Received April 22, 2018, accepted May 18, 2018, date of publication May 28, 2018, date of current version June 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2841051

Classification of Multi-Class BCI Data by Common Spatial Pattern and Fuzzy System

THANH NGUYEN[1], IMALI HETTIARACHCHI[1], AMIN KHATAMI[1], LEE GORDON-BROWN[2], CHEE PENG LIM[1], AND SAEID NAHAVANDI[1]

[1]Institute for Intelligent Systems Research and Innovation, Deakin University, Waurn Ponds, VIC 3216, Australia
[2]Royal Australian Air Force Base, East Sale, VIC 3852, Australia

Corresponding author: Thanh Nguyen (thanh.nguyen@deakin.edu.au)

ABSTRACT Improving classification accuracy of motor imagery-based brain computer interface (MI-BCI) systems has been discussed widely in the BCI research community. Analyses of multi-class MI data are challenging because feature extraction and classification of these data are more difficult as compared with those applied to binary-class data. This paper introduces a new model for multi-class MI-BCI data classification. The approach encompasses two main stages: feature extraction and fuzzy classification. In the feature extraction phase, a common spatial pattern algorithm is employed to extract significant discriminant features from the multi-class data. The obtained features serve as inputs to a fuzzy classifier that is designed by fusing the fuzzy standard additive model and particle swarm optimization method. The proposed fuzzy model aims to handle uncertainty, noise, and outliers existed in MI-BCI data. Experimental studies are carried out using two benchmark BCI competition data sets. Our results show the effectiveness of the fuzzy classifier against its competing techniques. The proposed model paths the way for paralysed or disordered patients to use movement imagery to control different assistive devices in their daily activities.

INDEX TERMS Brain computer interface, common spatial pattern, electroencephalography (EEG), fuzzy systems, motor imagery, particle swarm optimization.

I. INTRODUCTION

BCI is a system that creates a direct communication between the human brain and a hardware device [1]. An MI-based BCI system replaces the use of muscle related movements with device commands generated by translating brain signals. Such systems have enormous potential when using in locked-in patients such as after stroke or movement related disorders for rehabilitation purposes [2]–[4]. In addition to health-care, other non-medical applications of BCI systems include gaming, human-computer interaction, and human performance monitoring [5]–[7].

When it comes to assistive technology using BCI systems, the MI-based BCI systems have gained a lot of attention from the BCI research community. MI-based systems are mainly used in two application categories: as replacement of motor action or as a rehabilitation system helping to restore normal motor action [8]. In the former, a locked-in patient can communicate or control a device such as a wheel chair or prosthesis through brain activities based on movement imagery of hands, feet, fingers or tongue.

The key components of a BCI system are shown in Fig. 1. The most common input is information of the brain activity through electroencephalography (EEG) signals. EEG is an imaging modality, which records the brain's electrical signals using scalp mounted electrodes. Due to its non-invasive nature, high time-resolution and low-cost, EEG is preferred over other imaging techniques such as electrocorticography (ECoG), intracortical neuron recording and functional magnetic resonance imaging (fMRI). The EEG input is then

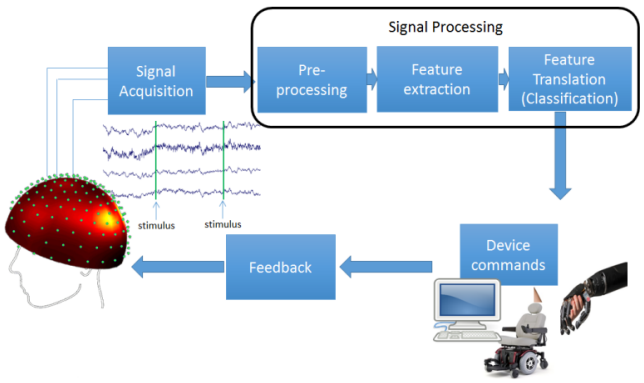

FIGURE 1. The main elements of a BCI system.

processed through a signal processing pipeline including signal pre-processing, feature extraction, and feature classification. The classifier output is finally converted to a device command.

In the case of MI-BCI systems, the EEG signals carry uniquely characterised electro-physiological information referred to as sensorimotor rhythms (SMR). The brain oscillations in the mu band (7-13 Hz) and beta band (13-30Hz) belong to SMRs. During actual motor activities, due to the intention of movement or MI, an amplitude modulation of SMRs takes place [9]. This modulation occurs in two forms: an increase in the amplitude of SMR is referred to as event-related synchronization (ERS), while the term event-related desynchronization (ERD) refers to suppression of amplitude of the SMR [10]. During MI, the mu and beta rhythms show an immediate ERD, followed by ERS in the beta band, while the mu rhythms are still suppressed [6], [10].

The MI-BCI systems utilise SMRs as control signals in the system. The modulation of SMRs is decoded by the BCI system during the feature extraction stage. Feature extraction refers to deriving the discriminative patterns from the brain signal to differentiate between different MI classes. Towards this end, various feature extraction techniques including band power, principle component analysis (PCA) and independent component analysis (ICA) have been used [11]. Time-frequency methods such as fast-Fourier transform (FFT), autoregressive modelling (AR) and wavelet transform (WT) and their variants, e.g. short-time FFT, adaptive autoregressive modelling (AAR) [12] and wavelet packet transform (WPT) [13] also have been used to handle inherent non-stationarity in the EEG signals. These methods have been shown to capture the SMR modulation effectively in MI-BCI research.

Once the features are derived, the classification stage utilises the CSP features to recognise the class a user's movement imagery belongs to. In offline BCI systems, during a training phase, the classifier learns the boundaries in a pre-labelled feature space. Then, in the evaluation or test phase, the trained classifier is used to recognise the class the new (unseen) features belong to. Many classification algorithms are available in the literature, e.g. linear discriminant analysis (LDA) [11], support vector machines (SVM) [14], naïve Bayes (NB) [15], k-nearest neighbour (KNN) [15], artificial neural networks (ANN) [16], and fuzzy logic [4], [17]–[20].

Specifically, Fabien *et al.* [17] extracted power features from both electrodes C3 and C4 in alpha and beta bands to form the feature sets as inputs of a fuzzy system for two-class (binary) MI classification. Xu *et al.* [18] used WT to construct feature sets based on a three-level decomposition tree where sub-bands of the wavelet tree corresponding to mu and beta rhythms of channels C3 and C4 were selected. The extracted feature sets were fed into a binary fuzzy SVM classifier with the radial basis function kernel. On the other hand, Prasad *et al.* [4] implemented a two-class type-2 fuzzy logic classifier to distinguish left and right imagination tasks using features that were extracted based on estimates of the spectral power of C3 and C4 signals within the adjusted mu and beta bands. Likewise, Herman *et al.* [20] designed a type-2 fuzzy logic classifier for binary MI task classification where time frequency representation, namely the short time Fourier transform, of EEG signals was used for feature set construction. Particularly, WT was used as the feature extraction method and fuzzy models were proposed for classifying EEG data in our previous studies [21], [22].

In terms of feature extraction, there is a distinguished difference between feature set in this paper and those of the aforementioned studies, which were based on spectral power of EEG signals. In contrast, in this study, we extract the spatial patterns derived from the common spatial pattern (CSP) algorithm to form the feature set. The CSP algorithm has an advantage that its underlying principle has close correspondence to the ERD/ERS phenomenon during MI. During MI, ERS and ERD show ipsilateral and contralateral spatial variation, respectively. The CSP algorithm designs spatial filters, which maximise the variance between two classes of input signals. Due to the proven efficiency of the CSP features, we extend the original two-class version of CSP to multi-class CSP in this paper.

In terms of classifier, the abovementioned studies, including our methods presented in [21] and [22], were limited to binary classification problems. To deal with this shortcoming, in this study, we propose a multi-class MI data classification approach using a multi-output fuzzy logic system (FLS). The fuzzy system is designed by fusing with the metaheuristic population-based particle swarm optimization (PSO) method to increase the classification performance. Several competing approaches are employed to compare with the PSO-based multi-class fuzzy system. The next section summarises CSP and its extensions to deal with multi-class MI data. Section III describes the proposed FLS and its learning algorithm using PSO. Experiments and results are respectively presented in sections IV and V, followed by conclusions in section VI.

II. COMMON SPATIAL PATTERN FEATURE EXTRACTION

A. BINARY-CLASS CSP ALGORITHM

This section describes the CSP algorithm and the feature set derivation for a two-class problem. Let C_1 and C_2 be the estimates of the covariance matrices of classes 1 and 2, respectively. The CSP algorithm makes use of the simultaneous diagonalization of the two covariance matrices, C_1 and C_2. Mathematically, this can be achieved by solving the eigenvalue decomposition problem,

$$C_1 W = (C_1 + C_2) W D \tag{1}$$

where $W \in \mathbb{R}^{N \times N}$ is the CSP projection matrix, which yields the features whose variances are optimal for distinguishing two classes of EEG measurements. The rows of W are stationary spatial filters, and the common spatial patterns can be obtained from the rows of W^{-1}.D is a diagonal matrix that contains the eigenvalues of C_1.

For each i^{th} trial of the multichannel EEG signal, $E_i \in \mathbb{R}^{N\times T}$ are transformed into a low-dimensional subspace with a projection matrix, W. N represents the number of EEG channels and T denotes the number of samples per channel. The linear transformation of the i^{th} trial is

$$Z_i = WE_i \tag{2}$$

where $Z_i \in \mathbb{R}^{N\times T}$ denotes the spatially filtered signals, which maximize the difference in the variance of two classes of EEG signals. A small subset of N spatially filtered signals is generally used for the formation of the feature vector [23]–[25]. The subset selection is based on m pairs of the first and last rows of Z_i. Let $Z_f \in \mathbb{R}^{2m\times T}$ be the first and last rows of Z_i, the variance of Z_f forms the feature vector for the i^{th} trial,

$$f_i = log\left(\frac{var(Z_f)}{\sum_{i=1}^{2m} var(Z_f)}\right) \tag{3}$$

where $f_i \in \mathbb{R}^{2m}$. In the present study, we use $m = 2$ for all experimental data sets.

B. MULTI-CLASS EXTENSION OF CSP

The multi-class extension of the originally proposed binary CSP algorithm [26], [27] decomposes the multi-class problem into several multi-class problems. The multi-class extension includes the divide-and-conquer (DC) approach, one-versus-rest (OVR), CSP within classifier, and pair-wise schemes [24], [25], [28], [29].

For the multi-class problems, we use the OVR approach. Using the binary CSP algorithm, the features that discriminate one class versus the rest of the classes are calculated. Then, the features are concatenated to form the feature vector for the i^{th} trial:

$$F_i = [f_i^1 f_i^2 \ldots f_i^{N_C}]^T \tag{4}$$

where $F_i \in \mathbb{R}^{1\times(2m\times N_C)}$, $f_i^k \in \mathbb{R}^{2m}$ denotes the binary-class features for the k^{th} class versus the rest for the i^{th} trial, N_c is the number of classes in the multi-class problem, and superscript T denotes the matrix transpose.

III. MULTI-CLASS FUZZY SYSTEM

A. FLS STRUCTURE

A FLS consists of four basic components: fuzzifier, fuzzy rule base (knowledge base), inference engine, and defuzzifier, which are generally diagrammed in Fig. 2.

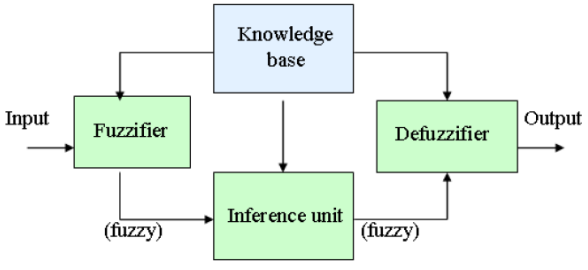

FIGURE 2. Four basic components of a fuzzy logic system.

The standard additive model, which was introduced by Kosko [30], featuring Mamdani fuzzy rules, sum-product inference, and centre of gravity (centroid) defuzzifier, is investigated in this research. This FLS contains m fuzzy rules, mapping from the if-part, R^n, to the then-part, R^p [31]. The rule is often presented in the "IF-THEN" format, i.e., "$IFX_j^1 = A_j^1\ AND\ \ldots\ AND\ X_j^n = A_j^n\ THEN\ Y_j^1 = B_j^1\ AND\ \ldots\ AND\ Y_j^p = B_j^p$". Fuzzy sets characterizing the if-part, $A_j \subset R^n$, can be chosen from several types, e.g. triangle, trapezoid, Gaussian, Cauchy, Sinc, or logistic [32]. The then-part fuzzy sets, $B_j \subset R^p$, can also be one of these types. The centroid and volume of B_j^k, denoted as c_j^k and V_j^k respectively, are employed to compute the kth output, $F^k(x)$, of the FLS using the centroid defuzzifier method:

$$\begin{aligned} F^k(x) &= Centroid\left(\sum_{j=1}^m w_j a_j(x) B_j^k\right) \\ &= \frac{\sum_{j=1}^m w_j a_j(x) V_j^k c_j^k}{\sum_{j=1}^m w_j a_j(x) V_j^k} \end{aligned} \tag{5}$$

where w_j is the weight of the jth rule, and $a_j(x)$: $R^n \rightarrow [0, 1]$ is the joint set function of the if-part, A_j, that integrates the membership degrees of n dimensions of the input, $x \in R^n$, by the expression $a_j(x) = a_j^1(x_1)\ldots a_j^n(x_n)$. To reduce the computational load, the volumes (areas), V_j^k, and the rule weights, w_j, can be fixed to unity: $V_1^k = \ldots = V_m^k = 1$ and $w_1 = \ldots = w_m = 1$. Equation (5) can be shortened to:

$$F^k(x) = \frac{\sum_{j=1}^m a_j(x) c_j^k}{\sum_{j=1}^m a_j(x)} \tag{6}$$

The FLS in this paper is designed to handle multi-class problems. Therefore, each training data sample is characterized by a vector with the length of $n + p$, where n represents the number of features (corresponding to the number of FLS inputs), and p is the number of outputs, which is equal to the number of class labels. The class labels are encoded by binary arrays where the position of 1 in the arrays indicates the label of the class, while all other positions are set to 0. The array length is thus equivalent to the number of class labels. As an example, for a four-class problem, the first class is encoded by the array [1 0 0 0], while the second class is represented by [0 1 0 0], and so on. The final output of the FLS is assigned equal to the index corresponding to the maximum value among the estimated outputs. In other words, when a test sample is fed into the multi-class FLS, all outputs are calculated, and the maximum output designates the output class label of the sample.

The vector of length $n + p$ that characterizes each training data sample is used to construct a fuzzy rule. In this study, we use a Gaussian membership function represented by its centre and width to characterize an if-part fuzzy set [32]. From (6), we see that centres and widths of the if-part fuzzy sets $A_j^1, A_j^2, \ldots, A_j^n$ and the centroids c_j^k, where $k = 1, 2, \ldots, p$, are adjustable parameters of the jth fuzzy rule. Therefore, there are $2n + p$ adjustable parameters

for each fuzzy rule. Centres of the if-part fuzzy sets and centroids c_j^k are initialized equal to corresponding elements of the vector that characterizes the training data sample. In addition, widths of the if-part fuzzy sets are initialized equivalent to the standard deviation values of the corresponding features of the training data set. After initializing, these parameters will be trained by the PSO algorithm described in the next subsection.

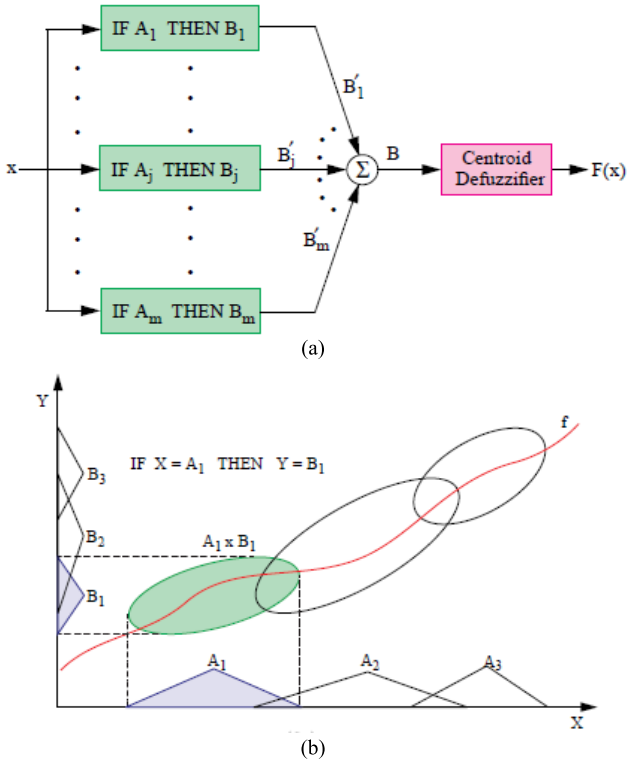

FIGURE 3. (a) Structure of the additive FLS, and (b) the graph of the approximand f is covered by fuzzy rules [33].

Fig. 3 elucidates the configuration of the additive FLS and how its fuzzy rules can cover the graph of the approximand f. This process often suffers the curse of exponential rule explosion. Parsimonious design approaches are normally adopted to cure this problem [33]. Dickerson and Kosko [34] introduced the two-step learning process for the additive FLS including unsupervised learning and supervised learning. Unsupervised learning on one hand is performed by clustering methods and its results are used to construct if-then fuzzy rules. Supervised learning on the other hand is based on the gradient descent algorithm to tune parameters of the fuzzy rules.

In [35] and [36], we used a three-step method that includes adaptive vector quantization or fuzzy c-means clustering and genetic algorithm for constructing and optimizing the FLS rule structure. In this paper, we propose the use of a stochastic learning process, i.e. PSO [37], to find the optimal if-then fuzzy rule structure as well as tuning the FLS parameters in one step. This single-step optimization scheme imposes a lower computational cost than the three-step learning method used in Nguyen *et al.* [35], [36]. The idea of using PSO is that via this stochastic search algorithm, the redundant fuzzy rules can be disregarded, and only a certain number of useful rules are reserved in the FLS. At the same time, the parameters of fuzzy rules are tuned to minimize the errors between the FLS outputs and the real values. This helps reduce computational expense and increase the generalization of the FLS. Details of the multi-class FLS learning procedure by PSO are described in the next subsection.

B. TRAINING FLS BY THE PSO ALGORITHM

PSO is one of the swarm intelligence methods based on social behaviours and dynamic movements of bird flocking or fish schooling [38]. Numerous agents (particles) constitute a swarm that moves around in the search space to look for the best solution. The position of a particle is used as a candidate solution of the optimization problem.

PSO begins by generating the initial particles and assigning them random initial velocities. It calculates the objective function, denoted as f, at each particle location, and identifies the best (lowest) function value and the best location. Each particle observes its best solution, or personal best, *pbest*, as well as the best value of any particle, or global best, *gbest*. Each particle changes its position based on its current position, its current velocity, the distance between its present position and *pbest*, as well as the distance between its present position and *gbest*. At each time step, every particle stochastically accelerates toward its *pbest* and *gbest*. Every particle then iteratively updates the particle location, velocity, and neighbours. The iteration continues until a stopping criterion is met [39].

The PSO algorithm is applied to optimize the rule structure and train the parameters of the multi-class FLS. Fuzzy rules are coded by an array whose values are ranged from 0 to 1. The array's length is equal to the initial number of fuzzy rules. A value greater than or equal to 0.5 indicates that the corresponding rule is selected, whilst a value smaller than 0.5 implies that the corresponding rule is disregarded. It means that approximately 50 per cent of fuzzy rules are chosen as the most optimal rules for building the fuzzy system. Consequently, a candidate solution (particle position) is characterized by a vector that includes two parts, as illustrated in Fig. 4. The first part is the mentioned binary array coding m fuzzy rules. The second part includes all parameters of the m fuzzy rules.

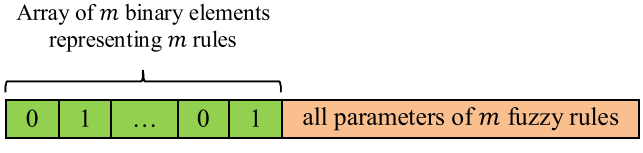

FIGURE 4. A vector characterizing a candidate solution of the PSO algorithm.

The PSO optimization problem is performed based on the training data to minimize the objective (error) function:

$$error = \sqrt{\frac{\sum_{i=1}^{N} (O_i - o_i)^2}{N_S}} \tag{7}$$

where O_i is the output calculated by FLS, o_i is the real output of the ith training data sample, whilst N_S is the total number of training data samples. The Matlab implementation of PSO, ''particleswarm,'' is used in this study, where the swarm size is set equal to 50, and the maximum number of iterations is set to 100.

IV. EXPERIMENTAL STUDIES

Two data sets consisting of multiple subjects of four-class MI EEG recordings are used in our study. The four MI classes correspond to imaginary movement of the right hand, left hand, tongue, and feet/foot. The data set IIIa [40] from the BCI competition III consists of EEG recordings of 3 subjects. The second data set comprising 9 subjects is taken from the BCI competition IV, data set IIa [25], [41]. These data sets were provided by the Laboratory of Brain-Computer Interfaces (BCI-Lab), Graz University of Technology.

The subjects participated in the studies were guided to perform four MI tasks following a trial-based visual cue when the EEG signals were recorded. A description of the data sets, including the trial-based paradigm design and the pre-processing step used in our study, is presented in the next section.

A. DATA SET IIIa FROM THE BCI COMPETITION III

1) PARADIGM

Fig. 5 (a) shows the paradigm used in the study. An empty 2s long black screen was followed by an auditory signal (beep tone) as well as the appearance of a fixation cross ('+') on the screen at $t = 2s$. The visual cue for the subject appeared at $t = 3s$, and lasted till $t = 4.25s$, for 1.25 s. An arrow pointing to the left, right, upwards, or downwards appeared, corresponding to the imagined movement of the left hand, right hand, tongue, or foot, respectively. The subjects were guided to carry out the movement imagination until the fixation cross vanished at $t = 7s$. No feedback was provided to each subject. Each trial lasted 10.24s, which included an inter-trial interval of 3.24 s between the trials.

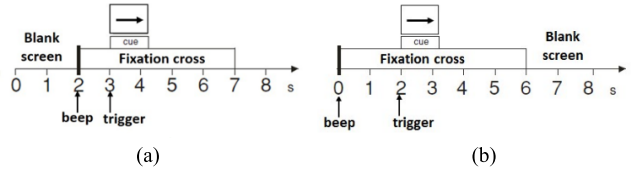

FIGURE 5. Experimental paradigm of the two data sets considered in our study (a) data set IIIa - BCI competition III, and (b) data set IIa - BCI competition IV.

Each of the four cues was showed ten times within each run in a random order. The data set acquired from subject K3 consisted of 9 runs, which resulted in total of 360 trials, with 90 trials per class. The data sets of K6 and L1 consisted of 6 runs each. For the latter two subjects, the number of trials per class was 60, giving a total of 240 trials. The visual cues corresponding to the left hand, right hand, foot and tongue imagery movement were assigned as class labels '1,' '2,' '3,' and '4' respectively. The training and evaluation data for subject K3 comprised 180 trials each, with 45 trials for each class. Regarding subjects K6 and L1, the training and evaluation data sets consisted of 120 trials each, with 30 trials for each class.

2) PRE-PROCESSING

The data sets of the three subjects (K3, K6 and L1) were recorded using a 64-channel Neuroscan EEG amplifier. Out of the 64 channels, 60 EEG electrodes were placed on the scalp as brain signal recording sites. The left mastoid served as the reference, whilst the right mastoid was treated as the ground. The data samples were captured at a sampling rate of 250 Hz, and filtered between 1 and 50Hz with a notch filter, in order to subdue line noise.

The artefact trials were included in the data sets for our analysis. A time interval of 0-8 seconds was taken for each trial. The band-pass (Wp) filter in the frequency range of 8-30Hz is applied for all trials using a 9^{th} order zero-phase Chebyshev type-II filter.

B. DATA SET IIa FROM THE BCI COMPETITION IV

1) PARADIGM

The timing used in the experiment is shown in Fig. 5(b). A short auditory signal together with the appearance of a fixation cross ('+') in a black screen at $t = 0$ marked the beginning of each trial. The black screen with the fixation cross prolonged 2s. At $t = 2s$, the visual cue was presented with an arrow pointing to the left, right, upwards or downwards as a guide for the subjects to perform imagination either left hand, right hand, tongue or foot movement correspondingly. The arrow lasted for 1.25s from 2s to 3.25s. The subjects were asked to conduct the movement imagination until the fixation cross vanished at $t = 6s$. No feedback was provided to each subject. A short break was followed before the start of the next trial.

The left hand, right hand, both feet and tongue imagination are labelled as classes '1,' '2,' '3' and '4' respectively. Two sessions were recorded for each subject on two different days. Each session comprises 6 runs. During one run, four classes were presented in a random order for 12 times. This added up to 48 trials per run and 288 trials per session, with 72 trials belonging to each class. As such, the training and evaluation data for each subject consisted of 288 trials each.

2) PRE-PROCESSING

Twenty-two EEG channels and 3 monopolar electrooculogram (EOG) channels were employed to record the data. The data were recorded with the left and right mastoids

serving as reference and ground correspondingly. The data were captured at 250Hz, notch filtered at 50Hz, and band-pass filtered between 0.5-100Hz. In this study, only the pre-processed raw data samples from the 22 EEG channels were used.

The raw data set provided in the GDF format was loaded by functions of the BioSig toolbox [42]. The data samples were extracted in a time interval of 2.5s prior to 6.5s after the onset of the fixation point (*extract_time_segment*) from both training and evaluation session data from the competition. A zero-phase 10^{th} order Chebyshev type-II filter was used to band-pass filter the extracted data in the SMR bands, i.e. 7-35Hz.

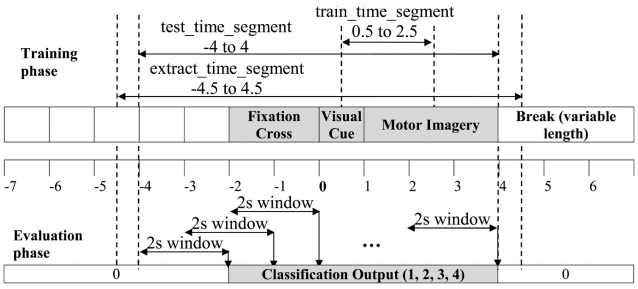

FIGURE 6. Data extraction protocol for the data set IIa. The protocol is adopted from [26].

Fig. 6 illustrates the training (*train_time_segment*) and evaluation (*test_time_segment*) segments used in this study, based on [25]. The training data for our study were taken from the interval 0.5-2.5s after the onset of the visual cue of the training session data supplied in the competition. The test data were taken from the test session data supplied in the competition in the time interval of 2s prior to 4s after the onset of the fixation cross. Because a continuous and causal output is needed in the competition, we employed a 2s time moving window, and the outcome of each window was recorded at the end of the window. The window was shifted by 10 samples in each iteration so that there were 150 windows in total [19].

V. RESULTS AND DISCUSSIONS

During the competition, both the data sets required a continuous output to evaluate the system performance. Based on this continuous output, the overall system performance was judged in accordance with the maximum accuracy and maximum kappa scores [43] achieved on the unseen evaluation data sets.

A. PERFORMANCE EVALUATION METHODS

The diagonal elements of the confusion matrix denote the true positive (TP) for each class c, $TP_c = M_{cc}$. The accuracy rate of the multi-class classification is the sum of individual class c's accuracy,

$$Accuracy = \frac{1}{N_T} \sum_{i=1}^{N_C} M_{ii} \tag{8}$$

where, N_T is the total number of trials in the evaluation data set.

TABLE 1. Confusion matrix for a N_C class problem. M_{ij} denotes the classification output for a system.

		Predicted class			
		1	2	⋯	N_C
Actual class	1	M_{11}	M_{12}	⋯	M_{1N_C}
	2	M_{21}	M_{22}	⋯	M_{2N_C}
	⋮	⋮	⋮	⋱	⋮
	N_C	M_{N_C1}	M_{N_C2}	⋯	$M_{N_CN_C}$

Cohen's kappa [43] is given as,

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \tag{9}$$

where p_0 is the observed accuracy rate and p_e is the chance expected agreement. In the case of four classes ($N = 4$) with an equal number of samples per class, $p_e = \frac{1}{N} = 0.25$.

We compare the proposed PSO-based FLS with numerous competing approaches including multi-class LDA, NB, KNN, ensemble AdaBoostM2, and SVM. We apply these algorithms by using their Matlab multi-class fitting functions, namely, fitcdiscr, fitcnb, fitcknn, fitensemble, and fitcecoc, respectively. For the ensemble procedure, we select AdaBoostM2 as the classification algorithm, decision tree as the learner, with the ensemble learning cycle number fixed to 100. Regarding the multi-class SVM, we employ the coding scheme one-vs-all with a binary SVM learner using the Gaussian kernel [19].

For a comprehensive evaluation of the performance of the proposed approach against other competing techniques, we carry out the cross-validation method on the training set of each experimental data set. The 10-fold cross-validation method is used, and the outcomes in terms of average and standard deviation are presented. To examine the statistical significance of the differences between the proposed FLS with each of the competing methods, we implement the pair-wise sign test described in [44]. This is a useful method to compare algorithms by counting the number of cases on which a method is the overall winner. The number of winning cases is distributed following a binomial distribution. When the number of cases is large, the number of wins under the null hypothesis is distributed based on the normal distribution $N(n/2, \sqrt{n}/2)$. This allows the use of the z-test to determine whether an algorithm is significantly better than another with $p < 0.05$ if the number of wins is greater than or equal to $n/2 + 1.96 \cdot \sqrt{n}/2$. Table 2 presents the critical number of wins required to obtain the significance levels of $\alpha = 0.05$ and $\alpha = 0.1$. A method performs significantly better than another when its performance is better for at least the number of cases shown in each row.

TABLE 2. Critical values for the two-tailed sign test at $\alpha = 0.05$ and $\alpha = 0.1$, adapted from [44].

#Cases	3	4	5	6	7	8	9
$\alpha = 0.05$	3	4	5	6	7	7	8
$\alpha = 0.1$	3	4	5	6	6	7	7

B. RESULTS ON COMPETITION TEST DATA SETS

1) DATA SET IIIa, COMPETITION III: 4-CLASS EEG DATA

Outputs of classifiers are provided in a continuous form and the maximum accuracy rates and kappa scores for $t < 7s$ are recorded as the performance indicator of each subject. Tables 3 and 4 present the maximum accuracy rates and kappa scores of each classification method for each subject. The continuous output is computed for each 1s window, whilst the window is shifted by 10 samples (40ms) in each iteration. This results in 175 windows and the outcome is drawn against the starting point of each window. Fig. 7 displays the accuracy rates and kappa scores attained by classifiers on the test set of subject K3. FLS shows a good performance in most of the windows. The ensemble AdaBoostM2 produces the worst performance, especially at the duration of $4s < t < 7s$ [19].

TABLE 3. Accuracy rates of classifiers - data set IIIa, competition III.

Subjects	LDA	NB	KNN	Ensemble	SVM	FLS
K3	0.900	0.883	0.894	0.878	0.883	0.911
K6	0.583	0.567	0.575	0.575	0.592	0.583
L1	0.717	0.783	0.742	0.717	0.783	0.833
Average	0.733	0.744	0.737	0.723	0.753	0.776

TABLE 4. Kappa coefficients of classifiers - data set IIIa, competition III.

Sub.	LDA	NB	KNN	Ensemble	SVM	FLS	CW [36]
K3	0.867	0.844	0.859	0.837	0.844	0.882	0.82
K6	0.444	0.422	0.433	0.433	0.456	0.444	0.76
L1	0.622	0.711	0.656	0.622	0.711	0.778	0.80
Ave.	0.644	0.659	0.649	0.631	0.670	0.701	0.79

In the last columns of Table 4, we have included the competition winning (CW) results in terms of the kappa score from [40]. When considering the average kappa score for the three subjects, the competition winner [40] is the best among the compared methods. However, it should be noted that due to the variations of different pre-processing steps, a direct comparison with the competition results is impossible. The focus of the study is to compare the proposed multi-class FLS with other competing classifiers. A fair comparison would be to keep the pre-processing and feature extraction procedures the same, and compare one classifier with its competing methods [11], [45], [46].

The classifiers on average achieve the best outcomes for subject K3, and the worst outcomes for subject K6. This is reasonable because K3 is an experienced BCI subject, while K6 is a naïve BCI subject. Tables 3 and 4 show that the

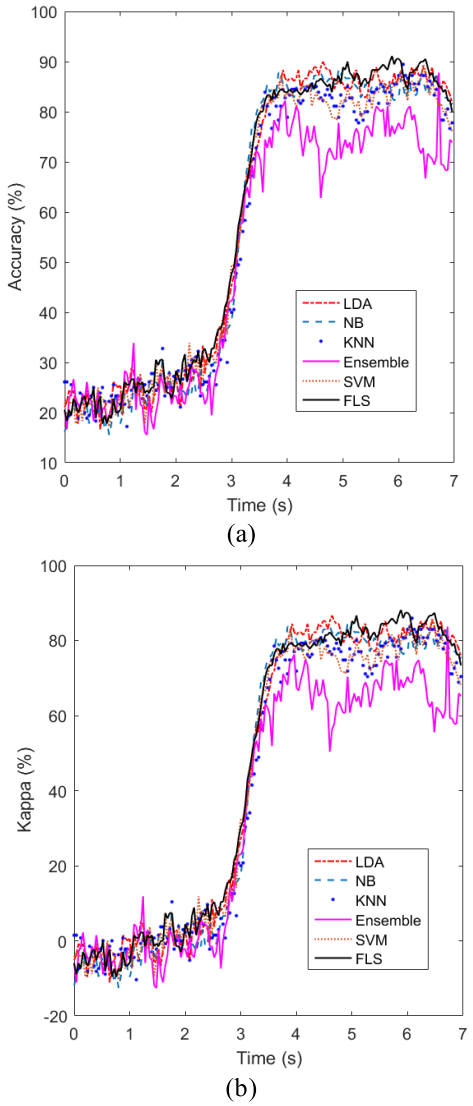

FIGURE 7. Accuracy (a) and kappa scores (b) for the test data of the subject K3. The similarity of these two charts indicates that accuracy and kappa coefficient can be replaced by each other for measuring performance of BCI systems.

proposed FLS yields the highest average performance among the competing methods. FLS obtains an average accuracy rate of 0.776 across three subjects, and an average kappa score of 0.701. SVM is the second-best method, with 0.753 average accuracy rate and 0.670 average kappa score. With the average accuracy rate of 0.723 and average kappa score of 0.631, AdaBoostM2 appears to be the worst classifier.

2) DATA SET IIa, COMPETITION IV: 4-CLASS EEG DATA

The results of each classifier in terms of the maximum accuracy rates and kappa scores on the test set of each subject are assembled in Tables 5 and 6 correspondingly.

From Tables 5 and 6, FLS yet again is the best method among six classification methods. The average accuracy rate of the FLS across 9 subjects is 0.650, while its average kappa

TABLE 5. Classifier accuracy rates - data set IIa, competition IV.

Subjects	LDA	NB	KNN	Ensemble	SVM	FLS
1	0.722	0.639	0.670	0.594	0.688	0.719
2	0.444	0.493	0.455	0.465	0.483	0.531
3	0.781	0.757	0.747	0.726	0.760	0.764
4	0.632	0.632	0.504	0.590	0.618	0.667
5	0.403	0.399	0.358	0.399	0.413	0.392
6	0.403	0.438	0.399	0.351	0.410	0.424
7	0.788	0.705	0.674	0.656	0.778	0.733
8	0.785	0.802	0.760	0.708	0.816	0.802
9	0.767	0.785	0.726	0.750	0.792	0.816
Average	0.636	0.628	0.588	0.582	0.640	0.650

TABLE 6. Kappa coefficients of classifiers - data set IIa, competition IV.

Sub.	LDA	NB	KNN	Ensemble	SVM	FLS	CW [22]
1	0.630	0.519	0.560	0.458	0.583	0.625	0.556
2	0.259	0.324	0.273	0.287	0.310	0.375	0.310
3	0.708	0.676	0.662	0.634	0.681	0.685	0.704
4	0.509	0.509	0.338	0.454	0.491	0.556	0.444
5	0.204	0.199	0.144	0.199	0.218	0.190	0.222
6	0.204	0.250	0.199	0.134	0.213	0.232	0.199
7	0.718	0.607	0.565	0.542	0.704	0.644	0.606
8	0.713	0.736	0.681	0.611	0.755	0.736	0.759
9	0.690	0.713	0.634	0.667	0.722	0.755	0.722
Ave.	0.515	0.504	0.451	0.443	0.520	0.533	0.502

score is 0.533. SVM achieves the second-best performance with an average accuracy rate of 0.640, and an average kappa score of 0.520. The ensemble AdaBoostM2 method obtains the lowest accuracy and kappa score of 0.582 and 0.443, respectively.

The last column of Table 6 contains the kappa scores of the competition IV winning method reported in [25]. The results of the winner of the BCI competition IV data set IIa are based on a modified-version of the base CSP algorithm, namely the filter bank common spatial pattern (FBCSP) algorithm. Unlike the CSP algorithm, which applies wide band filtering on the EEG recordings in the SMR range (7-30Hz), the FBCSP algorithm performs subject specific frequency band selection. This has shown to increase the BCI system performance as compared with using the CSP features, owing to more relevant features are selected in the classification stage. Since our focus is on the classifier stage, we have employed the wide band filtered CSP features in this study.

It is worth noting that the FLS has outperformed the winner of competition IV. The average kappa score of the FLS across 9 subjects is 0.533, which is much higher than 0.502 from the competition winner (Table 6).

To further evaluate the classifiers, we report in Figs. 8 and 9 the processing time of classifiers including training and classifying the entire test set of the data set IIIa, competition III and data set IIa, competition IV, respectively. It is noticed that the time taken by classifiers for data set IIIa is significantly higher than that for data set IIa. This is because in data set IIIa, the classifiers need to be trained in each of 175 windows whilst in data set IIa, classifiers just need to be trained once

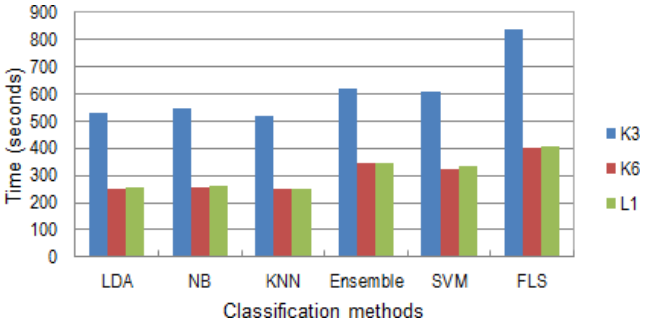

FIGURE 8. Processing time of classifiers for subjects K3, K6 and L1 in data set IIIa, competition III.

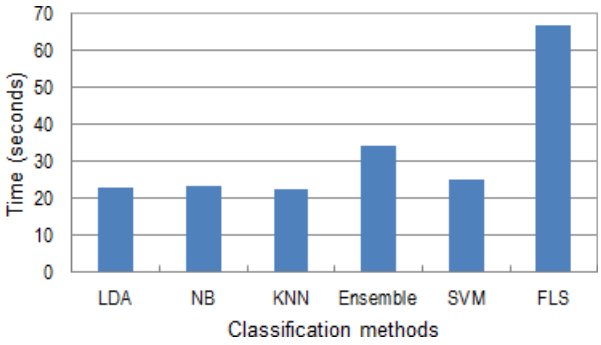

FIGURE 9. Average processing time across 9 subjects of classifiers in data set IIa, competition IV.

and they are then used to evaluate for all 150 windows. In Fig. 8, we present processing time for each subject because data of subject K3 contain 360 trials whilst data of subjects K6 and L1 contain only 240 trials each (see subsection IV.A). In Fig. 9, we report the average processing time of classifiers across 9 subjects because data of every subject contain the same number of trials: 288 for training and 288 for evaluation (see subsection IV.B). In general, the time taken by FLS is relatively higher than that by the competing methods. This is a disadvantage of the proposed FLS. However, it can be acceptable because the proposed method yields higher accuracy, and the training process is normally performed offline as it had been done in BCI competitions [1], [25], [40], [41].

C. CROSS-VALIDATION RESULTS

1) DATA SET IIIa COMPETITION III: 4-CLASS EEG DATA

The training data set of subject K3 encompasses 180 trials, with 45 trials in each class. The training sets of subjects K6 and L1 include 120 trials each, with 30 trials for each class. The 10-fold cross-validation procedure is applied to these training sets and the average accuracy, kappa scores and their standard deviations (in brackets) are presented in Tables 7 and 8 respectively.

Across 3 subjects, the FLS achieves the highest performance with an average accuracy rate of 0.865, and an average kappa score of 0.899. NB is the second-best method, with an average accuracy rate of 0.810, and an average kappa score

TABLE 7. Cross-validation accuracy results - data set IIIa, competition III.

Subjects	LDA	NB	KNN	Ensemble	SVM	FLS
K3	0.963 (0.039)	0.919 (0.064)	0.926 (0.035)	0.919 (0.055)	0.933 (0.042)	0.918 (0.082)
K6	0.611 (0.059)	0.678 (0.133)	0.678 (0.097)	0.589 (0.075)	0.633 (0.075)	0.756 (0.115)
L1	0.833 (0.059)	0.833 (0.059)	0.756 (0.070)	0.867 (0.070)	0.811 (0.105)	0.922 (0.092)
Average	0.803 (0.052)	0.810 (0.085)	0.786 (0.068)	0.791 (0.066)	0.793 (0.074)	0.865 (0.096)

TABLE 8. Cross-validation kappa results - data set IIIa, competition III.

Sub.	LDA	NB	KNN	Ensemble	SVM	FLS
K3	0.972 (0.029)	0.939 (0.049)	0.944 (0.026)	0.939 (0.041)	0.950 (0.032)	0.939 (0.061)
K6	0.708 (0.044)	0.758 (0.100)	0.758 (0.073)	0.692 (0.056)	0.725 (0.056)	0.817 (0.086)
L1	0.875 (0.044)	0.875 (0.044)	0.817 (0.053)	0.900 (0.053)	0.858 (0.079)	0.942 (0.069)
Ave.	0.852 (0.039)	0.857 (0.064)	0.840 (0.051)	0.844 (0.050)	0.844 (0.056)	0.899 (0.072)

of 0.857. In terms of stability, LDA produces the smallest standard deviation values, with 0.052 and 0.039 for the accuracy and kappa score, respectively. Comparing among three subjects, the classification methods attain the highest performance for subject K3 and the lowest performance for subject K6. This is consistent with the results achieved on the test sets of the competition data.

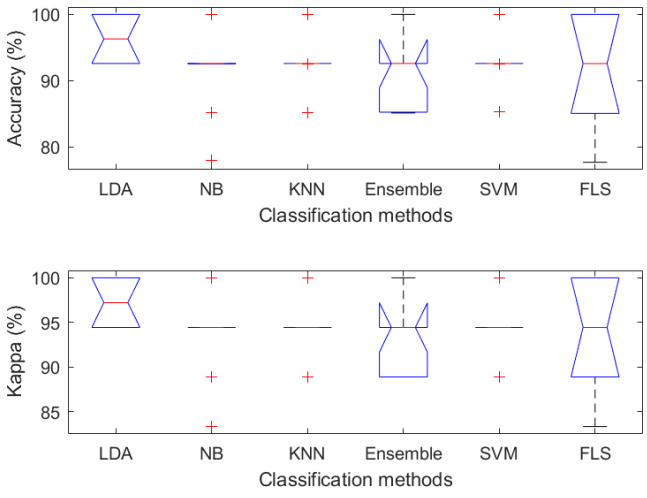

FIGURE 10. Box plot for K3 (the easiest data-processing subject among 3 subjects).

Figs. 10 and 11 present the box plots showing distributions of the cross-validation results for subjects K3 and K6, respectively. The two-tailed sign test fails to derive a statistically significant dominance of the FLS in this cross-validation experiment. It is because FLS wins only 2 cases out of 3 cases against its competing methods (see Table 2). However, for subject K6 (the most naïve subject in terms of the BCI experience level), FLS obtains the best performance among the competing methods although this is the most difficult

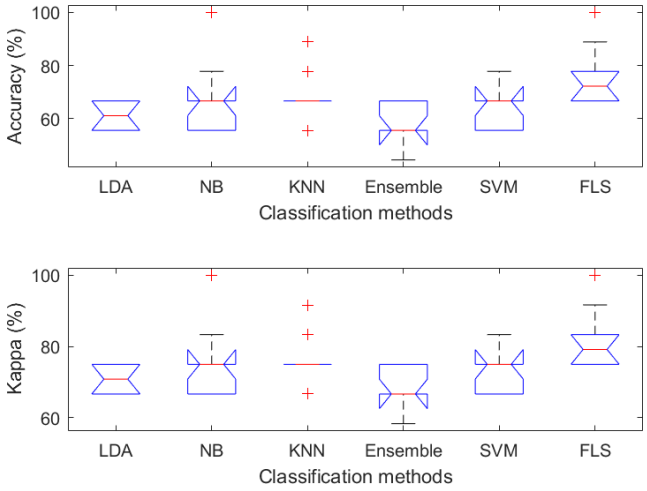

FIGURE 11. Box plot for K6 (the most difficult data-processing subject among 3 subjects).

classification data set among 3 subjects. The same implication can be derived from subject L1, who has the medium level of BCI experience, i.e. FLS performs better than the competing methods. It is obvious that when more outliers and noise exist in the EEG data, the FLS performs better than other methods, due to its capability of handling these uncertainties.

2) DATA SET IIa, COMPETITION IV: 4-CLASS EEG DATA

In cross-validation procedure, the validation data samples were taken during the time interval of 2s prior to 4s after the onset of the fixation cross from the training session data. For each of the 9 subjects, the training set contains 288 trials, with 72 trials for each class. The 10-fold cross-validation method splits the data set into a training set of 259 trials and a test set of 29 trials.

Similar to the results obtained from the previous data set, FLS attains the best performance with 0.726 average accuracy rate and 0.654 average kappa score. KNN and ensemble AdaBoostM2 depict the lowest average accuracy rate and kappa score, namely, 0.680 and 0.593, respectively. LDA and SVM show the equivalent mediocre performance with an accuracy rate of 0.712 and a kappa score of 0.635. The proposed FLS performs better than KNN and ensemble AdaBoostM2, and attains more stable results than those from the competing methods. This is shown by the standard deviation results: FLS obtains the accuracy standard deviation at 0.068 while those of KNN and ensemble AdaBoostM2 are 0.079 and 0.070, respectively (see Table 9). The kappa results presented in Table 10 also show a similar trend. FLS obtains an average kappa standard deviation of 0.088, which is smaller than those of KNN and AdaBoostM2 of 0.100 and 0.089, respectively.

We present the cross-validation results in [25], which used the conventional CSP features, in the last column of Table 10. The cross-validation results of the competition winner are

TABLE 9. Cross-validation accuracy results - data set IIa, competition IV.

Sub.	LDA	NB	KNN	Ensemble	SVM	FLS
1	0.782 (0.039)	0.793 (0.053)	0.796 (0.089)	0.764 (0.070)	0.836 (0.056)	0.814 (0.053)
2	0.661 (0.074)	0.646 (0.078)	0.618 (0.067)	0.636 (0.050)	0.643 (0.051)	0.675 (0.074)
3	0.879 (0.038)	0.857 (0.041)	0.836 (0.061)	0.854 (0.039)	0.857 (0.045)	0.882 (0.067)
4	0.505 (0.096)	0.510 (0.068)	0.414 (0.140)	0.471 (0.130)	0.462 (0.105)	0.486 (0.102)
5	0.507 (0.041)	0.504 (0.064)	0.464 (0.067)	0.518 (0.070)	0.500 (0.034)	0.529 (0.060)
6	0.532 (0.068)	0.514 (0.094)	0.536 (0.075)	0.493 (0.077)	0.529 (0.063)	0.557 (0.063)
7	0.811 (0.048)	0.782 (0.068)	0.789 (0.074)	0.732 (0.059)	0.850 (0.044)	0.843 (0.068)
8	0.889 (0.054)	0.907 (0.045)	0.857 (0.061)	0.807 (0.074)	0.893 (0.045)	0.893 (0.077)
9	0.846 (0.058)	0.821 (0.077)	0.796 (0.072)	0.843 (0.066)	0.836 (0.061)	0.857 (0.051)
Ave.	0.712 (0.057)	0.704 (0.065)	0.679 (0.079)	0.680 (0.070)	0.712 (0.056)	0.726 (0.068)

TABLE 10. Cross-validation kappa results - data set IIa, competition IV.

Sub.	LDA	NB	KNN	Ensemble	SVM	FLS	CW [22]
1	0.760 (0.052)	0.724 (0.070)	0.729 (0.119)	0.686 (0.093)	0.781s (0.075)	0.752 (0.070)	0.644 (0.064)
2	0.548 (0.099)	0.529 (0.104)	0.491 (0.090)	0.514 (0.067)	0.524 (0.067)	0.567 (0.099)	0.423 (0.056)
3	0.838 (0.051)	0.810 (0.055)	0.781 (0.082)	0.805 (0.052)	0.810 (0.059)	0.843 (0.090)	0.797 (0.070)
4	0.505 (0.096)	0.510 (0.068)	0.414 (0.140)	0.471 (0.130)	0.462 (0.105)	0.486 (0.102)	0.365 (0.053)
5	0.343 (0.054)	0.338 (0.085)	0.286 (0.090)	0.357 (0.093)	0.333 (0.045)	0.371 (0.080)	0.215 (0.046)
6	0.376 (0.091)	0.352 (0.125)	0.381 (0.100)	0.324 (0.102)	0.371 (0.083)	0.410 (0.085)	0.280 (0.049)
7	0.748 (0.064)	0.710 (0.091)	0.719 (0.099)	0.643 (0.079)	0.800 (0.059)	0.791 (0.090)	0.629 (0.064)
8	0.852 (0.073)	0.876 (0.060)	0.810 (0.081)	0.743 (0.098)	0.857 (0.059)	0.857 (0.103)	0.774 (0.069)
9	0.795 (0.078)	0.762 (0.103)	0.729 (0.095)	0.791 (0.088)	0.781 (0.082)	0.810 (0.067)	0.719 (0.067)
Ave.	0.635 (0.073)	0.623 (0.085)	0.593 (0.100)	0.593 (0.089)	0.635 (0.071)	0.654 (0.088)	0.5381 (0.060)

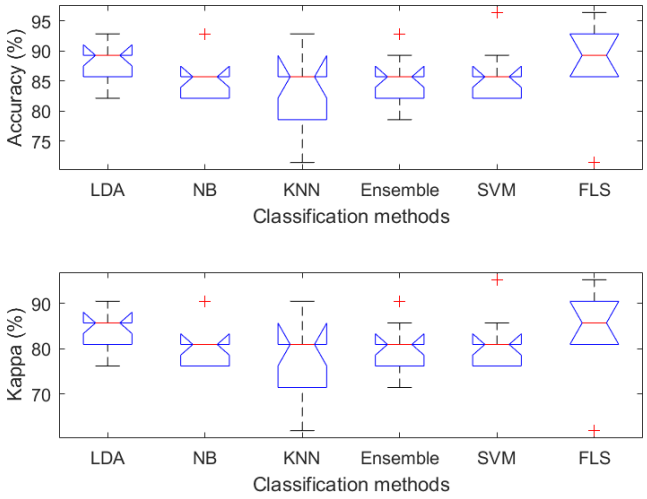

FIGURE 12. Distributions of cross-validation results on subject 3 (the best performance subject).

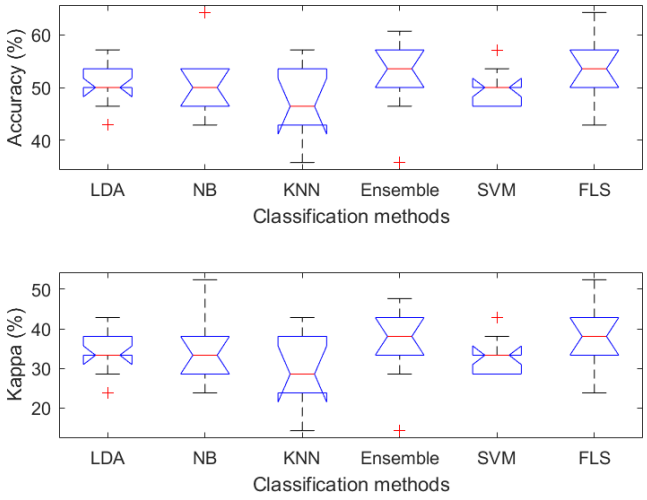

FIGURE 13. Distributions of cross-validation results on subject 5 (the worst performance subject).

dominated by those of the FLS and other competing classifiers in our study.

Figs. 12 and 13 display the box plots of classifier performance with respect to the best and the worst subjects, which are subjects 3 and 5 respectively. In line with the results in Tables 9 and 10, FLS achieves the highest performance for subject 3, and ensemble AdaBoostM2 yields the best performance for subject 5.

TABLE 11. Summary of statistical sign tests between FLS and competing methods.

FLS	LDA	NB	KNN	Ensemble	SVM
Wins (+)	8	7	9	9	6
Loses (-)	1	2	0	0	3
Detected differences (α)	0.05	0.1	0.05	0.05	–

Table 11 summarises the statistical pairwise comparison results between the proposed FLS and each competing method. FLS demonstrates a significant dominance over LDA, KNN and ensemble AdaBoostM2, at the significance level of $\alpha = 0.05$, and over NB at the significance level of $\alpha = 0.1$. Comparing between FLS and SVM, there are 6 cases where FLS performs better than SVM, and one case (i.e. subject 8) where FLS and SVM produce an equivalent performance, with the accuracy rate of 0.893 and kappa score of 0.857 (see Tables 9 and 10). However, we view FLS is inferior to SVM in the case of subject 8 because it has larger standard deviations as compared with those of SVM, i.e., 0.077 against 0.045 and 0.103 against 0.059 in terms of accuracy and kappa score, respectively. However, FLS outperforms KNN and ensemble AdaBoostM2 for all 9 subjects.

VI. CONCLUSIONS

This paper has introduced the fusion of an FLS with the PSO algorithm as a classifier for handling multi-class BCI-based EEG data. Multi-class CSP has been used as the feature extraction method to extract significant features as the inputs to different classifiers. The proposed FLS outperforms other competing classification methods, including LDA, NB, KNN, AdaBoostM2 and SVM, in the experimental studies.

The learning process of an FLS is generally time-consuming, especially for a multi-output structure. Owing to its robust optimization capability, PSO has served as an efficient learning algorithm for the proposed multi-class FLS. The application of PSO has reduced the computational expense of the multi-class FLS and improved its classification capability. The results of the cross-validation method on two benchmark MI data sets have confirmed the effectiveness of the proposed FLS in this study.

This research has demonstrated that CSP is able to deconstruct complex signals and handle noise of MI-based EEG data. In addition, the uncertainty handling capability of the proposed FLS has been confirmed. From the experimental results, it is strongly believed that our approach has a great potential to deal with data collected from different types of BCI applications, e.g. in the medical field, neuro-ergonomics or smart environments, games or entertainment sectors.

The use of CSP feature extraction for MI data has been successfully verified with various classifiers. As mentioned in Eq. (4), we chose $m = 2$ and therefore the feature set includes 16 features. For further research, it would be useful to increase the value of m and generate a larger number of features. Once more features are available, different feature selection methods can be explored to select the best features for classification. Another research direction would be to explore more efficient feature extraction methods, for example FBCSP. The use of the FBCSP algorithm would improve the performance of CSP because it performs autonomous selection of discriminative subject-specific frequency ranges for band-pass filtering of EEG recordings.

REFERENCES

[1] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, "Characterization of four-class motor imagery EEG data for the BCI-competition," *J. Neural Eng.*, vol. 2, no. 4, pp. L14–L22, 2005.

[2] N. Birbaumer, "Breaking the silence: Brain–computer interfaces (BCI) for communication and motor control," *Psychophysiology*, vol. 43, no. 6, pp. 517–532, 2006.

[3] J. Conradi, B. Blankertz, M. Tangermann, V. Kunzmann, and G. Curio, "Brain–computer interfacing in tetraplegic patients with high spinal cord injury," *Int. J. Bioelectromagn.*, vol. 11, no. 2, pp. 65–68, 2009.

[4] G. Prasad, P. Herman, D. Coyle, S. McDonough, and J. Crosbie, "Applying a brain-computer interface to support motor imagery practice in people with stroke for upper limb recovery: A feasibility study," *J. Neuroeng. Rehabil.*, vol. 7, no. 1, p. 60, 2010.

[5] J. van Erp, F. Lotte, and M. Tangermann, "Brain-computer interfaces: Beyond medical applications," *Computer*, vol. 45, no. 4, pp. 26–34, 2012.

[6] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[7] M. Ahn, M. Lee, J. Choi, and S. C. Jun, "A review of brain-computer interface games and an opinion survey from researchers, developers and users," *Sensors*, vol. 14, no. 8, pp. 14601–14633, 2014.

[8] R. Ortner, D. C. Irimia, J. Scharinger, and C. Guger, "A motor imagery based brain-computer interface for stroke rehabilitation," *Stud. Health Technol. Informat.*, vol. 181, pp. 319–323, Sep. 2012.

[9] H. Yuan and B. He, "Brain–computer interfaces using sensorimotor rhythms: Current state and future perspectives," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 5, pp. 1425–1435, May 2014.

[10] G. Pfurtscheller and F. L. Da Silva, "Event-related EEG/MEG synchronization and desynchronization: Basic principles," *Clin. Neurophysiol.*, vol. 110, no. 11, pp. 1842–1857, 1999.

[11] 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, p. R1, 2007.

[12] I. T. Hettiarachchi, T. T. Nguyen, and S. Nahavandi, "Multivariate adaptive autoregressive modeling and Kalman filtering for motor imagery BCI," in *Proc. IEEE Int. Conf. Syst., Man, (SMC)*, Oct. 2015, pp. 3164–3168.

[13] I. T. Hettiarachchi, T. T. Nguyen, and S. Nahavandi, "Motor imagery data classification for BCI application using wavelet packet feature extraction," in *Proc. Int. Conf. Neural Inf. Process.*, Nov. 2014, pp. 519–526.

[14] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek, "Optimizing the channel selection and classification accuracy in EEG-based BCI," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 6, pp. 1865–1873, Jun. 2011.

[15] A. Sharmila and P. Geethanjali, "DWT based detection of epileptic seizure from EEG signals using naive Bayes and k-NN classifiers," *IEEE Access*, vol. 4, pp. 7716–7727, 2016.

[16] A. Ahangi, M. Karamnejad, N. Mohammadi, R. Ebrahimpour, and N. Bagheri, "Multiple classifier system for EEG signal classification with application to brain–computer interfaces," *Neural Comput. Appl.*, vol. 23, no. 5, pp. 1319–1327, 2013.

[17] L. Fabien, L. Anatole, L. Fabrice, and A. Bruno, "Studying the use of fuzzy inference systems for motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 2, pp. 322–324, Jun. 2007.

[18] Q. Xu, H. Zhou, Y. Wang, and J. Huang, "Fuzzy support vector machine for classification of EEG signals using wavelet-based features," *Med. Eng. Phys.*, vol. 31, no. 7, pp. 858–865, 2009.

[19] T. Nguyen, I. K. A. Hettiarachchi, S. M. Salaken, A. Bhatti, and S. Nahavandi, "Multiclass EEG data classification using fuzzy systems," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6.

[20] P. A. Herman, G. Prasad, and T. M. McGinnity, "Designing an interval type-2 fuzzy logic system for handling uncertainty effects in brain–computer interface classification of motor imagery induced EEG patterns," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 1, pp. 29–42, Feb. 2017.

[21] T. Nguyen, S. Nahavandi, A. Khosravi, D. Creighton, and I. Hettiarachchi, "EEG signal analysis for BCI application using fuzzy system," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–8.

[22] T. Nguyen, A. Khosravi, D. Creighton, and S. Nahavandi, "EEG signal classification for BCI applications by wavelets and interval type-2 fuzzy logic systems," *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4370–4380, 2015.

[23] C. Guger, H. Ramoser, and G. Pfurtscheller, "Real-time EEG analysis with subject-specific spatial patterns for a brain-computer interface (BCI)," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 4, pp. 447–456, Dec. 2000.

[24] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing optimal spatial filters for single-trial EEG classification in a movement task," *Clin. Neurophysiol.*, vol. 110, no. 5, pp. 787–798, 1999.

[25] 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, p. 39, Mar. 2012.

[26] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.

[27] M. I. Khalid *et al.*, "Epileptic MEG spikes detection using common spatial patterns and linear discriminant analysis," *IEEE Access*, vol. 4, pp. 4629–4634, 2016.

[28] G. Dornhege, B. Blankertz, G. Curio, and K. R. Müller, "Increase information transfer rates in BCI by CSP extension to multi-class," in *Proc. 16th Int. Conf. Neural Inf. Process. Syst.* Cambridge, MA, USA: MIT Press, Dec. 2003, pp. 733–740.

[29] Z. Y. Chin, K. K. Ang, C. Wang, C. Guan, and H. Zhang, "Multi-class filter bank common spatial pattern for four-class motor imagery BCI," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Sep. 2009, pp. 571–574.

[30] B. Kosko, *Fuzzy Engineering*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.
[31] S. Mitaim and B. Kosko, "The shape of fuzzy sets in adaptive function approximation," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 4, pp. 637–656, Aug. 2001.
[32] S. Mitaim and B. Kosko, "What is the best shape for a fuzzy set in function approximation?" in *Proc. 5th IEEE Int. Conf. Fuzzy Syst.*, vol. 2, Sep. 1996, pp. 1237–1243.
[33] S. M. Zhou and J. Q. Gan, "Constructing accurate and parsimonious fuzzy models with distinguishable fuzzy sets based on an entropy measure," *Fuzzy Sets Syst.*, vol. 157, no. 8, pp. 1057–1074, 2006.
[34] J. A. Dickerson and B. Kosko, "Fuzzy function approximation with ellipsoidal rules," *IEEE Trans. Syst., Man, B, Cybern.*, vol. 26, no. 4, pp. 542–560, Aug. 1996.
[35] T. Nguyen, A. Khosravi, D. Creighton, and S. Nahavandi, "Classification of healthcare data using genetic fuzzy logic system and wavelets," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 2184–2197, 2015.
[36] T. Nguyen, A. Khosravi, D. Creighton, and S. Nahavandi, "Hierarchical gene selection and genetic fuzzy system for cancer microarray data classification," *PLOS ONE*, vol. 10, no. 3, e0120364, 2015.
[37] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*. New York, NY, USA: Springer 2011, pp. 760–766.
[38] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Human Sci.*, vol. 1. New York, NY, USA, Oct. 1995, pp. 39–43.
[39] A. R. Kumar and L. Premalatha, "Security constrained multi-objective congestion management in transactional based restructured electrical network using bacterial foraging algorithm," in *Proc. Circuits, Power Comput. Technol. (ICCPCT)*, Mar. 2013, pp. 63–67.
[40] B. Blankertz *et al.*, "The BCI competition III: Validating alternative approaches to actual BCI problems," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 153–159, Jun. 2006.
[41] M. Tangermann *et al.*, "Review of the BCI competition IV," *Frontiers Neurosci.*, vol. 6, p. 55, Jul. 2012.
[42] C. Vidaurre, T. H. Sander, and A. Schlögl, "BioSig: The free and open source software library for biomedical signal processing," *Comput. Intell. Neurosci.*, vol. 2011, Dec. 2010, Art. no. 935364.
[43] J. Cohen, "A coefficient of agreement for nominal scales," *Edu. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, 1960.
[44] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.
[45] O. AlZoubi, I. Koprinska, and R. A. Calvo, "Classification of brain-computer interface data," in *Proc. 7th Australas. Data Mining Conf.*, Nov. 2008, pp. 123–131.
[46] Koprinska, "F eature selection for brain-computer interfaces," in *Proc. sPacific–Asia Conf. Knowl. Discovery Data Mining*. Berlin, Germany: Springer, Apr. 2009, pp. 106–117.

THANH NGUYEN received the Ph.D. degree in mathematics and statistics from Monash University, Australia, in 2013. He is currently a Research Fellow with the Institute for Intelligent Systems Research and Innovation, Deakin University, Australia.

He was appointed as a Visiting Scholar with the Computer Science Department, Stanford University, California, USA, in 2015. He has published various peer-reviewed papers in the field of computational and artificial intelligence. His current research interests include applied statistics and interactive machine learning.

Dr. Nguyen was a recipient of the Alfred Deakin Post-Doctoral Research Fellowship in 2016.

IMALI HETTIARACHCHI received the Ph.D. degree from Deakin University, Australia, in 2013. She is currently a Research Fellow with the Institute for Intelligent Systems Research and Innovation, Deakin University, Australia. Her research interests include signal processing, brain compute interface systems and human performance modeling using multi modal biophysical signals, including electroencephalography, electrocardiography, and respiration.

AMIN KHATAMI received the Ph.D. degree in engineering from Deakin University, Australia. He is currently doing research in the area of intelligent systems with the Institute for Intelligent Systems Research and Innovation, Deakin University. His primary research interests include optimization, decision support, and design of the computational intelligent-based systems for complex systems.

LEE GORDON-BROWN received the B.Eng. (Mech) degree from the Swinburne University of Technology, Australia, and the Ph.D. degree from Monash University, Australia. He was an Operations Researcher in industry and with Monash University, Deakin University, and The University of Melbourne. He has consulting experience both in engineering and in econometrics, particularly in decision support, constrained optimization, and information technology. He is currently with the Royal Australian Air Force Base, East Sale, VIC, Australia.

CHEE PENG LIM has over 20 years experience in CI research, especially in theoretical development and practical application of data-based learning models. He is currently a Professor with the Institute for Intelligent Systems Research and Innovation, Deakin University, Australia. He has published over 350 refereed papers, received seven best paper awards in international conferences, and 10 prizes in international product competitions.

SAEID NAHAVANDI received the Ph.D. from Durham University, U.K., in 1991. He is currently an Alfred Deakin Professor, a Pro Vice-Chancellor (Defence Technologies), the Chair of engineering, and the Director of the Institute for Intelligent Systems Research and Innovation, Deakin University. He has published over 600 papers in various international journals and conferences. His research interests include modeling of complex systems, robotics, and haptics.

He is a fellow of Engineers Australia and the Institution of Engineering and Technology.

He is the Co-Editor-in-Chief of the IEEE Systems Journal, an Associate Editor of the IEEE/ASME Transactions on Mechatronics and the IEEE Transactions on Systems, Man and Cybernetics: Systems, and an Editorial Board Member of the IEEE Access.