Pattern Recognition 76 (2018) 715–726

Contents lists available at ScienceDirect

Pattern Recognition

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

Sparse Support Matrix Machine

Qingqing Zheng[a,*], Fengyuan Zhu[a], Jing Qin[b], Badong Chen[c], Pheng-Ann Heng[a]

[a] *Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong*
[b] *Centre of Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong*
[c] *Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China*

ARTICLE INFO

Article history:
Received 10 April 2017
Revised 27 September 2017
Accepted 6 October 2017
Available online 9 October 2017

Keywords:
Classification
Support vector machine
Matrix analysis
Sparse
Low rank

ABSTRACT

Modern technologies have been producing data with complex intrinsic structures, which can be naturally represented as two-dimensional matrices, such as gray digital images, and electroencephalography (EEG) signals. When processing these data for classification, traditional classifiers, such as support vector machine (SVM) and logistic regression, have to reshape each input matrix into a feature vector, resulting in the loss of structural information. In contrast, modern classification methods such as support matrix machine capture these structures by regularizing the regression matrix to be low-rank. These methods assume that all entities within each input matrix can serve as the explanatory features for its label. However, in real-world applications, many features are redundant and useless for certain classification tasks, thus it is important to perform feature selection to filter out redundant features for more interpretable modeling. In this paper, we tackle this issue, and propose a novel classification technique called *Sparse Support Matrix Machine* (SSMM), which is favored for taking both the intrinsic structure of each input matrix and feature selection into consideration simultaneously. The proposed SSMM is defined as a hinge loss for model fitting, with a new regularization on the regression matrix. Specifically, the new regularization term is a linear combination of nuclear norm and ℓ_1 norm, to consider the low-rank property and sparse property respectively. The resulting optimization problem is convex, and motivates us to propose a novel and efficient generalized forward-backward algorithm for solving it. To evaluate the effectiveness of our method, we conduct comparative studies on the applications of both image and EEG data classification problems. Our approach achieves state-of-the-art performance consistently. It shows the promise of our SSMM method on real-world applications.

1. Introduction

Classification is an important research topic in the area of machine learning and pattern recognition, with wide range of empirical applications [1]. Traditional classification approaches such as support vector machine (SVM) [2] and logistic regression [3] are originally designed for input samples represented as vectors or scalars. However, modern technologies and scientific applications are frequently producing datasets where samples are naturally represented as two-dimensional matrices instead of vectors. Examples include digital images, with quantized color values at a number of pixels of rows and columns, and electroencephalogram (EEG) signals with voltage fluctuation at multiple channels over a period of time [4]. When using classical classifiers on these matrix-form data, we have to reshape them into vectors for preprocessing, which would destroy the topological structural information embedded in each input matrix, e.g., the spatial relationship between nearby pixels for image data [5], and the correlation between different channels for EEG data [6]. Moreover, when a matrix is reshaped into a vector, its dimension can be extremely high, resulting in the serious curse of dimensionality problem, especially when the sample size is limited.

To tackle these issues, several works have been proposed to classify data in matrix form directly, and to explore the correlation between columns and rows for each input matrix [7,8]. Wolf et al. [5] proposed a rank-*k* SVM model, which assumed the regression matrix was a sum of *k* rank-one orthogonal matrices. Dyrholm et al. [9] proposed a bilinear logistic regression model, decomposing the regression matrix as a product of two rank-*k* matrices. Pirsiavash et al. [10] further extended this model and proposed a bilinear classifier, by employing hinge loss for model fitting. These methods all take advantage of the low-rank assumption to exploit the correlation between columns and rows among each matrix.

[*] Corresponding author.
E-mail addresses: qqzheng@cse.cuhk.edu.hk (Q. Zheng), fyzhu@cse.cuhk.edu.hk (F. Zhu), harry.qin@polyu.edu.hk (J. Qin), chenbd@mail.xjtu.edu.cn (B. Chen), pheng@cse.cuhk.edu.hk (P.-A. Heng).

https://doi.org/10.1016/j.patcog.2017.10.003
0031-3203/© 2017 Elsevier Ltd. All rights reserved.

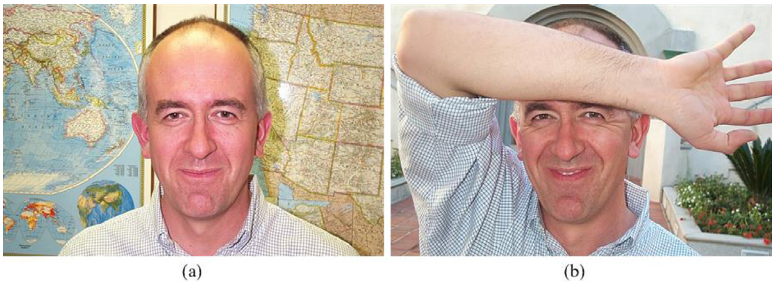


Fig. 1. Two images of the same person from the Caltech Face dataset.

However, the rank of the regression matrix is difficult to be predetermined in these methods, resulting in tedious parameter tuning procedures. To address this issue, Kobayashi and Otsu [11] derived a novel bilinear SVM model by introducing nuclear norm to determine the rank of regression matrix automatically. Luo et al. [12] extended [11] and proposed a spectral elastic net regularization, which combines both Frobenius norm and nuclear norm to constrain the regression matrix.

While all these works made good effort to take the correlation between columns and rows of the regression matrix into consideration under a low-rank assumption, and achieved satisfactory performance, one of the limitations of these methods is that they simply considered all the entities in each input matrix as explanatory factors, which is irrational in practical applications. In real-world classification tasks, many features are redundant and useless for certain classification tasks, and only a small subset of them are important to explain the label of each sample [13]. Let's take two images for face classification as an example to demonstrate this. Fig. 1 shows two such images of the same person with different postures. Specifically, in Fig. 1(b), the person put his forearm on his forehead, which makes this image quite different from Fig. 1(a). However, these two images are still in the same category. This indicates the features of posture cannot serve as explanatory factors for the label of this image. Similar phenomena also occur in other classification tasks as well. In a word, only a small subset of features should be considered to make the classifier more interpretable. Moreover, for applications like EEG data classification, where the sample size is rather small, it could be beneficial in model fitting to make the model simpler by only considering a small subset of useful features. Unfortunately, even though the low-rank assumption used in previous matrix classification approaches can also somehow control the model complexity, it cannot be used to perform feature selection to select these useful features. Thus, it could be insufficient to only consider the low-rank property when dealing with the matrix classification tasks.

To tackle the issue of feature selection, some approaches have been proposed to introduce the sparse property to the regression matrix. Previous methods developed for sparse modeling mostly rely on the use of ℓ_1-norm as a constraint [14]. This idea has been applied to many traditional classification approaches for data in vector form, including SVM [15] and logistic regression [16], to sparsify the coefficient vectors for better classification performance with simpler and more interpretable model. However, this idea has not been applied in the context of matrix classification.

In this paper, we address the aforementioned issues by introducing the sparse property to the problem of matrix classification, and propose a novel classifier called *Sparse Support Matrix Machine* (SSMM). In our SSMM, instead of just considering the correlation of each input matrix as previous approaches did, we assume the regression matrix is not only low-rank representable, but also with the features of sparsity. Note that, even though both low-rank and sparse properties can control the topological structural information of a matrix, they can actually be seen as two orthogonal concepts (consider a diagonal matrix, which has full rank but is highly sparse; or a matrix with row vectors being the same, which is rank-1 but not sparse at all, see Fig. 2 as an example). More specifically, we study the sum of nuclear norm and ℓ_1-norm of the regression matrix as the regularization term, to control its low-rank and sparse properties respectively. We also employ the hinge loss to maximize the margin between matrices belonging to different classes for our SSMM method, due to its desirable capability of sparseness and robustness in modeling. In this way, our approach is not only favored for the ability to classify data in matrix form without loss of the structural information, but also able to perform feature selection for better classification performance.

The optimization problem for SSMM is convex, but the combination of hinge loss, ℓ_1-norm and nuclear norm makes the problem nontrivial to be solved directly. To tackle this issue, we split the problem into sub-problems with the Generalized Forward-Backward (GFB) splitting approach [17], and develop a novel and effective algorithm to solve the optimization problem efficiently. To evaluate the effectiveness of our method, we apply the SSMM to image classification and EEG data classification problems, where samples of each dataset can be represented as matrices naturally. Comparing with state-of-the-art approaches, our SSMM method achieves superior performance. It shows the effectiveness and the strong empirical value of SSMM in real-world applications.

The contributions of this paper can be summarized as follows:

- We propose a novel classifier called SSMM for classification problem involved explanatory features that are two-dimensional matrices. Compared with existing classifiers, the proposed method can simultaneously leverage the inherent structural information within matrix-form data and select useful features, and hence improve the classification performance.
- We propose a novel objective function based on regularized risk minimization framework by regularizing the combination of nuclear norm and ℓ_1 norm of the regression matrix, and develop an efficient solver based on GFB splitting framework to solve it. We also provide a theoretical guarantee for the global convergence and analyze the excess risk statistically.
- We extensively evaluate the proposed SSMM on four real datasets. The results show that SSMM achieves state-of-the-art generalization performance in image classification and single-trial EEG classification tasks.

2. Notations

In this section we introduce notations and preliminaries that will be used later in this work. Following standard conventions, we represent scalar values by lowercase letters (e.g. x), vectors by bold lowercase letters (e.g. $\mathbf{x}$), and matrices by bold uppercase letters (e.g. $\mathbf{X}$). For a matrix $\mathbf{X} \in \mathbb{R}^{m \times d}$, there exists a singular value decomposition (SVD) of the form $\mathbf{X} = \mathbf{U\Sigma V^T}$,

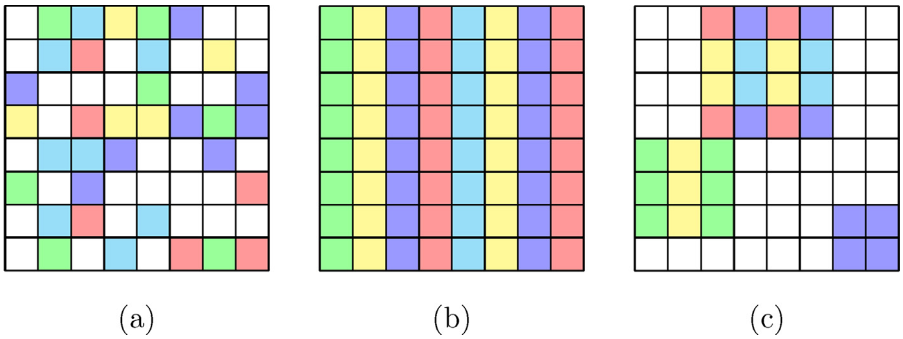


Fig. 2. Three matrices with special structures: (a) sparse; (b) low rank; (c) simultaneous sparse and low rank. Various colors denote different numerical values and white color represents zero.

where $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbb{R}^{d\times r}$ are both unitary matrix, and $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2, \cdots, \sigma_r)$, in which the diagonal entities are singular values satisfying $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$. Here, the number of non-negative singular values r denotes the rank of $\mathbf{X}$ and $r \leq \min(m, d)$.

Also, we set $||\mathbf{X}||_* = \sum_{i=1}^{r} \sigma_i$ to represent the nuclear norm of a matrix $\mathbf{X}$, and $||\mathbf{X}||_1 = \sum_{i,j} |x_{i,j}|$ as the ℓ_1 norm.

We further introduce the singular value thresholding operator, which will be used later in this paper to derive the solver of SSMM model.

Definition 1. For any $\tau \geq 0$, the singular value thresholding operator is well defined as follows:

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

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

This operator $\mathcal{D}_\tau$ shrinks the singular values of $\mathbf{X}$ with a soft-thresholding rule. In the literature [18,19], such a transformation is also called singular value shrinkage operator, which is widely used for low-rank matrix completion [18].

3. Problem formulation and related works

In principle, the proposed SSMM is a regularized binary matrix classifier, which can not only take correlation of columns and rows in each matrix into consideration, but also perform feature selection to remove redundant features for more interpretable modeling. In order to facilitate description, we first formulate the matrix classification problem as follows.

Given a set of training samples $\{\mathbf{X}_i, y_i\}_{i=1}^{n}$, $\mathbf{X}_i \in \mathbb{R}^{m\times d}$ is the i_{th} input matrix and $y_i \in \{1, -1\}$ is its corresponding true label. We aim to train a function $f : \mathbb{R}^{m\times d} \to \mathbb{R}$ with the given data, which can successfully identify the category of a newly arrived data.

To tackle the classification issue with classical vector-based approaches, a heuristic method is to stack a matrix-form data $\mathbf{X}_i$ into a vector first, and then train a classification model with the set of vectorized data. One of these classical classifiers is the soft marginal SVM model [20]; it can be trained by optimizing the following energy function:

$$\min_{\mathbf{w},b} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\{1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)\}_+, \tag{2}$$

where $\mathbf{x}_i = vec(\mathbf{X}_i)$ represents the vectorized data of matrix $\mathbf{X}_i$, $\{1-u\}_+ = \max(0, 1-u)$ denotes the hinge loss function to maximize the margin between vectorized data points of different categories, $\mathbf{w} \in \mathbb{R}^{md}$ is the regression parameter, $b \in \mathbb{R}$ is an offset term, and $C \in \mathbb{R}$ denotes a penalty parameter.

From the viewpoint of computation, Eq. (2) is equivalent to the following formulation, to perform matrix classification directly:

$$\min_{\mathbf{W},b} \frac{1}{2}tr(\mathbf{W}^T\mathbf{W}) + C\sum_{i=1}^{n}\{1 - y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]\}_+. \tag{3}$$

Since $tr(\mathbf{W}^T\mathbf{W}) = vec(\mathbf{W})^T vec(\mathbf{W})$ and $tr(\mathbf{W}^T\mathbf{X}_i) = vec(\mathbf{W})^T vec(\mathbf{X}_i)$, it indicates directly performing classification with Eq. (3) cannot capture the intrinsic structure of each input matrix effectively. In a word, even though the reshaping operations to transform matrices into vectors can be efficient, the structural information within each matrix could be destroyed, resulting in the loss of information. Thus, it is of research interest in how to preserve and take full advantage of intrinsic structural information within each matrix, such as the correlation between rows and columns, when training classifiers for matrix-form data. With the structural information into consideration, it is expected that the classification performance can be improved.

To take the structural information into consideration, one intuitive way is to capture the correlation within each matrix by imposing a low-rank constraint on $\mathbf{W}$. Several approaches have been proposed to tackle this problem in this way, including the low-rank SVM [5] and bilinear SVM [10]. However, these methods require the latent rank of $\mathbf{W}$ to be pre-specified manually for different applications. Kobayashi and Otsu [11] addressed this issue by introducing the variational form of nuclear norm [21] by decomposing $\mathbf{W} = \mathbf{W}_p\mathbf{W}_q$ to determine the rank automatically, but the resulting objective function is non-convex, which affects the robustness of this algorithm due to the existence of local optima. In order to solve this problem, Luo et al. [12] proposed a novel model called support matrix machine (SMM), with the optimization problem formulated as follows:

$$\arg\min_{\mathbf{W},b} \frac{1}{2}tr(\mathbf{W}^T\mathbf{W}) + \tau||\mathbf{W}||_* + C\sum_{i=1}^{n}\{1 - y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]\}_+. \tag{4}$$

Here, $\frac{1}{2}tr(\mathbf{W}^T\mathbf{W}) + \tau||\mathbf{W}||_*$ is called spectral elastic net regularization, which is employed to capture the correlation within each matrix. Note that, the nuclear norm $||\mathbf{W}||_*$ is used as a regularization term to control the rank for $\mathbf{W}$ because determining the rank of a matrix can be NP-hard [22] while the nuclear norm is known as the best convex approximation of the rank of $\mathbf{W}$ [6,19].

The SMM method is capable of capturing the latent structure within each matrix effectively. However, one of the main shortcomings of this model is that it performs classification based on all the entities of each matrix, which not only makes the model complicated but also imperils the classification performance because a lot of redundant and useless information is involved in the classification procedure. To tackle this issue, an intuitive thought is to employ a feature selection mechanism which takes only a subset of entities that encode representative features of a matrix to improve the classification performance. While taking full advantage of structural information of a matrix and meanwhile integrating a feature selection mechanism may derive more interpretable and representative features for better classification performance of matrix-form data, to our knowledge, we are not aware of any previous matrix-form classifier that simultaneously and elegantly harnesses these two important properties. In this regard, we develop the proposed SSMM, aiming at leveraging both low-rank property and sparsity property of the regression matrix to simultaneously achieve the preservation of structural information and the selection of representative features for better classification of matrix-form data.

4. The proposed SSMM

In this section, we introduce the proposed SSMM, which is, in principle, a novel matrix classifier that simultaneously considers the correlation information encoded in the input matrices and selects more representative features by removing redundant information using sparsity property. A novel and efficient algorithm based on GFB splitting framework is further derived to solve the SSMM.

4.1. Sparse Support Matrix Machine

4.1.1. The model

It is well known that hinge loss provides a tight and convex upper bound on the 0/1 indicator function. With the large margin principal, it is favored for its robustness and sparseness in prediction performance of binary classification problems. In this regard, we adopt the hinge loss function in our SSMM. In order to simultaneously preserve structural information and extract discriminative features, we impose both low-rank and sparse constraints on the regression matrix **W**. In particular, we present the objective function of our SSMM method as:

$$\arg\min_{\mathbf{W},b} \gamma||\mathbf{W}||_1 + \tau||\mathbf{W}||_* + \sum_{i=1}^{n}\{1 - y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]\}_+, \tag{5}$$

with $\mathbf{X}_i, \mathbf{W} \in \mathbb{R}^{m\times d}$. This formulation incorporates the hinge loss and constraints on regression matrix **W** for matrix classification. The regularization term on **W** is a linear combination of ℓ_1 norm $||\mathbf{W}||_1$ to control the sparseness and nuclear norm $||\mathbf{W}||_*$ to capture the correlation within each input matrix. Specifically, the ℓ_1 norm encourages **W** to be sparse by serving as a convex surrogate for the number of nonzero entries in **W**. Meanwhile, the nuclear norm, which is a convex approximation for rank of a matrix, encourages **W** to be low-rank. Since $tr(\mathbf{W}^T\mathbf{X}_i) = vec(\mathbf{W})^T vec(\mathbf{X}_i)$, when **W** is constrained to be sparse, we can implicitly perform feature selection for the input matrix by enforcing the coefficients of useless features to be 0. Thus, by setting $\tau = 0$, our model degenerates to the sparse SVM [15] if we stack the feature matrices into vector form. The combination of these two constraints yields a desirable regression matrix which is simultaneously sparse and low-rank, and thus is capable of well capturing the intrinsic structure of each input matrix and effectively selecting explanatory features for more interpretable modeling.

4.1.2. Remarks on the SSMM

The proposed SSMM is the first matrix-form classifier to simultaneously select the discriminative features and make full use of the inherent structrual information within the input matrices. This benefits from the novel regularization which is a combination of ℓ_1 norm and nuclear norm on the regression matrix **W**.

Though the combination of these two norms has been studied in machine learning areas, especially for the matrix recovery problem, the existing works are different from ours in terms of motivation or formulation. Among these methods, one stream assumes that the original input matrix can be decomposed as the summation of two different matrices, one is low-rank and the other is sparse. One pioneering work is the robust principle component analysis (RPCA) [23], which recovered the low-rank matrix contaminated by additive sparse outliers. Gu et al. [24] extended the RPCA model with more general factorization and proposed a nonconvex optimization algorithm for large scale problems. Both [25] and [26] further recovered the input matrix from the combination of a low rank component and a sparse one with compressive measurements. Inspired by RPCA, Guan et al. proposed a MahNMF model [27] which robustly estimated the low-rank component and the sparse component of the non-negative matrix. It modeled the heavy-tailed Laplacian noise by minimizing the Manhattan distance between a non-negative matrix and the product of two non-negative low-rank factor matrices. The works [28,29] further studied the statistical performance of the MahNMF in the viewpoint of the statistical learning theory. Different from the proposed SSMM, all these works decompose the estimated matrix to be a linear combination of a low-rank matrix **L** and a sparse matrix **S**, while SSMM regularizes the model parameter **W** to be simultaneously sparse and low-rank.

In the other stream, a few of studies estimate a matrix to be simultaneously sparse and low-rank. Most of them are on the matrix completion problem [30–32] with the smooth fidelity loss term. For example, Parekh et al. [32] estimated the sparse and low-rank matrix from the noisy obervations with the smooth Frobenius loss term. In addition, all these works are unsupervised and have been applied to applications like link prediction [33] but not yet in the classification task. Therefore, these approaches are inherently different from the proposed SSMM, which is a matrix-based classification method with the non-smooth hinge loss to estimate the constrained regression matrix.

To our best knowledge, the only matrix classification approach that has considered the sparsity property is the work [34], which was based on the bilinear logistic regression approach [9] with the assumption that the rank of regression matrix **W** is known. It further assumed that $\mathbf{W} = \mathbf{W}_a\mathbf{W}_b$, and enforced both $\mathbf{W}_a$ and $\mathbf{W}_b$ to be sparse by incorporating ℓ_1 norm as constraints. However, the resulting optimization is biconvex with respect to $\mathbf{W}_a$ and $\mathbf{W}_b$, it may get stuck in local minima. In addition, since the product of two sparse matrices is not guaranteed to be sparse, this method [34] cannot perform feature selection for matrix classification, resulting in a significant difference from the SSMM.

4.2. Learning algorithm

We currently present an efficient algorithm to solve the optimization problem of SSMM proposed in Eq. (5). Consider the hinge loss, ℓ_1 norm and nuclear norm in Eq. (5) are all non-smooth but convex, which is difficult to be solved together, it is intuitive to split the objective function into a sum of sub-problems which can be solved more easily.

$$\arg\min_{\mathbf{W},b} F = \arg\min_{\mathbf{W},b} h(\mathbf{W}, b) + \sum_{k=1}^{2} f_k(\mathbf{W}), \tag{6}$$

with $h = \sum_{i=1}^{n}\{1 - y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]\}_+$, $f_1 = \tau||\mathbf{W}||_*$ and $f_2 = \gamma||\mathbf{W}||_1$.

F is a sum of three lower semicontinuous and convex functions with respect to **W** and b in a real Hilbert space $\mathcal{H}$. Thus, there exists at least an optimal value of F and the set of minimizers of F

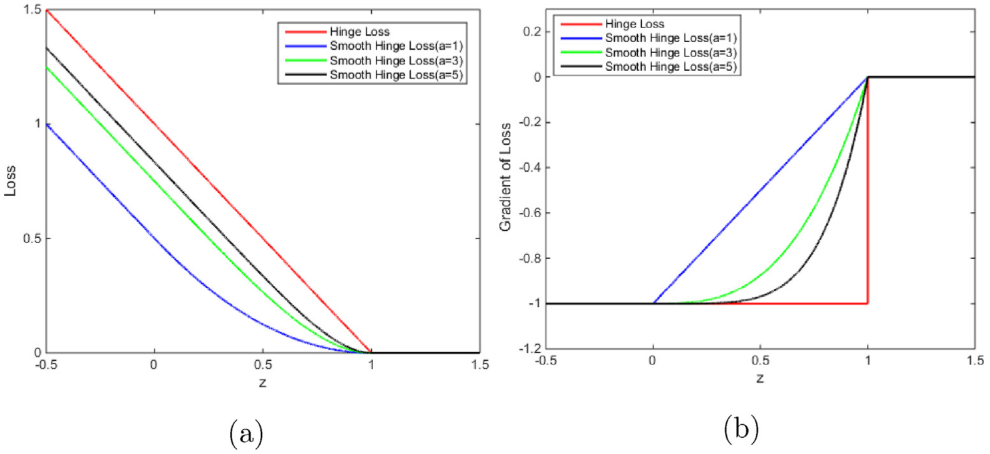


Fig. 3. Shown are the (a) loss values and (b) gradients of loss functions for hinge loss and generalized smooth hinge loss with different smooth parameters α.

verifies

$$\mathbf{0} \in \partial h + \sum_{k=1}^{2} \partial f_k, \tag{7}$$

where ∂ denotes the subdifferential operator.

However, each term involved in SSMM is non-smooth and non-differentiable, the conventional proximal algorithms cannot be applied to this problem directly. The Nesterov method in [6] requires the loss function h has a Lipschitz-continuous gradient and can only handle one convex non-negative constraint. Traditional splitting algorithms like the *alternating direction method of multipliers* (ADMM) [35] are inherently built for optimization problems with at most two separable convex non-differentiable terms. In addition, the direct extensions of ADMM for optimization problems with multiple convex terms do not necessarily converge for certain problems [36]. While the GFB splitting [17] also requires the gradient of h to be Lipschitz-continuous, it can tackle arbitrary $k > 0$ convex non-differentiable terms with simple proximal operators. In this regard, we are motivated to develop an efficient solver based on the GFB splitting framework. To pursue a loss function with Lipschitz-continuous gradient, we first smooth the loss function h by approximating it with a generalized smooth hinge loss h_α [37] with

$$\sum_{i=1}^{n} h_\alpha(z_i) = \begin{cases} \frac{\alpha}{\alpha+1} - z_i, & \text{if } z_i \le 0, \\ \frac{1}{\alpha+1}(z_i)^{\alpha+1} + \frac{\alpha}{\alpha+1} - z_i, & \text{if } 0 < z_i < 1, \\ 0, & \text{if } z_i \ge 1, \end{cases} \tag{8}$$

where $z_i = y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]$. As shown in Fig. 3(a), $h_\alpha(z_i)$ is zero for $z_i \ge 1$ and has a constant negative slope for $z_i \le 0$. When $0 < z_i < 1$, h_α has smooth transition with slope between (−1,0). In this regard, h_α not only shares similarities to hinge loss, such as sparisity, but also benefits from the differentiability. Then the gradient of h_α with respect to $\mathbf{W}$ can be easily computed by the chain rule with

$$\begin{aligned} \nabla_{\mathbf{W}} h_\alpha &= \frac{dh_\alpha}{dz}\frac{\partial z}{\partial \mathbf{W}} \\ &= \sum_{i=1}^{n} \begin{cases} -y_i\mathbf{X}_i, & \text{if } z_i \le 0, \\ (z_i^\alpha - 1)y_i\mathbf{X}_i & \text{if } 0 < z_i < 1, \\ 0, & \text{if } z_i \ge 1. \end{cases} \end{aligned} \tag{9}$$

Note that the gradients of hinge loss and the smooth ones are only different between the interval of (0, 1), as shown in Fig. 3(b).

With the approximated gradient of h, the GFB learning procedure can be implemented in an iterative manner. In each iteration, it individually evaluates the resolvent of ∂f_k (denoted as $J_{\partial f_k}$) at various points of $\mathcal{H}$, and determines the regression matrix $\mathbf{W}$ by the linear combination of the resolvents $J_{\partial f_k}$.

To tackle $J_{\partial f_k}$, we introduce two auxiliary variables $\mathbf{Z}_1$ and $\mathbf{Z}_2$, corresponding to regularizer f_1 and f_2 respectively. The update of auxiliary variables $\mathbf{Z}_k$ is

$$\mathbf{Z}_{k,t+1} = \mathbf{Z}_{k,t} + \lambda_t (J_{\frac{\theta}{\omega_k}\partial f_k}(2\mathbf{W}_t - \mathbf{Z}_{k,t} - \theta\nabla_{\mathbf{W}} h_\alpha) - \mathbf{W}_t), \tag{10}$$

where $t \in \mathbb{N}$, $\theta > 0$ denotes the step size, $\omega_k \in [0, 1]$ ($\sum \omega_k = 1$) denotes the weight of $\mathbf{Z}_k$, λ_t denotes the relaxation parameter and $J_{\frac{\theta}{\omega_k}\partial f_k}$ denotes the resolvent of $\frac{\theta}{\omega_k}\partial f_k$. The resolvent of the maximal monotone operator ∂f_k is equivalent to the proximal operator $prox_{f_k}$ [38], and all such resolvent operators are firmly non-expensive computationally.

Within each iteration, the resulting auxiliary variables $\{\mathbf{Z}_1, \mathbf{Z}_2\}$ can be projected to the regression matrix $\mathbf{W}$ by linear combination with

$$\mathbf{W}_{t+1} = \sum_k \omega_k \mathbf{Z}_{k,t+1}, \tag{11}$$

Once $\mathbf{W}$ gets updated, bias b can be easily calculated by the gradient descent algorithm with

$$b_{t+1} = b_t - \theta\nabla_b h_\alpha \tag{12}$$

In summary, the proposed learning algorithm for our SSMM is demonstrated in Algorithm 1 [1].

Algorithm 1: Generalized forward-backward algorithm for SSMM.

```
Input  : Training data $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$, sparsity coefficient $\gamma$ and
         low-rank coefficient $\tau$, weights $\omega_1, \omega_2$, smooth
         parameter $\alpha$
Output: $\mathbf{W}, b$
Initialize $\mathbf{W}, \mathbf{Z}_1, \mathbf{Z}_2 = \mathbf{0}$
repeat
    Compute $\mathbf{G} \leftarrow \nabla_{\mathbf{W}} h_\alpha$ with Eq. (9)
    $\mathbf{Z}_1 \leftarrow \mathbf{Z}_1 + \lambda_t(prox_{\frac{\theta}{\omega_1}\gamma||\cdot||_1}(2\mathbf{W} - \mathbf{Z}_1 - \theta\mathbf{G}) - \mathbf{W})$
    $\mathbf{Z}_2 \leftarrow \mathbf{Z}_2 + \lambda_t(prox_{\frac{\theta}{\omega_2}\tau||\cdot||_*}(2\mathbf{W} - \mathbf{Z}_2 - \theta\mathbf{G}) - \mathbf{W})$
    $\mathbf{W} \leftarrow \sum_{k=1}^{2} \omega_k \mathbf{Z}_k$
    $b \leftarrow b - \theta\nabla_b h_\alpha$
until Convergence;
return $\mathbf{W}, b$
```

[1] The detailed implementation of the proximal operators can refer to supplementary.

To state the robustness of Algorithm 1, we further derive Theorem 1 to ensure its convergence.

Theorem 1. *Let $\epsilon_{1,\,t}$ be the error at t_{th} iteration when computing $\nabla_{\mathbf{W}} h_\alpha$, let $\epsilon_{2,\,k,\,t}$ be the error at t_{th} iteration when computing $J_{\frac{\theta}{\omega_k}\partial f_k}$. If*

1. $\nabla h_\alpha + \sum_{k=1}^{2} \partial f_k \neq \emptyset$,
2. $(\lambda_t)_{t\in\mathbb{N}} \in [0, 2]$ *and* $\sum_{t\in\mathbb{N}} \lambda_t(2-\lambda_t) = +\infty$,
3. $\sum_{t=0}^{+\infty} ||\epsilon_{1,t}|| < +\infty$ *and* $\sum_{t=0}^{+\infty} ||\epsilon_{2,k,t}|| < +\infty$

are satisfied, then the sequences of $(\mathbf{W}_t)_{t\in\mathbb{N}}$ in Eq. (11) converges weakly towards a solution of Eq. (6). Moreover, if $\forall t \in \mathbb{N}, \lambda_t \leq 1$, then $(\mathbf{W}_t)_{t\in\mathbb{N}}$ converges strongly to the unique minimizer of Eq. (6).

The proof is similar to [17] and interested readers can refer to Section 4 in [17].

4.2.1. Computational cost

We also analyze the time complexity of Algorithm 1. The main time cost of our solver (Algorithm 1) is to calculate the gradient of the smoothed hinge loss with respect to $(\mathbf{W}, b)$ (line 3 and line 7 in Algorithm 1). Given n training samples, it calculates n dot-products and each dot-product takes time $O(md)$, where m and d are the number of rows and columns of the matrices. Line 4 is the eigen decomposition, which takes $O(min(m^2\ d, md^2))$. Line 5 and line 6 take $O(md)$ to update $\mathbf{Z}_2$ and $\mathbf{W}$, respectively. In this regard, the time complexity of Algorithm 1 is $O(nmd) \times K$, where K is the iteration number. As a comparison, the SVM models implemented with LIBSVM library have time complexity scales between $O(n^2)$ and $O(n^3)$ [39]. The computational cost of each iteration in SMM model is dominated by the quadratic programming with respect to $\mathbf{W}$, which takes $O(n^2md)$ time complexity. Therefore, our method is much faster than SMM owing to the computational simplicity in each iteration.

4.3. Theoretical risk analysis

In this section, we theoretically analyze the excess risk of our SSMM for regularization with the combination of nuclear norm and ℓ_1 norm. In our theoretical analysis, we assume each entity within an input matrix independently obeys the standard Gaussian distribution following the setting in [40,41].

Given the optimization problem of SSMM defined in Eq. (5), we can simply rewrite it as follows:

$$\begin{aligned} &\arg\min_{\mathbf{W},b} \sum_{i=1}^{n} h(\mathbf{W}, b, \mathbf{X}_i, y_i), \\ &\qquad \text{s.t. } ||\mathbf{W}||_1 \leq c_0, \qquad ||\mathbf{W}||_* \leq c_1, \end{aligned} \tag{13}$$

for certain constants c_0 and c_1, with $h(\mathbf{W}, b, \mathbf{X}_i, y_i) = \{1 - y_i[tr(\mathbf{W}^T\mathbf{X}_i) + b]\}_+$ as the loss term for each input matrix with parameters $\mathbf{W}$ and b. With the Karush–Kuhn–Tuck (KKT) conditions

$$y_i\{tr[\mathbf{W}^T\mathbf{X}_i] + b\} - 1 = 0, \qquad i \in \{1, \cdots, n\}, \tag{14}$$

the loss function can be further simplified as

$$\hat{h}(\mathbf{W}, \hat{\mathbf{X}}_i, y_i) = \left\{1 - y_i\left[tr(\mathbf{W}^T\hat{\mathbf{X}}_i) + \frac{1}{n}\sum_{j=1}^{n} y_j\right]\right\}_+, \tag{15}$$

which is L-Lipschitz continuous, with $\hat{\mathbf{X}}_i = \mathbf{X}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{X}_j$, as $\mathbf{X}_i$ minus the empirical mean of all given data matrices, which tends to be 0 when n is large. As a result, entities of $\hat{\mathbf{X}}_i$ are i.i.d zero mean standard normal distributed [42].

Following [41], the standard form of empirical risk without the bias term for loss function in Eq. (13) can be formulated as

$$\tilde{R}(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n} \hat{h}(\mathbf{W}, \hat{\mathbf{X}}_i, y_i), \tag{16}$$

and the expected risk is defined as

$$R(\mathbf{W}) = \mathbb{E}_{(\mathbf{X},y)\sim\mu} \hat{h}(\mathbf{W}, \hat{\mathbf{X}}_i, y_i), \tag{17}$$

with μ as the probability distribution that the data points are sampled. Here, we set $\mathbf{W}^o$ as the optimal solution of the following problem to minimize the expected risk with

$$\mathbf{W}^o = \arg\min_{\mathbf{W}} R(\mathbf{W}), \qquad \text{s.t. } ||\mathbf{W}||_1 \leq c_0, ||\mathbf{W}||_* \leq c_1, \tag{18}$$

and $\tilde{\mathbf{W}}$ as the optimal solution to minimize the empirical risk as follows

$$\tilde{\mathbf{W}} = \arg\min_{\mathbf{W}} \tilde{R}(\mathbf{W}), \qquad \text{s.t. } ||\mathbf{W}||_1 \leq c_0, ||\mathbf{W}||_* \leq c_1. \tag{19}$$

In the following theorem, we provide an upper bound of the excess risk of the SSMM method, and its proof can be found in the supplementary material.

Theorem 2. *For any $\delta \in (0, 1)$, with probability at least $1-\delta$, the excess risk of SSMM classifier, for $\mathbf{W}^o$ with rank r, is bounded as*

$$R(\tilde{\mathbf{W}}) - R(\mathbf{W}^o) \leq \frac{2L\max\left(\frac{1}{r\sqrt{d}}c_0, c_1\right)}{\sqrt{n}} \cdot (\sqrt{m} + \sqrt{d}) + \sqrt{\frac{\ln(\frac{1}{\delta})}{2n}}. \tag{20}$$

5. Experimental results

We conduct extensive experiments to evaluate our method on two important empirical applications - image classification and single trial EEG classification, in which the data are naturally represented as matrices. In order to compare the performance of vector-based classifiers and matrix-based classifiers, we set two vector-based classifiers, i.e., SVM [20] and sparse SVM (SSVM) [15], as the baseline methods.

We further compare our SSMM with several state-of-the-art matrix classifiers, including regularized GLM (RGLM) [6], bilinear SVM (BSVM) [11] and SMM [12].

5.1. Experiment settings

We first introduce the settings of our experiments.

The smooth parameter is set as $\alpha = 3$ to trade off the smoothness and computational complexity. Following the settings in [30], we set the weights ω_k to be $\frac{1}{2}$ and the relaxation parameters λ_t to be constant along iterations and equal to 1.

There are still two free parameters γ and τ involved to control the trade-off between the regularization terms and the hinge loss. Details on the selection of these two parameters will be discussed later in this section. For the sake of fair comparison, the free parameters of all competitive methods are carefully tuned in order to obtain their best classification results. For vector-based classifiers, i.e., SVM and SSVM, we reshape each input matrix into a feature vector for model training.

5.2. Image classification

We first test our approach with the competitive ones in the application of image classification, which is a fundamental tool to analyze and understand images in the area of computer vision [43]. Supervised learning approaches to image classification can extract

Fig. 4. Some sample images from INRIA Person dataset. The first row displays images with people and the second row shows people-free ones. The backgrounds of the human images are similar to people-free images, which makes the task of human detection challenging.

Table 1
Summary of four datasets.

Datasets	Dimension	Train(#pos/#neg)	Test(#pos/#neg)
INRIA	160 × 96	2416/1218	1126/453
Caltch	320 × 280	147/71	131/86
BCI IV 2a	240 × 150	72 × 9/72 × 9	72 × 9/72 × 9
BCI IV 2b	150 × 24	200 × 9/200 × 9	160 × 9/160 × 9

Table 2
Classification accuracy on INRIA Person Dataset with different features.

INRIA	SVM	RGLM	SSVM	BSVM	SMM	Ours
Gray	0.8442	0.8581	0.8771	0.8619	0.8626	**0.8866**
LBP	0.9791	0.9848	0.9816	0.9835	0.9823	**0.9861**

the spectral signatures from the training samples in order to label newly captured images.

To test the effectiveness of SSMM in this application, we apply all methods on two real-world datasets: the INRIA Person [44] and the Caltech Front Face [45]. The summary information for these two datasets is shown in Table 1. It can be observed that, when stacking the input matrices into vectors, the dimension of each sample is much higher than the number of images within the training sets, which makes the problem much more difficult. We employ the classification accuracy to evaluate the performance of each method; higher value indicates that the performance is better.

5.2.1. INRIA person dataset

The INRIA Person Dataset[2] was proposed to detect whether or not people exist in an image, which can be easily interpreted as an image classification problem. It contains 2416 images with people and 1218 people-free ones for training, and 1126 images with people and 453 people-free samples for testing. Some sample images are displayed in Fig. 4. This task is challenging because the backgrounds are similar across all samples and the distribution of human is arbitrary and without any alignment. To extract the matrix-form data, we convert each color image into a 160 × 96 gray level one and use the pixel values as an input matrix without any advanced feature extraction techniques. The performance of all approaches on the testing set is shown in Table 2. It can be observed that all matrix classification methods can beat the SVM approach, which indicates that leveraging the correlation of each image data is meaningful empirically. Also, the SSVM method can beat the traditional SVM, which shows that performing feature selection is important to filter out redundant features. It is also clear that our SSMM method achieves superior performance over all competitive ones with a significant margin. Fig. 5 further demonstrates several images that has been misclassified by other classifiers, but only classified accurately by our SSMM method. These results demonstrated that addressing both the correlation and feature selection issues with low-rank and sparse constraints on the regression matrix is effective on this application.

We also compare the proposed method with others using matrix-form features by the Local Binary Patterns (LBP) [46], an efficient local descriptor to capture fine details of human appearance and texture. In this paper, we follow the same steps as in [47] to extract the LBP for each training and testing sample. Each 160 × 96 image is first divided into 10 × 6 subregions, per size 16 × 16 pixels. Within each subregion, the histogram of 59 uniform binary patterns is computed by thresholding 8 neighboring pixels in a circle of radius 2. Thereby, the feature matrix is formed as feature-vs-space with $\mathbf{X}_i \in \mathbb{R}^{59\times 60}$. The recognition accuracy rates of different compared methods are also reported in Table 2. As is expected, all classifiers benefit from using LBP features, which is more robust to variations in illumination and other changes, and thus obtain a large scale of improvement. In addition, all useless features can not be completely eliminated by the advanced feature extraction technique, thus the proposed method still achieves the best performance benefiting from leveraging the correlation within data and selecting important features.

Fig. 6 shows the convergence process of SSMM on this dataset. It verifies that our method converges fast to the global optimal value in hundreds of iterations. Similar phenomena also occur when using SSMM in other datasets. It shows the efficiency of our method to train the SSMM classifier for real-world applications. In addition, in the testing phase, all methods are quite fast. Most methods, including our SSMM, take less than 0.001 second to classify an input sample. It fulfills the requirement of most applications. Fig. 7

5.2.2. Caltech face dataset

We further test the performance of different methods on the Caltech Face Dataset[3], which is used for gender recognition. It contains 435 images of size 592 × 896 on human face with various expressions under different illumination conditions and backgrounds. Some sample images are displayed in Fig. 7. In this dataset, it can be observed that all images share similar features in terms of human face outlines, and the difference of genders lies only in small details such as hair and eyes. In addition, to distinguish the gender of a human in each image, most of the pixels related to background are useless. Thus, it would be important to take feature selection into consideration when training a classifier.

Similar to [45], we also randomly select 147 male and 71 female images as training set, and set the rest 131 male and 86 female images as testing set. We first detect the faces by Viola–Jones face detector, which efficiently outputs a bounding box indicating

[2] http://pascal.inrialpes.fr/data/human/.

[3] http://www.vision.caltech.edu/Image_Datasets/faces/.

Fig. 5. Some cases have right results only with SSMM.

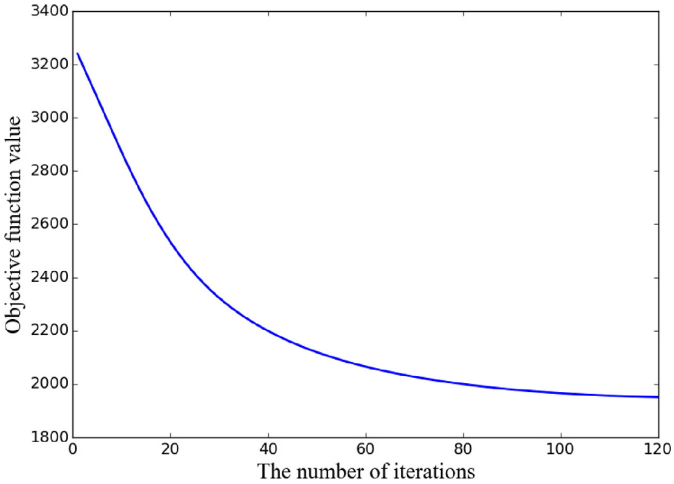


Fig. 6. Convergence process for SSMM.

Table 3
Classification performance on Caltech Face Dataset with different features.

Caltech	SVM	RGLM	SSVM	BSVM	SMM	Ours
Gray	0.9539	0.9770	0.9677	0.9862	0.9816	**0.9908**
LBP	0.9816	0.9908	0.9862	0.9908	**0.9954**	**0.9954**

the predicted face[4]. Similar to INRIA Person, we resize each face within the bounding box to size 320 × 280 and extract the gray pixel values as the input matrix-form features. The prediction results are shown in Table 3. It can be seen that the proposed SSMM achieves superior performance compared with other state-of-the-art classifiers. This is because the changing illumination or backgrounds have low-rank property, while the discriminant face features exist sparse structures. In this case, neither sparse model (*e.g.*, SSVM) nor low-rank model (*e.g.*, BSVM and SMM) is sufficient to capture the underlying structure entirely. Moreover, it also indicates that our method is robust to different disturbances like lighting, expressions and backgrounds. Because the dimension (320 × 280 = 89,600) in this dataset is much larger than the sample size (435), SVM-like matrix classifiers (e.g., BSVM, SMM, SSMM) with hinge loss are more robust for high dimensional data and have better generalization performance than RGLM. It should be noted that the proposed classifier can also reduce the computational cost in the testing phase because the regression matrix has less elements due to the sparse and low-rank structure. The number of iterations of SSMM on this dataset is 88.

We further compare the performance of different methods using the LBP features. Each 320 × 280 cropped and resized face region is first divided into a regular 10 × 10 grid of cells, per size 32 × 28 pixels. Following the same steps in Section 5.2.1, each feature matrix embedding feature-space correlationship is extracted as $\mathbf{X}_i \in \mathbb{R}^{59\times100}$. As is shown in Table 3, all classifiers benefit from using efficient LBP features and all matrix classifiers outperform the vector ones consistently. In addition, our method SSMM achieves the same competitive performance as SMM does.

5.3. Single trial EEG classification

To further evaluate our method, we apply our approach to the application of EEG data classification, which can benefit modern EEG-based Brain-Computer Interface (BCI) as a potential communication system without any requirement of peripheral muscular activity [48,49]. The BCI can extract and recognize the brain signals, and perform certain activities accordingly. This procedure highly relies on the classification of EEG data, to recognize different brain activity patterns [50]. The EEG signals can be intuitively represented as two-dimensional matrices, with high correlation among the rows and columns within each sample, which could be effectively captured by the matrix classification methods. The main challenge is that the sample size is rather small comparing with the sample dimensionality in EEG data, while there always exist redundant features within each input matrix [6]. To evaluate our method on this application, we conduct comparative experiments

[4] We use the OpenCV implementation of the Viola–Jones's face detector. Since there is only one face in each image, we reduce the false alarms by reducing the maximum of search scale iterations. The detector missed only 8 faces out of all 435 images in our experiment, and we have manually labeled these eight bounding boxes.

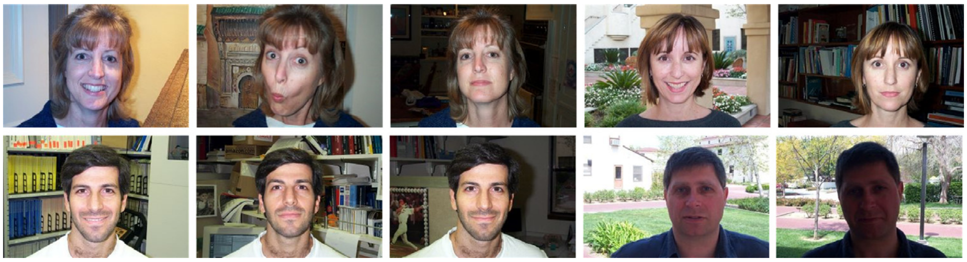
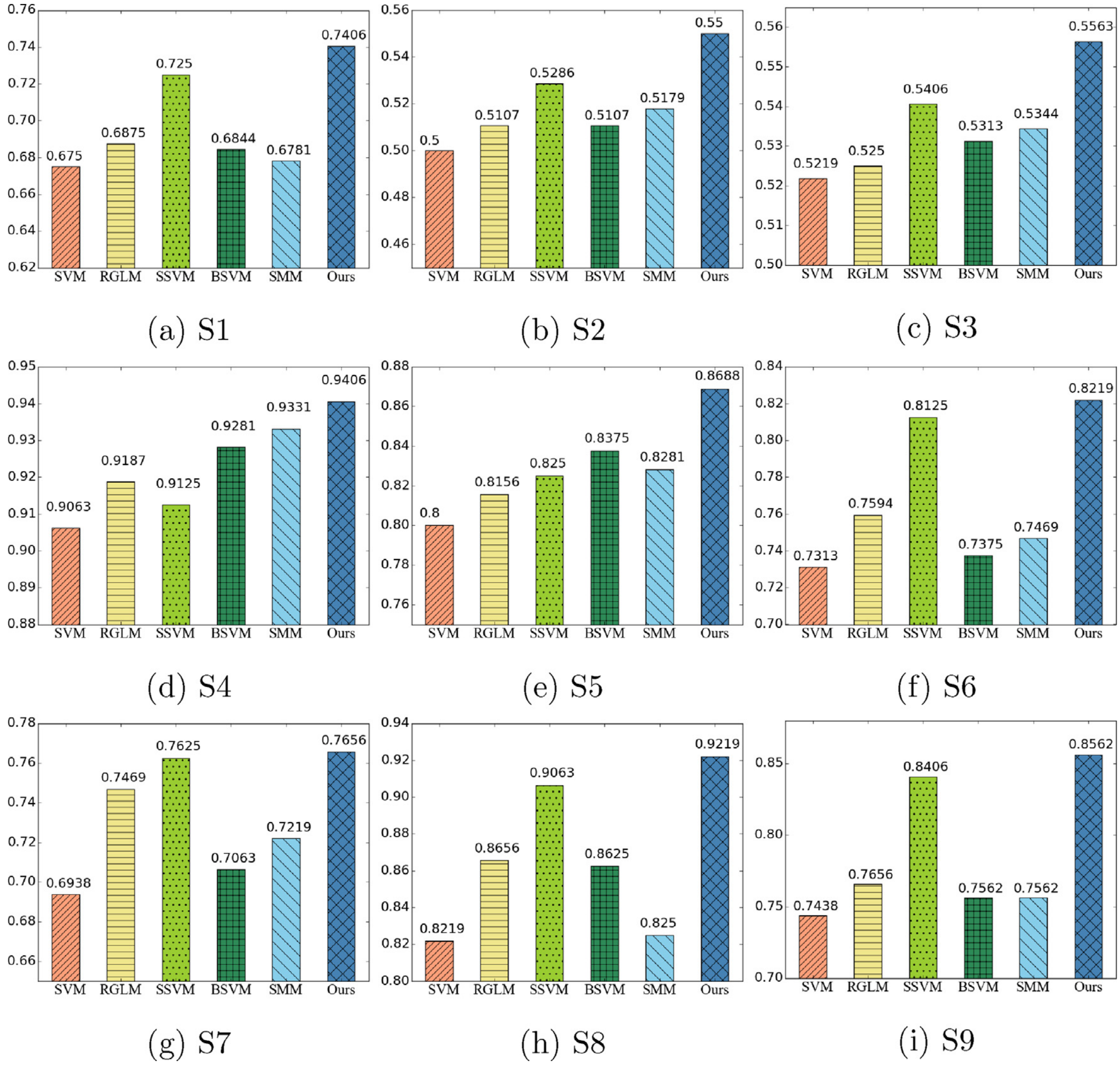


Fig. 7. Some sample images from Caltech Face dataset. The first row shows female faces and the second one shows male faces. The dataset is of high difficulty due to different face appearance, expressions, lighting conditions, and backgrounds within each class of images.



(a) S1 (b) S2 (c) S3

(d) S4 (e) S5 (f) S6

(g) S7 (h) S8 (i) S9

Fig. 8. Classification performance on Dataset 2b of BCI Competition IV.

on the Dataset 2a and 2b [51] of BCI Competition IV. The summary of these two datasets is listed in Table 1. It can been seen that, for both datasets, if each data matrix is stacked into a feature vector, the dimension of each data can highly exceed the available sample size in the training set, resulting in a challenging classification problem.

5.3.1. *Dataset 2a of BCI Competition IV*

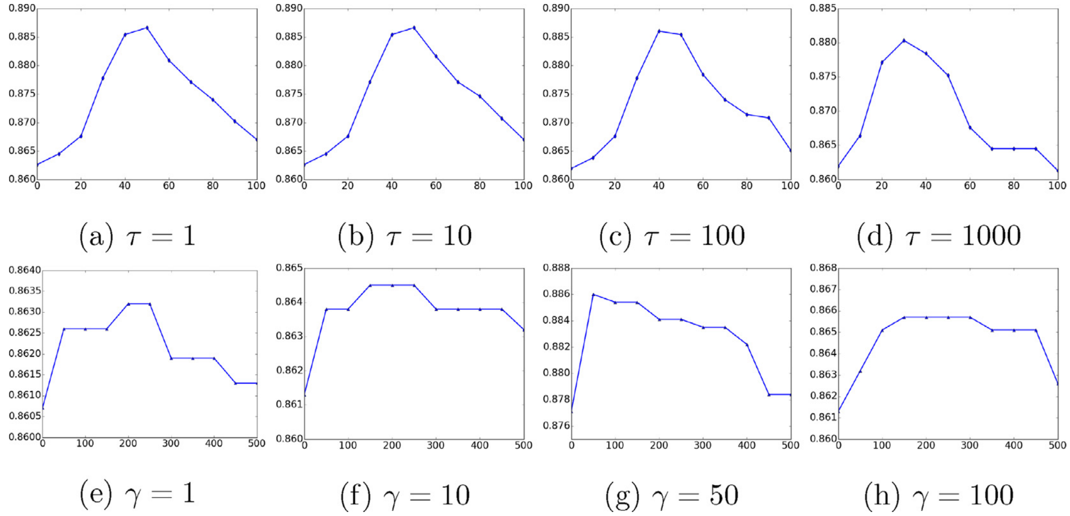
We first evaluate each method on the Dataset 2a of BCI Competition IV[5], which records single-trial EEG brain waves from nine healthy subjects performing four motor imagery tasks, namely, left-hand (L), right-hand (R), feet (F) and tongue (T). It comprises of training and testing data in two sessions conducted on different days. There are 72 trials per motor imagery task and 288 trials in total per session for each subject. This dataset can be used to train classifiers to recognize the true label of a newly captured motor imagery trial.

In this experiment, we use the following preprocessing and feature extraction procedures to obtain the matrix features for classification. Firstly, the EEG artifacts are removed with linear regression. Following [51], we employ filter banks to remove unrelated sensorimotor signals and a multi-class common spatial patterns (CSP) to select dominant channels for each motor imagery task. Then the EEG signals are down sampled from 250 Hz/s to 50 Hz/s to reduce the EEG time dimensionality and computation cost [52]. To extract the most dominant matrix-form features, we choose time-domain parameters (TDP) [53] for its good performance and low computational cost. To evaluate the single-trial binary classification performance, we separate the four-class matrix data and generate $C_4^2 = 6$ binary data subsets, namely, L-vs-R, L-vs-F, L-vs-T, R-vs-F, R-vs-T and F-vs-T. Note that, we train for all methods in the paradigm of single subject and average the prediction results of nine subjects for each subset. The testing performance is reported in Table 4. It is clear that the proposed method stably outperforms all competitive classifiers on all binary matrix subsets,

Table 4
Classification performance for Dataset 2a of BCI Competition IV.

Dataset	SVM	RGLM	SSVM	BSVM	SMM	Ours
L-vs-R	0.8048	0.8225	0.8187	0.8140	0.8102	**0.8318**
L-vs-F	0.8711	0.8892	0.8866	0.8812	0.8781	**0.8989**
L-vs-T	0.8588	0.8773	0.8650	0.8812	0.8742	**0.8951**
R-vs-F	0.8773	0.8696	0.8897	0.8873	0.8843	**0.9020**
R-vs-T	0.8688	0.8618	0.8704	0.8804	0.8773	**0.8974**
F-vs-T	0.8009	0.8130	0.8003	0.8133	0.8094	**0.8356**

[5] http://www.bbci.de/competition/iv/#dataset2a.



(a) $\tau = 1$ (b) $\tau = 10$ (c) $\tau = 100$ (d) $\tau = 1000$

(e) $\gamma = 1$ (f) $\gamma = 10$ (g) $\gamma = 50$ (h) $\gamma = 100$

Fig. 9. Classification results versus free parameters. Y axis denotes prediction accuracy, X axis denotes different values of γ (in the first row) and τ (in the second row).

which shows its strong efficiency in the task of EEG signal classification. This is due to the fact that EEG signals are usually highly correlated, and the useful features are rather sparse. It also indicates that our method can effectively capture these intrinsic structures. Taking the advantage of these features, SSMM can beat all other methods that cannot capture the low-rank and sparse features simultaneously. Over all the data subsets, our method converges to global optimal values no more than 350 iterations.

5.3.2. *Dataset 2b of BCI Competition IV*

We further test each method on the Dataset 2b of BCI Competition IV[6], which is proposed to train classifiers for the detection of motor imagery with left and right hand from nine healthy subjects. For each subject, five sessions are provided, whereby the first two sessions recorded without feedback are used for training and the last three sessions with feedback are used for testing.

To extract feature in matrix form from the raw data, we employ the same preprocessing and feature extraction techniques used for the dataset 2a of BCI IV as introduced in the previous section. The results of all algorithms on the testing set are reported in Fig. 8. It shows that all the matrix classifiers consistently outperform SVM on all subjects and our method achieves best performance compared with other matrix classifiers. This indicates that with the consideration of correlation among rows and columns, and feature selection for each data matrix, the classification results for EEG data can be enhanced. In addition, the SSVM with sparsity features can also achieve superior performance compared with matrix classifiers except ours on most of subjects, it shows that performing feature selection makes sense for the EEG classification. Due to the simultaneous consideration of both low-rank and sparse properties, our method yields competitive performance consistently for all the subjects, even for small sample problems. Note that for all subsets, our method converges to global optimums with iterations less than 240.

5.4. *Influence of free parameters*

We further test the influence of parameter settings for SSMM on matrix classification. The two free parameters τ and γ in our SSMM method are proposed to capture the correlation of data matrix and the feature selection behaviors respectively. When τ is set to be 0, our method degenerates to the SSVM; and when γ is 0, our method can be interpreted as the BSVM. To test the influence of them, we first set τ to be fixed and tune γ into different values between [0, 100]. Then, we fix the value of γ and tune the parameter τ accordingly. Fig. 9 shows the classification performance of SSMM on the INRIA dataset with different parameter settings. It can be observed that, when either τ or γ is set to be larger than 0, the performance is consistently better than when they are set to be zero. The performance is rather consistent with $\gamma \in (0, 100]$ and $\tau \in (0, 500]$, and can achieve best performance with appropriate γ/τ via cross validation. Similar phenomena also occur when using SSMM on other datasets. It shows that both the low-rank assumption and the sparse modeling of our SSMM method are effective and can enhance the classification performance. Thus, it can be indicated that our idea to consider both correlation among matrices, and perform feature selection simultaneously is reasonable in real-world applications.

6. Conclusion

In this paper, we propose a novel SSMM method for the problem of matrix classification, which can be defined as a hinge loss for model fitting, plus the combination of nuclear norm and the ℓ_1 norm. It is the first method to simultaneously capture the intrinsic structure for each matrix and select useful features for more interpretable modeling. Though the optimization problem is convex, the hinge loss, nuclear norm and ℓ_1 norm are all non-smooth, which makes it difficult to solve. We tackle this issue, and derive an efficient algorithm based on the framework of generalized forward-backward splitting to solve it. We further conduct extensive comparative experiments on real-world applications, i.e., image classification and EEG classification. For both tasks, our approach achieves the state-of-the-art performance. It shows that taking both the intrinsic structure of each input matrix and feature selection into consideration does make sense, and can benefit empirical classification tasks. It also shows the effectiveness of our SSMM method, and its promise to the real-world applications.

This paper also casts light on several future works based on the current SSMM method. To begin with, our method is designed for binary classification problem, with the assumption that labels on different samples are independent with each other. However, it is common to see multi-class classification problems for data in matrix form, or structural information involved within the label of each data sample. We are interested to take these issues into consideration to develop advanced matrix classifiers for multi-class

[6] http://www.bbci.de/competition/iv/#dataset2b.

[54] or structural classification [55]. Another issue is that, the free parameters γ, τ to control the sparseness and low-rank properties respectively for regularization are required to be fine tuned for our SSMM method. We are looking forward to developing automatic model selection techniques to choose these parameters [6].

Acknowledgments

This work is supported by the National Basic Program of China, 973 Program (Project No. 2015CB351706), the National Natural Science Foundation of China (Project No. 61233012), the Research Grants Council of the Hong Kong Special Administrative Region (Project No. CUHK 14225616) and a grant from the Guangdong Natural Science Foundation (No. 2016A030313047).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.patcog.2017.10.003.

References

[1] F. Wu, X.-Y. Jing, X. You, D. Yue, R. Hu, J.-Y. Yang, Multi-view low-rank dictionary learning for image classification, Pattern Recognit. 50 (2016) 143–154.
[2] X.-Y. Wang, T. Wang, J. Bu, Color image segmentation using pixel wise support vector machine classification, Pattern Recognit. 44 (4) (2011) 777–787.
[3] J. Bootkrajang, A. Kabán, Learning kernel logistic regression in the presence of class label noise, Pattern Recognit. 47 (11) (2014) 3641–3655.
[4] S. Sanei, J.A. Chambers, EEG Signal Processing, John Wiley & Sons, 2013.
[5] L. Wolf, H. Jhuang, T. Hazan, Modeling appearances with low-rank svm, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–6.
[6] H. Zhou, L. Li, Regularized matrix regression, J. R. Stat. Soc. 76 (2) (2014) 463–483.
[7] R. Tomioka, K. Aihara, Classifying matrices with a spectral regularization, in: Proceedings of the 24th International Conference on Machine Learning, ACM, 2007, pp. 895–902.
[8] X. Gao, L. Fan, H. Xu, A novel method for classification of matrix data using twin multiple rank SMMs, Appl. Soft Comput. 48 (2016) 546–562.
[9] M. Dyrholm, C. Christoforou, L.C. Parra, Bilinear discriminant component analysis, J. Mach. Learn. Res. 8 (May) (2007) 1097–1111.
[10] H. Pirsiavash, D. Ramanan, C.C. Fowlkes, Bilinear classifiers for visual recognition, in: Advances in Neural Information Processing Systems, 2009, pp. 1482–1490.
[11] T. Kobayashi, N. Otsu, Efficient optimization for low-rank integrated bilinear classifiers, in: European Conference on Computer Vision (ECCV), Springer, 2012, pp. 474–487.
[12] L. Luo, Y. Xie, Z. Zhang, W.-J. Li, Support matrix machines, in: The International Conference on Machine Learning (ICML), 2015.
[13] R. He, W.-S. Zheng, B.-G. Hu, X.-W. Kong, Two-stage nonnegative sparse representation for large-scale face recognition, IEEE Trans. Neural Netw. Learn. Syst. 24 (1) (2013) 35–46.
[14] Z. Han, J. Jiao, B. Zhang, Q. Ye, J. Liu, Visual object tracking via sample-based adaptive sparse representation (adaSR), Pattern Recognit. 44 (9) (2011) 2170–2183.
[15] J. Zhu, S. Rosset, T. Hastie, R. Tibshirani, 1-Norm support vector machines, Adv. Neural Inf. Process. Syst. 16 (1) (2004) 49–56.
[16] S. Aseervatham, A. Antoniadis, É. Gaussier, M. Burlet, Y. Denneulin, A sparse version of the ridge logistic regression for large-scale text categorization, Pattern Recognit. Lett. 32 (2) (2011) 101–106.
[17] H. Raguet, J. Fadili, G. Peyré, A generalized forward-backward splitting, SIAM J. Imaging Sci. 6 (3) (2013) 1199–1226.
[18] J.-F. Cai, E.J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (4) (2010) 1956–1982.
[19] E.J. Candès, B. Recht, Exact matrix completion via convex optimization, Foundations of Computational Mathematics 9 (6) (2009) 717–772.
[20] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 27.
[21] J. Qian, L. Luo, J. Yang, F. Zhang, Z. Lin, Robust nuclear norm regularized regression for face recognition with occlusion, Pattern Recognit. 48 (10) (2015) 3145–3159.
[22] J. Wang, M. Wang, X. Hu, S. Yan, Visual data denoising with a unified schatten-p norm and q norm regularized principal component pursuit, Pattern Recognit. 48 (10) (2015) 3135–3144.
[23] E.J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis? J. ACM 58 (3) (2011) 11.
[24] Q. Gu, Z. Wang, H. Liu, Low-rank and sparse structure pursuit via alternating minimization, in: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016, pp. 600–609.
[25] T. Zhou, D. Tao, Godec: randomized low-rank & sparse matrix decomposition in noisy case, in: Proceedings of the 28th International Conference on Machine Learning, 2011, pp. 33–40.
[26] A.E. Waters, A.C. Sankaranarayanan, R. Baraniuk, SpaRCS: recovering low-rank and sparse matrices from compressive measurements, in: Advances in Neural Information Processing Systems, 2011, pp. 1089–1097.
[27] N. Guan, D. Tao, Z. Luo, J. Shawe-Taylor, MahNMF: Manhattan non-negative matrix factorization, (2012) arXiv:1207.3438.
[28] T. Liu, D. Tao, On the performance of manhattan nonnegative matrix factorization, IEEE Trans. Neural Netw. Learn. Syst. 27 (9) (2016) 1851–1863.
[29] T. Liu, D. Tao, D. Xu, Dimensionality-dependent generalization bounds for k-dimensional coding schemes, Neural Comput. (2016).
[30] E. Richard, P.-A. Savalle, N. Vayatis, Estimation of simultaneously sparse and low rank matrices, (2012) arXiv:1206.6474.
[31] S. Oymak, A. Jalali, M. Fazel, Y.C. Eldar, B. Hassibi, Simultaneously structured models with application to sparse and low-rank matrices, IEEE Trans. Inf. Theory 61 (5) (2015) 2886–2908.
[32] A. Parekh, I.W. Selesnick, Improved sparse low-rank matrix estimation, (2016) arXiv:1605.00042.
[33] E. Richard, S. Gaïffas, N. Vayatis, Link prediction in graphs with autoregressive features., J. Mach. Learn. Res. 15 (1) (2014) 565–593.
[34] J.V. Shi, Y. Xu, R.G. Baraniuk, Sparse bilinear logistic regression, (2014) arXiv:1404.4104.
[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends Mach. Learn. 3 (1) (2011) 1–122.
[36] C. Chen, B. He, Y. Ye, X. Yuan, The direct extension of admm for multi-block convex minimization problems is not necessarily convergent, Math. Program. 155 (1–2) (2016) 57–79.
[37] J.D. Rennie, N. Srebro, Fast maximum margin matrix factorization for collaborative prediction, in: Proceedings of the 22nd International Conference on Machine Learning, ACM, 2005, pp. 713–719.
[38] C.L. Byrne, Iterative Optimization in Inverse Problems, CRC Press, 2014.
[39] T. Joachims, Training linear svms in linear time, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 217–226.
[40] K. Wimalawarne, R. Tomioka, M. Sugiyama, Theoretical and experimental analyses of tensor-based regression and classification, Neural Comput. 28 (4) (2016) 686–715.
[41] A. Maurer, M. Pontil, Excess risk bounds for multitask learning with trace norm regularization, in: Conference on Learning Theory (COLT), vol. 30, 2013, pp. 55–76.
[42] Y.L. Tong, The Multivariate Normal Distribution, Springer Science & Business Media, 2012.
[43] C. Barat, C. Ducottet, String representations and distances in deep convolutional neural networks for image classification, Pattern Recognit. 54 (2016) 104–115.
[44] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, IEEE, 2005, pp. 886–893.
[45] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE, 2003, pp. II–264.
[46] M. Heikkilä, M. Pietikäinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recognit. 42 (3) (2009) 425–436.
[47] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, Y. Ma, Toward a practical face recognition system: robust alignment and illumination by sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2) (2012) 372–386.
[48] F. Qi, Y. Li, W. Wu, Rstfc: a novel algorithm for spatio-temporal filtering and classification of single-trial eeg, IEEE Trans. Neural Netw. Learn. Syst. 26 (12) (2015) 3070–3082.
[49] S. Hu, H. Wang, J. Zhang, W. Kong, Y. Cao, R. Kozma, Comparison analysis: granger causality and new causality and their applications to motor imagery, IEEE Trans. Neural Netw. Learn. Syst. 27 (7) (2015) 1429–1444.
[50] H. Zhang, H. Yang, C. Guan, Bayesian learning for spatial filtering in an eeg-based brain–computer interface, IEEE Trans. Neural Netw. Learn. Syst. 24 (7) (2013) 1049–1060.
[51] K.K. Ang, Z.Y. Chin, C. Wang, C. Guan, H. Zhang, Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b, Front. Neurosci. 6 (2012) 39.
[52] F. Lotte, A tutorial on eeg signal-processing techniques for mental-state recognition in brain–computer interfaces, in: Guide to Brain-Computer Music Interfacing, Springer, 2014, pp. 133–161.
[53] C. Vidaurre, N. Krämer, B. Blankertz, A. Schlögl, Time domain parameters as a feature for eeg-based brain–computer interfaces, Neural Netw. 22 (9) (2009) 1313–1319.
[54] H. Guo, W. Wang, An active learning-based svm multi-class classification model, Pattern Recognit. 48 (5) (2015) 1577–1597.
[55] K. Kim, D. Lee, Inductive manifold learning using structured support vector machine, Pattern Recognit. 47 (1) (2014) 470–479.

Qingqing Zheng is currently a Ph.D. student in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. Her research interests include machine learning, computer vision and brain computer interfaces.

Fengyuan Zhu received the Ph.D. degree in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. His research interests include machine learning theory, computer vision, data mining.

Jing Qin is currently an assistant professor in School of Nursing, The Hong Kong Polytechnic University. His research interests include virtual/augmented reality for healthcare and medicine training, medical image processing, deep learning, visualization and human-computer interaction and health informatics.

Badong Chen is currently a professor at the Institute of Artificial Intelligence and Robotics (IAIR), Xi'an Jiaotong University. His research interests are in system identification and control, information theory, machine learning, and their applications in cognition and neuroscience.

Pheng-Ann Heng is currently a professor in the Department of Computer Science and Engineering, CUHK. He is also the director of the Research Center for Human-Computer Interaction, SIAT, Chinese Academy of Sciences. His research interests include virtual reality, visualization, medical imaging, human-computer interfaces, rendering and modeling, interactive graphics, and animation.