Robust Support Matrix Machine for Single Trial EEG Classification

Qingqing Zheng, Fengyuan Zhu, *Member, IEEE*, and Pheng-Ann Heng, *Senior Member, IEEE*

***Abstract*— Electroencephalogram (EEG) signals are of complex structure and can be naturally represented as matrices. Classification is one of the most important steps for EEG signal processing. Newly developed classifiers can handle these matrix-form data by adding low-rank constraint to leverage the correlation within each data. However, classification of EEG signals is still challenging, because EEG signals are always contaminated by measurement artifacts, outliers, and non-standard noise sources. As a result, existing matrix classifiers may suffer from performance degradation, because they typically assume that the input EEG signals are clean. In this paper, to account for intra-sample outliers, we propose a novel classifier called a robust support matrix machine (RSMM), for single trial EEG data in matrix form. Inspired by the fact that empirical EEG signals contain strong correlation information, we assume that each EEG matrix can be decomposed into a latent low-rank clean matrix plus a sparse noise matrix. We simultaneously perform signal recovery and train the classifier based on the clean EEG matrices. We formulate our RSMM in a unified framework and present an effective solver based on the alternating direction method of multipliers. To evaluate the proposed method, we conduct extensive classification experiments on real binary EEG signals. The experimental results show that our method has outperformed the state-of-the-art matrix classifiers. This paper may lead to the development of robust brain–computer interfaces (BCIs) with intuitive motor imagery and thus promote the broad use of the noninvasive BCIs technology.**

***Index Terms*— Brain computer interfaces, electroencephalograph, motor imagery, support matrix machine, robust classification.**

I. Introduction

BRAIN computer interfaces (BCIs) provide an advanced technology that translates the intent of a user into computer command by recognizing a task-related neuronal activity, and thereby can establish direct nonmuscular communication between a human brain and external devices. Such systems are promising for rehabilitation of environmental control abilities for severely disabled people [1], [2], as well as for promotion of productivity in many applications for healthy people [3]. Among various non-invasive measurements of neuronal activity, electroencephalography (EEG) is most widely used to record electrical activity of the brain from the scalp due to its simplicity and high temporal resolution [4]. Practically, EEG signals record the voltage fluctuations at several electrodes over a time segment during mental tasks, see Fig. 1. From the viewpoint of pattern recognition, EEG-based BCIs perform detection of EEG signals associated with specific mental tasks, which is referred to as single trial EEG classification problem. Single trial EEG classification can be very challenging due to poor characteristics of EEG caused by a variety of sources, e.g., measurement artifacts, outliers and non-standard noises [5]. To alleviate these problems, spatial filtering has been developed as a preprocessing technique to explore the discriminative spatial patterns and eliminate uncorrelated informations. To present, one of the most effective algorithms is the common spatial pattern (CSP) [6]. The basic idea of CSP is to seek optimal spatial projection such that the ratio of filtered variance between the two classes is maximized (or minimized) [7]. However, CSP and its various extensions [8]–[14] only make use of the inter-sample statistic information [15] (e.g., average, covariance), and thus cannot address the corruptions within each sample, namely the *intra-sample* outliers. While as claimed in [16], even a small number of outliers within the extracted features can arbitrarily bias the estimation of model parameters.

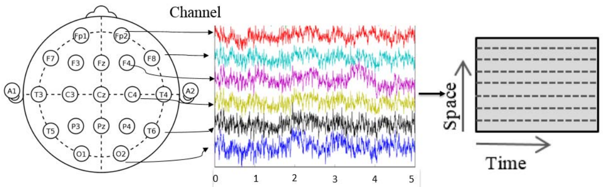


Fig. 1. EEG signals acquisition.

Besides intra-sample outliers, the extracted matrix-form features can not be classified by the traditional classifiers directly for the single trial EEG classification. The extracted features for each EEG sample containing rich spatial-temporal patterns can be naturally represented as a matrix with strong correlation among rows and columns (Fig. 1). To process these features, traditional linear or nonlinear classifiers, such as Fisher's linear discriminative analysis (LDA) [17], [18], support vector machine (SVM) [19], [20] and logistic regression [21], [22], have to reshape the matrix-form features into a vector or further extract advanced features in vector form,

Manuscript received April 14, 2017; revised November 18, 2017; accepted January 8, 2018. Date of publication January 17, 2018; date of current version March 6, 2018. This work was supported in part by the National Basic Program of China through the 973 Program under Project 2015CB351706, in part by the National Natural Science Foundation of China under Project 61233012, and in part by the Research Grants Council of the Hong Kong Special Administrative Region under Project CUHK 14225616. *(Corresponding author: Qingqing Zheng.)*

Q. Zheng and F. Zhu are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: qqzheng@cse.cuhk.edu.hk).

P.-A. Heng is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, and also with the Guangdong Provincial Key Laboratory of Computer Vision and Virtual Reality Technology, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China.

Digital Object Identifier 10.1109/TNSRE.2018.2794534

which would inevitably destroy the latent structure among each matrix-form data, resulting in degraded performance.

Recently, matrix classification methods address this issue and have achieved better performance, by directly processing data in matrix form with the structural correlation within EEG signals well preserved [23]. Such matrix classifier methods introduce certain constraints on the regression matrix to leverage the correlation within EEG signals. Examples include rank-k SVM [24] and bilinear classifiers [25], [26], which incorporate the low-rank property into the regression matrix. However, these methods require the rank of the regression matrix to be pre-determined, resulting in difficult and tedious tuning procedures. With the development of low-rank matrix analysis, nuclear norm was studied and used for low-rank approximation [27], [28]. Zhou and Li [29] proposed a novel logistic regression model by introducing nuclear norm for low-rank integrated regularization. Luo *et al.* [30] proposed the support matrix machine model by deriving a spectral elastic net regularization, which is a linear combination of nuclear norm and Frobenius norm of the regression matrix. However, all existing matrix classification methods assume the input EEG signals to be noise free, and thus lack robustness to noises and outliers in real-world applications.

In this paper, we propose a novel matrix classifier called "*Robust Support Matrix Machine*" (RSMM) for single trial EEG classification, which can effectively address the aforementioned issues by simultaneously eliminating outliers within EEG signals, and training a matrix classifier using the clean data. For feature recovery, our RSMM method is motivated by the important fact that rows and columns among each EEG matrix involve rich structural correlation information [31]. At the same time, outliers and artifacts are usually observed with large amplitudes [32], [33]. Taking advantage of these properties, we recover clean features by decomposing each EEG signal into a low-rank representable matrix, plus a sparse matrix for intra-sample outliers. For classifier training, we employ hinge loss in the energy function due to its desirable properties in sparseness and robustness modeling. Similar to [30], we also introduce nuclear norm of the regression matrix as a regularization to capture the global correlation among recovered clean EEG matrices. The proposed RSMM integrates feature recovery and classifier training into a unified framework. In this way, the matrix classifier is trained based on clean features, thus robust against outliers; and the improved classification performance would in turn provide efficient feedback for artifact removal. The resulting optimization problem for RSMM is non-smooth and non-differentiable, we further derive an efficient proximal solver based on the alternating direction method of multipliers (ADMM) framework. Finally, we conduct extensively comparative experiments on three real-world EEG datasets to evaluate the proposed RSMM model. The experimental results show that our method has outperformed the state-of-the-art matrix classifiers.

The rest of this paper is organized as follows. In Section II, we briefly give notations and preliminaries that run throughout the paper. In Section III, we present the proposed RSMM and the learning algorithm based on ADMM in detail. In Section IV, we conduct extensive experimental analysis to justify our method. Finally, we conclude our work in Section V.

II. Notations and Preliminaries

We use bold uppercase letters to denote matrices (i.e., $\mathbf{X}$), bold lowercase letters to denote column vectors (i.e., $\mathbf{x}$) and non-bold letters to denote scalar variables (i.e., x). For a matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$, we let $rank(\mathbf{X}) = r$ denote the rank of $\mathbf{X}$ where $r \leq min(d_1, d_2)$. The singular value decomposition (SVD) of $\mathbf{X}$ is denoted as $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2, \cdots, \sigma_r)$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$; $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$ are matrices with orthonormal columns; $\mathbf{V}^T \in \mathbb{R}^{r \times d_2}$ denotes matrix transpose of $\mathbf{V}$. The Frobenius norm is represented as $||\mathbf{X}||_F = \sqrt{\sum_{i=1}^{d_1} \sum_{j=1}^{d_2} X_{ij}^2}$; the ℓ_1 norm is $||\mathbf{X}||_1 = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} |X_{ij}|$; the nuclear (trace) norm is $||\mathbf{X}||_* = \sum_{k=1}^{r} \sigma_k$; and the inner product between $\mathbf{A}$, $\mathbf{B} \in \mathbb{R}^{d_1 \times d_2}$ is $tr(\mathbf{A}^T\mathbf{B}) = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} A_{ij} B_{ij}$.

Definition 1: For any positive scalar $\tau > 0$, the singular value thresholding (SVT) operator is well defined as

$$\mathcal{D}_\tau(\mathbf{X}) = \mathbf{U}\mathcal{S}_\tau(\mathbf{\Sigma})\mathbf{V}^T, \tag{1}$$

where $\mathcal{S}_\tau(\mathbf{\Sigma}) = diag(\{\sigma_i - \tau\}_+)$ and $\{u\}_+ = max(u, 0)$.

The operator $\mathcal{D}_\tau(\mathbf{X})$ shrinks the singular values of $\mathbf{X}$ with a soft-thresholding rule. It is also called the singular value shrinkage operator in the literatures [27], [34].

We further introduce the proximal operators, which have served as an important component in proximal algorithms for solving convex but non-smooth optimization problems [35].

Definition 2: For any positive scalar λ, the proximal operator $\mathbf{prox}_f : \mathbb{R}^{d_1 \times d_2} \rightarrow \mathbb{R}^{d_1 \times d_2}$ of a convex function $f : \mathbb{R}^{d_1 \times d_2} \rightarrow \mathbb{R}$ at $\mathbf{X}$ is defined by

$$prox_f(\mathbf{X}) = \arg\min_{\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}} \frac{1}{2}||\mathbf{Z} - \mathbf{X}||_F^2 + \lambda f(\mathbf{Z}). \tag{2}$$

Specifically, if $f = ||\mathbf{Z}||_*$, the proximal operator for the nuclear norm is given by

$$prox_{\lambda||\cdot||_*} = \mathcal{D}_\lambda(\mathbf{X}). \tag{3}$$

If $f = ||\mathbf{Z}||_1$, the proximal operator for the ℓ_1 norm is given by

$$prox_{\lambda||\cdot||_1}(\mathbf{X}) = sgn(\mathbf{X}) \circ \max\{|\mathbf{X}| - \lambda, 0\}, \tag{4}$$

where $sgn(\cdot)$ denotes sign function and $\circ$ is an element wise product operator.

III. Method

In this section, we first introduce the proposed RSMM for single trial EEG classification, which simultaneously captures correlation within each EEG matrix and removes sparse outliers during the training procedure. We further derive the learning algorithm for RSMM based on the ADMM framework.

A. Robust Support Matrix Machine

Given a set of training data $\{\mathbf{X}_i, y_i\}_{i=1}^n$, $\mathbf{X}_i \in \mathbb{R}^{d_1 \times d_2}$ denotes the i_{th} trial of input EEG signals corrupted by sparse outliers, where d_1 represents the number of channels and d_2 is the number of time points; $y_i \in \{1, -1\}$ denotes its corresponding true label. We are motivated to propose a novel matrix classifier to predict the label of a new observation with high accuracy.

Single trial EEG classification is very challenging due to poor characteristics of EEG signals contaminated by various sources of noises. Though preprocessing techniques are employed to remove unrelated artifacts and noises of known distribution, it can not clearly eliminate all non-standard noises, such as gloss or spiky noises. Motivated by the fact that clean EEG signals always contain strong correlated spatial-temporal structural information [31], we assume that each EEG sample can be decomposed additively as $\mathbf{X}_i = \mathbf{L}_i + \mathbf{S}_i$, where $\mathbf{L}_i \in \mathbb{R}^{d_1 \times d_2}$ denotes low-rank clean signal and $\mathbf{S}_i \in \mathbb{R}^{d_1 \times d_2}$ is sparse intra-sample outliers. It is promising to facilitate the classifier training using the clean signals $\{\mathbf{L}_i\}_{i=1}^n$ instead of the contaminated EEG matrices $\{\mathbf{X}_i\}_{i=1}^n$. For classifier training, it is well known that hinge loss used in SVM enjoys "max-margin" principle as well as robustness and sparseness properties. Thus, we adopt hinge loss in matrix form for classifier modeling. To account for the correlation shared among all training data and sparse outliers but of large amplitudes, we incorporate a low rank constraint into the regression matrix as a regularization. Based on the advanced rank minimization, we employ nuclear norm for low rank and ℓ_1 norm for sparseness approximation, leading to a novel optimization problem for the proposed RSMM as follows:

$$
\begin{aligned}
\min_{\mathbf{W}, b, \{\mathbf{L}_i, \mathbf{S}_i\}_{i=1}^n} & \sum_{i=1}^{n} \{1 - y_i[tr(\mathbf{W}^T \mathbf{L}_i) + b]\}_+ + \lambda_1 ||\mathbf{W}||_* \\
& + \sum_{i=1}^{n} (\lambda_2 ||\mathbf{L}_i||_* + \lambda_3 ||\mathbf{S}_i||_1) \\
s.t. \quad & \forall\ i, \ \mathbf{X}_i = \mathbf{L}_i + \mathbf{S}_i. \qquad (5)
\end{aligned}
$$

Here, $\mathbf{W} \in \mathbb{R}^{d_1 \times d_2}$ denotes the regression matrix normal to the class hyperplane, b denotes the offset of the hyperplane from the origin along $\mathbf{W}$. λ_1, λ_2 and λ_3 are positive scalars penalizing nuclear norm of $\mathbf{W}$ and $\mathbf{L}_i$, and ℓ_1 norm of $\mathbf{S}_i$.

From a computational perspective, the hinge loss in matrix form $\{1 - y_i[tr(\mathbf{W}^T \mathbf{L}_i) + b]\}_+$ is equivalent to conventional hinge loss used in SVM when converting $\mathbf{W}$ and $\mathbf{L}_i$ to vectors, since $tr(\mathbf{W}^T \mathbf{L}_i) = vec(\mathbf{W})^T vec(\mathbf{L}_i)$. However, the nuclear norm can not be equivalently reformulated into a vector form. Thus by leveraging the low-rank approximation of $\mathbf{W}$ and the inner product $tr(\mathbf{W}^T \mathbf{L}_i)$, RSMM is able to capture the global correlation within the clean input EEG matrices.

Note that matrix decomposition has been studied in the literatures, especially for the problem of matrix completion [36], [37]. Among them, one classic work is robust principle component analysis (RPCA) [38], which decomposes the highly corrupted measurement as the summation of a low-rank matrix and a sparse one in an unsupervised manner. RPCA has been applied in many application areas such as motion detection [39], image alignment [40], etc. However, this kind of matrix decomposition has neither been explored in contaminated EEG signals nor in single trial EEG classification problem. If we skip the matrix decomposition step to recover the corrupted signals ($\lambda_2 = \lambda_3 = 0$), the proposed RSMM would degenerate to the bilinear SVM (BSVM) [41]. In addition, our method embeds the decomposition of contaminated EEG signals in the procedure of computing the regression matrix interactively, namely we recover the clean signals and remove the sparse outliers in a supervised manner (see the red line arrows in Fig. 2). Therefore, RSMM is by no means a simple combination of RPCA to remove sparse outliers and BSVM to train a matrix classifier based on recovered data. Though [42] can address intra-sample outliers for logistic regression with a similar matrix decomposition strategy, this approach still requires to reshape each data matrix into a vector for eliminating noise and outliers, resulting in a significant difference from our method.

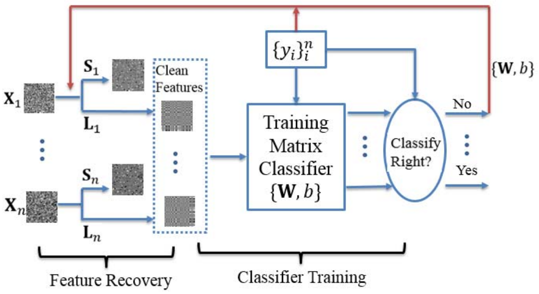


Fig. 2. The training framework of the proposed RSMM. Blue line arrows indicate feature recovery and classifier training procedures, while red line arrows indicates feedback of the classifier for feature recovery if the data is misclassified.

Another related work - matrix factorization has been applied successfully to EEG classification [43]. Among these works, they first concatenate multiple trials of EEG signals into a matrix-form data $\mathbf{X} \in \mathbb{R}^{m \times d}$ and assume that $\mathbf{X}$ can be decomposed into the product of two low-rank factors in the form of $\mathbf{X} \approx \mathbf{AS}$, where $\mathbf{A} \in \mathbb{R}^{m \times r}$ contains bases in its columns and $\mathbf{S} \in \mathbb{R}^{r \times d}$ is the associated encoding variables matrix. Lee *et al.* [44] employed nonnegative matrix factorization (NMF) to $\mathbf{X}$ to extract discriminative features. Lee and Choi [45] also proposed a group NMF method to explore the common bases reflecting intra-subject and inter-subject variations for multitask learning. Inspired by [45], Seichepine *et al.* [46] studied a soft co-factorization scheme accounting for possible local discrepancies across modalities or channels. Except for feature selection with NMF, Damon *et al.* [47] further applied NMF in a Gaussian source framework for single-channel EEG artifact removal. However, all these methods required multiple trials from a subject or multiple subjects to construct $\mathbf{X}$ for blind source separation. In addition, they only consider inter-sample information, while ignore intra-sample outliers. Once the basis matrix and encoding matrix are extracted, classifiers are trained independently based on the vector-form features within the encoding matrix, which is definitely different from RSMM for matrix-form data.

B. Solver

There are four variables $\{\mathbf{W}, b, \mathbf{L}, \mathbf{S}\}$ involved in the proposed RSMM, and the joint minimization of the resulting optimization problem is rather difficult since hinge loss, nuclear norm and ℓ_1 norm in Eq. (5) are neither smooth nor differentiable. Thus we develop an iterative learning algorithm based on alternating direction method of multipliers (ADMM) framework [35], to decouple the joint optimization problem into several easier subproblems. In each iteration, we alternatively optimize one variable with the proximal operators, which is firmly non-expensive computationally.

To decouple the hinge loss and nuclear norm with respect to $\mathbf{W}$ in RSMM, we first introduce an auxiliary variable $\mathbf{Z}$ and equivalently rewrite Eq. (5) as

$$\begin{aligned}\min_{\mathbf{W},\mathbf{Z},b,\{\mathbf{L}_i,\mathbf{S}_i\}_{i=1}^n} & \sum_{i=1}^n h(\mathbf{W},b,\mathbf{L}_i)+\lambda_1||\mathbf{Z}||_* \\ & +\sum_{i=1}^n(\lambda_2||\mathbf{L}_i||_*+\lambda_3||\mathbf{S}_i||_1), \\ s.t.\quad & \forall\ i,\ \mathbf{X}_i=\mathbf{L}_i+\mathbf{S}_i,\ \ \mathbf{W}=\mathbf{Z}, \end{aligned} \tag{6}$$

where $h(\mathbf{W},b,\mathbf{L}_i)=\{1-y_i[tr(\mathbf{W}^T\mathbf{L}_i)+b]\}_+$.

The above constrained problem can be efficiently optimized using Augmented Lagrangian Multiplier (ALM) algorithm. The key of ALM method is to search for a saddle point of the augmented Lagrangian function instead of solving the original constrained optimization problem. For problem in Eq. (6), the augmented Lagrangian function is given by

$$\begin{aligned}&\mathcal{L}(\mathbf{W},\mathbf{Z},b,\mathbf{L}_i,\mathbf{S}_i,\mathbf{V},\mathbf{M}_i) \\ &\quad=\sum_{i=1}^n h(\mathbf{W},b,\mathbf{L}_i)+\lambda_1||\mathbf{Z}||_*+tr[\mathbf{V}^T(\mathbf{Z}-\mathbf{W})] \\ &\qquad+\frac{\mu_1}{2}||\mathbf{Z}-\mathbf{W}||_F^2+\sum_{i=1}^n\{\lambda_2||\mathbf{L}_i||_*+\lambda_3||\mathbf{S}_i||_1 \\ &\qquad+tr[\mathbf{M}_i^T(\mathbf{X}_i-\mathbf{L}_i-\mathbf{S}_i)]+\frac{\mu_2}{2}||\mathbf{X}_i-\mathbf{L}_i-\mathbf{S}_i||_F^2\},\end{aligned} \tag{7}$$

where $\mathbf{V}$, $\{\mathbf{M}_i\}_{i=1}^n\in\mathbb{R}^{d_1\times d_2}$ are Lagrange multipliers, and μ_1 and μ_2 are the positive penalty parameters. For appropriate choice of $\mathbf{V}$ and $\{\mathbf{M}_i\}_{i=1}^n$, and sufficiently large constants μ_1 and μ_2, the augmented Lagrangian function in Eq. (7) has the same minimizer as the original problem in Eq. (5). For the ease of exposition, ALM algorithm iteratively estimates the optimal solutions by minimizing Eq. (7) in two steps, and subsequently updates Lagrange multipliers accordingly with

$$\begin{aligned}(\mathbf{W}^{(k)},\mathbf{Z}^{(k)},b^{(k)})&=\arg\min_{\mathbf{W},\mathbf{Z},b}\mathcal{L}(\mathbf{W},\mathbf{Z},b,\mathbf{L}_i^{(k-1)},\mathbf{V}^{(k-1)}), \\ (\mathbf{L}_i^{(k)},\mathbf{S}_i^{(k)})&=\arg\min_{\mathbf{L}_i,\mathbf{S}_i}\mathcal{L}(\mathbf{W}^{(k)},b^{(k)},\mathbf{L}_i,\mathbf{S}_i,\mathbf{M}_i^{(k-1)}), \\ \mathbf{V}^{(k)}&=\mathbf{V}^{(k-1)}+\mu_1(\mathbf{Z}^{(k)}-\mathbf{W}^{(k)}), \\ \mathbf{M}_i^{(k)}&=\mathbf{M}_i^{(k-1)}+\mu_2(\mathbf{X}_i-\mathbf{L}_i^{(k)}-\mathbf{S}_i^{(k)}),\end{aligned} \tag{8}$$

where $k\in\mathbb{N}$ represents the index of iteration. In this way, the first step in Eq. (8) denoted as *"classifier training"*, estimates the model parameters of the matrix classifier based on clean EEG signals; while the second step further adjusts the removal procedure of intra-sample outliers according to the newly estimated model parameters, thus we call it *"feature recovery"* step. We are motivated to solve them one by one in the following.

Algorithm 1: The Learning Algorithm for RSMM

```
Input  : Training data $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$, input coefficients
         $\lambda_1$, $\lambda_2$ and $\lambda_3$
Output: $\mathbf{W}$, $b$, $\{\mathbf{L}_i, \mathbf{S}_i\}_{i=1}^n$
Initialize: $\mathbf{W}^{(0)} = \mathbf{0}$, $\mathbf{L}_i^{(0)} = \mathbf{X}_i$, $\mathbf{S}_i^{(0)} = \mathbf{X}_i - \mathbf{L}_i^{(0)}$,
  $\mathbf{V}^{(0)} = \mathbf{0}$, $\mathbf{M}_i^{(0)} = \mathbf{0}$, hyperparameters $\rho = 1.1$,
  $\mu_1^{(0)} = \mu_2^{(0)} = 1$, k=1
while not converge do
    Update $\mathbf{Z}^{(k)}$ with Eq. (10)
    Update $\mathbf{W}^{(k)}$ and $b$ with Eq. (12)
    Update $\{\mathbf{L}_i^{(k)}\}_{i=1}^n$ with Eq. (16)
    Update $\{\mathbf{S}_i^{(k)}\}_{i=1}^n$ with Eq. (17)
    Update Lagrange multipliers and coefficients with
    Eq. (18)
    $k = k + 1$
end
return $\mathbf{W}^{(k)}$,$b^{(k)}$,$\{\mathbf{L}_i^{(k)}, \mathbf{S}_i^{(k)}\}_{i=1}^n$
```

1) Classifier Training: The first step in Eq. (8) is difficult to solve directly due to the coupled terms. A common strategy is to minimize $\mathcal{L}$ against the unknowns $\mathbf{W}$, $\mathbf{Z}$, b one at a time. To update $\mathbf{Z}$, we have the following theorem.

Theorem 1: For any positive scalars λ_1 and μ_1, let $g(\mathbf{Z})$ denote $\{\lambda_1||\mathbf{Z}||_*+tr(\mathbf{V}^T\mathbf{Z})+\frac{\mu_1}{2}||\mathbf{Z}-\mathbf{W}||_F^2\}$ and define

$$\mathbf{Z}'=\frac{1}{\mu_1}\mathcal{D}_{\lambda_1}(\mu_1\mathbf{W}-\mathbf{V}). \tag{9}$$

Then we have $\mathbf{0}\in\partial g(\mathbf{Z}')$.

To minimize $\mathcal{L}$ w.r.t $\mathbf{Z}$ is equivalent to minimize $g(\mathbf{Z})$. Since $g(\mathbf{Z})$ is convex w.r.t. $\mathbf{Z}$, based on Theorem 1, we derive one of the optimal solutions for $g(\mathbf{Z})$ with

$$\hat{\mathbf{Z}}=\frac{1}{\mu_1}\mathcal{D}_{\lambda_1}(\mu_1\mathbf{W}-\mathbf{V}). \tag{10}$$

To derive the update of $\mathbf{W}$ and b, we have the following theorem.

Theorem 2: For any positive μ_1, one of the solution of

$$\arg\min_{\mathbf{W},b}\sum_{i=1}^n h(\mathbf{W},b,\mathbf{L}_i)-tr(\mathbf{V}^T\mathbf{W})+\frac{\mu_1}{2}||\mathbf{Z}-\mathbf{W}||_F^2 \tag{11}$$

is

$$\begin{aligned}\hat{\mathbf{W}}&=\frac{1}{\mu_1}(\mu_1\mathbf{Z}+\mathbf{V}+\sum_{i=1}^n\alpha_i^*y_i\mathbf{L}_i), \\ \hat{b}&=\frac{1}{n}\sum_{i=1}^n(y_i-tr(\hat{\mathbf{W}}^T\mathbf{L}_i)),\end{aligned} \tag{12}$$

where $\boldsymbol{\alpha}^*\in\mathbb{R}^n$ is the solution of the following constrained quadratic programming problem:

$$\begin{aligned}&\arg\max_{\boldsymbol{\alpha}}-\frac{1}{2}\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha}+\mathbf{q}^T\boldsymbol{\alpha}, \\ &s.t.\mathbf{0}\le\boldsymbol{\alpha}\le\mathbf{1},\quad\sum_{i=1}^n\alpha_iy_i=0.\end{aligned} \tag{13}$$

Here the elements in $\mathbf{K} \in \mathbb{R}^{n\times n}$ *and* $\mathbf{q} \in \mathbb{R}^{n}$ *are defined as*

$$\begin{aligned} K_{ij} &= \frac{1}{\mu_1} y_i y_j tr(\mathbf{L}_i^T \mathbf{L}_j), \\ q_i &= 1 - \frac{1}{\mu_1} y_i tr[(\mu_1 \mathbf{Z} + \mathbf{V})^T \mathbf{L}_i]. \end{aligned} \tag{14}$$

From Eqs. (12), (13) and (14), we find that the estimation of $\mathbf{W}$ and b is based on the recovered EEG matrices $\{\mathbf{L}_i\}_{i=1}^n$ instead of the contaminated signals $\{\mathbf{X}_i\}_{i=1}^n$. Thus the model parameters of RSMM are robust against intra-sample outliers.

2) Feature Recovery: Similarly, we minimize the function $\mathcal{L}$ against $\mathbf{L}_i$ and $\mathbf{S}_i$ one by one. To derive the update of $\mathbf{L}_i$ and $\mathbf{S}_i$, we obtain the following theorems.

Theorem 3: One of the solution for

$$\begin{aligned} \arg\min_{\mathbf{L}_i} \; & h(\mathbf{W}, b, \mathbf{L}_i) + \lambda_2 ||\mathbf{L}_i||_* - tr(\mathbf{M}_i^T \mathbf{L}_i) \\ & + \frac{\mu_2}{2} ||\mathbf{X}_i - \mathbf{L}_i - \mathbf{S}_i||_F^2 \end{aligned} \tag{15}$$

is

$$\hat{\mathbf{L}}_i = \frac{1}{\mu_2} \mathcal{D}_{\lambda_2}(\mathbb{I}_{y_i(tr(\mathbf{W}^T \mathbf{L}_i)+b)<1} y_i \mathbf{W} + \mu_2(\mathbf{X}_i - \mathbf{S}_i) + \mathbf{M}_i) \tag{16}$$

where $\mathbb{I}_{y_i(tr(\mathbf{W}^T \mathbf{L}_i)+b)<1}$ *is a flag variable and equals to* 1 *if* $y_i(tr(\mathbf{W}^T \mathbf{L}_i) + b) < 1$ *or otherwise equals to* 0.

When $y_i(tr(\mathbf{W}^T \mathbf{L}_i) + b) < 1$, it can be regarded that the observation $\mathbf{L}_i$ is misclassified with the current model parameters $\mathbf{W}$ and b. In this regard, the clean signals $\{\mathbf{L}_i\}_{i=1}^n$ are gradually recovered under the influence of model parameters in a supervised manner.

Theorem 4: One of the solution of the following problem

$$\arg\min_{\mathbf{S}_i} \;\; \lambda_3 ||\mathbf{S}_i||_1 - tr(\mathbf{M}_i^T \mathbf{S}_i) + \tfrac{\mu_2}{2} ||\mathbf{X}_i - \mathbf{L}_i - \mathbf{S}_i||_F^2$$

is

$$\hat{\mathbf{S}}_i = \frac{1}{\mu_2} sgn(\mathbf{Y}_i) \circ \max(|\mathbf{Y}_i| - \lambda_3, 0), \tag{17}$$

where $\mathbf{Y}_i = \mu_2(\mathbf{X}_i - \mathbf{L}_i) + \mathbf{M}_i$.

The proofs of all above theorems are attached in appendices.

Then the Lagrange multipliers and coefficients can be updated with

$$\begin{aligned} \mathbf{V}^{(k)} &= \mathbf{V}^{(k-1)} + \mu_1^{(k-1)}(\mathbf{Z}^{(k)} - \mathbf{W}^{(k)}), \\ \mathbf{M}_i^{(k)} &= \mathbf{M}_i^{(k-1)} + \mu_2^{(k-1)}(\mathbf{X}_i - \mathbf{L}_i^{(k)} - \mathbf{S}_i^{(k)}), \\ \mu_1^{(k)} &= \rho \mu_1^{(k-1)}, \\ \mu_2^{(k)} &= \rho \mu_2^{(k-1)}. \end{aligned} \tag{18}$$

Here $\mathbf{V}^{(k)}$ and $\mathbf{M}_i^{(k)}$ are sequences of estimates of the Lagrange multipliers of the constraints $\mathbf{Z} = \mathbf{W}$ and $\mathbf{X}_i = \mathbf{L}_i + \mathbf{S}_i$, respectively. $\mu_1^{(k)}$ and $\mu_2^{(k)}$ are sequences of monotonically increasing positive scalar parameters bounded away from 0. $\rho \geq 1$ is a positive scalar that accelerates the convergence of Lagrange multipliers and thus the whole algorithm. We initialize the values of μ_1, μ_2 and ρ similar to [40]. Our solver is summarized in Algorithm 1.

3) Computational Cost: We also analyze the time complexity of Algorithm 1. Given n training samples of size $d_1 \times d_2$ each, line 2 and line 4 in Algorithm 1 compute the eigen decomposition for $\mathbf{Z}$ and $\{\mathbf{L}_i\}_{i=1}^n$, which take time $O(\min(d_1^2 d_2, d_1 d_2^2))$ and $O(\min(n d_1^2 d_2, n d_1 d_2^2))$, respectively. Line 3 computes the quadratic programming with respect to W, which takes time $O(n^2 d_1 d_2)$. Line 5 calculates n dot-product and each dot-product takes $O(d_1 d_2)$. In practice, either the dimension d_1 and d_2 is low, *e.g.*, high-dimensional EEG features are extracted at a few channels ($d_1 \ll d_2$). Thus, the main time cost of our solver is dominated by the quadratic programming in Line 3. In this regard, the time complexity of Algorithm 1 is $O(n^2 d_1 d_2) \times K$, where K is the iteration number.

C. RSMM for EEG Testing Data

Since the EEG data are contaminated by noises and outliers, we also need to take this issue into consideration when using trained RSMM for EEG data classification. Given a set of EEG data $\{\mathbf{X}_{te}\}_{te=1}^{n_{te}}$ without labels for testing, we also need to decompose each of them into a low-rank component and a sparse noise part by optimizing the following problem

$$\begin{aligned} \min_{\{\mathbf{L}_{te}, \mathbf{S}_{te}\}_{te=1}^{n_{te}}} \;& ||\mathbf{L}_{te}||_* + \gamma \, ||\mathbf{S}_{te}||_1, \\ s.t. \;\; & \mathbf{X}_{te} = \mathbf{L}_{te} + \mathbf{S}_{te}, \end{aligned} \tag{19}$$

where $\mathbf{L}_{te}$ denotes the low rank component, $\mathbf{S}_{te}$ is the sparse matrix for outliers, and γ is a positive scalar adding penalty for sparse noise. Note that, Eq. (19) is equivalent to RPCA and can be solved efficiently [35]. For consistency in training and testing phase, γ is determined by $\gamma = \frac{\lambda_3}{\lambda_2}$. After the matrix decomposition, only the clean part $\mathbf{L}_{te}$ is used for classification. Once the model parameters $(\mathbf{W}, b)$ is learned, we can predict the label for $\mathbf{X}_{te}$ with

$$y_{te} = sgn(tr(\mathbf{W}^T \mathbf{L}_{te}) + b). \tag{20}$$

IV. Experiment

In this section, we apply the proposed RSMM to single trial EEG data classification on three public EEG datasets: Dataset IVa of BCI Competition III, Dataset IIb and IIa of BCI Competition IV. To demonstrate the advantages of RSMM which jointly consider feature recovery and classifier training, we set the simple combination of RPCA as a preprocessing step and BSVM [41] as a matrix classifier (RPCA+BSVM) to be a benchmark. The method RPCA+BSVM adopts the same decomposition technique (RPCA) for both training and testing data, then trains classifiers independently based on the decomposed matrix-form data. For comparison, we further compare the performance of RSMM with the benchmark vector classifier SVM [48], and three state-of-the-art matrix classifiers, namely, regularized-GLM (RGLM) [29], bilinear SVM (BSVM) [41] and support matrix machine (SMM) [30].

A. EEG Data Description

The *Dataset IVa of BCI Competition III (Exp. 1)* [49] contains 118-channel EEG signals recorded from five subjects

TABLE I
Summary of Three Datasets

EEG Datesets	#subset	#positive	#negative	dimension
BCI III Dataset - IVa	5	140	140	120 × 300
BCI IV Dataset - IIb	9	360 ± 20	360 ± 20	24 × 150
BCI IV Dataset - IIa	54	72	72	240 × 150

TABLE II
The Testing Accuracy of Different Methods in Exp. 1

Subject	RPCA+BSVM	SVM	RGLM	BSVM	SMM	Ours
aa	0.7321	0.7321	0.7053	0.7500	0.7411	**0.7589**
al	0.9821	0.9821	0.9821	**1**	**1**	**1**
av	0.6837	0.6633	0.6648	0.6837	0.6734	**0.6990**
aw	0.7054	0.7054	0.7143	0.7175	0.7366	**0.8259**
ay	0.6984	0.6627	0.7024	0.6984	0.6944	**0.7579**
avg	0.7603	0.7491	0.7538	0.7699	0.7691	**0.8083**

when performing right-hand or foot motor imagery (MI). The sampling rate is $100Hz$. There are 280 trials for each subject, among which 168(112), 224(56), 84(196), 56(224), 28(252) trials are selected as training (testing) data respectively for subject aa, al, av, aw and ay.

The *Dataset IIb of BCI Competition IV (Exp. 2)* [50] records 3 bipolar-channel EEG signals from nine subjects (denoted as $B01 - B09$) involving left hand or right hand MIs. The sampling rate of the signals is $250Hz$. There are about 400 trials used for training and 320 trials for testing for each subject.

The *Dataset IIa of BCI Competition IV (Exp. 3)* [51] consists of 22-channel EEG signals from nine subjects (denoted as $A01 - A09$) involving four-class MIs related to left hand (L), right hand (R), feet (F) and tongue (T). The sampling rate of the signals is $250Hz$. Both training and testing sets contain 72 trials per motor imagery task for each subject. To evaluate the single trial binary classification performance, we decompose the four-class data and generate $C_4^2 = 6$ binary subsets for each subject, thereby we obtain a total of $6 \times 9 = 54$ binary subsets.

In this paper, we consider the time interval of $[0.5, 3.5]s$ after visual cue in each trial for dataset 1, and $[1, 4]s$ for dataset 2 and 3. Following the same settings in [52], we first remove the unrelated sensorimotor rhythms and artifacts with Chebyshev Type II filter. Then we perform spatial filter for detecting Event-Related Desynchronization/Synchronization (ERD/ERS) using the CSP algorithm. We also downsample the signals to $50Hz$ for dataset 2 and 3 to reduce the computational cost [31]. To extract feature in matrix form, we use time domain parameter (TDP) algorithm [53] [1] due to its robust performance [54]. For these three experiments, a total of $5 + 9 + 54 = 68$ binary EEG subsets are obtained. The main information of these datasets are summarized in Table I.

The proposed RSMM and competitive algorithms are run on each subset data for classification. The ratio of training and testing trials (n/n_{te}) is determined by the competition setup.[2] The performance is evaluated by calculating the ratio of trials correctly classified to the total number of testing trials, and higher ratio denotes the better classification performance.

B. Experiment Settings

We further introduce the settings for our experiments. There are several parameters needed to be determined in our proposed RSMM, including λ_1, λ_2, λ_3 and γ. To build the good model, here we employ tenfold cross validation to select optimal parameters. We first randomly partition the training trials into ten equal sized subsets. Of the ten subsets, nine subsets are used to remove intra-sample outliers and train classification models, and the remaining subset is used as validation data to evaluate the trained models. The procedure is repeated ten times such that each of the ten subsets is used exactly once as the validation data. For each pair of parameters, the specific values in the candidate sets that achieve the highest average classification accuracy across the 10 repetitions, is determined as optimal values. Specifically, the candidate sets are defined as follows: $\lambda_1 \in \{0.01, 0.1, 1, 10, 100\}$, $\lambda_2 \in \{0.001, 0.01, 0.1, 1, 5\}$ and $\lambda_3 \in \{0.0001, 0.001, 0.01, 0.1, 1\}$, and γ is determined by $\gamma = \frac{\lambda_3}{\lambda_2}$ for consistency in training and testing phases. For fair comparison, we use the online codes and select the parameters via cross validation for other algorithms with $\epsilon \in \{0.001, 0.005, 0.01, 0.05, \cdots, 0.5, 1\}$ for SVM, $C \in \{0.01, 0.1, 1, 10\}$, $t \in \{100, 300, 500\}$ for BSVM, $\alpha \in \{1, 2, 5, 10, 20, 50, 100\}$ for RGLM and $C \in \{0.01, 0.1, 1, 10, 100\}$, $\tau \in \{0.001, 0.01, 0.1, 1, 10\}$ for SMM. Note that for SVM, we concatenate each feature matrix into a vector before feeding it into the classifier.

C. Experimental Results Analysis

We first show the classification accuracies of different algorithms in each experiment, and further analyze the compared results and study the convergence process of our method.

1) Result on Exp. 1: We first evaluate our method on the Dataset IVa of BCI Competition III. For all compared algorithms, the results are listed in Table II. It shows that all matrix classifiers yield better average performance than the benchmark SVM for feature vectors, and the proposed RSMM outperforms other competitive methods. Specifically, RSMM greatly enhances the classification accuracy for subject aw and ay, which have quite small ratio of training sets (20% for subject aw and 10% for ay). This is because our method can remove the sparse outliers and leverage the low rank structure embedded in each data matrix for model training, leading to avoidance of over fitting even with small training size.

As an example, we visualize $\mathbf{X}_1$, $\mathbf{L}_1$, $\mathbf{S}_1$ and the regression matrix $\mathbf{W}$ for subject 'al' in Fig. 3. $\mathbf{W}$ is of the same size of $\mathbb{R}^{120\times300}$ as the input matrices. Fig. 3 shows that the input matrix $\mathbf{X}_1$ is of full rank. Both $\mathbf{W}$ and $\mathbf{L}_1$ are low rank due to the regularization of nuclear norm, and $\mathbf{S}_1$ is sparse due to the ℓ_1 norm.

2) Result on Exp. 2: We also evaluate our RSMM and other algorithms on the Dataset IIb of BCI Competition IV and the

[1] TDP is defined as a time-varying log-power of the first p_{th} derivatives of the signals. Here we empirically set $p = 5$. The code for TDP is available in BioSig-toolbox at http://biosig.sourceforge.net/download.html.

[2] http://www.bbci.de/competition/iii/ and http://www.bbci.de/competition/iv/

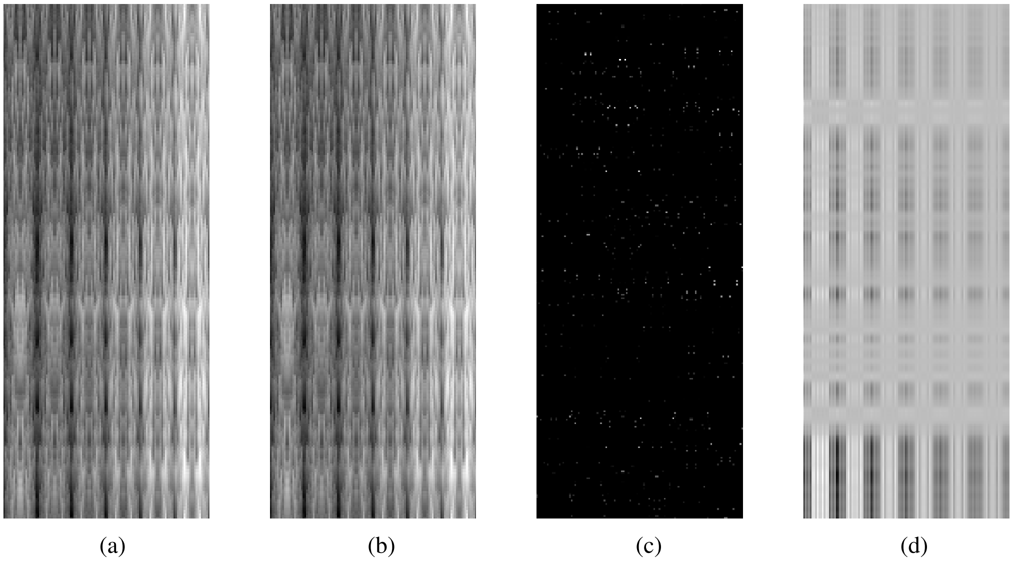


Fig. 3. Visualization of (a) the input matrix X_1 with $rank = 120$, (b) the latent clean feature L_1 with $rank = 60$, (c) the sparse outlier matrix S_1 and (d) the regression matrix W with $rank = 54$.

TABLE III
THE TESTING ACCURACY OF DIFFERENT METHODS IN EXP. 2

Subject	RPCA+BSVM	SVM	RGLM	BSVM	SMM	Ours
B01	0.6844	0.6750	0.6875	0.6844	0.6781	**0.7250**
B02	0.5143	0.5000	0.5107	0.5107	0.5179	**0.5643**
B03	0.4969	0.5219	0.525	0.5313	0.5344	**0.5563**
B04	0.9281	0.9063	0.9187	0.9581	0.9331	**0.9719**
B05	0.7844	0.8000	0.8156	0.8375	0.8281	**0.8844**
B06	0.7156	0.7313	0.7594	0.7375	0.7469	**0.7875**
B07	0.6938	0.6938	0.7469	0.7063	0.7219	**0.7750**
B08	0.8187	0.8219	0.8656	0.8625	0.8438	**0.9188**
B09	0.7531	0.7438	0.7656	0.7562	0.7562	**0.8344**
avg	0.7099	0.7104	0.7328	0.7283	0.7289	**0.7797**

accuracies on testing sets are shown in Table III. It can be observed that all the matrix classifiers except RPCA+BSVM, improve the performance compared with SVM on all the subsets. This implies that the structural information considered in the matrix classifier is helpful to EEG classification. Particularly, our method achieves better performance with large margin compared with other matrix classifiers. This is because our method can leverage the structural information for feature representation and reduce the adverse influence of outliers simultaneously for the classifier training. The RPCA+BSVM achieves mediocre results, though it is also a matrix classifier with RPCA as a preprocessing step. This may be because in the preprocessing step, the blind decomposition of RPCA without guidance from the labels could result in discarding part of structural information or remaining sparse outliers, which degrades the performance of the BSVM classifier.

3) Result on Exp. 3: We further evaluate the performance of all the algorithms on the Dataset IIa of BCI Competition IV, with results shown in Fig. 4. It can be observed that all the matrix classifiers beat SVM in most cases, which again explains the usefulness of matrix classifiers for EEG classification. As expected, the proposed RSMM yields the highest testing accuracy on most subsets consistently. This is because the sparse noise existing in each data matrix may contaminate the structural information, resulting in poor performance of matrix classifiers. While our method removes the sparse noise and leverages the low rank representation in a supervised manner during training phrase, which thus enhances the performance for EEG classification.

As an example, Fig. 5 shows the convergence process of RSMM on subset B05. It shows that our method converges to the an optimal value in dozens of iterations. Similar results also occur when using RSMM on other subsets.

D. Discussion

From the experimental results on the three public EEG datasets listed above, the following questions arise: 1) How statistically significant is the performance improvement of the proposed RSMM? 2) How does the classification performance of RSMM change according to the free parameters? and 3) Does it have obvious influence on the classification performance with different ratios of training and testing data?

1) Statistical Significance Testing: To investigate whether the improvement in classification performances of the proposed RSMM is at a significant level on each dataset, we conduct a pairwisely two-tailed t-test to compare the classification accuracies of RSMM with those of state-of-the-art classifiers and present the results in Table IV. In this table, we highlight the p-values in boldface if they are less than 0.05. It is clear that most of the cases show that the null hypothesis can be rejected with 95% confidence level. Thus it can be concluded that our method significantly outperforms the competing methods.

2) Influence of Free Parameters: We also consider the influence of free parameters on the performance of RSMM. There are three free parameters λ_1, λ_2 and λ_3, where λ_1 is the weight for nuclear norm of regression parameter, λ_2 is the weight for nuclear norm of feature matrices, and λ_3 is the penalty for outliers in the feature matrices. It is observed that, when

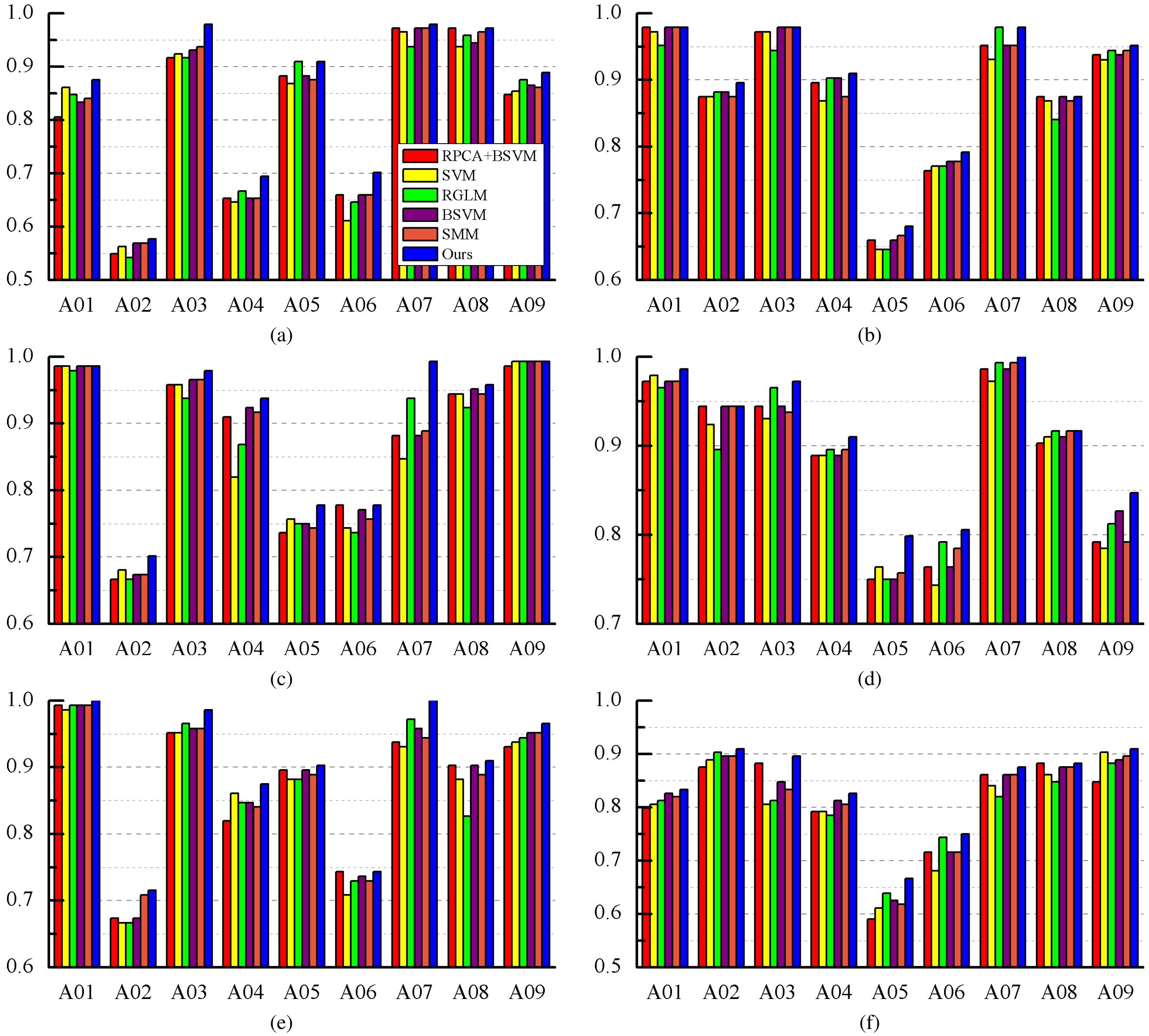


Fig. 4. The classification performance of different methods in Exp. 3. The Y axis represents the testing accuracy. (a) Left hand vs right hand. (b) Left hand vs feet. (c) Left hand vs tongue. (d) Right hand vs feet. (e) Right hand vs tongue. (f) Feet vs tongue.

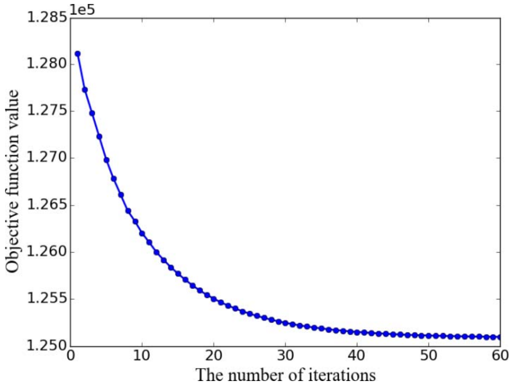


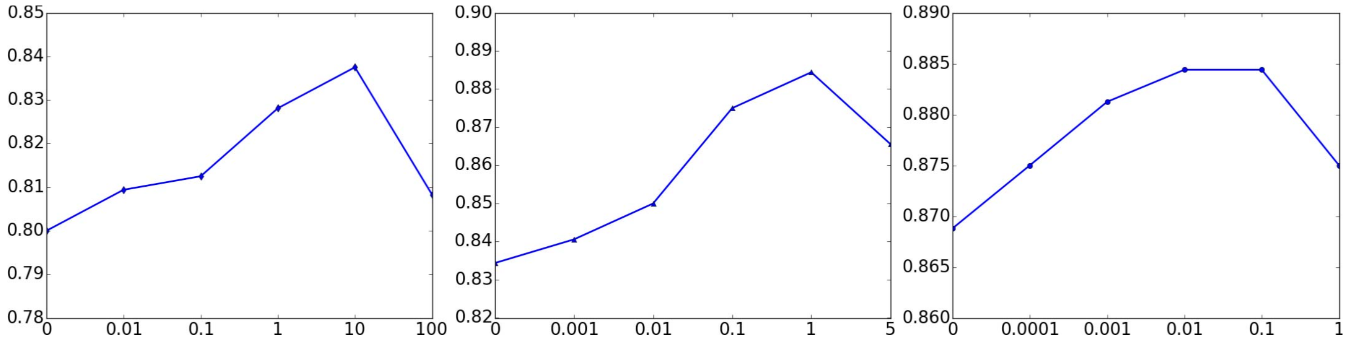
Fig. 5. The converge process.

$\lambda_1 = \lambda_2 = \lambda_3 = 0$, the nuclear norm and ℓ_1 norm in the objective function are inactive and our method degenerates to SVM without regularization; when $\lambda_1 > 0, \lambda_2 = \lambda_3 = 0$, our method degenerates to BSVM. To study the influence of each parameter, we fix the other two parameters and obtain the corresponding test accuracy. Fig. 6 shows the influence of each parameter on subject B05 in Exp. 2.

When $\lambda_2 = \lambda_3 = 0$, with the increase of λ_1, the test accuracy is enhanced accordingly, implying that taking the inter-sample structure information into account can improve the classification accuracy. Also, for certain positive value of λ_1, the test accuracy reaches its optimal value, which is equal to that of BSVM. As λ_1 keeps increasing, the testing accuracy begins to decrease. This is because when λ_1 is too large, most of the singular values in the regression parameter would be set to zero and some structure information of inter samples would be discarded. Similar phenomena also occur for λ_2 and λ_3, where suitable λ_2 and λ_3 can well leverage the intra-sample structure information and remove the intra-sample outliers for EEG data, leading to better performance. Similar tendencies of the free parameters also occur when applying RSMM to other subset data.

TABLE IV
STATISTICAL SIGNIFICANCE (p-VALUES) COMPARISON BETWEEN RSMM AND OTHER CLASSIFIERS

Datasets	Exp.1	Exp.2	Exp.3(LvsR)	Exp.3(LvsF)	Exp.3(LvsT)	Exp.3(RvsF)	Exp.3(RvsT)	Exp.3(FvsT)
Ours vs RPCA+BSVM	0.07209	**0.00002**	**0.00163**	**0.00332**	**0.03696**	**0.00287**	**0.00723**	**0.00277**
Ours vs SVM	**0.04428**	**0.000008**	**0.00162**	**0.00203**	**0.04532**	**0.00188**	**0.00057**	**0.00195**
Ours vs RGLM	**0.02615**	**0.00002**	**0.00184**	**0.00225**	**0.00135**	**0.00739**	**0.00440**	**0.00295**
Ours vs BSVM	0.13104	**0.000009**	**0.00038**	**0.01129**	0.07927	**0.00329**	**0.00376**	**0.00247**
Ours vs SMM	0.07315	**0.00003**	**0.00086**	**0.00903**	**0.03655**	**0.01198**	**0.00331**	**0.00387**



(a) (b) (c)

Fig. 6. The influence of free parameters on the classification performance for B05 in Exp. 2. The x-axis denotes the value of free parameter, and y-axis denotes testing accuracy. (a) λ_1 ($\lambda_2 = \lambda_3 = 0$). (b) λ_2 ($\lambda_1 = 10$, $\lambda_3 = 0.01$).(c) λ_3 ($\lambda_1 = 10$, $\lambda_2 = 1$).

TABLE V
THE TESTING ACCURACY OF DATASET IVa OF BCI COMPETITION III WITH DIFFERENT RATIOS OF TRAINING AND TESTING DATA

Subject/ratio	$n : n_{te} = 1 : 1$	$n : n_{te} = 2 : 1$	$n : n_{te} = 3 : 1$
aa	0.7810	0.7857	**0.7905**
al	0.9762	0.9857	**0.9905**
av	**0.7714**	0.7429	0.7476
aw	0.8143	0.8337	**0.8429**
ay	0.9571	0.9665	**0.9714**
avg	0.8600	0.8629	**0.8686**

3) Influence of Different Ratios of $n : n_{te}$: We further consider the influence of different ratios of training and testing trials on the testining performance of RSMM. Take the Dataset IVa of BCI Competition III as an example, which consists of 280 trials for each subject, we randomly partition the dataset into 4 equal sized subsets. The last subset is remained as testing set, we respectively select first, first two and first three subsets as the training set. In this way, the ratio of $n : n_{te}$ is set to be 1 : 1, 2 : 1 and 3 : 1 accordingly. Then we train classifiers with training data and evaluate the classification accuracy on the testing data. We repeat this process for five times and average the results. The averaged results are presented in Table V. From Table V, it is observed that in the most cases, the testing classification performance can be improved as the ratio of training data increases. This is because the size of training trials (n = 70, 140, 210) is far smaller than the dimensionality ($d = 120 \times 300 = 36000$). The increase of training samples can help to determine the variables within the regression matrix $\mathbf{W}$. However, in a few cases, such as subject 'av', which has relative mediocre classification performance, the classification performance is even degraded as the training data increases. This may be because of the non-stationarity of EEG signals caused by the poor performance of brain signal self-modulation. When the newly increased training data has different noise distribution, the model parameters can be biased. Therefore, it can be concluded that for subject with better ability to self-modulate his/her brain signals, the classification performance of RSMM can be improved with the increase of ratio of training data.

V. CONCLUSION

We present a novel RSMM method for single trial EEG data classification, which is a matrix classifier that can not only take the intrinsic structural information for each data matrix into consideration, but also eliminate outliers for each EEG data during model training. An ADMM based method is further derived to train the model and recover each EEG signal simultaneously. To evaluate the performance of RSMM, we conduct extensive experiments, and our method achieves the best performance, which shows the effectiveness of RSMM in real world EEG classification tasks. Though the proposed RSMM is applied to EEG data processing, it is general and robust enough to be used in other systems involving classification of low-rank data with intra-sample outliers, such as two-dimensional digital imaging, flow cytometry, etc.

This paper also casts light on several future works based on the current RSMM method. To begin with, we extract the matrix-form features using the TDP algorithm to preserve structural information, under the assumption that structural information is embeded in the channel by time matrices. However, there is also other feature extraction techniques that arrange the features into high-order tensors, such as in the form of channel × channel × band [55]. It is of interest to explore the structural information embeded in tensors, which is also more complicated than that in matrices due to the multilinear ranks. Secondly, the free parameters λ_1, λ_2 and λ_3 that control the regularization for feature recovery and classifier training are required to be fine tuned via cross validation. We are looking forward to developing automatic model selection to choose these parameters.

Appendix

Proof

Proof of Theorem 1: To prove Theorem 1, we first briefly introduce the concept of subgradient of a convex function $f : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}$. We define that $\mathbf{\Lambda} \in \mathbb{R}^{d_1 \times d_2}$ is a subgradient of f at point $\mathbf{X}_0$, denoted as $\mathbf{\Lambda} \in \partial f(\mathbf{X}_0)$, if

$$f(\mathbf{X}) \geq f(\mathbf{X}_0) + tr(\mathbf{\Lambda}^T(\mathbf{X} - \mathbf{X}_0)) \tag{21}$$

for all $\mathbf{X}$. To update $\mathbf{Z}$, by fixing the other parameters, the optimization problem in Eq. (5) in paper is equivalent to $\arg\min_{\mathbf{Z}} g(\mathbf{Z})$, where

$$g(\mathbf{Z}) = \lambda_1 ||\mathbf{Z}||_* + tr(\mathbf{V}^T\mathbf{Z}) + \frac{\mu_1}{2}||\mathbf{Z} - \mathbf{W}||_F^2. \tag{22}$$

Here $g(\mathbf{Z})$ is strongly convex due to the quadratic term. It is obvious there exists an optimal minimizer $\hat{\mathbf{Z}}$, and we need to prove $\hat{\mathbf{Z}} = \frac{1}{\mu_1}\mathcal{D}_{\lambda_1}(\mu_1\mathbf{W} - \mathbf{V})$. Now $\hat{\mathbf{Z}}$ minimizes $f(\mathbf{Z})$ if and only if $\mathbf{0}$ is a subgradient of the function f at the point $\hat{\mathbf{Z}}$, namely,

$$\mathbf{0} \in \lambda_1 \partial ||\hat{\mathbf{Z}}||_* + \mathbf{V} + \mu_1(\hat{\mathbf{Z}} - \mathbf{W}), \tag{23}$$

where $\partial ||\hat{\mathbf{Z}}||_*$ is the set of subgradients of the nuclear norm. Let $\mathbf{Z}$ be an arbitrary matrix and denote its SVD decomposition as $\mathbf{U\Sigma V}^T$. It follows the literature [27], [34], [56] that

$$\begin{aligned}\partial ||\mathbf{Z}||_* = \{&\mathbf{UV}^T + \mathbf{P} : \mathbf{P} \in \mathbf{R}^{d_1 \times d_2}, \\ &\mathbf{U}^T\mathbf{P} = \mathbf{0}, \mathbf{PV} = \mathbf{0}, ||\mathbf{P}||_F \leq 1\}.\end{aligned} \tag{24}$$

To prove $\hat{\mathbf{Z}} := \frac{1}{\mu_1}\mathcal{D}_{\lambda_1}(\mu_1\mathbf{W} - \mathbf{V})$ satisfies Eq. (23), we decompose $\mu_1\mathbf{W} - \mathbf{V}$ into two components as

$$\mu_1\mathbf{W} - \mathbf{V} = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^T + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T, \tag{25}$$

where $\mathbf{U}_0$ and $\mathbf{V}_0$ ($\mathbf{U}_1$ and $\mathbf{V}_1$) are singular vectors associated with singular values greater than (not greater than) λ_1. Thus $\hat{\mathbf{Z}} = \frac{1}{\mu_1}\mathbf{U}_0(\mathbf{\Sigma}_0 - \lambda_1\mathbf{I})\mathbf{V}_0^T$ and we can obtain

$$\begin{aligned}\mu_1(\mathbf{W} - \hat{\mathbf{Z}}) - \mathbf{V} &= \mu_1\mathbf{W} - \mathbf{V} - \mu_1\hat{\mathbf{Z}} \\ &= \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^T + \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T - (\mathbf{U}_0(\mathbf{\Sigma}_0 - \lambda_1\mathbf{I})\mathbf{V}_0^T) \\ &= \lambda_1(\mathbf{U}_0\mathbf{V}_0^T + \frac{1}{\lambda_1}\mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T)\end{aligned} \tag{26}$$

Compare with Eq. (24), we define $\mathbf{P} = \frac{1}{\lambda_1}\mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^T$. It is easy to verify that $\mathbf{U}_0\mathbf{P}^T = \mathbf{0}$, $\mathbf{PV}_0 = \mathbf{0}$, and $||\mathbf{P}||_F \leq 1$ since the diagonal matrix $\mathbf{\Sigma}_1$ is bounded by λ_1. Thus we have $\mu_1(\mathbf{W} - \hat{\mathbf{Z}}) - \mathbf{V} \in \lambda_1\partial||\hat{\mathbf{Z}}||_*$. □

Proof of Theorem 2: The optimization problem in Eq. (8) in paper is equivalent to

$$\begin{aligned}&\arg\min_{\mathbf{W},b} \frac{\mu_1}{2}||\mathbf{Z} - \mathbf{W}||_F^2 - tr(\mathbf{V}^T\mathbf{W}) + \sum\nolimits_{i=1}^n \xi_i \\ &s.t. \quad \forall\ i,\ \ y_i(tr(\mathbf{W}^T\mathbf{L}_i) + b) \geq 1 - \xi_i, \xi_i \geq 0.\end{aligned} \tag{27}$$

To solve the optimization in Eq. (27), we construct the Lagrangian function with

$$\begin{aligned}\mathcal{L}'(\mathbf{W}, b, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\beta}) = &\frac{\mu_1}{2}||\mathbf{Z} - \mathbf{W}||_F^2 - tr(\mathbf{V}^T\mathbf{W}) + \sum\nolimits_{i=1}^n \xi_i \\ &- \sum_{i=1}^n \alpha_i\{y_i[tr(\mathbf{W}^T\mathbf{L}_i) + b] - 1 + \xi_i\} \\ &+ \sum\nolimits_{i=1}^n \beta_i\xi_i,\end{aligned} \tag{28}$$

where $\boldsymbol{\alpha}, \boldsymbol{\beta} \geq \mathbf{0}$ are the Lagrange multipliers. Setting the derivatives of $\mathcal{L}'$ with respect to $\boldsymbol{\xi}$ to zero, we have

$$\nabla_{\boldsymbol{\xi}}\mathcal{L}' = 1 - \alpha_i - \beta_i = 0. \tag{29}$$

Then $\beta_i = 1 - \alpha_i \geq 0$ implies $\alpha_i \leq 1$. Setting the derivatives of $\mathcal{L}'$ with respect to $\mathbf{W}$ and b to zero, we obtain

$$\nabla_{\mathbf{W}}\mathcal{L}' = \mu_1(\mathbf{W} - \mathbf{Z}) - \mathbf{V} - \sum\nolimits_{i=1}^n \alpha_i y_i\mathbf{L}_i = 0, \tag{30}$$

This implies the optimal $\hat{\mathbf{W}}$ satisfies

$$\hat{\mathbf{W}} = \frac{1}{\mu_1}(\mu_1\mathbf{Z} + \mathbf{V} + \sum\nolimits_{i=1}^n \alpha_i y_i\mathbf{L}_i) \tag{31}$$

As for the derivative for b, we have

$$\nabla_b\mathcal{L}' = \sum\nolimits_{i=1}^n \alpha_i y_i = 0. \tag{32}$$

Substituting Eq. (29 - 32) into Eq. (28), we obtain

$$\begin{aligned}&\mathcal{L}'(\mathbf{W}, b, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\beta}) \\ &= \frac{\mu_1}{2}||\mathbf{Z} - \mathbf{W}||_F^2 - \sum_{i=1}^n \alpha_i\{y_i[tr(\mathbf{W}^T\mathbf{L}_i) + b] - 1\} - tr(\mathbf{V}^T\mathbf{W}) \\ &\quad //\text{eliminate } \boldsymbol{\xi} \text{ with Eq. (29)} \\ &= \frac{\mu_1}{2}||\mathbf{Z} - \mathbf{W}||_F^2 - \sum_{i=1}^n \alpha_i\{y_i tr(\mathbf{W}^T\mathbf{L}_i)\} + \sum_{i=1}^n \alpha_i - tr(\mathbf{V}^T\mathbf{W}) \\ &\quad //\text{eliminate } b \text{ with Eq. (32)} \\ &= \frac{\mu_1}{2}||\mathbf{Z} - \frac{1}{\mu_1}(\mu_1\mathbf{Z} + \mathbf{V} + \sum_{i=1}^n \alpha_i y_i\mathbf{L}_i)||_F^2 \\ &\quad - \sum_{i=1}^n \frac{\alpha_i}{\mu_1}\{y_i tr[(\mu_1\mathbf{Z} + \mathbf{V} + \sum_{j=1}^n \alpha_j y_j\mathbf{L}_j)^T\mathbf{L}_i]\} + \sum_{i=1}^n \alpha_i \\ &\quad - tr(\mathbf{V}^T\frac{1}{\mu_1}(\mu_1\mathbf{Z} + \mathbf{V} + \sum_{i=1}^n \alpha_i y_i\mathbf{L}_i)) \\ &\quad //\text{eliminate } \mathbf{W} \text{ with Eq. (31)} \\ &= \frac{1}{2\mu_1}||\mathbf{V} + \sum_{i=1}^n \alpha_i y_i\mathbf{L}_i||_F^2 - \sum_{i=1}^n y_i tr(\mathbf{Z}^T\mathbf{L}_i)\alpha_i \\ &\quad - \sum_{i=1}^n \frac{y_i tr(\mathbf{V}^T\mathbf{L}_i)}{\mu_1}\alpha_i - \sum_{i=1}^n\sum_{j=1}^n \frac{y_i y_j tr(\mathbf{L}_i^T\mathbf{L}_j)}{\mu_1}\alpha_i\alpha_j \\ &\quad + \sum_{i=1}^n \alpha_i - tr(\mathbf{V}^T\mathbf{Z}) - \frac{1}{\mu_1}tr(\mathbf{V}^T\mathbf{V}) - \sum_{i=1}^n \frac{y_i tr(\mathbf{V}^T\mathbf{L}_i)}{\mu_1}\alpha_i \\ &= -\sum_{i=1}^n\sum_{j=1}^n \frac{y_i y_j tr(\mathbf{L}_i\mathbf{L}_j)}{2\mu_1}\alpha_i\alpha_j \\ &\quad + \sum_{i=1}^n (1 - \frac{y_i tr[(\mu_1\mathbf{Z} + \mathbf{V})^T\mathbf{L}_i]}{\mu_1})\alpha_i + Const.\end{aligned} \tag{33}$$

Here *Const* is a constant and $Const = -tr(\mathbf{V}^T\mathbf{Z}) - \frac{1}{2\mu_1}tr(\mathbf{V}^T\mathbf{V})$. We obtain the above equations by minimizing $\mathcal{L}'$ with respect to $\mathbf{W}$ and b. Recall the constraint in Eq. (32) and $\alpha_i \geq 0$, we get the dual optimization problem of Eq. (8) in the paper and denote its optimal solution as $\hat{\boldsymbol{\alpha}}$. Then $\hat{\mathbf{W}}$ can be obtain with Eq. (31). As for the optimal solution of b, we consider the KKT conditions, which provide

$$\hat{\alpha_i}\{y_i tr[(\hat{\mathbf{W}}^T\mathbf{L}_i) + b] - 1 + \hat{\xi_i}\} = 0,$$
$$\beta_i\hat{\xi_i} = 0 \qquad (34)$$

For any $0 < \hat{\alpha_i} < 1$ and corresponding $\beta_i > 0$, we have $\hat{\xi_i} = 0$ and $y_i\{tr(\hat{\mathbf{W}}^T\mathbf{L}_i) + b\} = 1$. Then we have $\hat{b} = y_i - tr(\hat{\mathbf{W}}^T\mathbf{L}_i)$. In practice, we employ the optimal b with the averaging solution:

$$b = \frac{1}{n}\sum_{i=1}^{n}\{y_i - tr(\hat{\mathbf{W}}^T\mathbf{L}_i)\}. \qquad (35)$$

□

Proof of Theorem 3 Let $H(\mathbf{L}_i)$ denotes $h(\mathbf{W}, b, \mathbf{L}_i) + \lambda_2||\mathbf{L}_i||_* - tr(\mathbf{M}_i^T\mathbf{L}_i) + \frac{\mu_2}{2}||\mathbf{X}_i - \mathbf{L}_i - \mathbf{S}_i||_F^2$. Then the optimization problem in Eq. (12) of this paper has optimal minimizer since all terms in H with respect to $\mathbf{L}_i$ are convex. And $\hat{\mathbf{L}}_i$ minimizes $H(\mathbf{L}_i)$ if and only if $\mathbf{0} \in \partial H(\hat{\mathbf{L}}_i)$ is satisfied. Here the subgradient of $H(\mathbf{L}_i)$ is

$$\partial H(\mathbf{L}_i) = \partial h + \lambda_2\partial||\mathbf{L}_i||_* - \mathbf{M}_i - \mu_2(\mathbf{X}_i - \mathbf{L}_i - \mathbf{S}_i), \qquad (36)$$

where $\partial||\mathbf{L}_i||_*$ is the subgradient for nuclear norm and ∂h can be extended as

$$\partial h = \begin{cases} -y_i\mathbf{W}, & \text{if } y_i[tr(\mathbf{W}^T\mathbf{L}_i) + b] < 1 \\ 0, & \text{if } y_i[tr(\mathbf{W}^T\mathbf{L}_i) + b] \geq 1 \end{cases} \qquad (37)$$

Thus the optimal minimizer $\hat{\mathbf{L}}_i$ satisfies

$$\mathbf{0} \in \lambda_2\partial||\mathbf{L}_i||_* - \mathbb{I}_{y_i[tr(\mathbf{W}^T\mathbf{L}_i)+b]<1}y_i\mathbf{W} - \mathbf{M}_i + \mu_2(\mathbf{L}_i + \mathbf{S}_i - \mathbf{X}_i) \qquad (38)$$

With the proximal operator, we can obtain

$$\hat{\mathbf{L}}_i = \frac{1}{\mu_2}\mathcal{D}_{\lambda_2}(\mathbb{I}_{y_i[tr(\mathbf{W}^T\mathbf{L}_i)+b]<1}y_i\mathbf{W} + \mathbf{M}_i + \mu_2(\mathbf{X}_i - \mathbf{S}_i)), \qquad (39)$$

which concludes the proof. □

Proof of Theorem 4 To prove Theorem 4, we first reformulate the optimization problem as

$$\arg\min_{\mathbf{S}_i}\frac{\mu_2}{2}||\mathbf{S}_i - (\mathbf{X}_i - \mathbf{L}_i) - \frac{1}{\mu_2}\mathbf{M}_i||_F^2 + \lambda_3||\mathbf{S}_i||_1 + Const' \qquad (40)$$

where $Const'$ is a constant and $Const' = tr[\mathbf{M}_i^T(\mathbf{X}_i - \mathbf{L}_i)] + \frac{1}{2\mu_2}tr(\mathbf{M}_i^T\mathbf{M}_i)$. Based on the proximal operator defined in Section. 2 in the paper, the solution of optimization problem in Eq. (40) can be derived by the proximal operator with

$$\hat{\mathbf{S}}_i = prox_{\frac{\lambda_3}{\mu_2}||\cdot||_1}(\mathbf{X}_i - \mathbf{L}_i + \frac{1}{\mu_2}\mathbf{M}_i)$$
$$= \frac{1}{\mu_2}sgn(\mathbf{Y}_i) \circ \max(|\mathbf{Y}_i| - \lambda_3, 0), \qquad (41)$$

where $\mathbf{Y}_i = \mu_2(\mathbf{X}_i - \mathbf{L}_i) - \mathbf{M}_i$. □

References

[1] N. Birbaumer and L. G. Cohen, "Brain–computer interfaces: Communication and restoration of movement in paralysis," *J. Physiol.*, vol. 579, no. 3, pp. 621–636, 2007.

[2] K. K. Ang *et al.*, "A large clinical study on the ability of stroke patients to use an eeg-based motor imagery brain–computer interface," *Clin. EEG Neurosci.*, vol. 42, no. 4, pp. 253–258, 2011.

[3] L. C. Parra *et al.*, "Spatiotemporal linear decoding of brain state," *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 107–115, Jan. 2008.

[4] M. I. Al-Kadi, M. B. I. Reaz, and M. A. M. Ali, "Evolution of electroencephalogram signal analysis techniques during anesthesia," *Sensors*, vol. 13, no. 5, pp. 6605–6635, 2013.

[5] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 4, no. 2, p. R1, 2007.

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

[7] B. Blankertz, M. Kawanabe, R. Tomioka, F. U. Hohlefeld, V. V. Nikulin, and K.-R. Müller, "Invariant common spatial patterns: Alleviating nonstationarities in brain–computer interfacing," in *Proc. NIPS*, 2007, pp. 113–120.

[8] X. Li, C. Guan, H. Zhang, and K. K. Ang, "A unified Fisher's ratio learning method for spatial filter optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 11, pp. 2727–2737, Nov. 2017.

[9] M. Congedo, L. Korczowski, A. Delorme, and F. L. da Silva, "Spatiotemporal common pattern: A companion method for ERP analysis in the time domain," *J. Neurosci. Methods*, vol. 267, pp. 74–88, Jul. 2016.

[10] A. S. Aghaei, M. S. Mahanta, and K. N. Plataniotis, "Separable common spatio-spectral patterns for motor imagery BCI systems," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 1, pp. 15–29, Jan. 2016.

[11] H.-I. Suk and S.-W. Lee, "A novel Bayesian framework for discriminative feature extraction in brain–computer interfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 2, pp. 286–299, Feb. 2013.

[12] H. Wang and X. Li, "Regularized filters for L1-norm-based common spatial patterns," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 2, pp. 201–211, Feb. 2016.

[13] W. Wu, Z. Chen, X. Gao, Y. Li, E. N. Brown, and S. Gao, "Probabilistic common spatial patterns for multichannel EEG analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 639–653, Mar. 2015.

[14] M. Kawanabe, W. Samek, K.-R. Müller, and C. Vidaurre, "Robust common spatial filters with a maxmin approach," *Neural Comput.*, vol. 26, no. 2, pp. 349–376, 2014.

[15] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek, "Optimizing spatial filters by minimizing within-class dissimilarities in electroencephalogram-based brain–computer interface," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 4, pp. 610–619, Apr. 2013.

[16] G. Repovš, "Dealing with noise in eeg recording and data analysis," *Inf. Med. Slovenica*, vol. 15, no. 1, pp. 18–25, 2010.

[17] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Müllers, "Fisher discriminant analysis with kernels," in *Proc. IEEE Signal Process. Soc. Workshop Neural Netw. Signal Process. (IX)*, Aug. 1999, pp. 41–48.

[18] F. Qi, Y. Li, and W. Wu, "RSTFC: A novel algorithm for spatio-temporal filtering and classification of single-trial EEG," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3070–3082, Dec. 2015.

[19] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[20] C.-H. Chuang, L.-W. Ko, Y.-P. Lin, T.-P. Jung, and C.-T. Lin, "Independent component ensemble of EEG for brain–computer interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 2, pp. 230–238, Mar. 2014.

[21] D. W. Hosmer, Jr., and S. Lemeshow, *Applied Logistic Regression*. Hoboken, NJ, USA: Wiley, 2004.

[22] H. Zeng and A. Song, "Optimizing single-trial EEG classification by stationary matrix logistic regression in brain–computer interface," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2301–2313, Nov. 2016.

[23] R. Tomioka and K. Aihara, "Classifying matrices with a spectral regularization," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 895–902.

[24] L. Wolf, H. Jhuang, and T. Hazan, "Modeling appearances with low-rank SVM," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–6.

[25] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, “Bilinear classifiers for visual recognition,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1482–1490.

[26] M. Dyrholm, C. Christoforou, and L. C. Parra, “Bilinear discriminant component analysis,” *J. Mach. Learn. Res.*, vol. 8, pp. 1097–1111, May 2007.

[27] E. J. Candès and B. Recht, “Exact matrix completion via convex optimization,” *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[28] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino, “Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2488–2495.

[29] H. Zhou and L. Li, “Regularized matrix regression,” *J. Roy. Statist. Soc., Ser. B, Statist. Methodol.*, vol. 76, no. 2, pp. 463–483, 2014.

[30] L. Luo, Y. Xie, Z. Zhang, and W.-J. Li, “Support matrix machines,” in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 938–947.

[31] F. Lotte, “A tutorial on EEG signal-processing techniques for mental-state recognition in brain–computer interfaces,” in *Guide to Brain-Computer Music Interfacing*. London, U.K.: Springer, 2014, pp. 133–161.

[32] P. Li, P. Xu, R. Zhang, L. Guo, and D. Yao, “L1 norm based common spatial patterns decomposition for scalp EEG BCI,” *BioMed. Eng. OnLine*, vol. 12, no. 1, p. 77, 2013.

[33] Y. Zhang, G. Zhou, J. Jin, Q. Zhao, X. Wang, and A. Cichocki, “Sparse Bayesian classification of EEG for brain–computer interface,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2256–2267, Nov. 2016.

[34] J.-F. Cai, E. J. Candes, and Z. Shen, “A singular value thresholding algorithm for matrix completion,” *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[35] N. Parikh and S. Boyd, “Proximal algorithms,” *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[36] T. Zhou and D. Tao, “GoDec: Randomized low-rank & sparse matrix decomposition in noisy case,” in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 33–40.

[37] A. E. Waters, A. C. Sankaranarayanan, and R. Baraniuk, “SpaRCS: Recovering low-rank and sparse matrices from compressive measurements,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1089–1097.

[38] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?” *J. ACM*, vol. 58, no. 3, p. 11, May 2011.

[39] T. Bouwmans and E. H. Zahzah, “Robust PCA via principal component pursuit: A review for a comparative evaluation in video surveillance,” *Comput. Vis. Image Understand.*, vol. 122, pp. 22–34, May 2014.

[40] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, “RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2233–2246, Nov. 2012.

[41] T. Kobayashi and N. Otsu, “Efficient optimization for low-rank integrated bilinear classifiers,” in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2012, pp. 474–487.

[42] D. Huang, R. Cabral, and F. De la Torre, “Robust regression,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 363–375, Feb. 2016.

[43] Y.-X. Wang and Y.-J. Zhang, “Nonnegative matrix factorization: A comprehensive review,” *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1336–1353, Jun. 2013.

[44] H. Lee, A. Cichocki, and S. Choi, “Nonnegative matrix factorization for motor imagery eeg classification,” in *Artificial Neural Networks—ICANN*. Berlin, Germany: Springer, 2006, pp. 250–259.

[45] H. Lee and S. Choi, “Group nonnegative matrix factorization for EEG classification,” in *Proc. 12th Int. Conf. Artif. Intell. Statist.*, 2009, pp. 320–327.

[46] N. Seichepine, S. Essid, C. Févotte, and O. Cappé, “Soft nonnegative matrix co-factorization,” *IEEE Trans. Signal Process.*, vol. 62, no. 22, pp. 5940–5949, Nov. 2014.

[47] C. Damon, A. Liutkus, A. Gramfort, and S. Essid, “Non-negative matrix factorization for single-channel eeg artifact rejection,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 1177–1181.

[48] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[49] G. Dornhege, B. Blankertz, G. Curio, and K. R. Müller, “Boosting bit rates in noninvasive EEG single-trial classifications by feature combination and multiclass paradigms,” *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 993–1002, Jun. 2004.

[50] R. Leeb, F. Lee, C. Keinrath, R. Scherer, H. Bischof, and G. Pfurtscheller, “Brain–computer communication: Motivation, aim, and impact of exploring a virtual apartment,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 4, pp. 473–482, Dec. 2007.

[51] M. Naeem, C. Brunner, R. Leeb, B. Graimann, and G. Pfurtscheller, “Seperability of four-class motor imagery data using independent components analysis,” *J. Neural Eng.*, vol. 3, no. 3, p. 208, 2006.

[52] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, “Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b,” *Frontiers Neurosci.*, vol. 6, no. 1, p. 39, 2012.

[53] C. Vidaurre, N. Krämer, B. Blankertz, and A. Schlögl, “Time domain parameters as a feature for EEG-based brain–computer interfaces,” *Neural Netw.*, vol. 22, no. 9, pp. 1313–1319, 2009.

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

[55] K. Wimalawarne, R. Tomioka, and M. Sugiyama, “Theoretical and experimental analyses of tensor-based regression and classification,” *Neural Comput.*, vol. 28, no. 4, pp. 686–715, Apr. 2016.

[56] A. S. Lewis, “The mathematics of eigenvalue optimization,” *Math. Program.*, vol. 97, nos. 1–2, pp. 155–176, 2003.