401, Apr. 2017.
- [4] R. Li, T. Potter, W. Huang, and Y. Zhang, “Enhancing performance of a hybrid EEG-fNIRS system using channel selection and early temporal features,” *Frontiers Hum. Neurosci.*, vol. 11, p. 462, Sep. 2017.
- [5] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, “A review of classification algorithms for EEG-based brain–computer interfaces,” *J. Neural Eng.*, vol. 4, no. 2, pp. R1–R13, Jan. 2007.
- [6] Q. She, H. Gan, Y. Ma, Z. Luo, T. Potter, and Y. Zhang, “Scale-dependent signal identification in low-dimensional subspace: Motor imagery task classification,” *Neural Plasticity*, vol. 2016, p. 7431012, Nov. 2016.
- [7] Y. Ma, X. Ding, Q. She, Z. Luo, T. Potter, and Y. Zhang, “Classification of motor imagery EEG signals with support vector machines and particle swarm optimization,” *Comput. Math. Methods Med.*, vol. 2016, p. 4941235, May 2016.
- [8] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, “Extreme learning machine: Theory and applications,” *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, 2006.
- [9] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, “Extreme learning machine for regression and multiclass classification,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.
- [10] G. Huang, G.-B. Huang, S. Song, and K. You, “Trends in extreme learning machines: A review,” *Neural Netw.*, vol. 61, pp. 32–48, Jan. 2015.
- [11] L. Duan, M. Bao, J. Miao, Y. Xu, and J. Chen, “Classification based on multilayer extreme learning machine for motor imagery task from EEG signals,” *Procedia Comput. Sci.*, vol. 88, pp. 176–184, 2016.
- [12] S. Ding, N. Zhang, X. Xu, L. Guo, and J. Zhang, “Deep extreme learning machine and its application in EEG classification,” *Math. Problems Eng.*, vol. 2015, p. 129021, Nov. 2015.
- [13] Y. Peng and B.-L. Lu, “Discriminative manifold extreme learning machine and applications to image and EEG signal classification,” *Neurocomputing*, vol. 174, pp. 265–277, Jan. 2016.
- [14] Y. Zhang, J. Jin, X. Wang, and Y. Wang, “Motor imagery EEG classification via Bayesian extreme learning machine,” in *Proc. IEEE Int. Conf. Inf. Sci. Technol.*, Dalian, China, May 2016, pp. 27–30.
- [15] B. Varadarajan, D. Yu, L. Deng, and A. Acero, “Using collective information in semi-supervised learning for speech recognition,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Taipei, Taiwan, Apr. 2009, pp. 4633–4636.
- [16] S. Chen, S. Z. Li, S. Su, D. Cao, and R. Li, “Online semi-supervised compressive coding for robust visual tracking,” *J. Vis. Communun. Image Represent.*, vol. 25, no. 5, pp. 793–804, Jul. 2014.
- [17] H. Gan, N. Sang, and R. Huang, “Self-training-based face recognition using semi-supervised linear discriminant analysis and affinity propagation,” *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 31, no. 1, pp. 1–6, Jan. 2014.
- [18] M. Belkin, P. Niyogi, and V. Sindhwani, “Manifold regularization: A geometric framework for learning from labeled and unlabeled examples,” *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Nov. 2006.
- [19] H. Gan, N. Sang, and X. Chen, “Semi-supervised kernel minimum squared error based on manifold structure,” in *Proc. Int. Symp. Neural Netw.*, vol. 7951. Dalian, China: Springer-Verlag, 2013, pp. 265–272.
- [20] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, “Semi-supervised and unsupervised extreme learning machines,” *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2405–2417, Dec. 2014.
- [21] T. Yang and C. E. Priebe, “The effect of model misspecification on semi-supervised classification,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 2093–2103, Oct. 2011.

- [22] Y.-F. Li and Z.-H. Zhou, "Improving semi-supervised support vector machines through unlabeled instances selection," in *Proc. 25th AAAI Conf. Artif. Intell.* San Francisco, CA, USA: AAAI Press, 2011, pp. 386–391.
- [23] Y.-F. Li and Z.-H. Zhou, "Towards making unlabeled data never hurt," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 175–188, Jan. 2015.
- [24] K. P. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Proc. Conf. Adv. Neural Inf. Process. Syst.* Cambridge, MA, USA: MIT Press, 1999, pp. 368–374.
- [25] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th Int. Conf. Mach. Learn.* San Francisco, CA, USA: Morgan Kaufmann, 1999, pp. 200–209.
- [26] Y. Wang and S. Chen, "Safety-aware semi-supervised classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 11, pp. 1763–1772, Nov. 2013.
- [27] H. Gan, Z. Luo, Y. Sun, X. Xi, N. Sang, and R. Huang, "Towards designing risk-based safe Laplacian regularized least squares," *Expert Syst. Appl.*, vol. 45, pp. 1–7, Mar. 2016.
- [28] M. Hu and H. Liang, "Search for information-bearing components in neural data," *PLoS ONE*, vol. 9, no. 6, p. e99793, Jun. 2014.
- [29] C. Frogner, C. Zhang, H. Mabohi, M. Araya-Polo, and T. Poggio, "Learning with a Wasserstein loss," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2015, pp. 2053–2061.
- [30] D. Dua and E. K. Taniskidou. (2017). *UCI Machine Learning Repository*. School of Information and Computer Science, University of California, Irvine, CA, USA, [Online]. Available: <http://archive.ics.uci.edu/ml>
- [31] F. Schroff, T. Treibitz, and E. Christiansen. (2013). *Yale Face Database*. University of California, San Diego, CA, USA, [Online]. Available: <http://vision.ucsd.edu/content/yale-face-database>.
- [32] S. Melacci and M. Belkin, "Laplacian support vector machines trained in the primal," *J. Mach. Learn. Res.*, vol. 12, pp. 1149–1184, Jul. 2011.
- [33] H. Gan, N. Sang, and R. Huang, "Manifold regularized semi-supervised Gaussian mixture model," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 32, no. 4, pp. 566–575, Apr. 2015.
- [34] M. Tangermann et al., "Review of the BCI competition IV," *Frontiers Neurosci.*, vol. 6, no. 1, p. 55, 2012.
- [35] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b," *Frontiers Neurosci.*, vol. 6, no. 1, p. 39, 2012.
- [36] M. Meng, J.-Q. Zhu, Q.-S. She, Y.-L. Ma, and Z.-Z. Luo, "Two-level feature extraction method for multi-class motor imagery EEG," *Acta Automatica Sinica*, vol. 42, no. 12, pp. 1915–1922, 2016.
- [37] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, Jan. 1991.
- [38] Q.-S. She, Y.-L. Ma, M. Meng, X.-G. Xi, and Z.-Z. Luo, "Noise-assisted MEMD based relevant IMFs identification and EEG classification," *J. Central South Univ.*, vol. 24, no. 3, pp. 599–608, Mar. 2017.
- [39] C. L. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.



**BO HU** received the B.S. degree in electrical engineering and automation and the M.S. degree in pattern recognition and intelligent system from Hangzhou Dianzi University, Hangzhou, China. His research interests include semi-supervised learning, and EEG signal processing.



**HAITAO GAN** received the B.S. and M.S. degrees in information and computing science from Wuhan Polytechnic University, Wuhan, China, and the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology, Wuhan. He is currently an Associate Professor with the School of Automation, Hangzhou Dianzi University, Hangzhou, China. His research interests include machine learning, pattern recognition, biomedical signal processing.



**YINGLE FAN** received the B.S. degree in industrial automation and the Ph.D. degree in biomedical engineering from Zhejiang University, Hangzhou, China. He is currently a Professor with the School of Automation, Hangzhou Dianzi University, Hangzhou, China. His research interests include neural information coding, biological vision mechanism, and brain-computer interaction.



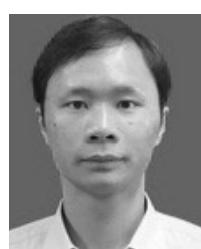
**THINH NGUYEN** received the B.S. degree in biomedical engineering from the University of Houston, Houston, TX, USA, where he is currently pursuing the Ph.D. degree in biomedical engineering. His research interests include computational neuroimaging, focused in EEG and fMRI algorithm development, signal processing, and machine learning.



**THOMAS POTTER** received the B.S. degree in neuroscience from Trinity University, San Antonio, TX, USA, and the M.S. degree in biomedical engineering from The University of Texas at San Antonio, San Antonio, TX, USA. He is currently pursuing the Ph.D. degree in biomedical engineering with the University of Houston. His research interests include multimodal imaging, brain-machine interfaces, and tissue-biomaterial interactions.



**YINGCHUN ZHANG** received the Ph.D. degree in electrical engineering from Zhejiang University, China. He completed his post-doctoral training in biomedical engineering at the University of Minnesota, Minneapolis, MN, USA. He is currently an Assistant Professor with the Department of Biomedical Engineering, University of Houston, Houston, TX, USA. His research interests include multimodal neuroimaging and neural engineering, and their applications rehabilitation. He is a recipient of the NIH Pathway to Independence (K99/R00) Award and the 12th Annual Delsys Prize for the Innovation in Electromyography.



**QINGSHAN SHE** received the B.S. and M.S. degrees in materials science and engineering from the Lanzhou University of Technology, Lanzhou, and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China. He is currently an Associate Professor with the School of Automation, Hangzhou Dianzi University, Hangzhou. His research interests include machine learning and pattern recognition, biomedical signal processing, and brain-computer interface and its applications.