Applied Soft Computing Journal 89 (2020) 106071

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Semi-supervised learning quantization algorithm with deep features for motor imagery EEG Recognition in smart healthcare application

Minjie Liu [a,1], Mingming Zhou [a,1], Tao Zhang [b,c,*], Naixue Xiong [d]

[a] School of Nursing, Taihu University of Wuxi, Qianrong street No.68, Wuxi, Jiangsu Province 214064, China
[b] Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Land and Resources, Shenzhen, China
[c] Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214000, China
[d] School of Computer Science and Technology, Tianjin University, Tianjin, China

ARTICLE INFO

Article history:
Received 29 August 2019
Received in revised form 24 December 2019
Accepted 3 January 2020
Available online 15 January 2020

Keywords:
Convolutional neural networks
Semi-supervised classification
EEG Recognition
Smart healthcare
Cartesian *K*-means

ABSTRACT

This paper depicts a novel semi-supervised classification model with convolutional neural networks (CNN) for EEG Recognition. The performance of popular machine learning algorithm usually rely on the number of labeled training samples, such as the deep learning approaches,sparse classification approaches and supervised learning approaches. However, the labeled samples are very difficulty to get for electroencephalography(EEG) data. In addition, most deep learning algorithms are usually time-consuming in the process of training. Considering these problems, in this article, a novel semi-supervised quantization algorithm based on the cartesian *K*-means algorithm is proposed, which named it as the semi-supervised cartesian *K*-means (SSCK), we use the CNN models pre-trained on motor imagery samples to create deep features, and then we applied it for motor imagery (MI) data classification. Unlike the traditional semi-supervised learning models that labeled information can be directly casted into the model training, label information can only be implicitly used in the semi-supervised learning strategy, in the semi-supervised learning algorithm, supervised information is integrated into the quantization algorithm by resorting a supervised constructed laplacian regularizer. Experimental results over four popular EEG datasets substantiate the efficiency and effectiveness of our proposed semi-supervised cartesian *K*-means.

1. Introduction

In recent years, deep learning technology has been widely used in image recognition, speech recognition, natural language processing and other fields [1–6]. With the development of smart healthcare, images gradually occupy a key role in recognizing the categories of the diseases and then in providing an accurate prediction results for patients. The brain is the body's central nervous system and behavior control center [7]. People mainly depend on brain reaction to get command of their bodies with respect to external stimulus. The brain consciousness of patients with dyskinesia is normal, but the motor intention cannot be transferred [8,9]. Electroencephalograph (EEG) is an efficient and convenient technique for analyzing and studying brain electrical activities. In most intelligent human–computer interaction systems,such as brain computer interface (BCI), we find EEG signals are very easy to get, and they are frequently used to conduct the detection operations [10]. EEG signals are translated into environmental control signals through the BCI system. Activity in different positions of the brain respond to different body movements and activity of imagination. The ipsilateral (relative to the subject's unilateral limb) and contralateral sensorimotor cortex will display appearances referred to as event-related synchronization and event-related desynchronization, respectively [11]. This laid a foundation for the classification of electrophysiological motor imagery (MI) representation signals. Motor imagery is one of multitudinous examples which depend upon numerous existing patterns of the EEG signal.

Recently, with the development of machine learning (deep learning,semi-supervised learning and other quantization learning approaches [12–17]) and smart healthcare, MI EEG data classification problem has triggered broad discussion in BCI area [18,19]. However, MI EEG recognition is usually difficulty due to the samples difference of the same class and time-consuming. While on the other hand, approximate nearest neighbor(ANN) search algorithm has grown into a hot research topic for its high retrieve performance on the large-variation datasets. ANN search is aim to find those instances whose Euclidean distance

* Corresponding author at: Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214000, China.
E-mail address: taozhang@jiangnan.edu.cn (T. Zhang).

[1] Minjie Liu and Mingming Zhou contributed equally to this work and should be considered co-first authors.

https://doi.org/10.1016/j.asoc.2020.106071
1568-4946/

are relatively small. In the ANN research to get the Euclidean distance between the query vector q and all the vectors involving lots of computations which is infeasible for the large scale and high dimensional cases.

Many machine learning algorithms have been proposed and improved to eliminate those computations, such as k-d tree [20], multiple k-d trees and priority search [21] as well as cover tree and the trinary-projection tree et al. Hashing is also a very attractive method in the ANN search work, it deal it by converting the database vectors into short codes, storage cost is small, making the in-memory search feasible and the distance computation cost is also reduced. Highly effective hashing approaches have been designed, such as typical locality sensitive hashing [22, 23], kernelized locality-sensitive hashing [24], semi-supervised hashing [25,26], K-means hashing [27], supervised hashing [28, 29].

Quantization also plays important role in the ANN search research work, it deal the job by adopting the data representation strategy. Unsupervised algorithm is firstly used to implement the data clustering task to enable original data to get labels. Then the instances in each cluster are represented or reconstructed. Then distance calculation between the query data and base data can be converted into distance computation between the centers of clustering in which class the query data and base data belong to respectively. In other words, the distance between the query and database vector can be approximated equal to the distance among the centers of clustering in which cluster the query and base data vector belong to respectively. Product Quantization [30] is an algorithm that the original vector space is decomposed into cartesian product of several low-dimensional vector Spaces, and the low-dimensional vector Spaces obtained by decomposition are quantified respectively. The computational complexity of violent exhaustion is a very high, which is obviously not desirable. Therefore, for the similar search scenario of high-dimensional data with large amount of data, we need some efficient approximate nearest neighbor search technologies, and PQ is one of them.

Since the quantization algorithm had been proposed, many extended algorithms have been developed to enhance its search performance. Cartesian K-means [31] extend the PQ algorithm and impose an appropriate regulatory factor to a linear mapping vector of the classical K-means objective function, which make the optimizing procedure become more efficient and effective than the traditional K-means by adopting the Lloyds algorithm [32]. Unlike cartesian K-means optimizing with respect to sub-codebook only, OPQ [33] optimize with respect to sub-codebook and space decomposition both to find the optimal space decomposition scheme and rotation matrix simultaneously. By balancing the eigenvalue of the covariance matrix, OPQ can obtain the optimal space decomposition, however, strong multimodal distributions are prone to errors during decomposition [34]. Optimal cartesian K-means [35] (OCK-means) is different from the previous coding scheme where one subcode word is choosed [35], OCK-means encodes data points using subvectors of multiple subcode words, performance are significantly improved in the ANN research experiments, same algorithm is also proposed in [36]. In recent years, the research about approximate nearest neighbor (ANN) has spread all over the pattern recognition area. However, existing typical ANN-based approaches are designed for static databases only. Liu et al. [37] have demonstrated the benefit of making full use of random kernel functions in a PQ strategy, they proposed a Kernelized product quantization method through decomposing corresponding implicit eigenspace. Xu et al. [38] constructed appropriate incoming streaming data through updating quantization codebook and presented an effective online product quantization (OPQ) algorithm.

All the above mentioned algorithms have obtained an significant improvement on the research of product quantization algorithm, however, all of these algorithms are still belong to the unsupervised learning framework, which may explicitly restrict the performance of these algorithms. To effectively reduce the quantization error in each subspace and improve the retrieval performance, we propose semi-supervised cartesian K-means algorithm (SSCK). It firstly build a laplacian matrix ground on the labeled data, the similarity computation between two samples who share the same labels will be given a large value, while data belong to different clusters will be assigned a small value. Then add the laplacian regularizer to the specific loss term to construct the more discriminative cartesian K-means objective function. We also design corresponding optimization algorithm to solve our constructed model. The main contributions of this paper are summarized as the following:

- In view of the number of labeled samples is very small, we propose a semi-supervised cartesian K-means algorithm with deep features extracted from CNN in EEG data.
- Labeled data is integrated in the quantization step to provide the additional constraints to promote the data reconstruction ability.
- Laplacian matrix is build based on the label data and is added to the optimal reverse prediction model to get more discriminative cartesian K-means model.
- Strategy to optimize the semi-supervised cartesian K-means is given to enable the function to get an minimum value.

The reminder of our paper is stated as following: We review the knowledge of cartesian K-means and the optimal reverse prediction algorithm in Section 2. In Section 3 we depict the semi-supervised cartesian K-means algorithm, then the supervised cartesian K-means are presented. Section 4 gives the interpretation and implementation detail of the algorithm. Section 5 gives the detailed descriptions of the algorithm design and results with quantitative comparison with other algorithms. In the end, Section 6 summarizes the main points of this paper.

2. Related work

Previous to our proposed algorithm, we will briefly introduce previous MI EEG data classification scheme, the optimal reverse prediction and cartesian K-means algorithm.

2.1. MI EEG classification

In this subsection, we briefly review existing methods and models for EEG MI classification. Through the investigation and research of MI EEG classification methods, we found that feature extraction and recognition are two key factors. These two kinds of methods have been used for the MI EEG classification tasks. Many researchers focus on feature extraction stage with respect to MI EEG classification. Currently, the feature extraction algorithms mainly include Filter bank common spatial patterns (FBCSPs), short time Fourier transform (STFT), wavelet transform (WT) and common spatial pattern (CSP) and other approaches [39–42]. Feature recognition strategies mainly relied on linear discriminant analysis (LDA), support vector machine (SVM) and other classifiers [43–45].

The current approaches for feature extraction have been progressing at a dramatic pace, but they may do not work well in some specific areas. Many studies have shown that compared with traditional feature extraction methods, adaptive auto regressive works better in extracting MI EEG feature [46]. For SIFT feature descriptors, the frequency resolution and time resolution are determined by the window size. A larger window size means

lower time resolution and higher frequency resolution. In contrast, a smaller window size result in higher time solution and lower frequency resolution. The limited window size will cause a certain frequency leakage, especially related to the discrete-time Fourier transform. Therefore, it is not possible to obtain both time and frequency resolution. Wavelet transform has excellent time–frequency localization characteristics and can be used for multi-scale analysis. But at present, most researchers manually screen wavelet coefficient EEG features by using statistical properties such as average, maximum, minimum and standard deviation. Although the moving image signal pattern is not based on prior knowledge, its irregular motion will make the signal to be unpredictable.

Lately, there have been high attention on CNN and RNN method, which are putting forward to reach excellent capability on the MI EGG data [47]. And relevant comparative tests about DBN and SVM have been carried out [48] to solve the problems of two MI classification. The result showed that DBN has an advantage over SVM. CNN was also used in [49], for recognition of MI EEG. RNN and CNN also were executed formerly in [50], in order to find the cognitive cases from MI EEG datasets by proposing the multidimensional features. Autoencoders, which is a famous part of deep learning schemes, have been utilized in emotion recognition of EEG signals [51]. Plenty of studies have transformed EEG signals to images and also used many deep learning models which have high performance in classifying images. In order to maintain the structures of EEG data, including spatial, temporal and spectral, [52] had proposed a new kind of compound features. Three chosen frequency bands of the power spectrum were estimated by using every electrodes' EEG signals. And the short-time Fourier transform was used to transform data of EEG time series to 2D images in [48]. MI signals' spectral features were taken and 1D CNN [53] and a stacked autoencoder(SAE) [54] were also used to get better property for MI EEG dataset.

The above studies have tried to make full use of the essences of deep learning for EEG classification, however, this kind of model is not very suitable for other fields, such as image and signal processing. Therefore, research on conducting and designing some novel learning models for EEG MI classification is very urgent.

2.2. Optimal reverse prediction

In [55] Xu et al. proposed an optimal reverse prediction (ORP) algorithm whose objective function contains two terms: one is the conventional K-means algorithm formula — an unsupervised clustering algorithm where neither the cluster center matrix nor the label matrix are known, another one is the supervised learning term, which is similar with the conventional K-means formula but label is known previously. The label variable in the objective function adopt the 1-of-K encoding scheme. The optimal reverse prediction algorithm is usually resolved through iteratively computing the least square loss term with respect to the cluster center matrix and the unknown label matrix variables. The objective function is defined as:

$$\begin{aligned}\mathcal{L}(C,B) = &\min_{C}\min_{B} \operatorname{tr}((X^{(L)}-CY^{(L)})^T(X^{(L)}-CY^{(L)}))/N_L \\ &+ \eta^2 \operatorname{tr}((X^{(U)}-CB)^T(X^{(U)}-CB))/N_U\end{aligned} \tag{1}$$

where $X^{(L)} \in \mathbb{R}^{P\times N_L}$ and $Y^{(L)} \in \mathbb{R}^{K\times N_L}$ represent the training instances matrix and labels matrix respectively, $X^{(U)} \in \mathbb{R}^{P\times N_U}$ is the unlabeled data matrix, $B \in \mathbb{R}^{K\times N_U}$ is the unknown label matrix and η^2 is trading off parameter. P is the dimension of the instances, N_L and N_U indicate the labeled and unlabeled samples number respectively, K is the number of the clusters.

The optimal reverse prediction algorithm is a unification of several supervised and unsupervised training principles through the concept of optimal reverse prediction: predict the inputs from the target labels, optimizing both over model parameters and any missing labels. Supervised least squares, principal components analysis, K-means clustering and normalized graph-cut can all be expressed as instances of the same training principle. The optimal reverse prediction algorithm unifies multiple supervised and unsupervised training principles through the concept of optimal reverse prediction: give a appropriate prediction about the input samples, and optimize the model parameters without labels. Most machine learning algorithm (Supervised least squares, Kmean clustering and normalized graph cutting) can all be treated in the same training method.

2.3. Cartesian K-means

In [30], product quantization algorithm is proposed for ANN research task. High dimensionality input data space is averagely decomposed and represented as a special cartesian product of M lower dimensionality. Each space can form a codebook by using the conventional K-means algorithm. Hence there will generate K sub codewords for each sub-vector, by this means, the M sub-vectors will forms K^M clusters, while if we adopt the traditional method to encode the whole input data by K-means requires storage $O(K^M P)$. At the same time, the computing complexity can be reduced to $O(KP)$.

In PQ algorithm, the codewords in each subspace are generated by employing the K-means clustering which optimize the squared distortion errors (2) with respect to b and C iteratively. However, PQ algorithm does not present a method how to get the optimal space decomposition for the ANN search task.

$$\begin{aligned}&\sum_{\mathrm{x}\in\mathcal{X}} \min_{b\in\mathcal{H}_{1/k}} \|\mathrm{x}-Cb\|_2^2 \\ &s.t. \quad b\in\{0,1\}^k \text{ and } \|b\|=1 \\ &C_{:,i}^{\top}C_{:,j}=0 \quad \text{for} \quad i\neq j, \quad C_{:,i}^{\top}C_{:,i}>0\end{aligned} \tag{2}$$

Cartesian K-means [31] solve this problem by designing appropriate constraints on the column of the linear mapping vector C in (2) to implicitly adjust the instances' dimension information, which make the optimizing (2) with respect to b become more tractable in the orthogonal cartesian K-means, at the same time, to find better subspace partitioning in the cartesian K-means and achieves better ANN search performance, same algorithm optimized product quantization is also proposed in [56]. Orthogonal constraint on the cluster center guarantee the cluster center can be represented as $C \equiv RD$, where $R^{\top}R = RR^{\top} = I$, so (2) can be reformulated as (3). Minimizing (3) with respect to R, D and B, the obtained optimal rotation matrix R and cluster center D can assist (3) in getting a lower distortion error.

$$\begin{aligned}&\min_{R,D,B}\sum_i \left\| \mathrm{x}_i - R\begin{bmatrix} D^1 b_i^1 \\ \vdots \\ D^M b_i^M \end{bmatrix} \right\|_2^2 \\ &s.t. \quad R^TR=I \quad b_i^m\in\{0,1\}^{K\times 1}, \forall i,m \\ &\qquad \|b_i^m\|_1=1, \forall i,m\end{aligned} \tag{3}$$

where the number of the subspace is denoted as M.

3. Semi-Supervised Cartesian K-means (SSCK)

In the following part, we will describe discriminative semi-supervised cartesian K-means algorithm and then extend the Semi-supervised concept to other ANN search algorithms.

3.1. Semi-supervised Cartesian K-means

As presented in [55], the classical K-means, principal component analysis (PCA) and normalized cut [57,58] et al. can be considered as the special cases of the optimal reverse prediction algorithm. Based on this concept we can use (1) to substitute (3) in the quantization procedure and to give the semi-supervised cartesian K-means algorithm. Given a set $\{X^{(L)}, Y^{(L)}\} \in \mathbb{R}^{P\times N_L} \times \mathbb{R}^{K\times N_L}$ and unlabeled dataset $\{X^{(U)}\} \in \mathbb{R}^{P\times N_U}$, where P is the dimensionality of the instance, K indicates the total number of the quantization center, N_L and N_U indicate the labeled and unlabeled samples number respectively. Ground on input space decomposition strategy, the semi-supervised cartesian K-means (SSCK) is defined as:

$$\begin{aligned}\mathcal{L}_{SSCK} = &\min_{R,D,B} \|R^\top X^{(L)} - DY^{(L\star)}\|_F^2 + \eta^2\|R^\top X^{(U)} - DB\|_F^2 \\ &s.t.\ R^\top R = RR^\top = I\ \|y_i^m\| \in \{0,1\}^{K\times 1}, \|(y^{(L\star)})_i^m\|_1 = 1 \\ &\|b_i^m\| \in \{0,1\}^{K\times 1}, \|b_i^m\|_1 = 1, \forall i, m\end{aligned} \tag{4}$$

Based on the space decomposition concept, $X^{(L)}$, $X^{(U)}$, DY and DB can be individually represented as: $X^{(L)} = \begin{bmatrix} X_1^{(L)} \\ \vdots \\ X_M^{(L)} \end{bmatrix}$, $X^{(U)} = \begin{bmatrix} X_1^{(U)} \\ \vdots \\ X_M^{(U)} \end{bmatrix}$, $DY = \begin{bmatrix} D^1(Y^{(L\star)})^1 \\ \vdots \\ D^M(Y^{(L\star)})^M \end{bmatrix}$, $DB = \begin{bmatrix} D^1B^1 \\ \vdots \\ D^MB^M \end{bmatrix}$, $(Y^{(L\star)})^i = \left[(y^{(L\star)})_1^i \cdots (y^{(L\star)})_{N_L}^i\right]$, $B^i = \begin{bmatrix} b_1^i & \cdots & b_{N_U}^i \end{bmatrix}$ and $\|\cdot\|_F$ represents the Frobenius norm. From (4) given the labeled dataset $X^{(L)}$, $Y^{(L\star)}$ and other unlabeled dataset $X^{(U)}$, the discriminative cartesian K-means algorithm can be obtained. More specifically, the above (4) is only the quantization problem, a part of the cartesian K-means.

However, we cannot use (4) to quantize the labeled and unlabeled data right now, because we do not know the quantization label matrix $Y^{(L\star)}$.

When optimal reverse prediction used in the clustering task, the notation of labels matrix $Y^{(L)}$ in (1) denote the clustering label or classification label, which can be obtained from the known labeled data. However, when the optimal reverse prediction used in our proposed semi-supervised cartesian K-means (4), its role is to quantize or encode the data, label matrix $Y^{(L)}$ is used to indict the quantization label, but it is unknown.

In another words, the quantization label is different from the clustering label, clustering label can be directly obtained from the labeled data, but the quantization label cannot, so (4) cannot be used for our semi-supervised cartesian K-means.

To solve this problem, we introduce the laplacian regularizer term into the above semi-supervised cartesian K-means model (4) and get the following formulation:

$$\begin{aligned}\mathcal{L}_{SSCK} = &\min_{R,D,B,Y,\mu} \|R^\top X_\mu^{(L)} - DY^{(L)}\|_F^2 \\ &+ \eta^2\|R^\top X_\mu^{(U)} - DB\|_F^2 + \lambda \mathrm{tr}(Y^{(L)}L(Y^{(L)})^\top) \\ &s.t.\ R^\top R = RR^\top = I\ \|(y^{(L)})_i^m\| \in \{0,1\}^{K\times 1}, \\ &\|y_i^m\|_1 = 1\ \|b_i^m\| \in \{0,1\}^{K\times 1}, \|b_i^m\|_1 = 1, \forall i, m\end{aligned} \tag{5}$$

where we denote $X_\mu^{(U)} \equiv X^{(U)} - \mu(\mathbf{1}^{(U)})^\top$, $X_\mu^{(L)} \equiv X^{(L)} - \mu(\mathbf{1}^{(L)})^\top$, μ is the mean value vector of the input data. Both $Y^{(L)}$ and B are the quantization label and unknown, L represents the constrained laplacian term and $L = W - D$, the similarity matrix is denoted as W and $D_{ii} = \sum_j W_{i,j}$. We construct the similarity matrix W use the supervised method, entries W_{ij} will be assigned a large weight if x_i and x_j have the same clustering label and a small weight will be given if x_i and x_j belong to different clusters. In our proposed model, we construct the corresponding matrix by adopting the method presented in [59].

Optimizing (5) generally is an intractable job, for $Y^{(L)}$ and B are the discrete matrixes with 1-of-K encoding scheme, discrete optimization is difficult in the non-submodular problem. Here we provide two methods to deal with this difficulty. The first one: We used the exhaustive search algorithm which also adopted in [36,60]. The second one: We first relax the discrete variable $Y^{(L)}$ into the continuous and resort a constraint to $Y^{(L)}$, then optimize the cartesian K-means objective function with the constraint to get an optimal $\bar{Y}^{(L)}$, threshold method is adopted to get the discrete variable.

Update $Y^{(L)}$: To optimize (5) with respect to $Y^{(L)}$ using the exhaustive search method, we can rewrite (5) as the following(only those terms relevant to $Y^{(L)}$ are kept):

$$\begin{aligned}(5) &= \sum_{i=1}^{N_L} \|R^\top x_{\mu,i}^{(L)} - Dy_i^{(L)}\|_2^2 + \frac{\lambda}{2}\sum_{i,j=1}^{N_L} w_{ij}\|y_i^{(L)} - y_j^{(L)}\|_2^2 \\ &= \sum_{i=1}^{N_L}\left(\|R^\top x_{\mu,i}^{(L)} - Dy_i^{(L)}\|_2^2 + \frac{\lambda}{2}\sum_{j=1}^{N_L} w_{ij}\|y_i^{(L)} - y_j^{(L)}\|_2^2\right)\end{aligned} \tag{6}$$

Optimize (6) with respect to $y_i^{(L)}$, we can fix all the other $\{y^{(L)}\}_{j\neq i}$ and try x_i in all the clusters to find the optimal assignment which can guarantee the loss function to have the global minimum. Of course, optimize (6) with respect to $y_i^{(L)}$ with the exhaustive search is a very expensive, we need to compute the loss function (5)$2^8 \times N_L$ times in each iteration when adopt the 8-bit code length. For 32-bit, 64-bit and 128-bit we need compute the loss function (5) $4\times 2^8\times N_L$, $8\times 2^8\times N_L$ and $16\times 2^8\times N_L$ times respectively.

The second method we adopted to optimize (6) is to relax the discrete variable $Y^{(L)}$ into continuous and resort constraint to $Y^{(L)}$ firstly, then optimize the objective to the optimal value and threshold it finally. We rewrite (5) as following(only those terms relevant to $Y^{(L)}$ are kept):

$$\begin{aligned}(5) &= \sum_{i=1}^{N_L} \|R^\top x_{\mu,i}^{(L)} - Dy_i^{(L)}\|_2^2 + \frac{\lambda}{2}\sum_{i,j=1}^{N_L} w_{ij}\|y_i^{(L)} - y_j^{(L)}\|_2^2 \\ &= \sum_{i=1}^{N_L}\left(\|R^\top x_{\mu,i}^{(L)} - Dy_i^{(L)}\|_2^2 + \frac{\lambda}{2}\sum_{j=1}^{N_L} w_{ij}\|y_i^{(L)} - y_j^{(L)}\|_2^2\right) \\ &\text{s.t.:}\quad (y_i^{(L)})^\top\mathbf{1} = 1,\ i = 1,2,\ldots,N_L\end{aligned} \tag{7}$$

Denote the predicted $Y^{(L)}$ as $\bar{Y}^{(L)}$ and let $r_i = R^\top x_{\mu,i}^{(L)}$. Aim to get the optimal solution of $\bar{Y}^{(L)}$, we consider the Lagrange function $\mathfrak{L}(\bar{y}_i, \beta)$, which is defined as:

$$\begin{aligned}\mathfrak{L}(\bar{y}_i^{(L)}, \beta) = &\sum_{i=1}^{N_L}\|r_i - D\bar{y}_i^{(L)}\|_2^2 + \frac{\lambda}{2}\sum_{i,j=1}^{N_L} w_{ij}\|\bar{y}_i^{(L)} - \bar{y}_j^{(L)}\|_2^2 \\ &+\beta((\bar{y}_i^{(L)})^\top\mathbf{1} - 1)\end{aligned} \tag{8}$$

Let $\frac{\partial\mathfrak{L}}{\partial\bar{y}_i^{(L)}} = 0$, we have

$$\begin{aligned}\frac{\partial\mathfrak{L}}{\partial\bar{y}_i^{(L)}} &= C\bar{y}_i^{(L)} + \lambda\sum_{j=1}^{N_L} w_{ij}(\bar{y}_i^{(L)} - \bar{y}_j^{(L)}) + \beta\mathbf{1} = 0 \\ \frac{\partial\mathfrak{L}}{\partial\bar{y}_i^{(L)}} &= C^*\bar{y}_i^{(L)} - \lambda\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)} + \beta\mathbf{1} = 0\end{aligned} \tag{9}$$

where $C = (r_i\mathbf{1}^\top - D)^\top(r_i\mathbf{1}^\top - D)$ and $C^* = \begin{bmatrix} C_{11} - \sum_{j=1}^{N_L} w_{ij} & , C_{12} & , \dots & , C_{1n} \\ C_{21} - \sum_{j=1}^{N_L} w_{ij} & , C_{22} & , \dots & , C_{2n} \\ \vdots & \vdots & \dots & \vdots \\ C_{k1} - \sum_{j=1}^{N_L} w_{ij} & , C_{k2} & , \dots & , C_{kn} \end{bmatrix}$ Pre-multiply (9) by $\mathbf{1}^\top(C^*)^{-1}$, we can get:

$$1 - \mathbf{1}^\top\lambda(C^*)^{-1}\left(\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)}\right) + \mathbf{1}^\top\beta(C^*)^{-1}\mathbf{1} = 0 \tag{10}$$

From (10) we can get $\beta = \frac{\mathbf{1}^\top\lambda(C^*)^{-1}\left(\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)}\right)-1}{\mathbf{1}^\top(C^*)^{-1}\mathbf{1}}$.

From (9) we can get the optimal $\bar{y}_i^{(L)}$ with respect to (8) :

$$\bar{y}_i^{(L)} = (C^*)^{-1}\left(\lambda\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)} - \beta\mathbf{1}\right) \tag{11}$$

Substitute β into (11), we get the update equation of $\bar{y}_i^{(L)}$:

$$\bar{y}_i^{(L)} \leftarrow (C^*)^{-1}\left(\lambda\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)} - \frac{\mathbf{1}^\top\lambda(C^*)^{-1}\left(\sum_{j=1}^{N_L} w_{ij}\bar{y}_j^{(L)}\right) - 1}{\mathbf{1}^\top(C^*)^{-1}\mathbf{1}}\mathbf{1}\right) \tag{12}$$

Having obtained the predicted $\bar{y}_i^{(L)}$, we can then get the discrete $y_i^{(L)}$ by the following:

$$y_i^{(L)} \leftarrow \max(\bar{y}_i^{(L)}) \tag{13}$$

Update B: Having obtained the quantization $Y^{(L)}$, we can get the cluster center D by computing the average value of all labeled data $X^{(L)}$ in each quantization cluster. Based on the predicted cluster center D, the labels matrix B of the unlabeled data can be obtained by adopting the KNN clustering algorithm. To be more precisely, the solving process of B is described as follows:

$$\begin{aligned} D &\leftarrow \min_D \|R^\top X_\mu^{(U)} - DY^{(L)}\|_F^2 \\ B &\leftarrow \min_B \|R^\top X_\mu^{(U)} - DB\|_F^2 \end{aligned} \tag{14}$$

Update D: Based on the labeled and unlabeled data, considering their quantization label $Y^{(L)}$ and B, the clustering center D can be updated in the following way:

$$D \leftarrow \min_D \|R^\top X_\mu - DY\|_F^2 \tag{15}$$

where $X_\mu := \begin{bmatrix} X_\mu^{(L)} & \eta X_\mu^{(U)} \end{bmatrix}$ and $Y := \begin{bmatrix} Y^{(L)} & \eta B \end{bmatrix}$.

Update R: Many papers have been published to treat orthogonal constraint optimization difficulty like in [61]-a Crank–Nicolson-like update scheme to guarantee appropriate orthogonal constraints. In our experiment we still adopt the [62] algorithm as does cartesian K-means algorithm to solve it for its high performance and efficiency. Based on the

$$[U_R, S_R, V_R^\top] = SVD(X_\mu(Y^{(L)})^T D^T) \tag{16}$$

Then R can be obtained by:

$$R \leftarrow U_R V_R^T \tag{17}$$

Update μ : After we have got R, D, $Y^{(L)}$ and B, then we can update the μ in the following way:

$$\mu \leftarrow \text{mean}(X - RDY) \tag{18}$$

where $X = \begin{bmatrix} X^{(L)} & \eta X^{(U)} \end{bmatrix}$ and $Y = \begin{bmatrix} Y^{(L)} & \eta B \end{bmatrix}$.

We present our algorithm to the semi-supervised cartesian K-means in Table 1.

3.2. Other extensions

The semi-supervised concept can also be used to other quantization algorithms for ANN search task, such as optimized cartesian K-means (OCK) [56] and optimized product quantization (OPQ) [63]. For example, the semi-supervised PQ can be obtained by optimizing (19). The optimization process is easier than the semi-supervised cartesian K-means, because no columns orthogonal constraints to the cluster center matrix.

$$\begin{aligned} \mathcal{L}_{PQ} =& \|X_\mu^{(L)} - DY^{(L)}\|_F^2 + \eta^2\|X_\mu^{(U)} - DB\|_F^2 \\ &+ \lambda\text{tr}(Y^{(L)}L(Y^{(L)})^\top) \\ \mathbf{s.t.}\ & \|(y^{(L)})_i^m\| \in \{0,1\}^{K\times 1}, \|y_i^m\|_1 = 1 \\ & \|b_i^m\| \in \{0,1\}^{K\times 1}, \|b_i^m\|_1 = 1, \forall i, m \end{aligned} \tag{19}$$

Resort the semi-supervised concept to the optimized cartesian K-means algorithm is more complicated than the cartesian K-means (OCK), because the OCK algorithm adopt the multiple codeword to quantize the data. The semi-supervised optimized cartesian K-means can be obtained by solving the following (20).

$$\begin{aligned} \mathcal{L}_{OCK} =& \|X^{(L)} - R\hat{D}\hat{B}\|_F^2 + \eta^2\|X^{(U)} - R\hat{D}\hat{Y}^{(L)}\|_F^2 \\ &+ \lambda\text{tr}\left(Y^{(L)}L(Y^{(L)})^\top\right) \\ \mathbf{s.t.}\ & \|(y^{(L)})_i^m\| \in \{0,1\}^{K\times 1}, \|y_i^m\|_1 = C \\ & \|b_i^m\| \in \{0,1\}^{K\times 1}, \|b_i^m\|_1 = C, \forall i, m \end{aligned} \tag{20}$$

where $\hat{D} \triangleq \begin{pmatrix} \hat{D}^1 & & \\ & \ddots & \\ & & \hat{D}^m \end{pmatrix}$, $\hat{D}^m \triangleq \begin{pmatrix} \hat{D}^{m,1} & \dots & \hat{D}^{m,C} \end{pmatrix}$, $\hat{B} \triangleq \begin{pmatrix} \hat{B}^{1\top} & \dots & \hat{B}^{M\top} \end{pmatrix}^\top$, $\hat{B}^M \triangleq \begin{pmatrix} \hat{b}_1^m & \dots & \hat{b}_N^m \end{pmatrix}$, $\hat{b}_i^m \triangleq \begin{pmatrix} \hat{b}_i^{m,1\top} & \dots & \hat{b}_i^{m,C\top} \end{pmatrix}^\top$. Here $Y^{(L)}$ has as same notation as B and only B is given for simplicity.

Given $\eta^2 = 0$, we can get the supervised Cartesian K-means quantization algorithm, same concept can also been adopted in other quantization algorithms to obtain the supervised quantization algorithms such as supervised product quantization algorithms, supervised orthogonal K-means algorithm as well as supervised optimized Cartesian K-means algorithms. Generally speaking, supervised and semi supervised algorithms perform better than the unsupervised algorithms. If our semi-supervised concept is workable, this paper will pave a new research direction on the semi-supervised quantization algorithm.

In above model training, many effort are needed to get the optimal $Y^{(L)}$ and B based on objective function. In our research, we propose to adapt the exhaustive search and relax two methods to optimize the semi-supervised quantization objective function with respect to $Y^{(L)}$. Lots of computations are needed for the exhaustive search strategy, but global minimum can be guaranteed. Relax plus constrain method is more efficient, in this article we adopt the relax method to solve our semi-supervised cartesian K-means problem.

4. Interpretation and implementation details

The algorithm we propose is called semi-supervised Cartesian K-means (SSCK). It makes use of the deep CNN feature, which can fuse all features of the extracted convolutional by applying fully connected layers. The across time samples are input to EEG channels, then we will get a 2D feature map output after the execution of convolutions. After finishing the feature learning phase,

Table 1
Algorithm to Semi-Supervised Cartesian K-means.

Semi-Supervised Cartesian K-means
Input: Labeled samples $(X^{(L)}, Y^{(L)})$ and unlabeled samples $X^{(U)}$, balanced factors η, λ, β, maximum number of iterations T and convergence factor τ, original value $Y_0^{(L)}$ $t = 0$ **While** $\left(t \leq T \;\&\&\; \|\mathcal{L}((Y^{(L)})^{t+1}, B^{(t+1)}, R^{(t+1)}, D^{(t+1)}, \mu^{(t+1)}) - \mathcal{L}((Y^{(L)})^{t+1}, B^{(t+1)}, R^{(t)}, D^{(t)}, \mu^{(t)})\| \geq \tau\right)$ $t \leftarrow t + 1$; 1 **Update** $Y^{(L)}$: Update all columns of $Y^{(L)}$ separately. Fix the those columns $y_j^{(L)}$, $j \neq i$ and update $y_i^{(L)}$ based on the following: $y_i^{(L)} \leftarrow \max(\bar{y}_i^{(L)})$ Where $\beta = \frac{\mathbf{1}^\top \lambda (C^*)^{-1} \left(\sum_{j=1}^{N_L} w_{ij} \bar{y}_j^{(L)}\right) - 1}{\mathbf{1}^\top (C^*)^{-1} \mathbf{1}}$ and $\bar{y}_i^{(L)} \leftarrow (C^*)^{-1} \left(\lambda \sum_{j=1}^{N_L} w_{ij} \bar{y}_j^{(L)} - \frac{\mathbf{1}^\top \lambda (C^*)^{-1} \left(\sum_{j=1}^{N_L} w_{ij} \bar{y}_j^{(L)}\right) - 1}{\mathbf{1}^\top (C^*)^{-1} \mathbf{1}} \mathbf{1} \right)$ 2 **Update** B: $D \leftarrow \min_D \|R^\top X_\mu^{(U)} - DY^{(L)}\|_F^2$ $B \leftarrow \min_B \|R^\top X_\mu^{(U)} - DB\|_F^2$ 3 **Update** D: $D \leftarrow \min_D \|R^\top X_\mu - DY\|_F^2$ 4 **Update** R: $R \leftarrow U_R V_R^T$, where $[U_R, S_R, V_R^\top] = SVD(X_\mu (Y^{(L)})^T D^T)$ 5 **Update** μ: $\mu \leftarrow \text{mean}(X - RDY)$ **End while** **Output:** The predicted label D, R and μ

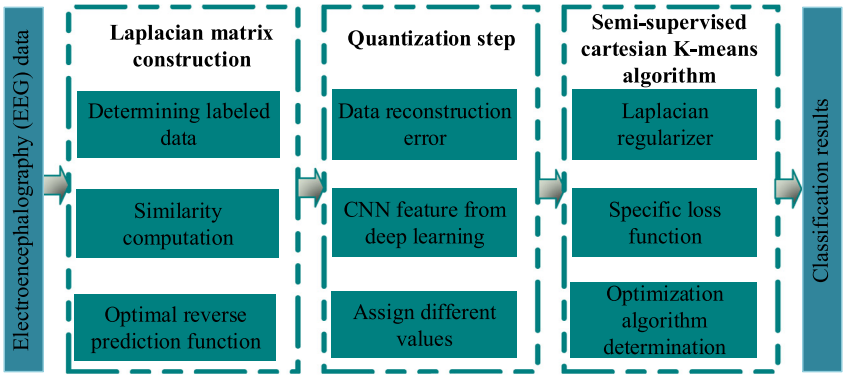


Fig. 1. Detailed process of our proposed algorithm.

in the next phase we execute the multilayer feature extraction and fusion phase. In the first phase of this model, the saw cropped input EEG signal is taken and Laplacian matrix construction is used to process these features. The features we used in the quantization process are taken from the convolution layers after the max pooling, which is in order to decrease the feature size with keeping relevant information extracted by the convolutional layers complete. Therefore, the feature maps are all taken out from Pool-1, Pool-2, Pool-3 and Pool-4 (max pooling) layers. Every pooling layers stand for convolutional layers at distinct levels of abstraction about CNN architecture. For example, the initial layers indicate simpler features and the more deep layers will get more complex features. In the end, we use a supervised constructed Laplacian regularizer to integrate the semi-supervised information into the quantization algorithm. Each detailed steps of our proposed algorithm are displayed in Fig. 1. In our experiment, we use the optimal reverse prediction model as the quantization distortion function and take the sample's label information into the quantization step, propose the semi-supervised cartesian K-means model to improve the previous cartesian K-means's performance. Optimal reverse prediction is a simple semi-supervised learning algorithm, many algorithms such as least square, PCA, K-means as well as Normal Cut are the special cases of it. In the semi-supervised cartesian K-means algorithm the optimal reverse prediction is used to quantize or encode the sample instances into short codeword, which is different from its application of classification task. In quantization step of ANN search, we generally quantize the data into 8 bits which results 256 clusters, however, the number of real label is usually not equal to 256.

So the first key problem in the semi-supervised cartesian K-means is to get the quantization label, which is a very hard problem. How to convert the label information of the labeled data into the quantization label? In this article, we innovatively introduced the laplacian regularizer factor to solve this problem. Given a dataset $\{X\} =: \{X^{(L)}, X^{(U)}\}$ includes a small amount labeled samples $X^{(L)} \in \mathbb{R}^{P \times N_L}$ and large amount of unlabeled samples $X^{(U)} \in \mathbb{R}^{P \times N_U}$. In order to transfer the clustering label information into the quantization label, we construct the laplacian matrix based on the supervised method. In the laplacian similarity matrix construction, samples who have the same labels will be assigned a large similarity weights and a lower similarity value should be given to those samples who do not share the same clustering label. Assuming we have found a method to optimize (5) with respect to $Y^{(L)}$, then the regularizer term $\mathbf{tr}(Y^{(L)} L (Y^{(L)})^T)$ in (5) will force those labeled samples who share the same label to intend to get the same quantization label $Y^{(L)}$ or the assigned quantization center have the minimal distance value. However, as we have mentioned above that optimizing (5) is a NP-hard

problem and intractable. Two kinds of algorithms have emerged to ease above difficult problem, the first one is to adopt the exhuastive search method which is very common used method and have been used in optimized cartesian K-means [56], composite quantization [36] and Sparse quantization [60]. The second one is to relax the discrete variable into continuous and resort the relative constraint to the variable to enable the objective optimization to be tractable.

Take the SEED dataset for example, we use Y_C denotes the clustering label of the digital number and $Y_C \in \{0, 1, \ldots, 9\}$. Y_Q denotes the quantization label and $Y_Q \in \{1, 2, 3, \ldots, 256\}$ and $Y^{(L)}$ is used to denote the quantization label matrix with 1-of-K encode to be estimated as mentioned above. In another word, $Y^{(L)}$ is constructed by using the 1-of-K encode scheme based on the Y_Q value. Y_C is known, while Y_Q and $Y^{(L)}$ are unknown.

The regularizer term $\mathbf{tr}(Y^{(L)}L(Y^{(L)})^T)$ in (5) can implicitly constrain the quantization label Y_Q keep close relationship to the clustering label Y_C and the estimated $Y^{(L)}$ can guarantee the those labeled data who share the same label in the quantization center as close as possible. So label information is indirectly conducted to the quantization label Y_Q from Y_C.

4.1. ANN search

For any query q, we search the base set $S = \{s_1, \ldots, s_N\}$ to find those instances that are most closest to q. If the query q is the real value vector, we use asymmetric quantizer distance method to compute the distance between the query instance q and base instance $s_i, i \in \{1, \ldots, N\}$ where the distance of $\|q - s_i\|$ can be approximate by $\|k(q) - k(s_i)\|$. If the query q is the encode word, symmetric quantizer distance (SQD) is used to estimate the distance between q and s_i where $\|k(q) - k(s_i)\|$ is used to approximate the distance $\|q - s_i\|$. Where $k(q)$ and $k(s_i)$ denote the reconstructed query vector q and base set vector s_i respectively. As for the orthogonal cartesian K-means and ITQ algorithms in the experiments we use the asymmetric hamming (AH) distance and symmetric hamming (SH) distance to compute the distance when the query data is original data and encode data respectively.

4.2. Computation analysis

Immediately after that, we will give the computation complexity of our algorithms in the training and information retrieval task.

In the model training stage, we solve the objective function to get $Y^{(L)}$, R, D, μ and B iteratively. Optimizing R, D, μ and B have as same computation complexity as the original Cartesian K-means algorithm, however, besides those computation cost, still more computations needed for our proposed semi supervised Cartesian K-means. As mentioned previously, optimizing $Y^{(L)}$ is really a NP-hard problem, exhaustive search is a very common used strategy to deal this problem. All the three quantization algorithm-composite quantization [36], sparse quantization [60] as well as the optimized product quantization [33,63] have adopted the exhaustive search strategy to optimize the label variable. In our proposed semi-supervised cartesian K-means, exhaustive search is also used to optimize the objective function with respect to $Y^{(L)}$. Exhaustive search method can guarantee our objective function to obtain the global minimum, however it is really a very expensive choice. In each iteration of the semi-supervised cartesian K-means optimization model, the objective function (5) is computed about $256 \times N_L \times L_C/8$ times, where N_L represents the labeled instance number and L_C indicates the code length. And in the loss function computation, the computation complexity is $O(PN(K + P + 1) + 2KN_L^2)$. Assuming that we adopt the 32-bit coding length, optimizing (5) with respect to $Y^{(L)}$, the computation complexity is about $O(4N_L(PN(K + P + 1) + 2KN_L^2))$ at least, where P is dimensionality of samples, K indicates the number of quantization center, here $K = 256$. Where $N = N_L + N_U$, N_L and N_U indicate the labeled and unlabeled samples number respectively.

Based the above analysis, we proposed another algorithm to substitute the exhaustive search method. We relax the discrete variable to continuous and resort the constrain to it, finally to threshold the continuous variable to get the 1-of-k code. In each iteration, the computation complexity is $O(N_L^4)$, which is the operation to compute the Moore $-$Penrose pseudo inverse matrix of $C^\star$.

Apparently, our proposed algorithm has less computation complexity than the exhaustive search algorithm. Because the pseudo inverse computation, our algorithm still has large computation cost. Developing new algorithm to own less computation complexity and enhance the model performance is our main job in the following research work.

5. Experiment

In this experiment, we apply Intel Xeon E5-2680 2.40 GHz CPUs with 12 cores and 64 GB RAM. For more efficient deep learning, we use TITAN RTX GPU with 24 GB memory. And we use the PyTorch which is an outstanding deep learning framework to construct CNN and MNE-Python to process EEG data. In order to estimate the advantage of our proposed algorithm SSCK, we choose three datasets to implement a range of ANN search tests: SEED [64], fifteen volunteers with EEG signals and three sentiments(positive, neutral and negative) about eye motions; DEAP [65], Electroencephalogram and 32 patients with peripheral physiological signals; BCI [66], the MI EEG evaluation were resorted from 9 subjects and partition into 4 classifications of physical activities(left hand, right hand, foot and tongue).

The previous works show that pre-trained CNN models on large datasets can be transferred to extract CNN features for other image datasets. We use the VGG-Net [67] to extract features. As for selecting the features from CNN, those belonging to the shallow layers contain too many dimensions and they are too sparse to get appropriate results for classification. Moreover, the features of the deepest layer is totally corresponding to the original dataset, which is hard to transfer to other tasks. Therefore we select some middle layers of CNN to extract features for the recognition task.

Selecting appropriate data is a key problem in this paper, for example,we can randomly select eight subject data for training and one for subject-specific testing. In this manner, the system is tested without seeing the subject beforehand and is a completely new test case for the system. The testing is challenging, and they also can be further generalized. These sets are retained until complete training and testing are over. Finally, We calculate the results by averaging the values obtained at each stage. We use the supervised learning strategy to extract effective feature, the softmax classification function uses the output from the feature extraction function, such that the CNN network can optimize both functions simultaneously.

We compare our semi-supervised cartesian K-means, with many typical and recent approaches: product quantization (PQ) [30], cartesian K-means (CKmeans) [31], Correlated Attention Network (CAN) [68], Kernelized product quantization (KPQ) [37], online product quantization (OPQ) [38], semi-supervised extreme learning machine (SS-ELM) [69], orthogonal K-means (OKmeans) and iterative quantization (ITQ) [70] as well as the composite quantization(CQ) [36].

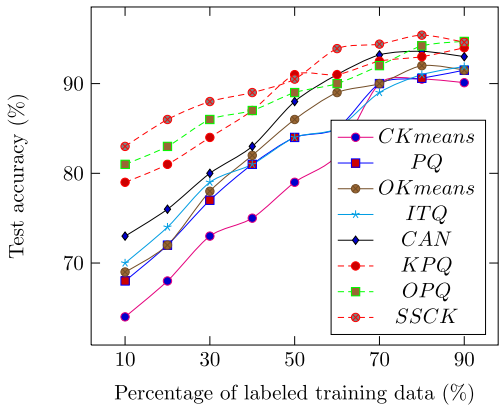


Fig. 2. Performance comparisons of eight methods on SEED dataset.

Table 2
An example of selection criteria.

Number	Emotion label	Film clips sources
1	Negative	Tangshan Earthquake
2	Negative	Back to 1942
3	Positive	Lost in Thailand
4	Positive	Flirting Scholar
5	Positive	Just Another Pandora Box
6	Neutral	World Heritage in China

Table 3
Comparisons of classification results using 32, 64, and 128 bits.

Algorithms	acc ± std (32 bits)	acc ± std (64 bits)	acc ± std (128 bits)
SSCK	**94.95 ± 0.68**	**95.16 ± 0.67**	**95.35 ± 0.70**
CKmeans	89.15 ± 1.17	89.78 ± 1.22	90.13 ± 1.18
PQ	90.73 ± 1.13	91.05 ± 1.08	91.54 ± 0.99
OKmeans	91.09 ± 1.08	91.58 ± 1.01	91.89 ± 0.98
ITQ	93.03 ± 0.81	93.38 ± 0.82	93.55 ± 0.79
CAN	93.25 ± 0.83	93.87 ± 0.79	94.03 ± 0.77
KPQ	93.85 ± 0.80	94.12 ± 0.78	94.21 ± 0.74
OPQ	93.95 ± 0.81	94.27 ± 0.77	94.76 ± 0.76

5.1. Experiments on SEED

The SEED dataset was were selected from material database as stimuli used in the experiments, ESI neural scanning system was used to record the eye movement signals with sampling rate of 1000 Hz and 62 channels in the electrode cap. SMI ETG eye-movement tracking glasses were used to record the eye movement signals. The signals recorded by the subjects when they watch the first 9 movie clips serve as the training dataset of each experiment, and the rest serve as the test dataset. 32 and 128 bits code length are used in the experiments. We used DE features as the eye movement features, which are the same as Lu et al. 2015 [64]. 32 and 128 bits code length are used in the experiments.

In Table 2 we have shown an example of selection criteria. The selection guidelines of film clips are: (a) the whole experiment length should not be too long to avoid fatigue of subjects; (b) the video can be easily understood; (c) these video should trigger a single target emotion. Each clip is about four minutes long and is carefully edited to create coherent emotions.

In Fig. 2 we present the recall rate experimental result comparisons of using SSCK and other typical and recent approaches on the SEED with 32 and 128 bits code length. From Fig. 2 we can see that the performance of SSCK surpass the other comparative algorithms on the SEED dataset. That is because we have found an appropriate method to optimize (4) with respect to Y. Cartesian K-means algorithm keep its priority as does in other dataset [31] and ITQ also have a better performance than the Cartesian K-means. That because we do not use the sift or gist feature to implement the search task and small amount samples used for model training. We only present the precision of 128 bits encode experimental result in 3, because the performance of all algorithms using the 32 and 128 bits have almost the same performance for using the pixel feature. Comparing the conventional cartesian K-means, our proposed semi-supervised cartesian K-means have improved the performance effectively, which shows the efficiency of our semi-supervised cartesian K-means model. Orthogonal K-means model gives the worst performance.

From Fig. 2 we can observe a curious phenomenon that the semi-supervised cartesian K-means algorithm have a better performance by using the symmetric quantize distance (SQD) than using the asymmetric quantizer distance (AQD) method, which is different from the previously conclusions. We think that is because we have introduced much locality information of the samples. Besides the recall rate experimental result comparisons, we also compare the classification performance (based on average accuracy and standard deviation acc ± std) of these algorithms on this data, detailed experimental results are given in 3. We can easily draw that the semi-supervised cartesian K-means' priority over the other algorithms. In this experiment the parameter α and η for SSCK are well adjusted, we set $\alpha = 0.2$ and $\eta = 0.5$.

5.2. Experiments on DEAP

The DEAP dataset is a collection of signals collected when participants watch a minute-long emotion music video. We chose 5 as the threshold and divided the experiments into two categories on account of the levels of excitement and titer. The downloaded preprocessing data is used in our experiment. Then, we selected 1000 samples of each class from the basic dataset and divided them into two parts, one as labeled data and the other as unlabeled data, to train our proposed semi-supervised cartesian k-mean. For the other comparative algorithms, we use the whole 10 000 instances to train the model. The comparisons of precision are given 4 and 5 respectively. The classification performance is measured based on average accuracy and standard deviation (acc ± std).

In Table 4 we have shown an example of our used samples, this file is available in Open-Office Calc (online ratings.ods), Microsoft Excel (online _ratings.xls), and Comma separated values (online _ratings.csv) formats. The scores were collected through an online self-assessment software as cited in [71]. Participants used Sam's manikin to rate arousal, valence and dominance on a discrete nine-point scale. Participants also rated their feelings through the emotional wheel tool (refer [8]).

In this experiment, it seems that all the algorithms can the AQ and SQ ANN search have the same performance and we only present the AQD method recall precision of all the algorithms using the 128 bits code length (see Fig. 3).

Orthogonal K-means and cartesian K-means have unsatisfying results, which are different from the results obtained on the SEED dataset. AQD and SQD method have the same performance we think that because we do not use large amount samples for the model training and the samples are the high dimensionality data. The experimental result comparisons between SSCK and cartesian K-means show the efficiency of the SSCK on the image retrieval task. The parameter value of α and η are given by $\alpha = 0.5$ and $\eta = 0.6$.

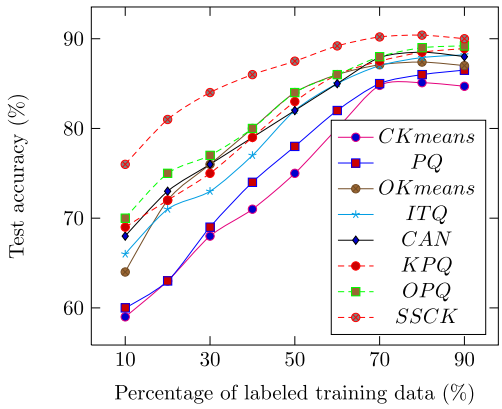


Fig. 3. Performance comparisons of eight methods on DEAP dataset.

Table 4
An example of dataset summary.

File name	Format	Section	Details
Online_ratings	Spreadsheet	Online self-assessment	Individual ratings.
Video_list	Spreadsheet	Both parts	Music videos in YouTube links
Participant_1	Spreadsheet	Experiment	All participants rated video.
Participant_2	Spreadsheet	Experiment	Participants' answers.
Face_video	Zip file	Experiment	Face video on the front.
Data_original	Zip file	Experiment	Raw physiological data.
Data_preprocessed	Python, Matlab	Experiment	Preprocessed physiological data.

Table 5
Comparisons of classification results using 32, 64, and 128 bits.

Algorithms	acc ± std (32 bits)	acc ± std (64 bits)	acc ± std (128 bits)
SSCK	**89.96 ± 1.51**	**90.16 ± 1.47**	**90.36 ± 1.41**
CKmeans	84.16 ± 2.11	84.73 ± 2.06	85.11 ± 2.03
PQ	85.23 ± 1.99	85.75 ± 1.98	86.51 ± 1.94
OKmeans	85.89 ± 1.96	86.58 ± 1.91	87.39 ± 1.88
ITQ	86.63 ± 1.91	87.38 ± 1.89	88.15 ± 1.85
CAN	87.29 ± 1.83	87.87 ± 1.87	88.51 ± 1.79
KPQ	88.05 ± 1.83	88.32 ± 1.81	88.89 ± 1.74
OPQ	88.55 ± 1.81	88.97 ± 1.77	89.16 ± 1.72

5.3. Experiments on BCI datasets

The dataset is a collection of EEG data from 9 subjects. The clue-based BCI paradigm contain four different motor representation tasks, namely, the motor imagination of the left hand (class 1), right hand (class 2), foot (class 3), and tongue (class 4). Each session contain six runs with a short break. A run contains 48 trials (12 for each of the four possible classes), resulting in a total of 288 trials per session.

The performance of different algorithms is analyzed for different proportion of labeled training samples on BCI dataset. In 288 trials of experimental training, testing set contains 28 samples randomly chose from the whole data, and 10%–90% of the remaining 260 samples were naturally selected as the labeled training samples, and the rest were used as the unlabeled training samples. Moreover, the 288 samples is also set as the test samples. The process by which we distribute samples for ten repeats. Fig. 4 gives the results of four representative subjects (A01, A02, A03,A04,A05,A06,A07, A08).

Our experiments randomly choose 50 samples of each object as the query and the reminder used as the base data. 500 samples of each class are then selected from the base dataset to train those comparative models such as cartesian K-means, orthogonal K-means, product quantization and iterative quantization algorithms. For SSCK model, the 1000 samples are averagely divided into two sections, one section acts as the labeled samples and the other is the unlabeled samples . The corresponding contrast experiments are summarized in Fig. 4 and Table 6 respectively. Experimental results show that SSCK perform better than its competitors on all the three experiments of using 32 and 128 bits code length respectively. Product quantization have the worst performance except by using 32 bits code experiment. That is because the feature occur a great change on the bits code and this kind of algorithm is not suitable any more, all the algorithms except the product quantization have adopted the rotation matrix to adjust the dimensionality of samples to reduce the errors. In this experiment we use the common spatial pattern(CSP) approach in [66] to implement the ANN search work, those algorithms who have adopted the rotation matrix strategy can implicitly reduce the objective function error and improve the ANN search performance, however, product quantization algorithm does not use the rotation matrix to adjust the feature and cannot get the satisfying result. Comparing with the cartesian K-means, our proposed SSCK can effectively improve the ANN search performance on the BCI dataset.

Unlike the circumstances presented in [31], all the algorithms use the image pixel as the feature cannot testify that the cartesian K-means priority over the iterative quantization and AQ (AH) and SQ (SH) method have almost the same performance.

Additionally, in order to emphasize the necessity of our research and make our experiments more complete, We have added more comparative experiments and statistical hypotheses to demonstrate the effectiveness of the algorithm. we compare the proposed model to four recent and representative algorithms, including hierarchical semi-supervised extreme learning machine method (HSS-ELM) [66], wavelet transform time–frequency image and convolutional network based approach (WTT-CNN) [72], Multilevel Weighted Feature Fusion Using Convolutional Neural Networks (MWF-CNN) [73], and novel fused convolutional neural network (FCNN) [74]. The classification performance is evaluated

Table 6
Classification accuracy on the BCI Dataset using 128 bits.

Algorithms	*SSCK*	*CKmeans*	*PQ*	*OKmeans*	*ITQ*	*SS − ELM*	*KPQ*	*OPQ*	*HSS − ELM*
A01	83.37	73.77	74.82	76.66	78.65	79.27	81.93	82.74	81.14
A02	51.04	43.17	44.57	46.24	47.61	48.68	48.78	49.84	49.86
A03	78.69	70.23	71.59	77.23	76.12	77.95	77.77	77.81	78.02
A04	63.93	59.09	58.55	60.29	61.53	62.99	63.78	62.89	63.33
A05	45.36	39.25	40.62	41.28	43.56	44.86	44.80	44.02	44.03
A06	50.33	41.35	43.90	45.39	46.59	48.95	48.81	48.98	49.44
A07	82.02	73.12	75.54	76.23	79.03	80.83	80.77	81.94	81.11
A08	82.06	74.03	76.51	78.21	80.51	81.21	81.76	81.87	81.49
A09	82.62	72.12	74.44	76.16	79.48	80.69	80.96	81.81	81.38
MC	68.82	60.68	62.28	64.19	65.90	67.27	67.70	67.99	67.76
MK	0.5886	0.5021	0.5232	0.5417	0.5571	0.5636	0.5756	0.5792	0.5701

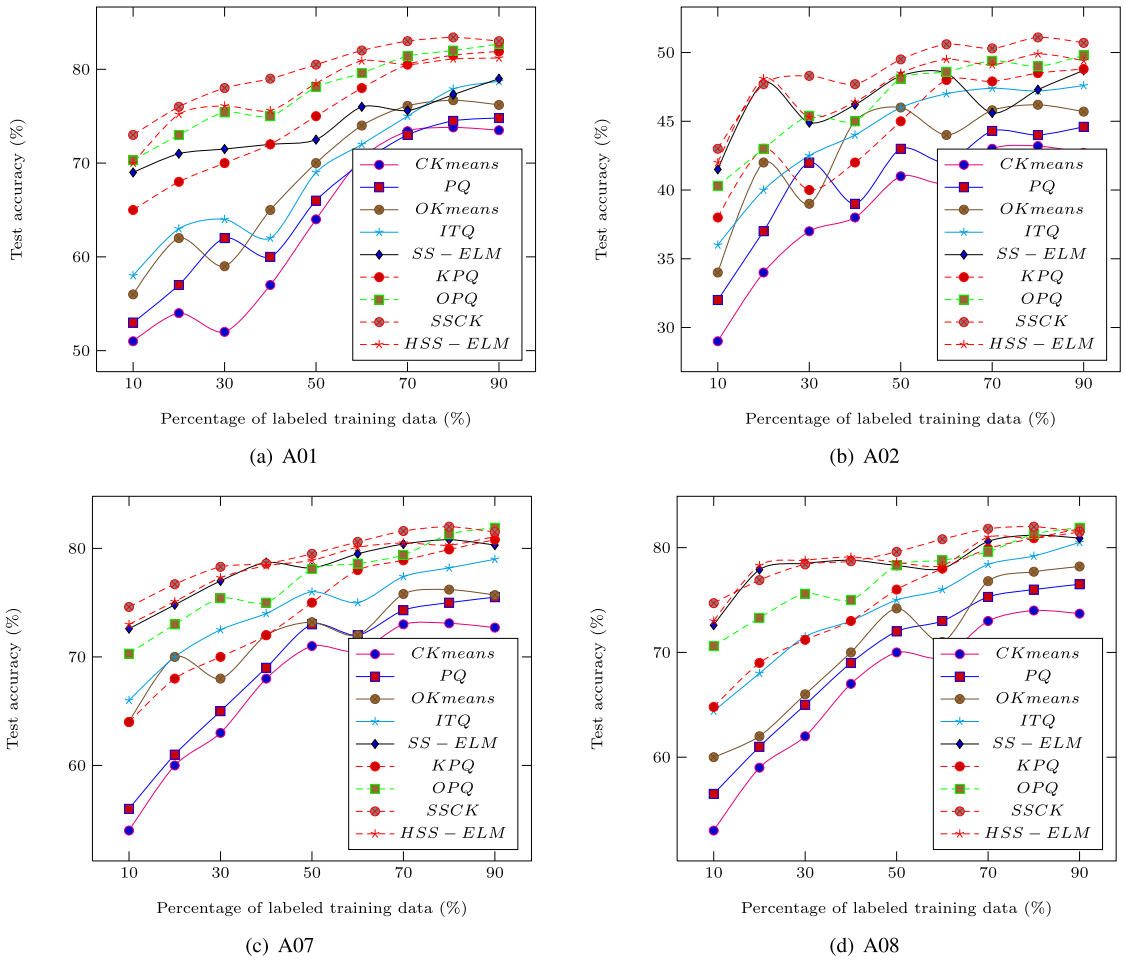


Fig. 4. Performance comparisons of nine approaches on BCI datasets.

Table 7
Comparisons of classification results and training time on SEED dataset.

Algorithms	acc ± std	Time (s)
OurAlgorithm	**95.35 ± 0.71**	58
HSS − ELM	92.16 ± 1.21	**28.175**
WTT − CNN	91.83 ± 1.39	2800
MWF − CNN	90.89 ± 1.46	2280
FCNN	91.63 ± 1.41	1300

Table 8
Comparisons of classification results and training time on DEAP dataset.

Algorithms	acc ± std	Time (s)
OurAlgorithm	**90.36 ± 1.41**	49
HSS − ELM	87.86 ± 1.71	**26.218**
WTT − CNN	87.83 ± 1.79	2710
MWF − CNN	87.17 ± 1.83	2190
FCNN	88.03 ± 1.65	1220

Table 9
Comparisons of classification results and training time on BCI dataset.

Algorithms	acc ± std	Time (s)
OurAlgorithm	**76.86 ± 1.78**	42
HSS − ELM	72.16 ± 2.21	**24.852**
WTT − CNN	71.89 ± 2.39	2500
MWF − CNN	73.68 ± 2.06	2056
FCNN	74.29 ± 1.91	1170

in terms of average accuracy and standard deviation (acc ± std). Tables 7–9 summarizes the classification results. From the results shown in above three tables, it is evidenced that our proposed algorithm achieved best performance, the pure supervised learning algorithms, such as HSS-ELM, WTT-CNN, MWF-CNN, and FCNN with deep architecture, present Inferior performances. And it demonstrates that our constructed semi-supervised cartesian K-means algorithm actually enhanced classification accuracy, suggesting effective exploitation of unlabeled data. It is also observed that our designed model with deep features outperformed the other existing deep learning algorithm on all three datasets, indicating the successful incorporation of deep network architecture to extract compact and high-level features. Despite the computational efficiency is not the best, compared with deep learning algorithms, the computation complexity of our algorithm is relatively very low on all of the datasets.

To give more statistical analysis, we perform $t-test$ for the precision obtained by HSS-ELM method, WTT-CNN method, MWF-CNN method, FCNN method and by our approach on SEED, DEAP and BCI dataset, under the null hypothesis using a significance level of 0.05. p$-value$ is found as 0.00082, 0.00102 and 0.00221 respectively, indicating that accuracy rate performed by our approach is indeed significantly better than above four recent proposed methods.

6. Conclusion

EI EMM recognition could promote the interaction between human and intelligent devices. We propose a semi-supervised

cartesian K-means architecture based on CNN for EEG MI classification. The kind of recognition model not only captures the local activities responding to emotion, but also exploits the interactions among intracranial nerve areas. In our research work, we have designed the semi-supervised cartesian K-means model based on the cartesian K-means and the orthogonal optimal reverse prediction algorithms. To promote the traditional cartesian K-means performance, we adopt the samples' label information to construct the similarity matrix. Those samples who share the same label are imposed a high weights value and a lower weight value are given to those who have a different labels. Those instances who have the same label will be assigned to the same or the nearest clustering in the quantization step. However, optimizing objective function with respect to the label matrixes is a relatively difficult problem. We optimize the equation separately and the performance is validated over three public datasets. Detailed experimental analysis indicates our proposed algorithms perform better than other comparative algorithms. We have obtained a good result on the research of the semi-supervised cartesian K-means algorithm, however, more effort are needed to solve the problems, such as finding a more suitable algorithm for optimizing the semi-supervised cartesian K-means objective function and better semi-supervised quantization algorithms to be proposed to improve the performance. By incorporating deep features of deep network model, it proves to be more effective to improve the classification and evaluation of MI EEG signals.

The main contributions of our work are mainly reflected in four aspects: In view of the number of labeled samples is very small, we propose a semi-supervised cartesian K-means algorithm with deep features extracted from CNN in EEG data; Labeled data is integrated in the quantization step to provide the additional constraints to promote the data reconstruction ability; Laplacian matrix is build based on the label data and is added to the optimal reverse prediction model to get more discriminative cartesian K-means model; Strategy to optimize the semi-supervised cartesian K-means is given to enable the function to get an minimum value. The framework proposed is designed to extract spectral, temporal features from EEG motor data while learning general spatially invariant characteristics of MI tasks. The multilayer feature fusion methods based on CNN have yet to be tested on other EEG datasets. Therefore, our method can also be used for other EEG applications.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106071.

CRediT authorship contribution statement

Minjie Liu: Investigation, Methodology, Software, Writing - original draft. **Mingming Zhou:** Investigation, Data curation, Methodology, Visualization. **Tao Zhang:** Conceptualization, Writing - review & editing, Supervision. **Naixue Xiong:** Visualization, Data curation, Validation.

Acknowledgments

This research was partly supported by National Science Foundation, China (No. 61702226, 21365008, 61562013), the Natural Science Foundation of Jiangsu Province, China (Grant no. BK20170200, BK20161135), the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Land and Resources, China (KF-2018-03-065), the Fundamental Research Funds for the Central Universities, China (JUSRP11854).

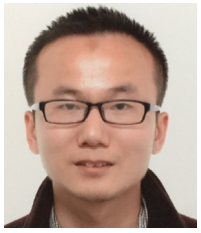
References

[1] G. Wenzhong, J. Li, C. Guolong, N. Yuzhen, C. Chengyu, A pso-optimized real-time fault-tolerant task allocation algorithm in wireless sensor networks, Trans. Parallel Distrib. Syst. 26 (2015) 3236–3249.
[2] S. Zhirong, P. P. C. Lee, S. Jiwu, G. Wenzhong, Encoding-aware data placement for efficient degraded reads in xor-coded storage systems: Algorithms and evaluation, Trans. Parallel Distrib. Syst. 29 (2018) 2757–2770.
[3] C. Yongli, J. Hong, W. Fang, H. Yu, F. Dan, G. Wenzhong, W. Yunxiang, Using high-bandwidth networks efficiently for fast graph computation, Trans. Parallel Distrib. Syst. 30 (2019) 1170–1183.
[4] H. Xing, G. Wenzhong, L. Genggeng, C. Guolong, Fh-oaos:a fast 4-step heuristic for obstacle-avoiding octilinear architecture router construction, ACM Trans. Des. Autom. Electron. Syst. 21 (2016) 1–31.
[5] H. Xing, L. Genggeng, G. Wenzhong, N. Yuzhen, C. Guolong, Obstacle-avoiding algorithm in x-architecture based on discrete particle swarm optimization for vlsi design, ACM Trans. Des. Autom. Electron. Syst. 20 (2015) 24–28.
[6] L. Bin, G. Wenzhong, X. Naixue, C. Guolong, V.V. Athanasios, Z. Hong, A pretreatment workflow scheduling approach for big data applications in multi-cloud environments, Trans. Netw. Serv. Manage. 13 (2016) 581–594.
[7] X. Kelvin, B. Jimmy, K. Ryan, C. Kyunghyun, C. A.C, Brain-computer interfaces: principles and practice, 2012.
[8] K.D. Sidney, M.K. Jacqueline, A review and meta-analysis of multimodal affect detection systems, ACM Comput. Surv. 47 (2015) 43–50.
[9] X. Kelvin, B. Jimmy, K. Ryan, C. Kyunghyun, C. A.C, Show, attend and tell: Neural image caption generation with visual attention, in: ICML, Vol. 14, 2015, pp. 77–81.
[10] P. Gert, N. Christa, Motor imagery and direct brain-computer communication, Proc. IEEE 89 (2001) 1123–1134.
[11] S. Saeid, A.C. Jonathon, EEG Signal Processing, John Wiley & Sons, 2007.
[12] Z. Weiping, G. Wenzhong, Y. Zhiyong, X. Haoyi, Multitask allocation to heterogeneous participants in mobile crowd sensing, Wirel. Commun. Mobile Comput. 2018 (2018) 721–731.
[13] M. Yuchang, X. Liudong, L. Yi-Kuei, G. Wenzhong, Efficient analysis of repairable computing systems subject to scheduled checkpointing, Trans. Dependable Secure Comput. 2018 (2018) 286–293.
[14] Y. Yang, L. Ximeng, Z. Xianghan, R. Chunming, G. Wenzhong, Efficient traceable authorization search system for secure cloud storage, Trans. Cloud Comput. Online Publ. 2018 (2018) 282–294.
[15] W. Shiping, G. Wenzhong, Sparse multi-graph embedding for multimodal feature representation, Trans. Multimedia 19 (2017) 1454–1466.
[16] L. Fangfang, G. Wenzhong, Y. Yuanlong, C. Guolong, A multi-label classification algorithm based on kernel extreme learning machine, Neurocomputing 260 (2016) 313–320.
[17] N. Yuzhen, C. Jianer, G. Wenzhong, Meta-metric for saliency detection evaluation metrics based on application preference, Multimedia Tools Appl. 77 (2018) 26351–26369.
[18] S. Wojciech, K. Motoaki, M. Klausrobert, Divergence-based framework for common spatial patterns algorithms, IEEE Rev. Biomed. Eng. 7 (2014) 50–72.
[19] S. Qingshan, G. Haitao, M. Yuliang, L. Zhizeng, Scale-dependent signal identification in low-dimensional subspace: motor imagery task classification, Neural Plast. 2016 (2016) 743–752.
[20] H.F. Jerome, J.L. Bentley, F. Raphael Ari, An algorithm for finding best matches in logarithmic expected time, ACM Trans. Math. Softw. (TOMS) 3 (3) (1977) 209–226.
[21] C. Silpa-Anan, R. Hartley, Optimised kd-trees for fast image descriptor matching, in: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1–8.
[22] A. Gionis, P. Indyk, R. Motwani, et al., Similarity search in high dimensions via hashing, in: VLDB, Vol. 99, 1999, pp. 518–529.
[23] M. Datar, N. Immorlica, P. Indyk, V.S. Mirrokni, Locality-sensitive hashing scheme based on p-stable distributions, in: Proceedings of the Twentieth Annual Symposium on Computational Geometry, 2004, pp. 253–262.
[24] B. Kulis, K. Grauman, Kernelized locality-sensitive hashing, IEEE Trans. Pattern Anal. Mach. Intell. 34 (6) (2012) 1092–1104.
[25] W. Jun, K. Sanjiv, C. Shih-Fu, Semi-supervised hashing for scalable image retrieval, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 3424–3431.
[26] T. Zhang, W. Jia, C. Gong, J. Sun, X. Song, Semi-supervised dictionary learning via local sparse constraints for violence detection, Pattern Recognit. Lett. 107 (2018) 98–104.
[27] H. Kaiming, F. Wen, J. Sun, K-means hashing: An affinity-preserving quantization method for learning binary compact codes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2938–2945.
[28] L. Wei, W. Jun, J. Rongrong, J. Yu-Gang, C. Shih-Fu, Supervised hashing with kernels, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, 2012, pp. 2074–2081.

[29] T. Zhang, W. Jia, X. He, J. Yang, Discriminative dictionary learning with motion weber local descriptor for violence detection, IEEE Trans. Circuits Syst. Video Technol. 27 (3) (2017) 696–709.
[30] H. Jegou, M. Douze, C. Schmid, Product quantization for nearest neighbor search, IEEE Trans. Pattern Anal. Mach. Intell. 33 (1) (2011) 117–128.
[31] M. Norouzi, D.J. Fleet, Cartesian k-means, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),, 2013, pp. 3017–3024.
[32] S. Lloyd, Least squares quantization in pcm, Inf. Theory IEEE Trans. 28 (2) (1982) 129–137.
[33] G. Tiezheng, H. Kaiming, K. Qifa, S. Jian, Optimized product quantization for approximate nearest neighbor search, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, 2013, pp. 2946–2953.
[34] Y. Kalantidis, Y. Avrithis, Locally optimized product quantization for approximate nearest neighbor search, in: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, 2014, pp. 2329–2336.
[35] W. Jianfeng, J. Song, X. Xu, H. Shen, S. Li, Optimized cartesian k-means, 2014.
[36] Z. Ting, C. Du, W. Jingdong, Composite quantization for approximate nearest neighbor search, in: ICML, 2014, pp. 838–846.
[37] J. Liu, Y. Zhang, J. Zhou, Kernelized product quantization, neurocomputing 235 (2017) 15–26.
[38] D. Xu, I.W. Tsang, Y. Zhang, Online product quantization, IEEE Trans. Knowl. Data Eng. 30 (2018) 2185–2198.
[39] S. Hui-Wen, F. Yun-Fa, X. Xin, Identification of eeg induced by motor imagery based on hilbert-huang transform, Acta Automat. Sinica (2015) 1686–1692.
[40] F. Yun-Fa, X. Bao-Lei, L. Yong-Cheng, Recognition of actual grip force movement modes based on movement-related cortical potentials, Acta Automat. Sinica (2014) 1045–1057.
[41] A.M. Ebrahim, J.M. Jerome, B.F. Paul, J.L. Brian, Wavelet common spatial pattern in asynchronous offline brain computer interfaces, Biomed. Signal Process. Control (2011) 121–128.
[42] H. Wei-Chun, L. Li-Fong, C. Chun-Wei, H. Yu-Tsung, L. Yi-Hung, Eeg classification of imaginary lower limb stepping movements based on fuzzy support vector machine with kernel-induced membership function, Int. J. Fuzzy Syst. (2017) 566–579.
[43] H. Lianghua, H. Die, W. Meng, W. Ying, D. Karen M. von, Z. MengChu, Common bayesian network for classification of eeg-based multiclass motor imagery bci, IEEE Trans. Syst. Man Cybern.: Syst. (2016) 843–854.
[44] I. Md Rabiul, T. Toshihisa, M. Islam, Multiband tangent space mapping and feature selection for classification of eeg during motor imagery, J. Neural Eng. (2018) 1333–1344.
[45] J.D. Lpez, V. Litvak, J.J. Espinosa, K. Friston, G. Barnes, Algorithmic procedures for bayesian meg/eeg source reconstruction in spm, NeuroImage 84 (1) (2014) 476–487.
[46] B. P. Hayes, J. K. Gruber, M. Prodanovic, A closed-loop state estimation tool for mv network monitoring and operation, IEEE Trans. Smart Grid 6 (4) (2015) 2116–2125.
[47] C. Eduardo, S. Miho, N. Isao, W. Yasuhiro, Convolutional neural networks with 3d input for p300 identification in auditory brain-computer interfaces, Comput. Intell. Neurosci. (2017) 16–25.
[48] Y.R. Tabar, U. Halici, A novel deep learning approach for classification of eeg motor imagery signals, J. Neural Eng. (2017) 159–169.
[49] X. An, D. Kuang, X. Guo, Y. Zhao, L. He, A deep learning method for classification of eeg data based on motor imagery, Intell. Comput. Bioinform. (2014) 203–210.
[50] H. Yang, S. Sakhavi, K.K. Ang, C. Guan, On the use of convolutional neural networks and augmented csp features for multi-class motor imagery of eeg signals classification, in: 2015 37th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society, 2015, pp. 2620–2623.
[51] V.J. Lawhern, A.J. Solon, N.R. Waytowich, Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces, J. Neural Eng. (2018) 369–374.
[52] Y. Zhang, Y. Qian, D. Wu, M.S. Hossain, A. Ghoneim, M. Chen, Emotion-aware multimedia system security, IEEE Trans. Multimed. (2018) 1251–1259.
[53] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient Based Learning Applied to Document Recognition, IEEE, 1998, pp. 2278–2324.
[54] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy layer-wise training of deep networks, Adv. Neural Inform. Process. Syst. (2007) 136–147.
[55] X. Linli, M. White, D. Schuurmans, Optimal reverse prediction: a unified perspective on supervised, unsupervised and semi-supervised learning, in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 1137–1144.
[56] W. Jianfeng, J. Song, X. Xu, H. Shen, S. Li, Optimized cartesian k-means, IEEE Trans. Knowl. Data Eng. 27 (1) (2015) 180–192.
[57] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[58] U. Von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.
[59] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Trans. Image Process. 20 (5) (2011) 1327–1336.
[60] Z. Ting, Q. Guo-Jun, T. Jinhui, W. Jingdong, Sparse composite quantization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4548–4556.
[61] W. Zaiwen, Y. Wotao, A feasible method for optimization with orthogonality constraints, Math. Program. 142 (1–2) (2013) 397–434.
[62] P.H. schonemann, A generalized solution of the orthogonal procrustes problem, Psychometrika 31 (1) (1966) 1–10.
[63] G. Tiezheng, H. Kaiming, K. Qifa, S. Jian, Optimized product quantization, 2014.
[64] Y. Lu, W. Zheng, B. Li, Combining eye movements and eeg to enhance emotion recognition, in: IJCAI'15 Proceedings of the 24th International Conference on Artificial Intelligence, 2015, pp. 1170–1176.
[65] Z. Yin, M. Zhao, Y. Wang, Recognition of emotions using multimodal physiological signals and an ensemble deep learning model, Comput. Methods Programs Biomed. 140 (2017) 93–110.
[66] Q. She, Z. Luo, T. Nguyen, Y. Zhang, A hierarchical semi-supervised extreme learning machine method for eeg recognition, Med. Biol. Eng. Comput. 57 (2019) 147–157.
[67] S. Christian, L. Wei, J. Yangqing, Going deeper with convolutions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9.
[68] J. Qiu, X. Li, K. Hu, Correlated attention networks for multimodal emotion recognition, in: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2018, pp. 2656–2660.
[69] G. Huang, S. Song, J. Gupta, Semi-supervised and unsupervised extreme learning machines, IEEE Trans. Cybern. 44 (2014) 2405–2417.
[70] G. Yunchao, S. Lazebnik, Iterative quantization: A procrustean approach to learning binary codes, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, 2011, pp. 817–824.
[71] S. Mohammad, P. Maja, P. Thierry, Multimodal emotion recognition in response to videos, Affect. Comput. Intell. Interact. (2015) 491–497.
[72] X. Baoguo, S. Linlin, W. Aiguo, Wavelet transform time-frequency image and convolutional network based motor imagery eeg classification, IEEE Access (2018) 1–11.
[73] A. Syed-Umar, A. Mansour, M. Ghulam, Multilevel weighted feature fusion using convolutional neural networks for eeg motor imagery classification, IEEE Access (2019) 16–26.
[74] P. Shutao, A novel fused convolutional neural network for biomedical image classification, Med. Biol. Eng. Comput. (2019) 107–121.

Minjie Liu received the bachelor's degree from Changjiang University, Jingzhou, China, in 2006, and the M.S. degree from the Physiology, Liaoning Medical University, Jinzhou, China, in 2013. She is currently a lecturer with the School of Nursing, Taihu University of Wuxi, Wuxi, China. Her research interests focus on understanding the mechanisms of electromagnetic activities in biological tissue and systems, computational modeling and analysis of organ systems to aid clinical diagnosis of dysfunction in the human body.

Mingming Zhou received the M.S. degree from Anatomy and histoembryology, Nantong University Medical School, Jiangsu Province, China, in 2004, and the Ph.D.degree from Molecular Medicine, Fudan University Medical School, Shanghai, China, in 2009. He is currently a associate professor with the School of Nursing, Taihu University of Wuxi, Wuxi, China. He has authored over twenty quality journal articles and conference papers. His research interests focus on understanding the mechanisms of electromagnetic activities in biological tissue and systems, computational modeling and analysis of organ systems to aid clinical diagnosis of dysfunction in the human body.



Tao Zhang received the bachelor's degree from Henan Polytechnic University, Jiaozuo, China, in 2008,and the Ph.D.degree from the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China, in 2016. He is currently a associate professor with the Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi, China. He has led many research projects (e.g.,the National Science Foundation and the National Joint Fund), He has authored over thirty quality journal articles and conference papers. His current research interests include medical image processing, medical data analysis, visual surveillance, scene understanding, behavior analysis, object detection,and pattern analysis.

Naixue Xiong is current a Professor in College of Intelligence and Computing, Tianjin University, China. He received his both PhD degrees in Wuhan University (about sensor system engineering), and Japan Advanced Institute of Science and Technology (about dependable sensor networks), respectively. Before he attended Tianjin University, he worked in Northeastern State University, Georgia State University, Wentworth Technology Institution, and Colorado Technical University (full professor about 5 years) about 10 years. His research interests include Cloud Computing, Security and Dependability, Parallel and Distributed Computing, Networks, and Optimization Theory.