



Efficient space learning based on kernel trick and dimension reduction technique for multichannel motor imagery EEG signals classification

Youkabed Amiri¹ · Hesam Omranpour¹

Received: 8 July 2022 / Accepted: 20 September 2023 / Published online: 14 October 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Electroencephalogram (EEG) signals show the electrical activity of the brain, which are one of the inputs of the brain–computer interface (BCI). The BCI provides the communication path between the brain and the computer. One of the critical applications of BCI is Motor imagery (MI). MI is a mental process that a person practices or simulates a particular movement without physically acting. BCI allows the person to communicate with their environment independently of peripheral muscles and nerves, using EEG brain signals by assistive devices such as wheelchairs, robotic arms, and computers. In this paper, a space learning concept is proposed for EEG motor imagery signal classification. Our innovation in the proposed method is to increase and then, reduce the data dimensions, which has led to learning the efficient space for signals classification. It is based on two techniques: Multi-Kernel Learning (MKL) and dimension reduction. The composite kernel is made of a combination of four kernels by The Heuristic MKL Algorithm. This algorithm uses heuristic rules to estimate the weight of kernels with high accuracy and very little computational complexity. The weight associated with each base kernel and its parameters is calculated by the Equilibrium Optimizer. Dimensions of data are reduced to avoid the curse of dimensions. In this step, the number of dimensions of reduced space and the mapping matrix are learned to reduce the dimensions of data linearly. We selected ELM, KNN, and SVM classifiers for classification. The BCI Competition dataset was used for evaluation, which consists of five subsets *aa*, *al*, *ay*, *aw*, *av*, and two classes of the right hand and right foot. The proposed method with the ELM was improved the average classification accuracy and standard deviation by 3.9% and 2.28, respectively, and achieved 91.4% accuracy. The lower standard deviation than other methods shows that our method is more robust than all other methods to subject variety. The proposed method is compared with twelve state-of-the-art methods and has shown higher accuracy than other methods such as the deep convolutional neural networks. The results show the superiority of the proposed method over other methods in the Wilcoxon signed test.

Keywords BCI system · Space learning · Kernel tricks · EEG · Optimization

1 Introduction

BCI applications have attracted the attention of many researchers recently [1–3]. One application of BCI is Motor imagery (MI) which creates a new communication method for disabled people that they have no dependence on muscle control. BCI allows the person to communicate with their environment independently of peripheral muscles and nerves, using EEG brain signals by assistive

devices such as wheelchairs, robotic arms, and computers. It improves the quality of life of people with disabilities [4, 5].

The EEG signal indicates the electrical activity of the brain. These signals are nonlinear and contain useful information about the state of the brain. These signals are recorded by electrodes placed on the scalp [5, 6]. EEG signals are used to diagnose problems such as Epilepsy [7], Emotion detection [8], Alzheimer’s [9] and lie detection [10].

MI is a mental process that a person practices or simulates a particular movement without physically acting [11]. In [12], the EEG signals of motor imagery are filtered by the CSP methods. Then, these data are transferred to the

✉ Hesam Omranpour
h.omranpour@nit.ac.ir

¹ Department of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

five-dimensional phase space. The PCA is used for feature extraction and phase space dimension reduction. Poincare sections are applied to analyze data trajectory in phase space. The parameters needed for the phase space reconstruction and the Poincare coefficients are adjusted via an evolutionary algorithm. Finally, signals are classified by SVM.

In [13], a method is proposed for EEG motor imagery classification with four classes. Features are obtained using different feature extraction methods, such as the Phase Locking Value (PLV) and the higher order Statistics (HOS) method. The nonlinear principal component analysis (NLPCA) method was employed for dimension reduction. Features are classified using backpropagation neural network (BPNN), k-Nearest neighbors (k-NN), and support vector machine (SVM). Classification accuracy by NLPCA and SVM for PLV and HOS features is higher than other classifiers.

In [14], conditional empirical mode decomposition (CEMD) is used for feature extraction. In CEMD, two conditions of correlation coefficients and relative energy occupancy rate are used to select IMFs. Then, EEG signals are reconstructed from selected IMFs. A model called 1DMSCNN is used for motor imagery EEG signals classification. This model is based on a convolutional neural network (CNN) with a multi-scale convolution feature extraction strategy. An intelligent wheelchair system is designed and implemented based on this proposed algorithm. Jana [15] is used Fourier transform (FFT) to feature extraction of the BCI Competition III dataset Iva. DNN, SVM, KNN, Naive Bayes, Random Forest and Decision Tree are used in the classification step. SVM is used with two Sigmoid and RBF kernels. KNN is used with parameter $k = 3$. The maximum depth of trees in the Decision Tree method is equal to 3. The DNN achieves the highest classification performance. Here, the DNN has two hidden layers that have 7 and 6 neurons, respectively. The BCI Competition III dataset Iva is used for evaluation.

The motivation for this paper is as follows. In the existing methods, the concepts of increasing and decreasing dimensions of data were not present simultaneously. On the other hand, the number of features in this application is not very large for each subject. Therefore, the data can be moved to another space to provide richer discriminant data and higher classification accuracy. The kernel functions commonly fall into two categories, global and local kernel functions. The global kernel function has strong extrapolation characteristics, but the local kernel function has better interpolating characteristics. The single kernel function has specific limitations. Therefore, we decided to present a new efficient space for learning based on Multi-kernel learning. Since each kernel function has parameters, we used the meta-heuristic algorithm to fine-

tune these parameters and dimension reduction. In this study, a space learning concept is proposed for multi-channel motor imagery EEG signals classification. Four kernels are employed to map the features extracted by CSP in a high-dimensional space that obtain better discriminant information. The four kernels are combined using the multi-kernel learning approach based on Meta-heuristic optimization. The weight and parameters associated with each base kernel in the composite kernel are adjusted using the optimizer. Data dimensions increase after mapping to the space generated by the kernels, and the curse of dimensions occurs. Thus, the dimensions of the data should be reduced. The dimensions number of reduced space and the mapping matrix is obtained by the optimizer.

After the dimensional reduction step, the data are fed as input to the classifier. Three classifiers of KNN, ELM, and SVM are used. The proposed method is applied to the BCI Competition III dataset Iva for evaluation. Experimental results show that the proposed method outperforms other methods. The rest of the paper is structured as follows. In Sect. 2, the basic concepts used in the paper are described. The proposed method is presented in Sect. 3. In Sect. 4, the proposed method is evaluated. Finally, the conclusion is presented in Sect. 5.

2 Literature review

In this section, the methods used in the proposed method are reviewed. CSP for feature extraction, Multi-Kernel Learning (MKL) for combining base kernels and mapping data to a higher-dimensional space are described. Equilibrium Optimizer (EO) is explained for finding optimal values for parameters. ELM, KNN, and SVM classifiers are described for EEG signal classification.

2.1 Common spatial pattern

Common spatial pattern (CSP) is a feature extraction method that extracts the features of each class by applying spatial filters to the input signals. CSP and its variants have been considered for feature extraction for the application of MI in recent years [16–18]. CSP is an effective method to extract features in the classification of BCI-related two-class movement patterns. Assuming $X_R, X_L \in R^{N \times M}$ EEG signals are recorded from two classes, N is the number of channels, and M is the number of samples per channel. First, the normalized spatial covariance matrix is calculated for each class:

$$C = \frac{EE'}{\text{trace}(EE')} \tag{1}$$

Trace (.) is the sum of the diagonal elements, ' is the transpose of the matrix, and E is X_L or X_R . The composite spatial covariance is obtained according to the following equation: decomposed into matrices of eigenvectors (U_c) and the corresponding diagonal matrix of eigenvalues (λ).

$$C_c = \bar{C}_L + \bar{C}_R = U_c \lambda U_c' \tag{2}$$

\bar{C}_L and \bar{C}_R are the average of the normalized covariance matrix of the right and left-hand classes. The diagonal matrix of eigenvalues λ is arranged in descending order. The whitening transformation matrix is obtained as:

$$p = \lambda^{-\frac{1}{2}} U_c' \tag{3}$$

The data are calculated using the white matrix as follows:

$$S_L = p \bar{C}_L p' \quad S_R = p \bar{C}_R p' \tag{4}$$

That S_R and S_L have common eigenvectors, and the sum of corresponding values is equal to the identity matrix:

$$S_L = B \lambda_L B', \quad S_R = B \lambda_R B', \quad \lambda_L + \lambda_R = I \tag{5}$$

The eigenvector with the largest eigenvalue corresponds to one class, and the eigenvector with the smallest eigenvalue corresponds to the other class. In the next step, the projection matrix is calculated:

$$W = B' p \tag{6}$$

The matrix W contains spatial filters that apply to the inputs and maximize the variance of the signals in the first class and minimize them in the other class at the same time. Then, it extracts first-class features from the filtered signals. So that the variance ratio between the two classes is maximized. The EEG signal of each class is filtered by the W matrix:

$$Z = WE \tag{7}$$

where E can be X_R or X_L [19, 20].

2.2 Multi-kernel learning (MKL)

The kernel function is represented as the inner product of the data points, which is interpreted as the similarity between the data points. Suppose $T = \{(x_i, y_i)\}_{i=1}^n$ is the training dataset, and n is the number of labeled features samples. x_i is i th sample, and y_i is the class label for samples where $y_i \in \{+1, -1\}$. M base kernels $\{K_1, K_2, \dots, K_M\}$ is made from samples where $K_m(x_i, x_j) = \langle \varphi_m(x_i), \varphi_m(x_j) \rangle$ is the m th basis kernel function, where transfers data to a higher dimension space (Hilbert space). The combination function consists of a set

$\mathbf{d} = (d_1, d_2, \dots, d_M)$. The MKL algorithm obtains composite kernel (\mathbf{k}_c) by tuning the parameters of this function. The combination function can be linear or nonlinear. The nonlinear function is rarely used in machine learning because it has high computational complexity.

$$\begin{aligned} \mathbf{k}_c &= \sum_{m=1}^M d_m K_m, \\ &\text{s.t. } d_m \geq 0 \\ &\sum_{m=1}^M d_m = 1 \end{aligned} \tag{8}$$

\mathbf{k}_c : Composite kernel, m th basis kernel, M : Number of base kernels and $d_m \in \mathbf{d}$ is the weight associated with m kernel is estimated by MKL algorithms [21–23].

2.3 Equilibrium optimizer (EO)

Like most metaheuristic algorithms, Equilibrium Optimizer (EO) uses the initial population to start the optimization process. In EO, each particle acts as a search agent with its position. There is no knowledge about the equilibrium state at the beginning of the optimization process. Four particles are considered as candidates to provide a particle search pattern. EO sets a large number for the fitness value of the Equilibrium candidates. The fitness value is calculated for population particles and compared with the fitness value of four candidate particles. If the fitness value of the selected particle is less than the candidate particle, the two particles will be replaced. Finally, the average of four candidate particles is calculated. The four candidate particles with their average construct the equilibrium pool set. The particle position updates according to the best available solutions (equilibrium candidates) until reaching the equilibrium state finally. The equilibrium state is the most optimal state of the algorithm, which is the final convergence state of the algorithm [24].

2.4 Classifier

SVM: Support vector machines (SVM) searches a hyperplane for data classification. This method selects the optimal the hyperplane that has the maximum margin with the data. Margin is the distance between the nearest examples and the hyperplane [8].

KNN: K-nearest neighbor (KNN) is one of the classifiers used by many researchers for EEG signals classification [25–27]. For a new test sample X , KNN finds the K nearest neighbor of X in the training set with distance metric. From this, K sample is done voting to determine the label of sample X . The X label is equal to the label with the maximum number of votes [28].

ELM: Extreme learning machine (ELM) is modeled as follows [29, 30]:

$$\sum_{j=1}^L \beta_j h(\mathbf{X}_i) = y_j, \quad i = 1, \dots, N \tag{9}$$

L : Hidden nodes, $h(\mathbf{X}_i)$: nonlinear feature map, β_j : the vector of weight that connects the j th hidden neuron to output neurons. Different functions are used to map fea-

dimensions problem. The dimension of data is reduced through the optimizer. The number of dimensions of reduced space and the mapping matrix are obtained using the optimizer. In phase 4, the data are used as the classifier input to determine the test sample label. Also, the code of the activation function and the number of hidden layer neurons in the ELM classifier is determined by the optimizer.

Algorithm 1. Algorithm of the proposed method

Input: EEG Data

Output: Output class of the right hand - the right foot

Phase1: pre-processing (8-30 HZ Band-pass filtering)

Phase2: Feature extraction (CSP)

Phase 3: Space learning: (Feature Map with kernels, parameters tuning and Dimension Reduction (EO))

Phase 4: classification: (SVM- ELM- KNN)

tures in ELM, such as sigmoid. The hidden layer output matrix (\mathbf{H}) is calculated as follows:

$$H(a_1, \dots, a_L, b_1, \dots, b_L, x_1, \dots, x_N) = \begin{pmatrix} h_1(x_1) & \dots & h_L(x_1) \\ \vdots & \ddots & \vdots \\ h_1(x_N) & \dots & h_L(x_N) \end{pmatrix} \in \mathbb{R}^{N \times L} \tag{10}$$

$a_j = [a_{j1}, a_{j2}, \dots, a_{jD}]^T \in \mathbb{R}^D$ is a weight vector that connects the input neurons to j th hidden neuron. b_j is the j th hidden node bias. These two parameters are calculated randomly. The output weight vector $\tilde{\beta}$ is calculated as follows:

$$\tilde{\beta} = H^\dagger \mathbf{y} \tag{11}$$

The symbol \dagger is represented by the Moore–Penrose generalized inverse and $\mathbf{y} = [y_1, \dots, y_N]^T$. The ELM decision function for a new test \hat{X} is as follows:

$$y = \text{sign}\left(h(\hat{X})\tilde{\beta}\right) \tag{12}$$

3 The proposed method

The proposed method consists of four phases. These phases are shown in Algorithm 1. First, the input signals are pre-processed. In phase 2, features are extracted by CSP. In the next phase, the data are mapped to another space using a combination of multi-kernels. The parameters and weight related to each kernel are calculated by the Meta-heuristic optimization method. Data mapping in the new space increases the dimensions of the data and causes the curse of

3.1 Pre-processing

By selecting the data from the appropriate frequency band, signals are obtained that lead to better feature extraction and more accurate classification. At this phase, the primary EEG signals are filtered by a Butterworth band-pass filter (BPF) in the range of 8–30 Hz. It leads to finding useful data for motor imagery. Also, BPF is used to remove low and high-frequency noise [31–33]. Studies are shown that MI-related event-related desynchronization (ERD) and event-related synchronization (ERS) phenomena occur in $\alpha(8–13)$ and $\beta(13–30)$ rhythms [34, 35]. ERD refers to a decrease in rhythmic activity associated with motor events. ERS refers to an increase in rhythmic activity [36].

Butterworth band-pass filter is performed by Transfer function coefficients (a, b) to the input signal. These coefficients are row vectors of length $2n+1$ for band-pass filters. n is the filter order. The order filter is equal to the maximum delay used to generate the output sample. Transfer function $H(z)$ is as follows [37]:

$$H(z) = \frac{b(1) + b(2)z^{-1} + \dots + b(n+1)z^{-n}}{a(1) + a(2)z^{-1} + \dots + a(n+1)z^{-n}} \tag{13}$$

As a result:

$$Y(z) = H(z)X(z) \tag{14}$$

$Y(z)$ is z-transform of the filter output signal filter. $X(z)$ is the z-transform of the filter input signal X . The input signal and filtered signal are related subject *aw* right hand, and foot classes are shown in Fig. 1.

3.2 Feature extraction

In this phase, the features were extracted from filtered signals by CSP with the parameter $M = 1$. The matrix $W = [w_1, w_2, \dots, w_{2M}] \in R^{N \times 2M}$ contains spatial filters. Feature vector a from input signal E is obtained (Eq. 15). The \log operator is used to normalize the distribution and reduce the range of changes in the obtained features [29].

$$a = \log \left[\text{var} \left(\hat{Z} \right) \right] \tag{15}$$

$\text{var}(\cdot)$ is the variance. For example, the output of this step, the feature vector of the two classes for the subject aw is shown in Fig. 2.

3.3 Space learning

In this phase, a learning space is devised that is based on multi-kernel and dimension reduction techniques. This phase includes Parameter tuning, Feature map with kernels, and Dimension reduction.

3.3.1 Feature map with kernels

As shown in Fig. 2, there is a lot of overlap between the obtained features of the two classes in the previous step. So the features are mapped to another space with high dimensions using the composite kernel. The kernel composite is a combination of base kernels in Table 1.

The Linear kernel is simple and does not require parameter tuning. The RBF kernel is a local kernel with good learning ability and is compatible with many conditions such as high dimension, low dimension, and large or

small sample. It requires few parameters compared to other kernel functions. Therefore, it is comfortable for regularization. The RBF has a wide convergence domain and poor generalization ability. The Polynomial kernel is a global kernel with low learning ability and high generalization ability. If the degree of a polynomial is too high, the generalization ability decreases and the problem of overfitting may occur. By trial and error method, we decided to use the RBF kernel twice because it better resulted. The RBF has a wide convergence domain and poor generalization ability. The Polynomial kernel is a global kernel with low learning ability and high generalization ability. If the degree of a polynomial is too high, the generalization ability decreases and the problem of overfitting may occur [38, 39]. By trial and error method, we decided to use the RBF kernel twice because it better resulted. To combining base kernels, it is needed to learning the parameter and weight-related each base kernel that is adjusted with a Meta-heuristic optimizer (Figs. 3, 4).

3.3.2 Parameters tuning, dimension reduction

The Meta-heuristic MKL algorithms can estimate the weight of kernels in a composite kernel (Eq. 8). These algorithms use Meta-heuristic rules to estimate the weight of kernels [21]. The weight of the base kernels and their parameters are adjusted through EO. EO has parameters such as generation rate used to enhance EO in exploration and exploitation, local optima avoidance. It also performs better than other algorithms such as PSO, GWO, GA, GSA, SSA, CMA-ES, and LSHADE [24]. In the EO, the number of particles as the population, the maximum number of iterations and the run number are considered 40, 25, and 5,

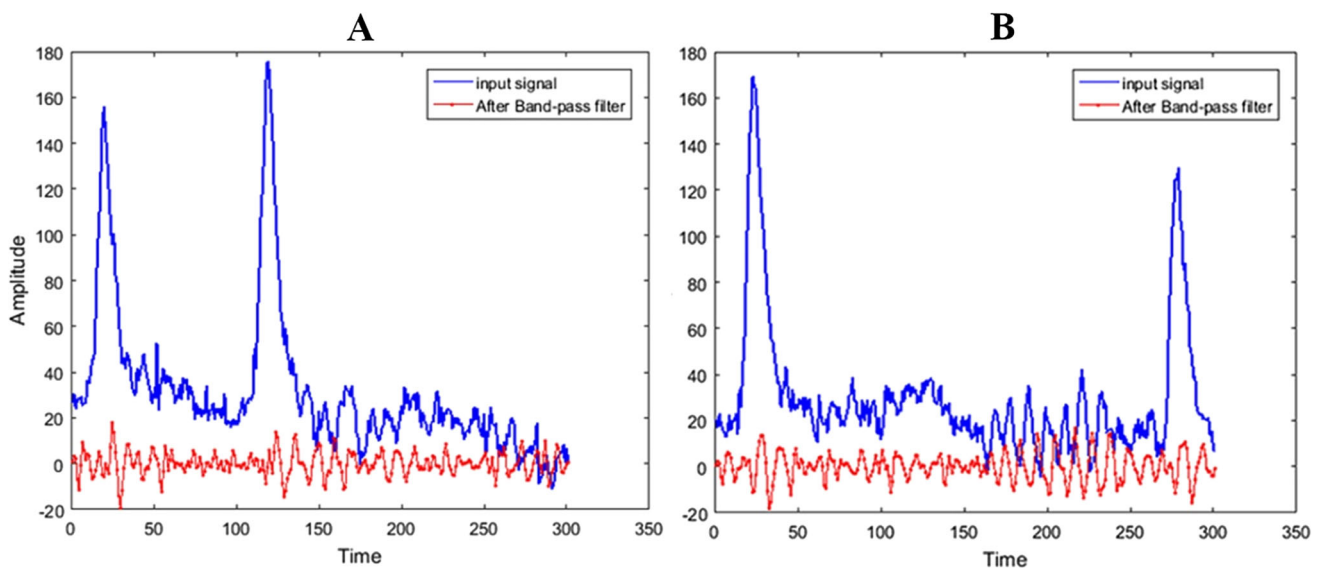


Fig. 1 A: Right-hand class signals from channel 1, B: Right-foot class signals from channel 1

respectively. The search interval is considered between $[-1, 1]$ so that the data space is the same. The dimensions of the data increase after mapping by the composite kernel. The number of reduced space dimensions and mapping matrix is obtained through EO. These are used for dimension reduction in data linearly. The number of dimensions is considered [1–10].

dimension reduction is calculated with Eq. 16. X is the output matrix of the composite kernel.

$$X_{new} = X \times W(:, 1 : \text{particle}(7)) \quad (16)$$

The fitness function for this method is summarized in Algorithm 2.

Algorithm 2. Fitness function

Input: feature vector of train data (x_i, y_i) , $i = \{1, \dots, N\}$
particle

Output: validation error

1: compute weights (particles (1:4)) & parameters (particle (5:6)) for kernel functions, (particle (7)) number of reduced Dimension, particle (8:) for W mapping matrix

2: $X = \sum_{m=1}^4 \text{particle}_m k_m(x_i, x_j)$ for $\forall i, j \in \{1, \dots, N\}$

3: Linear Dimension Reduction with $X_{new} = X \times W(:, 1 : \text{particle}(7))$

4: normalization X_{new}

5: compute validation error, train error, test error with SVM or KNN with 5×5 fold subset cross validation

3.3.2.1 Fitness function The inputs of the fitness function are the particle population and feature vectors of train data. The composite kernel matrix is made from four kernels $\{k_1, k_2, k_3, k_4\}$ from the input samples. The data are divided into three parts: Test, Train and Validation. One of the study's innovations is that the value of the fitness function is equal to the value of the validation error. Also, in this function, train and test errors are calculated. The coding of particles using SVM and KNN classifiers is equal to an array called particle that is as follows:

The first four particles are the weight associated with each base kernel in the composite kernel. Particle 5 is the

The coding of particles in fitness function using the ELM classifier is as follows:

Particle 7 denotes the code of the Activation function where presented in Table 2. For example, when Particle 7 is 1, the sigmoid function is selected. It is a changed scale in the range [1–5]. Particle 8 represents the number of hidden layer neurons that changed the scale to [2–50]. The dimension reduction is calculated linearly by Eq. 17.

$$X_{new} = X \times W(:, 1 : \text{particle}(9)) \quad (17)$$

The fitness function for this method is stated in Algorithm 3.

Algorithm 3. Fitness function

Input: feature vector of train data (x_i, y_i) , $i = \{1, \dots, N\}$
Particle

Output: validation error

1: compute weights (particles (1:4)) & parameters (particle (5:6)) for kernel functions, ELM parameters (particle (7:8)), number of reduced Dimension (particle (9)), W mapping matrix (particle (10 :))

2: $X = \sum_{m=1}^4 \text{particle}_m k_m(x_i, x_j)$ for $\forall i, j \in \{1, \dots, N\}$

3: Linear Dimension Reduction with $X_{new} = X \times W(:, 1 : \text{particle}(9))$

4: normalization X_{new}

5: compute validation error, train error, test error with ELM & 5×5 fold subset cross validation

value of the RBF parameter. It is used as an input for the two RBF kernels. The value of the polynomial is known as particle 6. Particle 7 shows the number of dimensions that data will reduce to that. Particle 8 is the W mapping matrix, which is used to the dimensions of the data linearly. The

3.4 Classification

In this step, the data are sent as input to the classifier to determine the test sample label. SVM, KNN and ELM are selected for classification. SVM is a fast classifier that is not sensitive to overtraining and high-dimensional data. It

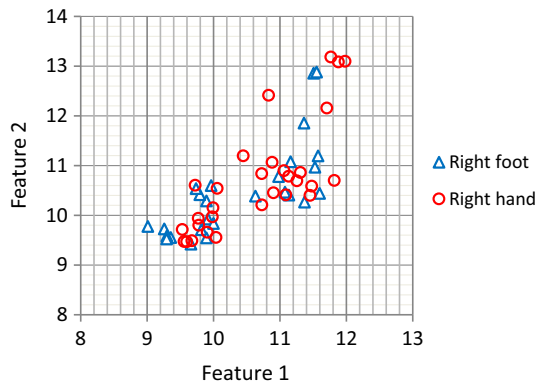


Fig. 2 The feature vector of two classes (Right foot, Right hand) for the subject *aw*

Table 1 Functions of the basic kernels used in the composite kernel

Kernel	Formula
Linear	$k(x, y) = x^T \cdot y + C$
Polynomial	$k(x, y) = (x^T \cdot y + c)^d$ <i>d</i> : polynomial degree
RBF	$k(x, y) = e^{(-\gamma \ x-y\ ^2)}, \gamma > 0$ $\ x - y\ $: norm of the distance is <i>x, y</i>

has good generalization performance. The KNN algorithm has no explicit training phase for classification, and all the work happens during prediction. It can learn nonlinear decision boundaries [40]. In this paper, the KNN classifier is used the majority voting method because it is less sensitive to noise data [28]. Here, KNN is with parameter $K = 1$. ELM has good scalability and fast learning speed. The weights of the hidden layer are tuning-free [29, 30]. In ELM, the input layer neurons are equal to the number of features. The number of output neurons is equal to the number of classes. The number of hidden layer neurons and the activation function are obtained through EO. Five activation functions are defined, as shown in Table 2.

Table 2 Activation functions used in the ELM Classifier

Code	Activation function	Formula
1	Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$
2	sinusoid	$f(x) = \sin(x)$
3	Hard Limit	$f(x) = \begin{cases} 1, x \geq 0 \\ -1, otherwise \end{cases}$
4	Triangular basis function	$f(x) = \begin{cases} 1 - abs(x), -1 \leq x \leq 1 \\ 0, otherwise \end{cases}$
5	Gaussian	$f(x) = e^{-x^2}$

4 Experimental results

4.1 Data description

BCI Competition III dataset Iva [41] is used to evaluate the proposed method. It contains EEG data from 118 electrodes with a sampling rate of 100 Hz for healthy subjects (*aa, al, aw, av, ay*). It contains EEG data measured at the sampling rate of 100 Hz from 118 electrodes with 10–20 International System of Electrodes. The EEG signals were recorded for five subjects (*aw, aa, al, ay, av*) during the imaginations of the right foot or hand movement. The number of trials was 280 for each subject [42]. Table 3 shows the number of labeled trial samples for each subject.

4.2 Experimental evaluation

In this section, the results of the proposed method are presented. The fivefold subset cross-validation method is used for the final evaluation. The data are divided into three parts test, train, and validation.

For example, the results on the subject *aw* with the KNN classifier are shown in Table 4.

In this table, the optimizer runs five times and searches for the optimal values in each run. The values of the kernel weight are considered in the range [0 1] because the weight of the kernel doesn't accept a negative value [21, 22].

	1	2	3	4	5	6	7	8
Particle	Linear kernel weight	RBF1 kernel weight	RBF2 kernel weight	Polynomial kernel weight	RBF parameter	Polynomial parameter	Dimension	W Mapping matrix

Fig. 3 Representation of particles using the KNN, SVM classifiers

	1	2	3	4	5	6	7	8	9	10
Particle	Linear kernel weight	RBF 1 kernel weight	RBF 2 kernel weight	Polynomial kernel weight	RBF parameter	Polynomial parameter	Code of Activation Function	Number of Hidden layer Neurons	Dimension	W Mapping matrix

Fig. 4 Representation of particles using ELM classifier

Table 3 Number of labeled trials for each subject for BCI Competition III dataset Iva

Subject	Number of trials	Number of Right-hand trails	Number of Right-foot trails
<i>aa</i>	168	80	88
<i>al</i>	224	112	112
<i>av</i>	84	42	42
<i>aw</i>	56	30	26
<i>ay</i>	28	18	10

Table 4 Results obtained by the proposed method with KNN classifier on subject *aw*

Run number	Optimal values obtained by Equilibrium Optimizer (EO)							Accuracy of Test (%)
	Linear weight	RBF no.1 weight	RBF no.2 weight	Polynomial weight	RBF parameter	Polynomial parameter	Dimension	
1	0.3455	0.6051	0.1231	0.0977	35	2	2	89.1
	0.9437	0.9327	0.8256	0.2896	28	3	5	100
	0.8329	0.4065	0.8624	0.613	14	2	3	98.2
	0.7642	0.0528	0.5221	0.2508	38	2	9	90.9
	0.3046	0.6458	0.7162	0.4028	13	3	7	90.9
2	0.6268	0.8791	0.4846	0.2963	37	2	3	100
	1	0.5291	1	0.2673	22	2	3	81.8
	0.8965	0.5235	0.7229	0.6732	35	2	7	89.1
	0.0217	0.3709	0.4049	0.3008	4	2	2	81.8
	0.6078	0.9311	0.2697	0.5624	6	3	2	81.8
3	0.3169	0.0734	0.2016	0.9119	19	2	4	100
	0.9398	0.8944	0.7375	0.1353	49	2	4	100
	0.7086	0.8054	0.6895	0.1618	39	3	3	100
	0.2014	0.7854	0.8831	0.0934	28	2	5	98.2
	0.6838	1	0.4331	0.6067	21	2	3	100
4	0.1424	0.0905	1	0.15	34	3	1	83.6
	0.1987	1	0.4189	0.8636	46	2	4	100
	1	0.794	1	0.5954	26	2	3	100
	0.1639	0.0707	0.65	0.1274	14	2	6	98.2
	1	0.4215	0.0085	0.1167	22	2	3	100
5	0.528	0.0749	0.8874	0.0084	11	3	1	100
	0.0700	0.3101	0.6712	0.0329	17	4	5	100
	0.5776	0.0163	0.4424	0.221	49	2	8	100
	0.0182	0.6644	0.7368	0.6617	20	2	4	90.9
	0.8004	0.183	0.2852	0.103	34	2	4	100
								Average
								95

Different experiments have shown that the higher the RBF kernel parameter, the lower its learning ability [38]. For this purpose, the value of this parameter is limited to 50 by the trial and error method. The value of this parameter is scaled to the range [1 50]. If the polynomial kernel parameter is high, the overfitting problem occurs [38]. The value of this parameter is considered by changing the scale in the range [2 4]. The Dimension column shows the number of dimensions that the data are reduced. This value

has changed the scale in the [1 10]. Another particle obtained by the optimizer is the mapping matrix (W). This matrix has dimensions to the value of Dimension. It is not presented in these tables. In the fitness function are calculated the validation error, Train, and Test in each run. Finally, the average of these errors and the accuracy of each are calculated.

Table 5 Results obtained by the proposed method with SVM classifier on subject *aw*

Run number	Optimal values obtained by Equilibrium Optimizer (EO)							Accuracy of Test (%)
	Linear weight	RBF no.1 weight	RBF no.2 weight	Polynomial weight	RBF parameter	Polynomial parameter	Dimension	
1	0.1146	0.7776	0.5897	0.0158	10	2	2	90.9
	0.0405	0.0614	0.0633	0.99	38	2	5	89.1
	0.0049	0.1844	0.1492	0.8085	30	3	9	90.9
	0.4676	0.7945	0.6849	0.428	13	2	4	90.9
	0.1918	0.2773	0.4606	0.336	11	2	4	89.1
2	0.1973	0.1989	0.0468	0.1016	26	2	2	100
	0.7391	0.8964	0.3934	0.1084	36	2	6	98.2
	0.5807	0.3292	0.7287	0.8881	26	3	4	98.2
	0.75	0.1813	0.8252	0.2032	30	3	8	100
3	1	0.3144	1	0.4313	45	2	3	98.2
	0.5146	0.6109	1	0.0107	28	3	3	100
	0.7175	0.1305	0.0684	0.5835	26	2	2	100
	0.1286	0.3084	0.4424	0.2227	43	2	5	100
	1	0.0855	1	0.1398	28	2	9	90.9
4	0.2363	0.3634	0.7502	0.2275	39	2	3	96.4
	0.954	0.9541	0.1924	0.7462	20	2	4	100
	1	0.806	0.0533	0.1847	27	3	5	100
	0.4818	0.369	0.2679	0.0011	29	3	9	100
	0.1516	0.7248	0.1217	0.5238	16	2	2	100
	0.9749	0.3824	0.6121	0.0419	42	2	4	100
5	0.8132	0.0748	0.2077	0.7834	24	2	7	90.9
	0.7757	0.1915	0.6867	0.8941	40	3	8	90.9
	0.6636	0.3311	0.277	0.8512	25	2	7	90.9
	0.4311	0.2632	0.5319	0.418	48	3	7	90.9
	0.747	0.4954	0.3712	0.1365	39	2	6	89.1
Average								95.4

The results on the subject *aw* with the SVM classifier are shown in Table 5. The descriptions of Table 5 columns are similar to Table 4.

The results on the subject *aw* with the ELM classifier are shown in Table 6. In this table, the Activation Function code column determines the activation function. A scale change is used to the range [1 5] for the value of this column. The number of hidden layer neurons column shows the optimal number of hidden layer neurons. If the number of neurons in the hidden layer is small, a classification error will occur, and if it is high, it will complicate the hidden layer. Therefore, this value is defined range [2 50] neurons by changing the scale.

The accuracy of the classification of the proposed method with three classifiers is shown in Table 7.

According to the table, the SVM has the highest accuracy on the subject *av*. ELM has higher accuracy on the subjects *aa*, *al* and *aw*. KNN has much more accuracy on

the subject *ay* with 99.5%. ELM has the highest average classifier accuracy, which is 91.4%. The KNN and ELM have the highest recall and precision values in *aa*, respectively. In the subject *al*, ELM has the highest value of recall and precision. The SVM classifier has the highest recall and precision values in the *av* subject. KNN and SVM have the highest recall and precision values in the *aw* subject, respectively. KNN has the highest recall and precision values in the *ay* subject with 99.7% and 99%, respectively. F-score was obtained 0.88 in *aa* subject by ELM and KNN. Also, KNN and SVM have an F-score of 0.99 in *ay* subject.

The time order of the training phase is $O(Iter \times n \times Tfit)$ where $Tfit$, $Iter$, n are the running time of the fitness function, the number of iterations and the number of particles, respectively. The time order is $O(Tfit)$ in the test phase. The advantages of the proposed method can be summarized as follows: New efficient space

Table 6 Results obtained by the proposed method with ELM classifier on subject *aw*

Run number	Optimal values obtained by Equilibrium Optimizer (EO)									Accuracy of Test (%)
	Linear weight	RBF no.1 weight	RBF no.2 weight	Polynomial weight	RBF parameter	Polynomial parameter	Activation Function code	Number of hidden layer neurons	Dimension	
1	0.7386	0.2982	0.5482	0.3583	48	2	3	27	3	89.1
	0.3646	0.4345	0.6885	0.9862	27	3	4	19	6	92.7
	0.7161	0.3882	0.8199	0.2169	26	3	1	12	7	89.1
	0.3978	0.3627	1	1	29	2	5	18	9	92.7
	0.3773	0.8955	0.476	0.7587	10	3	5	18	3	90.9
2	0.6768	0.0618	0.3299	0.0032	19	3	5	8	4	100
	0.5934	0.3143	0.57	0.4487	11	3	5	17	5	100
	0.2243	0.0831	0.0672	0.3732	22	2	1	16	4	94.5
	0.9172	0.5509	0.1862	0.0309	36	3	5	15	5	100
3	0.1391	1	1	0.5448	21	3	2	11	7	98.2
	0.0779	0.1452	0.525	0.6177	25	2	2	26	8	100
	0.4879	0.5772	0.9521	0.4898	10	2	5	50	10	94.5
	0.5859	0.4111	0.5625	0.6133	21	3	2	27	4	100
	0.7507	0.5889	0.6699	0.6382	36	2	4	34	1	100
4	0.1296	0.0486	0.2403	0.0779	26	2	1	28	3	89.1
	1	0.0007	1	1	39	3	2	16	5	94.5
	0.8108	0.6922	1	0.6688	23	3	5	26	5	94.5
	0.1447	1	0.3902	0.6649	39	2	1	39	2	100
	0.3509	0.8739	0.6842	0.7755	24	2	1	13	5	100
5	0.223	1	0.9352	0.1753	28	3	1	14	3	100
	0.2681	0.4861	0.7531	0.8388	40	3	5	18	5	98.2
	0.3662	0.722	0.9234	1	44	2	5	49	9	92.7
	1	0.8867	0.8106	0.6042	40	2	5	15	8	94.5
	0.5956	1	1	1	1	3	4	23	6	100
	0.0771	0.2369	0.0025	0.7973	36	3	2	4	3	92.7
										Average 95.9

Table 7 Classification performance evaluation criteria of the proposed method

Subjects	Accuracy (%)			Precision (%)			Recall (%)			F- score		
	SVM	KNN	ELM	SVM	KNN	ELM	SVM	KNN	ELM	SVM	KNN	ELM
aa	85.5	87.8	87.9	86	87.9	87.6	86	87.9	88.2	0.86	0.88	0.88
al	96.7	96.6	98.7	96.8	96.8	98.7	96.8	96.5	98.8	0.97	0.97	0.99
av	79.8	71.5	78.5	81.2	69.2	77.3	80.5	70.2	77.6	0.81	0.70	0.77
aw	95.4	95	95.9	96	94.8	95.8	95	96	95.9	0.95	0.95	0.96
ay	98.7	99.5	95.9	98.9	99	96.4	98.7	99.7	95.9	0.99	0.99	0.96
Average	91.2	90.1	91.4	91.8	89.5	91.2	91.4	90.1	91.3	0.9	0.9	0.9

learning based on Multi-kernel learning resulted in discriminant data and higher classification accuracy. The time order in the test phase is low. Also, one of the innovations is the value of the fitness function is equal to the value of

the validation error. Train error is not used as a fitness value to avoid the overfitting problem.

Table 8 Comparison of proposed methods with other methods

Methods	<i>aa</i>		<i>al</i>		<i>av</i>		<i>aw</i>		<i>ay</i>		Average Rank	Final Rank
	ACC (%)	Rank	ACC (%)	Rank	ACC (%)	Rank	ACC (%)	Rank	ACC (%)	Rank		
[29],(2018)	83.3	5	98.5	4	71.4	9	91.3	8	93.1	6	6.4	5
[43],(2019)	82.5	6	96.8	6	71.1	10	92.9	5	93.9	4	6.2	4
[44],(2020)	75.89	12	100	1	76.53	4	89.73	11	76.19	14	8.4	8
[45],(2020)	71.39	14	90.27	13	59.38	14	62.07	15	58.53	15	14.2	13
[46],(2020)	72.32	13	98.21	5	68.87	12	78.57	14	92.06	8	10.4	11
[47],(2020)	82.1	7	95	10	72.1	6	90.7	9	91.8	10	8.4	8
[48],(2021)	70	15	96.7	7	70	11	85	13	93.3	5	10.2	10
[49],(2021)	81.8	8	96.1	9	72.5	5	92.1	7	90.3	11	8	7
[50],(2021)	86.07	3	98.57	3	52.14	15	96.07	1	92.14	9	6.2	4
[51],(2022)	81.25	9	100	1	77.55	3	90.63	10	77.38	13	7.2	6
[52],(2022)	78.3	10	93.6	11	68.6	13	87.9	12	92.7	7	10.6	12
[53],(2022)	78.12	11	92.4	12	71.88	7	92.33	6	88.02	12	9.6	9
Method1	87.8	2	96.6	8	71.5	8	95	4	99.5	1	4.6	3
Method2	85.5	4	96.7	7	79.8	1	95.4	3	98.7	2	3.4	2
Method3	87.9	1	98.7	2	78.5	2	95.9	2	95.9	3	2	1

4.3 Compare with state-of-the-art methods

In this section, a comparison is made between the proposed method and the last state-of-the-art methods. Method 1 refers to the proposed method with the KNN classifier. Method 2 refers to the proposed method with the SVM classifier. Method 3 refers to the proposed method with the ELM classifier. According to Table 8, these methods are compared with twelve methods.

In [29], the multi-kernel Extreme learning machine (MKELM) method is presented for motor imagery classification. The combination of two Gaussian and polynomial kernels is used to map features extracted by CSP to the nonlinear feature space. Then, classification was performed by the ELM algorithm. In [43], a correlation-based channel selection (CCS) method is proposed to select the channels that contain more correlated information. Then, features are extracted by regularized CSP (RCSP). SVM with the RBF kernel is used for classification. In [44], a new deep architecture called the DSSMM (Deep Stacked Support Matrix Machine) is based on the principle of stacked generalization. DSSMM is constructed in a layer-by-layer technique. Each layer contains an SMM module that can maintain structural information between rows or columns in the EEG feature matrix. SMM can grasp the feature matrices' structural information. SMM can grasp the feature matrices' structural information.

In [45], P-LTCSP (PLV-modulated Local temporal common spatial patterns) method is proposed for feature extraction. P-LTCSP incorporate PLV (Phase locking value) into LTCSP. LTCSP is an effective method of

obtaining the temporally local manifold of EEG time series. PLV is applied to quantify the phase relationship between samples. PL to quantify the phase relationship between samples. PLV is used as the weight between two EEG samples. LDA classifier is used for motor imagery EEG signals classification. In [46], filter band CSP (FCCSP) is proposed for MI classification. FCCSP employs two regularization parameters in order to increase robustness and reduce the estimation variance. All EEG signal is divided into frequency sub-bands. The sub-band is divided using wavelet packet. Features were extracted from sub-bands by (Component regularized CSP) CRCSP. The final features selected by mRMR are fed to LDA for classification.

In [47], a method is proposed a bispectrum-based channel selection (BCS) for MI tasks classification. Bispectrum is a statistical analysis method used to analyze the interactions between EEG signals. Bispectrum for each channel is Computed in all trials. Channels without redundant information are selected based on the larger f-scores. F-score is based on the sum of logarithmic amplitudes (SLA) and the first-order spectral moment (FOSM) features from the bispectrum. Features extracted by CSP are classified by SVM. In [48], a binary harmony search algorithm (BHS) is proposed to channel selection. Harmony search (HS) is a new meta-heuristic optimizer. The BHS is binary coded, in which every harmony vector length is equal to the channels available number in the data set. If the decision variable holds a value of 1, the related channel is selected. The sparse representation-based classification (SRC), SVM, and LDA are performed on the

Fig. 5 Comparison of proposed Methods in terms of Accuracy with other methods

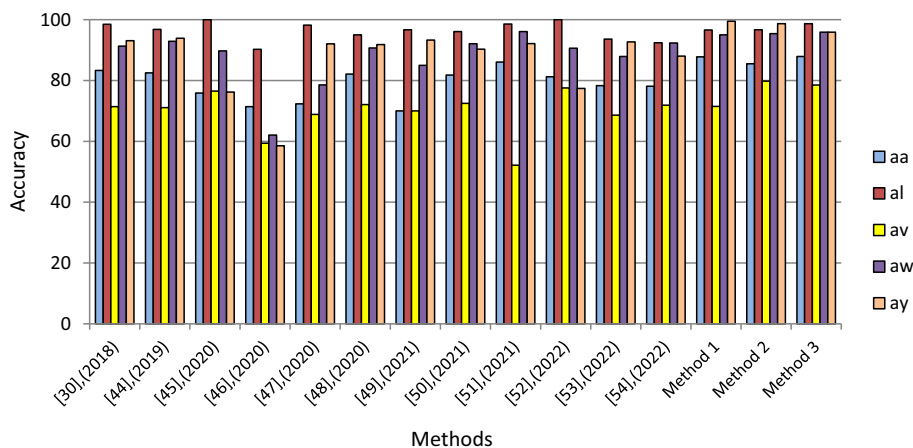


Table 9 Comparison of the average classification accuracy and Standard deviation of the proposed methods and other methods

Methods	Average		Standard deviation
	ACC (%)	Rank	
[29],(2018)	87.5	4	10.53
[43],(2019)	87.4	5	10.6
[44],(2020)	83.7	12	10.85
[45],(2020)	68.32	15	13.29
[46],(2020)	82.01	13	12.67
[47],(2020)	86.3	7	9.28
[48],(2021)	80.2	14	12.61
[49],(2021)	86.6	6	9.43
[50],(2021)	85	9	18.96
[51],(2022)	85.36	8	9.79
[52],(2022)	84.2	11	10.64
[53],(2022)	84.55	10	9.17
Method 1	90.1	3	11.24
Method 2	91.2	2	8.16
Method 3	91.4	1	8.25

CSP extracted features, which the BHS-SRC has higher accuracy.

In [49], a spatial-frequency-temporal (SFT)-3D CNN model is proposed for MI classification. The SFT-3DCNN model consists of 8 layers. First layer is the input layer. The three layers are the SFT convolution layers. Four layers are the fully connection layers and the layer of output. Novel 3D CNN with three fully connected layers is proposed for the extraction of SFT features and classification. In [50], common time–frequency-spatial patterns (CTFSP) are proposed to extract sparse CSP features from multi-band filtered EEG. First, EEG signals are pre-processed with a Butterworth band-pass filter of 8–30 Hz. Then, the EEG signals are divided into seven frequency bands. Features

are extracted from the frequency band by CSP. The most significant features are selected from frequency bands by LASSO. Classification is done by the voting result of three SVM classifiers.

The Deep Stacked Feature Representation (DSFR) method is proposed in [51]. The DSFR employed a set of feature decoding modules (FDMs). Each FDM includes a CSP and a support matrix machine (SMM). The architecture of DSFR has several layers. Each layer is an FDM, which needs to be fed with the predictions of all the previous layers and the original EEG feature to produce the EEG feature representation and prediction. In [52], the objective Firefly Algorithm (FA) is proposed to find an optimal EEG channel set. Then, the Channel set is ranked using fisher information index criteria. Regularized Common Spatial Pattern with Aggregation (RCSPA) is used for feature extraction. The RCSPA has two regularization parameters which control the bias-variance tradeoff among MI tasks. Regularized Support Vector Machine (SVM) is used for motor imagery tasks classification.

In [53], Multiobjective X-shaped Binary Butterfly Optimization Algorithm (MX-BBOA) is used to select EEG channels. The MX-BBOA method target to hold a balance between the classification accuracy and the number of channels. This method examines the butterfly’s natural behavior with dual sigmoid functions to solve the channel selection problem. Features are extracted by Multivariate Empirical Mode Decomposition (MEMD). SVM, Naive Bayes and Decision Tree were used for classification, and SVM has higher performance.

According to Table 8, Method 3, Method 2, and Method 1 have higher accuracy compared to other methods with rank 1, 2, 3, respectively. Method 3 and method 2 have higher accuracy in all subjects from [49] that used deep convolutional neural network. Method 1 has higher accuracy in four subjects from [49].

As you can see in Fig. 5, the proposed Method3 were performed better in subject *aa*. Method2 and Method1 have

Table 10 Results of Wilcoxon signed-rank test Method1 and other methods

Methods	P-value	Test number	Superior	Inferior	Similar
Method 1 vs. [29]	0.1875	5	4	1	0
Method 1 vs. [43]	0.125	5	4	1	0
Method 1 vs. [44]	0.3125	5	3	2	0
Method 1 vs. [45]	0.0625	5	5	0	0
Method 1 vs. [46]	0.125	5	4	1	0
Method 1 vs. [47]	0.125	5	4	1	0
Method 1 vs. [48]	0.125	5	4	1	0
Method 1 vs. [49]	0.1875	5	4	1	0
Method 1 vs. [50]	0.4375	5	3	2	0
Method 1 vs. [51]	0.4375	5	3	2	0
Method 1 vs. [52]	0.0625	5	5	0	0
Method 1 vs. [53]	0.125	5	4	1	0

Table 11 Results of Wilcoxon signed-rank test Method2 and other methods

Methods	P-value	Test number	Superior	Inferior	Similar
Method 2 vs. [29]	0.125	5	4	1	0
Method 2 vs. [43]	0.125	5	4	1	0
Method 2 vs. [44]	0.1875	5	4	1	0
Method 2 vs. [45]	0.0625	5	5	0	0
Method 2 vs. [46]	0.125	5	4	1	0
Method 2 vs. [47]	0.0625	5	5	0	0
Method 2 vs. [48]	0.125	5	4	0	1
Method 2 vs. [49]	0.0625	5	5	0	0
Method 2 vs. [50]	0.8125	5	2	3	0
Method 2 vs. [51]	0.1875	5	4	1	0
Method 2 vs. [52]	0.0625	5	5	0	0
Method 2 vs. [53]	0.0625	5	5	0	0

Table 12 Results of Wilcoxon signed-rank test Method3 and other methods

Methods	P-value	Test number	Superior	Inferior	Similar
Method 3 vs. [29]	0.0625	5	5	0	0
Method 3 vs. [43]	0.0625	5	5	0	0
Method 3 vs. [44]	0.125	5	4	1	0
Method 3 vs. [45]	0.0625	5	5	0	0
Method 3 vs. [46]	0.0625	5	5	0	0
Method 3 vs. [47]	0.0625	5	5	0	0
Method 3 vs. [48]	0.0625	5	5	0	0
Method 3 vs. [49]	0.0625	5	5	0	0
Method 3 vs. [50]	0.1875	5	5	0	0
Method 3 vs. [51]	0.1875	5	4	1	0
Method 3 vs. [52]	0.0625	5	5	0	0
Method 3 vs. [53]	0.0625	5	5	0	0

higher accuracy compared to other methods in *av*, *ay* subjects, respectively. Table 9 shows the comparison of the average classification accuracy of the proposed methods and other methods.

Method3, Method 2 and Method 1 have higher accuracy than other methods with 91.4%, 91.2% and 90.1, respectively. Standard deviation has been used for robustness. As you can see in Table 9, the proposed methods in Method 2 and Method3 have lower standard deviations and are more

robust. The proposed method by the ELM was improved the average classification accuracy and standard deviation by 3.9% and 2.28, respectively.

The Wilcoxon signed-rank test is a pairwise test that shows the differences between the behaviors of the two algorithms [54]. P-values are positive values between 0 and 1. The smaller the P-value, the better performance. The results of this test for method1 are shown (See Table 10).

As shown in Table 10, the Superior column shows the number of subjects that our method outperforms other methods. As Table 10 states, the number of subjects on that Method 1 performs better than [45] and [52] is 5. The number of subjects on that Method 1 performs better than [29, 43, 46–49] and [53] is 4. The number of subjects on that this method performs better than [44, 50] and [51] is 3. Therefore, the overall results show that Method 1 is better than twelve states of the art methods.

Experimental results of Wilcoxon signed-rank test Method 2 and other methods are shown in Table 11. The number of subjects on that Method 2 performs better than [45, 47, 49, 52] and [53] is 5. The number of subjects on that Method 2 performs better than [29, 43, 44, 46, 48] and [51] is 4.

The results of Wilcoxon signed-rank test Method 3 and other methods are given in Table 12. The number of subjects on that Method 3 performs better than [29, 43, 45–50, 52] and [53] is 5. The number of subjects on that Method 3 performs better than [44, 51] is 4. As you can see, Method 3 has superiority in other states of the art methods. The overall results show Method 3 is better than method1 and method 2.

5 Conclusion

In this study, efficient space learning based on kernel trick and dimension reduction was presented for multichannel motor imagery EEG signals. Dimension increase in data is done with the multi-kernel learning method. The parameters are optimized using the optimization method in this step. Also, dimension reduction is used to overcome the curse of dimension problem. The composite kernel is obtained to map the features extracted by CSP. The composite kernel was the combination of three types of the kernels, i.e., RBF, polynomial and linear kernel by Meta-heuristic MKL algorithm. The parameters associated with each base kernel and their weights in the composite kernel were calculated by the EO. After data mapping, dimensions of the data were reduced to maximum of ten dimensions. The number of reduced dimensions and mapping matrix were obtained using EO. Data dimension are reduced by mapping matrix linearly that its columns are equal to the number of reduced dimensions. Three classifiers of KNN,

SVM and ELM were selected for the proposed method. Also, the number of hidden layer neurons and the code of the classifier activation function were calculated by EO in ELM. The proposed method with ELM has higher accuracy than the other two classifiers. The results indicate the superiority of the proposed method to state-of-the-art methods. This method can be employed for EEG signals classification in other applications such as, epileptic diagnosis, emotion recognition and other MI signals classification.

Author contribution YA contributed to programmer validation, visualization, investigation, and writing—original draft. HO contributed to supervision, project administration, conceptualization, methodology, and writing—reviewing and editing.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

1. Acı Çİ, Kaya M, Mishchenko Y (2019) Distinguishing mental attention states of humans via an EEG-based passive BCI using machine learning methods. *Expert Syst Appl* 134:153–166
2. Kaur B, Singh D, Roy PP (2019) Age and gender classification using brain–computer interface. *Neural Comput Appl* 31(10):5887–5900
3. Alazrai R, Alwanni H, Daoud MI (2019) EEG-based BCI system for decoding finger movements within the same hand. *Neurosci Lett* 698:113–120
4. Wei CS, Wang YT, Lin CT, Jung TP (2018) Toward drowsiness detection using non-hair-bearing EEG-based brain-computer interfaces. *IEEE Trans Neural Syst Rehabil Eng* 26(2):400–406
5. Nicolas-Alonso LF, Gomez-Gil J (2012) Brain computer interfaces, a review. *Sensors* 12(2):1211–1279
6. Subha DP, Joseph PK, Acharya R, Lim CM (2010) EEG signal analysis: a survey. *J Med Syst* 34(2):195–212
7. Khan KA, Shanir PP, Khan YU, Farooq O (2020) A hybrid local binary pattern and wavelets based approach for EEG classification for diagnosing epilepsy. *Expert Syst Appl* 140:112895
8. Zhang Y, Zhang S, Ji X (2018) EEG-based classification of emotions using empirical mode decomposition and autoregressive model. *Multimed Tools Appl* 77(20):26697–26710
9. Yang S, Bornot JMS, Wong-Lin K, Prasad G (2019) M/EEG-based bio-markers to predict the MCI and alzheimer’s disease: a review from the ML perspective. *IEEE Trans Biomed Eng* 66(10):2924–2935
10. Saini, N., Bhardwaj, S., & Agarwal, R. (2019). Classification of EEG signals using hybrid combination of features for lie detection. *Neural Computing and Applications*, 1–11.
11. Jeannerod M (1995) Mental imagery in the motor context. *Neuropsychologia* 33(11):1419–1432

12. Pourali H, Omranpour H (2023) CSP-Ph-PS: Learning CSP-phase space and Poincare sections based on evolutionary algorithm for EEG signals recognition. *Expert Syst Appl* 211:118621
13. Gupta GS, Tripathi PR, Kumar S, Ghosh S, Sinha RK (2022) Prototype design for bidirectional control of stepper motor using features of brain signals and soft computing tools. *Biomed Signal Process Control* 71:103245
14. Tang X, Li W, Li X, Ma W, Dang X (2020) Motor imagery EEG recognition based on conditional optimization empirical mode decomposition and multi-scale convolutional neural network. *Expert Syst Appl* 149:113285
15. Jana, G. C., Shukla, S., Srivastava, D., & Agrawal, A. (2020). Performance Estimation and Analysis Over the Supervised Learning Approaches for Motor Imagery EEG Signals Classification. In *Intelligent Computing and Applications* (pp. 125–141). Springer, Singapore.
16. Jiang A, Shang J, Liu X, Tang Y, Kwan HK, Zhu Y (2020) Efficient CSP Algorithm With Spatio-Temporal Filtering for Motor Imagery Classification. *IEEE Trans Neural Syst Rehabil Eng* 28(4):1006–1016
17. Park Y, Chung W (2019) Frequency-optimized local region common spatial pattern approach for motor imagery classification. *IEEE Trans Neural Syst Rehabil Eng* 27(7):1378–1388
18. Park SH, Lee D, Lee SG (2017) Filter bank regularized common spatial pattern ensemble for small sample motor imagery classification. *IEEE Trans Neural Syst Rehabil Eng* 26(2):498–505
19. Baig MZ, Aslam N, Shum HP, Zhang L (2017) Differential evolution algorithm as a tool for optimal feature subset selection in motor imagery EEG. *Expert Syst Appl* 90:184–195
20. Wang, Y., Gao, S., & Gao, X. (2006, January). Common spatial pattern method for channel selection in motor imagery based brain-computer interface. In *2005 IEEE engineering in medicine and biology 27th annual conference* (pp. 5392–5395). IEEE.
21. Niazmardi S, Demir B, Bruzzone L, Safari A, Homayouni S (2017) Multiple kernel learning for remote sensing image classification. *IEEE Trans Geosci Remote Sens* 56(3):1425–1443
22. Gönen M, Alpaydın E (2011) Multiple kernel learning algorithms. *The Journal of Machine Learning Research* 12:2211–2268
23. Abbasnejad ME, Ramchandram D, Mandava R (2012) A survey of the state of the art in learning the kernels. *Knowl Inf Syst* 31(2):193–221
24. Faramarzi A, Heidarinejad M, Stephens B, Mirjalili S (2020) Equilibrium optimizer: A novel optimization algorithm. *Knowl-Based Syst* 191:105190
25. Dhongade, D. V., & Rao, T. V. K. H. (2017, March). Classification of sleep disorders based on EEG signals by using feature extraction techniques with KNN classifier. In *2017 International Conference on Innovations in Green Energy and Healthcare Technologies (IGEHT)* (pp. 1–5). IEEE.
26. Birjandtalab J, Pouyan MB, Cogan D, Nourani M, Harvey J (2017) Automated seizure detection using limited-channel EEG and non-linear dimension reduction. *Comput Biol Med* 82:49–58
27. Ozkan, Y., & Barkana, B. D. (2020, October). Multi-class Mental Task Classification Using Statistical Descriptors of EEG by KNN, SVM, Decision Trees, and Quadratic Discriminant Analysis Classifiers. In *2020 IEEE 5th Middle East and Africa Conference on Biomedical Engineering (MECBME)* (pp. 1–6). IEEE.
28. Chaovalitwongse WA, Fan YJ, Sachdeo RC (2007) On the time series k -nearest neighbor classification of abnormal brain activity. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 37(6):1005–1016
29. Zhang Y, Wang Y, Zhou G, Jin J, Wang B, Wang X, Cichocki A (2018) Multi-kernel extreme learning machine for EEG classification in brain-computer interfaces. *Expert Syst Appl* 96:302–310
30. Huang G, Huang GB, Song S, You K (2015) Trends in extreme learning machines: A review. *Neural Netw* 61:32–48
31. Shi TW, Chang GM, Qiang JF, Ren L, Cui WH (2023) Brain computer interface system based on monocular vision and motor imagery for UAV indoor space target searching. *Biomed Signal Process Control* 79:104114
32. Hong D, Man S, Martin JV (2016) A stochastic mechanism for signal propagation in the brain: Force of rapid random fluctuations in membrane potentials of individual neurons. *J Theor Biol* 389:225–236
33. Khanal, B., Pant, S., Pokharel, K., & Gaire, S. (2018, October). Mental State Prediction by Deployment of Trained SVM Model on EEG Brain Signal. In *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)* (pp. 82–85). IEEE.
34. Tang X, Wang T, Du Y, Dai Y (2019) Motor imagery EEG recognition with KNN-based smooth auto-encoder. *Artif Intell Med* 101:101747
35. Yu T, Xiao J, Wang F, Zhang R, Gu Z, Cichocki A, Li Y (2015) Enhanced motor imagery training using a hybrid BCI with feedback. *IEEE Trans Biomed Eng* 62(7):1706–1717
36. Pfurtscheller G (2001) Functional brain imaging based on ERD/ERS. *Vision Res* 41(10–11):1257–1260
37. Mackenroth, U. (2004). Rational Transfer Functions. In *Robust Control Systems* (pp. 17–40). Springer, Berlin, Heidelberg.
38. An-na, W., Yue, Z., Yun-tao, H., & Yun-lu, L. I. (2010, January). A novel construction of SVM compound kernel function. In *2010 International Conference on Logistics Systems and Intelligent Management (ICLSIM) (Vol. 3, pp. 1462–1465)*. IEEE.
39. Li X, Chen X, Yan Y, Wei W, Wang ZJ (2014) Classification of EEG signals using a multiple kernel learning support vector machine. *Sensors* 14(7):12784–12802
40. Lotte F, Congedo M, Lécuyer A, Lamarche F, Arnaldi B (2007) A review of classification algorithms for EEG-based brain-computer interfaces. *J Neural Eng* 4(2):R1
41. <http://www.bbci.de/competition/iii/datasetIVa>.
42. Blankertz B, Müller KR, Krusienski DJ, Schalk G, Wolpaw JR, Schlogl A, Birbaumer N (2006) The BCI competition III: Validating alternative approaches to actual BCI problems. *IEEE Trans Neural Syst Rehabil Eng* 14(2):153–159
43. Jin J, Miao Y, Daly I, Zuo C, Hu D, Cichocki A (2019) Correlation-based channel selection and regularized feature optimization for MI-based BCI. *Neural Netw* 118:262–270
44. Hang, W., Feng, W., Liang, S., Wang, Q., Liu, X., & Choi, K. S. (2020). Deep Stacked Support Matrix Machine Based Representation Learning for Motor Imagery EEG Classification. *Computer Methods and Programs in Biomedicine*, 105466.
45. Yu Z, Ma T, Fang N, Wang H, Li Z, Fan H (2020) Local temporal common spatial patterns modulated with phase locking value. *Biomed Signal Process Control* 59:101882
46. Guo Y, Zhang Y, Chen Z, Liu Y, Chen W (2020) EEG classification by filter band component regularized common spatial pattern for motor imagery. *Biomed Signal Process Control* 59:101917
47. Jin J, Liu C, Daly I, Miao Y, Li S, Wang X, Cichocki A (2020) Bispectrum-based channel selection for motor imagery based brain-computer interfacing. *IEEE Trans Neural Syst Rehabil Eng* 28(10):2153–2163
48. Shi B, Wang Q, Yin S, Yue Z, Huai Y, Wang J (2021) A binary harmony search algorithm as channel selection method for motor imagery-based BCI. *Neurocomputing* 443:12–25
49. Miao, M., Hu, W., & Zhang, W. (2021). A spatial-frequency-temporal 3D convolutional neural network for motor imagery EEG signal classification. *Signal, Image and Video Processing*, 1–8.

50. Miao Y, Jin J, Daly I, Zuo C, Wang X, Cichocki A, Jung TP (2021) Learning Common Time-Frequency-Spatial Patterns for Motor Imagery Classification. *IEEE Trans Neural Syst Rehabil Eng* 29:699–707
51. Liang, S., Hang, W., Yin, M., Shen, H., Wang, Q., Qin, J., ... & Zhang, Y. (2022). Deep EEG feature learning via stacking common spatial pattern and support matrix machine. *Biomedical Signal Processing and Control*, 74, 103531.
52. Tiwari, A., & Chaturvedi, A. (2022). Automatic EEG channel selection for multiclass brain-computer interface classification using multiobjective improved firefly algorithm. *Multimedia Tools and Applications*, 1–29.
53. Tiwari, A., & Chaturvedi, A. (2022). Automatic Channel Selection using Multiobjective X-shaped Binary Butterfly algorithm for Motor Imagery Classification. *Expert Systems with Applications*, 117757.
54. Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol Comput* 1(1):3–18

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.