Pattern Recognition 78 (2018) 307–317

Contents lists available at ScienceDirect

Pattern Recognition

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

# SVM based multi-label learning with missing labels for image annotation

Yang Liu[a], Kaiwen Wen[a], Quanxue Gao[a,*], Xinbo Gao[a], Feiping Nie[b]

[a] State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
[b] Center for Optical Imagery Analysis and Learning, Northwestern Polytechnical University, Xi'an 710072, China

ARTICLE INFO

*Article history:*
Received 27 December 2016
Revised 5 November 2017
Accepted 24 January 2018
Available online 1 February 2018

*Keywords:*
Multi-label learning
Missing labels
SVM
Image annotations

ABSTRACT

Recently, multi-label learning has received much attention in the applications of image annotation and classification. However, most existing multi-label learning methods do not consider the consistency of labels, which is important in image annotation, and assume that the complete label assignment for each training image is available. In this paper, we focus on the issue of multi-label learning with missing labels, where only partial labels are available, and propose a new approach, namely SVMMN for image annotation. SVMMN integrates both example smoothness and class smoothness into the criterion function. It not only guarantees the large margin but also minimizes the number of samples that live in the large margin area. To solve SVMMN, we present an effective and efficient approximated iterative algorithm, which has good convergence. Extensive experiments on three widely used benchmark databases in image annotations illustrate that our proposed method achieves better performance than some state-of-the-art multi-label learning methods.

## 1. Introduction

Recently, multi-label learning has gradually attracted significant attention in the fields of pattern analysis and machine learning and has been widely applied to many real-world applications, such as text categorization [1–3], image annotation [4–7] and facial action unit (AU) recognition [8–10]. For traditional multi-label learning, a basic assumption is that each training sample is completely labeled. In other words, if a training sample contains the labels car, road and sky, the user should provide these concepts for the image. In many applications, however, training labels are obtained via crowd-sourcing, which typically leads to label incompleteness [11]. In general, only a partial label set is available, while the assignments with other classes are missing. For example, the image is only tagged with the label *car* and label *road* but no label *sky*. In such scenario, we can not judge whether *sky* is a proper label for the sample. It is obvious that this scenario is different from the classic multi-label learning where all labels for training samples have been given. This kind of multi-label problem is also called the missing labels problem.

Although the classic multi-label learning has gained great breakthrough [12–16], multi-label learning with missing labels still remains in a preliminary state. The recent work of positive examples and negative examples labeling heuristic (PNLH) [17] learns a standard binary classifier from only positive and unlabeled data by two steps extraction and enlargement. Bayesian framework for multi-label classification using compressed sensing (BML-CS) [18] assumes a continuous probability model over the binary labels and it can be used to solve the multi-label learning with missing labels problem. However, all of these methods assume that all labels are independent and ignore the correlation between labels. To handle this problem, semi-supervised multi-label learning (SMSE2) [19] was proposed to construct two adjacent graphs on instance level and category level respectively to characterize the intrinsic structures and integrates them into the criterion function. The weak label learning (WELL) [20] method focuses on the case where examples only have a partial set of positive labels available. Bucak et al. [4] proposed the multilabel ranking with group lasso (ML-RGL) algorithm which formulates multilabel classification as a bipartite ranking problem. Although these methods consider the correlation between labels, all of which implicitly assume that missing labels are equivalent to negative labels. It is not reasonable if such an assumption always holds in actual datasets, and whenever it does not, treating missing label in discriminately as negative labels introduce undesirable bias to the learning problem [21]. For example, for a multi-label image, “tree” and “house” are its positive labels, while “blue” and “road” are negative labels. All other labels are missing labels. It's ground-truth positive labels

[*] Corresponding author.
*E-mail addresses:* 523240290@qq.com (Y. Liu), 1780406819@qq.com (K. Wen), qxgao@xidian.edu.cn (Q. Gao).

https://doi.org/10.1016/j.patcog.2018.01.022
0031-3203/© 2018 Elsevier Ltd. All rights reserved.

are “tree”, “house”, “green” and “home”. Label “home” is not shown in the label list, but we cannot simply think that the image should have a negative value for that label, because it has a positive value in the available label “house” that has strong semantic correlation with “home”. To solve this problem, Wu et al. [21] proposed multi-label learning with missing labels (MLML) method. In MLML, the positive labels, negative labels and missing labels are assigned to different values. However, linear logistic regression used in MLML is difficult to handle data with complex distributions. Furthermore, some methods take the process of imputing missing labels as part of the prediction model training [4,22], which are limited to supervised learning over the sparsely labeled data. A few others only take label imputation as a preprocessing step [23,24], which recover labels that are not optimized for the target prediction models.

Some algorithms based on support vector machines (SVM) [25] have been proposed to solve multi-label classification problems. In [26], a ranking-SVM approach is proposed to minimize the margin and the ranking loss [27]. Li et al. [28] proposed SVM based active learning method for multi-label image classification which made instance selection decision using Max Loss and Mean Max Loss strategies and determined the predicted labels of an unlabeled instance using binary SVM classifiers. These methods have got an improvement in the field of text categorization and image annotation, but none of them takes into consideration the correlation among different class labels, which is important for classification [29,30]. Moreover, they cannot be efficient for multi-label classification with missing labels problem.

In this paper, inspired by SVM, we proposed a novel approach SVMMN (SVM-based minimum number of samples which live in the margin area) for multi-label learning. Our proposed method aims to find a mapping function such that the margin is large with the smallest number of samples which live in the margin area. We integrate the smoothness of label corresponding to both example smoothness and class smoothness, which considers the correlation between different class labels, into the criterion function. To solve SVMMN, we describe an iteratively re-weighted least squares (IRWLS) [31] method, which has good convergence. The main contributions of this paper are three-folds:

- We present a new loss function which guarantees not only the large margin with the projected data but also the minimum number of samples which live in margin area.
- Our approach takes into account both example smoothness and label consistence when learning the mapping function.
- We propose a SVM based method for multi-label learning with missing label problems. Experimental results illustrate that our approach is superior to some existing approaches.

The remainder of this paper is organized as follows. We give a brief review of SVM based multi-label learning in Section 2. In Section 3, we propose the SVMMN and solve the optimization problem through an iteratively re-weighted least squares (IRWLS) method. Section 4 presents experimental results to validate the effectiveness of our proposed method, and conclusions are drawn in Section 5.

## 2. SVM based multi-label learning

SVM [25] is a well-known and widely used discriminant classifiers that finds the optimal hyperplane to separate data patterns into binary classes. The optimal separating hyperplane can be defined as the one giving the maximum margin between the support vectors, where support vectors are the samples that lie closest to the optimal separating hyperplane. After obtaining the hyperplane, new samples can be classified by determining on which side of the hyperplane they fall.

Multi-label SVM usually uses the one-against-rest scheme to realize multi-class classification by combining predictions of multiple binary SVM classifiers. This method is a computationally efficient and conceptually simple solution for multi-label classification. Assume that the input image data are represented by a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbf{R}^{d\times n}$ with each example $\mathbf{x}_i$ as a $d$-dimensionality row vector. Each example is associated with $m$ different classes $\{c_1, \ldots, c_m\}$. Denote by $\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$ the label matrix, $\mathbf{p}_i \in \{-1, +1\}^{m\times 1}$ is a label of the $i$th example. If $p_{ki} = 1$ $(k = 1, \ldots, m)$, it indicates that the instance $\mathbf{x}_i$ is assigned into the $k$th class; if $p_{ki} = -1$, the instance does not belong to the $k$th class. For the $k$th class, the corresponding binary SVM aims to seek project vector $\mathbf{w_k}$ and bias $b_k$ by solving the following standard quadratic optimization problem.

$$\min\left(\frac{1}{2}\|\mathbf{w}_k\|^2 + C\sum_{i=1}^{n}\xi_{ki}\right) \tag{1}$$
$$s.t.\quad p_{ki}(\mathbf{w}_k^T\mathbf{x}_i + b_k) \geq 1 - \xi_{ki},\ \ \xi_{ki} \geq 0,\ \ \forall i$$

where $\xi_{ki}$ is the slack variable and $C$ is the trade-off parameter. In Eq. (1), the samples from the $k$th class form the positive class and the remaining samples form the negative class. After obtaining the weight vector $\mathbf{w}_k$ and bias $b_k$, a binary classifier associated with the $k$th class is $f_k(\mathbf{x}_i) = \mathbf{w}_k^T\mathbf{x}_i + b_k$. If $f_k(\mathbf{x}^*) > 0$, the instance belongs to the $k$th class, otherwise, the instance does not belong to the $k$th class.

SVM based multi-label classification methods obtain large margin and good performance, but they do not consider the correlation among different class labels and they cannot deal with multi-label classification with missing labels problem efficiently. To handle this problem, inspired by SVM, we propose a novel multi-label learning with missing labels in Section 3.

## 3. SVM based multi-label learning with missing labels

In this section, we propose a novel approach SVMMN for multi-label learning with missing labels. For the traditional SVM, the constraints emphasize that all of the samples are approximately correctly separated. This constraint is very strong. It results in over-fitting and reduces the performance of SVM. To relax this constraint, we minimize the number of samples that lie in the margine in SVM. It can help improve the stableness and robustness of SVM. SVMMN aims to learn a mapping function which guarantees not only the large margin with the projected data but also the minimum number of samples which live in the margin area. To further improve performance, we make use of the sample smoothness and class smoothness in estimating the missing label. At last, we integrate these aforementioned three parts into the criterion function to build the model of SVMMN and then solve the solution by an iteratively re-weighted least squares (IRWLS) algorithm.

### 3.1. Loss function

Assume that we have a set of $n$ training sample images $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$, where $\mathbf{x}_i \in \mathbf{R}^d$ $(i = 1, 2, \ldots, n)$ is the $i$th training image. Denoted by label matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]$, where $\mathbf{y}_k \in \{-1, 0, +1\}^{m\times 1}$ is a label of the $k$th sample. $y_{ik} = 1$ means that the instance $\mathbf{x}_k$ is assigned into the $i$th class, $y_{ik} = -1$ shows that the instance does not belong to the $i$th class, and $y_{ik} = 0$ corresponds to a missing label for this instance.

It is commonly known that maximum margin is very important in pattern recognition and helps improve performance of data classification [32,33], and SVM i.e., Eq. (1) is a classical and widely used approach which can obtain large margin. SVM tries to look for the maximum margin from the aspect that all samples are outside the margin area instead of guaranteeing for a minimum number

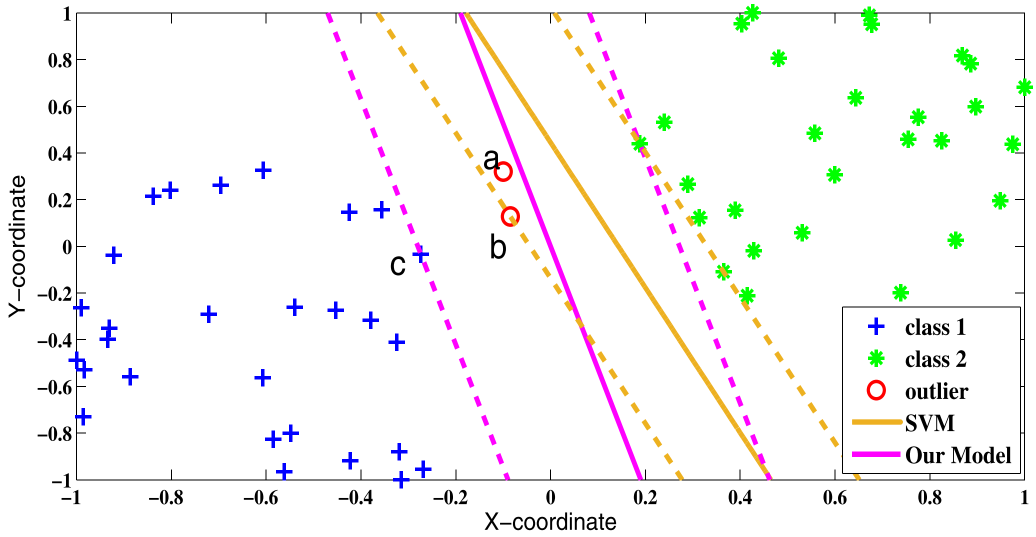

**Fig. 1.** The two classifications generated by SVM and our model.

of samples in the margin area. Moreover, a model combining SVM with sample or class smoothness directly will become very difficult to solve, which reduces the flexibility of SVM. To handle this problem, we modify the objective function (1) as follows.

$$\min \sum_{i}^{m} \frac{1}{2} err_i^T err_i + \frac{1}{2} tr\left(\mathbf{W}^T \mathbf{W}\right) \tag{2}$$

where $err_i$ is a column vector whose elements are composed of $f_{iq} - y_{iq}$, subscript $q$ is the index of sample that satisfies $f_{iq} \times y_{iq} < (1 - \xi_{iq})$, i.e. $q = \{k | 1 \le k \le n,\ f_{ik} \times y_{ik} < (1 - \xi_{ik})\}$, $f_{iq}$ is the abbreviation of the mapping function $f_{\mathbf{w}_i, b_i}(\mathbf{x}_q) = \mathbf{w}_i{}^T \mathbf{x}_q + b_i$. $\mathbf{w}_i \in \mathbf{R}^d$ represents the weight vector corresponding to the class $c_i$, and $b_i$ denotes the corresponding bias. $\mathbf{W} = [\mathbf{w}_1, \ldots \mathbf{w}_m] \in \mathbf{R}^{d \times m}$ is the weight matrix and the margin is equal to $2/tr(\mathbf{W}^T\mathbf{W})$.

It is easy to see that the first term in the objective function (2) minimizes the number of projected data in the margin area, and the second term guarantees the large margin with the weight matrix $\mathbf{W}$. Eqs. (1) and (2) have same purpose: find a maximum margin while trying to minimize the error magnitude for the points that are between the margins, but they solve the problem from different aspects. Eq. (1) wants to find the maximum margin under the condition that all samples outside the margin area. Eq. (2) tries to minimize the number of points between the margins while maximizing the margin.

To further show the difference, we construct two Gaussian classes with the covariance matrices being [0.07 −0.02; −0.02 0.17] and [0.06 0.02; 0.02 0.13], and means being [−0.63 -0.38] and [0.61 0.43], respectively. Each class consists of 28 2D samples as depicted in Fig. 1. Moreover, we also add additional outlier, i.e., [−0.10, 0.32] specified by “a” and [-0.09, 0.13] specified by “b” in Fig. 1. We use the same slack variables in SVM and our model (Eq. (2)). It is to see that traditional SVM regards the outlier “b” as a support vector, which results in over-fitting and reduces its performance. In contrast, our model selects the point “c” instead of “a” or “b” as a support vector and gets a larger and more reasonable margin than traditional SVM, which can help improving the stableness of classification. However, Eq. (2) does not consider either the correlation among different class labels or local geometric structure of samples, i.e., sample smoothness, which is defined in the following subsection.

### 3.2. Sample smoothness

It is commonly known that the local geometric structure of training data $\mathbf{X}$ can be characterized by adjacent graph $G = (\mathbf{X}, \mathbf{V})$ with vertex $\mathbf{X}$ and weight matrix $\mathbf{V}$, which is a symmetric matrix. The elements $V_{ij}$ characterize the relationship between $\mathbf{x}_i$ and $\mathbf{x}_j$ are often defined as follows.

$$V_{ij} = \exp \frac{-\left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2}{\sigma} \tag{3}$$

where $\sigma$ is a hyperparameter.

As in the previous discussion in Section 3.1, when learning the weight matrix $\mathbf{W}$, the objective function (2) cannot consider the local geometric structure of data, which can be preserved by the objective function (4).

$$\min \frac{1}{2} \sum_{i,j}^{n} V_{ij} \left\| \frac{\mathbf{z}_i}{\sqrt{d_i}} - \frac{\mathbf{z}_j}{\sqrt{d_j}} \right\|^2 \tag{4}$$

where $d_i = \sum_{j=1}^{n} V_{ij}$. $\mathbf{z}_i$ is the label vector for sample $\mathbf{x}_i$, which is estimated by our approach. $z_{ij}$ is the $j$th element of $\mathbf{z}_i$, which is defined as follows.

$$z_{ij} = \operatorname{sgn}\left(\mathbf{w}_i{}^T \mathbf{x}_j + b_i\right) \tag{5}$$

where $\operatorname{sgn}(a) = 1$ if $a > 0$ and $\operatorname{sgn}(a) = -1$ otherwise.

As can be seen in Eqs. (3) and (4), when $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighboring or similar samples, the objective function (4) with our choice of $V_{ij}$ incurs a heavy penalty if the estimated label vectors $\mathbf{z}_i$ and $\mathbf{z}_j$ are mapped far apart or not similar. Therefore, minimizing it is an attempt to ensure that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar, then $\mathbf{z}_i$ and $\mathbf{z}_j$ are similar as well. It illustrates that the objective function (4) well considers the example smoothness when learning weight vector $\mathbf{w}$ and bias $b$.

### 3.3. Class smoothness

Assume that $G_c = (\mathbf{Y}^T, \mathbf{Q})$ is a category graph which is built on label matrix $\mathbf{Y}^T$, where $\mathbf{Q}$ is a weight matrix which can be defined as follows.

$$Q_{ij} = \exp\left(-\varphi(1 - \cos(\mathbf{u}_i, \mathbf{u}_j))\right) \tag{6}$$

where $\varphi$ is a hyperparameter, $\mathbf{u}_i$ $(i = 1, 2, \ldots, m)$ is the $i$th column of $\mathbf{Y}^T$ and $\cos(\mathbf{u}_i, \mathbf{u}_j) = \langle \mathbf{u}_i, \mathbf{u}_j \rangle / \|\mathbf{u}_i\| \left\|\mathbf{u}_j\right\|$ computes the cosine similarity between $\mathbf{u}_i$ and $\mathbf{u}_j$.

The class-level smoothness of the label matrix $\mathbf{Y}$ is characterized by the following objective function.

$$\min \frac{1}{2}\sum_{i,j}^{m} Q_{ij}\left\| \frac{\tilde{\mathbf{z}}_i}{\sqrt{s_i}} - \frac{\tilde{\mathbf{z}}_j}{\sqrt{s_j}} \right\|^2 \tag{7}$$

where $\tilde{\mathbf{z}}_i (i = 1, 2, \ldots, m)$ is the $i$th column of $\mathbf{Z}^T$ and $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_n] \in \mathbf{R}^{m\times n}$, $s_i = \sum_{j=1}^{m} Q_{ij}$.

As can be seen in Eqs. (6) and (7), when $\mathbf{u_i}$ and $\mathbf{u_j}$ are similar, the objective function (7) with our choice of $Q_{ij}$ incurs a heavy penalty if the estimated label vectors $\tilde{\mathbf{z}}_\mathbf{i}$ and $\tilde{\mathbf{z}}_\mathbf{j}$ are not similar. Therefore, minimizing it is an attempt to ensure that if $\mathbf{u_i}$ and $\mathbf{u_j}$ are similar then $\tilde{\mathbf{z}}_\mathbf{i}$ and $\tilde{\mathbf{z}}_\mathbf{j}$ are similar as well. It illustrates that the objective function (7) well considers the label smoothness when learning weight vector $\mathbf{w}$ and bias $b$.

### 3.4. The objective function of SVMMN

By simple algebra, we have

$$\frac{1}{2}\sum_{i,j}^{n} V_{ij}\left\| \frac{\mathbf{z}_i}{\sqrt{d_i}} - \frac{\mathbf{z}_j}{\sqrt{d_j}} \right\|^2 = tr(\mathbf{Z}(\mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{V}\mathbf{D}^{-\frac{1}{2}})\mathbf{Z}^T) = tr(\mathbf{Z}\mathbf{L}\mathbf{Z}^T) \tag{8}$$

where $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{V}\mathbf{D}^{-\frac{1}{2}}$ is a symmetric matrix, which is called the normalized Laplacian matrix of graph $G$. $\mathbf{D} = diag(d_1, \ldots, d_n)$ is a diagonal matrix.

$$\frac{1}{2}\sum_{i,j}^{m} Q_{ij}\left\| \frac{\tilde{\mathbf{z}}_i}{\sqrt{s_i}} - \frac{\tilde{\mathbf{z}}_j}{\sqrt{s_j}} \right\|^2 = tr(\mathbf{Z}^T(\mathbf{I} - \tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{Q}\tilde{\mathbf{D}}^{-\frac{1}{2}})\mathbf{Z}) = tr(\mathbf{Z}^T\mathbf{H}\mathbf{Z}) \tag{9}$$

where $\mathbf{H} = \mathbf{I} - \tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{Q}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ is a symmetric matrix, which is called normalized Laplacian matrix of graph $G_c$. $\tilde{\mathbf{D}} = diag(s_1, \ldots, s_m)$ is a diagonal matrix.

As in the previous discussion, SVMMN aims to seek mapping function which can not only guarantee the large margin and minimum number of samples which live in the margin area but also considers both example smoothness and label smoothness. Substituting Eqs. (8) and (9) into the objective functions (4) and (7), respectively, and then integrating them into the objective function (2), the objective function of SVMMN becomes

$$\min \sum_{i}^{m} \frac{1}{2}{err_i}^T err_i + \frac{1}{2}tr\left(\mathbf{W}^T\mathbf{W}\right) + \beta tr\left(\mathbf{Z}\mathbf{L}\mathbf{Z}^T\right) + \gamma tr\left(\mathbf{Z}^T\mathbf{H}\mathbf{Z}\right) \tag{10}$$

where $\beta$ and $\gamma$ are two nonnegative constants that balance example smoothness and the label smoothness, and can be tuned by cross validation.

In the objective function (10), $\mathbf{Z}$ relates to the sign function, which is non-differentiable. This makes the numerical optimization of the objective function (10) be difficult. To handle this problem, we give a relaxation to the sign function as follows.

$$z_{ij} = \mathrm{sgn}\left(\mathbf{w}_i{}^T\mathbf{x}_j + b_i\right) \approx 2\sigma\left(\tau\left(\mathbf{w}_i{}^T\mathbf{x}_j + b_i\right)\right) - 1 \in [-1, 1] \tag{11}$$

where $\sigma(a) = 1/(1 + \exp(-a))$ denotes the sigmoid function. $\tau \geq 1$ is a parameter. When $\tau = 10$, the relaxation is illustrated in Fig. 2.

Now, we consider how to solve the objective function (10). At first, let

$$\mathbf{w}_i = \begin{bmatrix} b_i \\ \mathbf{w}_i \end{bmatrix}, \mathbf{x}_j = \begin{bmatrix} 1 \\ \mathbf{x}_j \end{bmatrix} \tag{12}$$

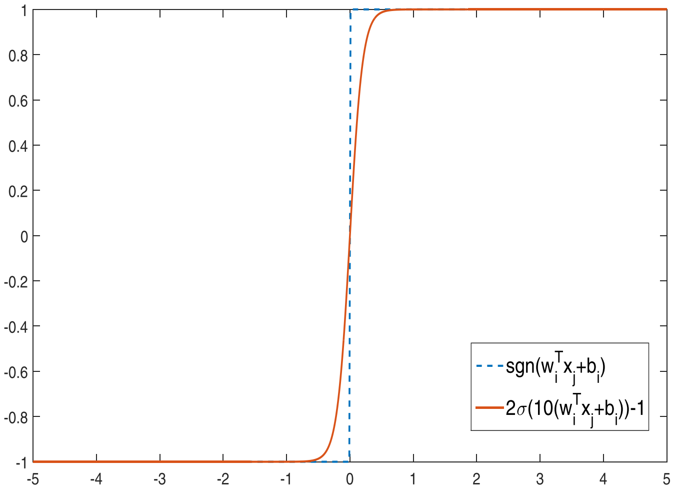

**Fig. 2.** The sign function and its relaxation.

Then the mapping function becomes $f_{\mathbf{w}_i,b_i}(\mathbf{x}_j) = \mathbf{w}_i{}^T\mathbf{x}_j$. Then we seek the solution of the following problem to optimize $\mathbf{w}_i$, .

$$\min_{\mathbf{w}_i}\left\{\frac{1}{2}err_i^T err_i + \frac{1}{2}\|\mathbf{w}_i\|^2 + \beta\frac{1}{2}\sum_{j,r}^{n} V_{jr}\left(\frac{z_{ij}}{\sqrt{d_j}} - \frac{z_{ir}}{\sqrt{d_r}}\right)^2 \right.$$
$$\left. + \gamma\frac{1}{2}\sum_{r\neq i}^{m}\sum_{j}^{n} Q_{ir}\left(\frac{z_{ij}}{\sqrt{s_i}} - \frac{z_{rj}}{\sqrt{s_r}}\right)^2\right\} \tag{13}$$

Thus $\mathbf{w}_i$ can be learned iteratively by solving Eq. (13). To simplify derivation, we denote the four terms in the objective function (13) as $A$, $B$, $C$, and $D$, respectively. Their gradients *w.r.t.* $\mathbf{w}_i$ are as follows.

$$\frac{\partial A}{\partial \mathbf{w}_i} = \widehat{\mathbf{X}} \cdot err_i \tag{14}$$

where $\widehat{\mathbf{X}} = \{\mathbf{x}_r | f_{ir} \times y_{ir} < (1 - \xi_{ir})\}$.

$$\frac{\partial B}{\partial \mathbf{w}_i} = \mathbf{w}_i \tag{15}$$

$$\frac{\partial C}{\partial \mathbf{w}_i} = \beta\sum_{j,r}^{n} V_{jr}\left(\frac{2\sigma_{ij} - 1}{\sqrt{d_j}} - \frac{2\sigma_{rj} - 1}{\sqrt{d_r}}\right)\left(\frac{2\sum_{j}^{n}\mathbf{x}_j \cdot \sigma_{ij}(1 - \sigma_{ij})}{\sqrt{d_j}} \right.$$
$$\left. - \frac{2\sum_{r}^{n}\mathbf{x}_r \cdot \sigma_{ir}(1 - \sigma_{ir})}{\sqrt{d_r}}\right) \tag{16}$$
$$= \frac{4\beta\sum_{j}^{n}\mathbf{x}_j \cdot \sigma_{ij}(1 - \sigma_{ij})}{\sqrt{d_j}}\left(\frac{2\sigma_{ij} - 1}{\sqrt{d_j}}\sum_{r}^{n} V_{jr} - \sum_{r}^{n}\frac{V_{jr}\left(2\sigma_{rj} - 1\right)}{\sqrt{d_r}}\right)$$

where $\sigma_{ij}$ is the shorthand notation for the function $\sigma(\tau\mathbf{w}_i{}^T\mathbf{x}_j)$.

$$\frac{\partial D}{\partial \mathbf{w}_i} = \gamma\sum_{r\neq i}^{m}\sum_{j}^{n} Q_{ir}\left(\frac{2\sigma_{ij} - 1}{\sqrt{s_i}} - \frac{2\sigma_{rj} - 1}{\sqrt{s_r}}\right)\frac{2\sum_{j}^{n}\mathbf{x}_j \cdot \sigma_{ij}(1 - \sigma_{ij})}{\sqrt{s_r}} \tag{17}$$
$$= \frac{2\gamma\sum_{j=1}^{n}\mathbf{x}_j \cdot \sigma_{ij}(1 - \sigma_{ij})}{\sqrt{s_i}}\left(\frac{2\sigma_{ij} - 1}{\sqrt{s_i}}\sum_{r\neq i}^{m} Q_{ir} - \sum_{r\neq i}^{m}\frac{Q_{ir}\left(2\sigma_{rj} - 1\right)}{\sqrt{s_r}}\right)$$

**Table 1**
Data statistics.

| Data set | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 |
|---|---|---|---|---|
| Example | 6775 | 7155 | 824 | 5495 |
| Class | 34 | 22 | 15 | 372 |
| Feature | 1000 | 1000 | 265 | 101938 |
| Avg. posi.-class/example | 3.81 | 1.46 | 6.79 | 13.26 |
| Posi. Label proportion(%) | 11.21 | 6.64 | 45.27 | 3.56 |

According to $\nabla \mathbf{w}_i = \frac{\partial A}{\partial \mathbf{w}_i} + \frac{\partial B}{\partial \mathbf{w}_i} + \frac{\partial C}{\partial \mathbf{w}_i} + \frac{\partial D}{\partial \mathbf{w}_i}$, we have

$$\nabla \mathbf{w}_i = \widehat{\mathbf{X}} \cdot err_i + \mathbf{w}_i + 2\sum_{j=1}^{n} \mathbf{x}_j \cdot \sigma_{ij}(1-\sigma_{ij}) \cdot \left( \frac{2\beta}{\sqrt{d_j}} \cdot \left[ \frac{2\sigma_{ij}-1}{\sqrt{d_j}} \sum_{r}^{n} V_{jr} - \sum_{r}^{n} \frac{V_{jr}(2\sigma_{rj}-1)}{\sqrt{d_r}} \right] + \frac{\gamma}{\sqrt{S_i}} \left[ \frac{2\sigma_{ij}-1}{\sqrt{S_i}} \sum_{r\neq i}^{m} Q_{ir} - \sum_{r\neq i}^{m} \frac{Q_{ir}(2\sigma_{rj}-1)}{\sqrt{S_r}} \right] \right) \tag{18}$$

Then, in the $t$th iteration, $\mathbf{w}_i$ is updated with gradient descent as follows.

$$\mathbf{w}_i^{(t+1)} \leftarrow \mathbf{w}_i^t - \alpha_t \nabla \mathbf{w}_i^t \tag{19}$$

where the step size $\alpha_t$ is determined by an optimal step search based on the Armijo rule [34].

## 4. Experiments

We validate SVMMN on four benchmark datasets(ESP Game [35], MIR Flickr [36], NUS-WIDE-Lite [37], Wiki10 [38]) and compare it with the recently proposed algorithms (MLML [21] and SLEEC [15]) and two popular methods (RBF kernel SVM and logistic regression with $l_1$ norm).In RBF kernel SVM and logistic regression, images, whose labels are known in training set, are used to learn classification hyperplane in the experiments. In our projective function, i.e., in Eq. (10), parameters $\beta$ and $\gamma$ are selected intervenient $[10^{-2}, 10^2]$ and $[10^{-2}, 10^2]$ via cross validation, respectively. To handle the data with more complex distributions, our model also be kernelized by RBF kernel. Besides, in order to evaluate the influence of example smoothness and class smoothness, we respectively set $\beta = 0$ and $\gamma = 0$ in Eq. (10), which denote by SVMMN$_{\beta=0}$ and SVMMN$_{\gamma=0}$ in our experiments. We set $\tau = 1$ in Eq. (11). The reason why we choose this value is discussed in Section 4.6. In addition, all experiments are performed on the Windows-7 operating systems (Intel Core i5-6500 CPU @ 3.40 GHz 8 GB RAM).

### 4.1. Datasets

We use the same setting as that in [4], i.e., some images with few classes or positive labels are removed in the experiments. To be specific, images, which have more than 5 positive labels with more than 300 samples per class, are selected as gallery in the ESP Game dataset [35]. Then, the gallery contains 6775 images sampled from 34 classes. In the MIR Flickr dataset [36], we select images, which have more than 3 positive labels, and classes, which have more than 300 samples, as gallery. Then, the gallery includes 7155 images sampled from 22 classes. In the NUS-WIDE-Lite dataset [37], we select images, which have than 5 images, classes, which have 150 samples, as gallery. Then, this gallery includes 824 images sampled from 15 classes. Moreover, we represent each image by 1000-dimensional vectors, which are composed of SIFT based Bag-of-words [39], in the ESP Game and MIR Flickr datasets. In the NUS-WIDE-Lite gallery, we use a 265-dimensional vectors to represent each image. Specifically, each feature vector consists of 128-dimensional wavelet texture, 73-dimensional edge direction histogram, and 64-dimensional color histogram. In order to verify the performance of our proposed algorithm on the large dataset, we do some experiments in Wiki10 dataset [38]. However, affected by experimental equipment, we select samples, which have more than 24 positive labels, and classes, which have more than 50 samples, as gallery which includes 5495 samples from 372 classes. To decrease computational burden, we employ PCA to recduce dimensionality to 100, which is the same as that in [21]. In our experiments, the ratios of positive labels to the whole ground-truth labels are respectively 11.21%, 6.64%, 45.27%, 3.56% in the ESP, MIR Flickr, NUS-WIDE-Lite and Wiki10 datasets. The average numbers of labels per sample in the ESP, MIR Flickr, NUS-WIDE-Lite and Wiki10 are 3.81, 1.46, 6.79 and 13.26, respectively. We randomly select 6100 images and 3623 images for training data and the remaining images for testing in the ESP gallery and MIR Flickr gallery, respectively. Table 1 lists the statistics of the afore mentioned three databases.

In NUS-WIDE-Lite dataset, we use 5-fold cross validation to validate the efficiency of our proposed method. Specifically, this database is randomly divided into 5 uniform folds, and 4 folds are used for training while the remaining one is used for testing. Consequently we obtain 5 results. We further repeat this process 2 runs to get different partitions. Finally, we have 10 results in this database and show the mean and corresponding standard deviation in Table 4. Moreover, to further validate the efficiency of our proposed method in the missing labels case, we construct the training data such that the portions of the labeled data vary from 20% (i.e., 80% missing labels) to 100% (i.e., no missing labels) for all databases. We repeat this process 10 times to get different missing labels and show the mean value and the corresponding standard deviation.

### 4.2. Evaluation metrics

In our experiments, two of the usually used metrics, i.e., average precision (AP) [40] and area under ROC curve (AUC) [4] are employed as evaluation criteria for multi-label ranking. AP can be calculated by the following formulation [40].

$$AP = \frac{1}{n}\sum_{i}^{n} \frac{1}{|S^i|} \sum_{s_r \in S^i} \frac{\left|\left\{ s_t \in S^i | rank(\mathbf{x}_i, s_t) < rank(\mathbf{x}_i, s_r) \right\}\right|}{rank(\mathbf{x}_i, s_r)} \tag{20}$$

where $S^i$ is the true positive label set of sample $\mathbf{x}_i$, and $rank(\mathbf{x}_i, s_r)$ denotes the rank of class $s_t$ in the ranking list of $\mathbf{x}_i$. The lager AP value is, the better classification performance is.

*AUC*: AUC [4] is the area under the ROC curve [41] of all classes. The larger AUC score is, the better performance is. The ROC curve characterizes the variation between true positive rate and false positive rate. The first $k$ labels, which varies from 1 to number of classes, are viewed as positive in the label ranking list, while the others are viewed as false. We take the mean for the AUC of more than one ROC curve, i,e, get the average value of this ten times result.

**Table 2**
AUC and AP results on the ESP Game dataset (mean(std) %). The best result in each column is highlighted in bold.

| Algorithms | AUC | | | | AP | | | |
|---|---|---|---|---|---|---|---|---|
| | 20% | 40% | 80% | 100% | 20% | 40% | 80% | 100% |
| Logistic | 70.31 | 70.51 | 71.21 | 71.28 | 35.11 | 35.19 | 35.91 | 35.57 |
| | (0.63) | (0.22) | (0.96) | (0.00) | (0.51) | (0.88) | (0.39) | (0.00) |
| KSVM | 69.62 | 69.96 | 70.02 | 69.95 | 35.32 | 35.64 | 36.07 | 36.18 |
| | (0.19) | (0.64) | (0.20) | (0.00) | (0.35) | (0.36) | (0.22) | (0.00) |
| MLML | 72.12 | 72.59 | 73.82 | 73.61 | 36.61 | 36.01 | 37.52 | 37.77 |
| | (0.76) | (0.22) | (0.39) | (0.00) | (0.59) | (0.66) | (0.23) | (0.00) |
| SLEEC | 56.99 | 59.50 | 62.20 | 69.10 | 17.19 | 18.47 | 19.80 | 30.07 |
| | (0.09) | (0.01) | (0.01) | (0.00) | (0.06) | (0.03) | (0.02) | (0.00) |
| SVMMN$_{\beta=0}$ | 74.14 | 74.78 | 75.17 | 75.30 | 38.90 | 39.41 | **39.68** | 39.69 |
| | (0.41) | (0.19) | (0.26) | (0.00) | (0.29) | (0.65) | **(0.37)** | (0.00) |
| SVMMN$_{\gamma=0}$ | 74.32 | 74.96 | 75.20 | 75.25 | 38.89 | 39.31 | 39.63 | 39.65 |
| | (0.55) | (0.21) | (0.51) | (0.00) | (0.36) | (0.51) | (0.29) | (0.00) |
| SVMMN | 73.45 | 74.42 | 75.19 | 75.31 | 38.41 | 39.12 | 39.54 | **39.71** |
| | (0.40) | (0.26) | (0.37) | (0.00) | (0.22) | (0.36) | (0.25) | **(0.00)** |
| KSVMMN | **74.90** | **75.36** | **75.41** | **75.54** | **39.08** | **39.52** | 39.53 | 39.60 |
| | **(0.30)** | **(0.08)** | **(0.07)** | **(0.00)** | **(0.47)** | **(0.22)** | (0.14) | (0.00) |

**Table 3**
AUC and AP results on the MIR Flickr dataset (mean(std) %). The best result in each column is highlighted in bold.

| Algorithms | AUC | | | | AP | | | |
|---|---|---|---|---|---|---|---|---|
| | 20% | 40% | 80% | 100% | 20% | 40% | 80% | 100% |
| Logistic | 59.61 | 59.77 | 59.94 | 59.96 | **27.45** | 27.56 | 27.58 | 27.65 |
| | (0.55) | (0.13) | (0.91) | (0.00) | **(0.50)** | (0.63) | (0.35) | (0.00) |
| KSVM | 56.65 | 57.06 | 57.58 | 57.66 | 22.83 | 22.93 | 23.24 | 23.19 |
| | (1.12) | (1.00) | (0.51) | (0.00) | (0.75) | (0.55) | (0.41) | (0.00) |
| MLML | 59.43 | 58.47 | 57.92 | 58.01 | 26.20 | 23.22 | 20.65 | 21.59 |
| | (0.43) | (0.32) | (0.40) | (0.00) | (0.50) | (0.43) | (0.20) | (0.00) |
| SLEEC | 55.15 | 56.16 | 55.93 | 57.15 | 20.08 | 21.19 | 20.75 | 23.31 |
| | (0.16) | (0.07) | (0.23) | (0.00) | (0.19) | (0.06) | (0.15) | (0.00) |
| SVMMN$_{\beta=0}$ | 59.95 | 61.52 | 62.88 | 63.55 | 26.21 | 26.71 | 27.83 | 28.11 |
| | (0.39) | (0.29) | (0.31) | (0.00) | (0.37) | (0.46) | (0.39) | (0.00) |
| SVMMN$_{\gamma=0}$ | 60.75 | 61.81 | 63.03 | 63.50 | 26.37 | 27.30 | 27.69 | 28.00 |
| | (0.46) | (0.19) | (0.22) | (0.00) | (0.29) | (0.36) | (0.24) | (0.00) |
| SVMMN | 60.69 | 62.16 | **63.35** | **63.56** | 26.92 | 27.36 | 28.11 | 28.13 |
| | (0.33) | (0.29) | **(0.34)** | **(0.00)** | (0.19) | (0.35) | (0.21) | (0.00) |
| KSVMMN | **61.24** | **62.50** | 63.10 | 63.45 | 27.01 | **27.93** | **28.18** | **28.62** |
| | **(0.77)** | **(0.16)** | (0.20) | (0.00) | (0.85) | **(0.74)** | **(0.36)** | **(0.00)** |

### *4.3. Results and analyses*

The annotation results are shown in Tables 2–5 in the aforementioned databases, respectively. As can be seen in the experimental results, logistic, KSVM (RBF kernel SVM) and SLEEC are overall inferior to MLML and our methods SVMMN and KSVMMN (RBF kernel SVMMN). The reason may be that logistic, KSVM and SLEEC ignore the information embedded in unlabeled data, which is important for multi-label learning. MLML is inferior to SVMMN and KSVMMN. This is probably because that in the criterion function, MLML employs logistic as the loss function. It is easy to see that, compared with KSVM, logistic function cannot guarantee the maximum margin, which encodes discriminative information for multi-label learning. In most cases, our method SVMMN is superior to the other methods (logistic regression, KSVM, MLML and SLEEC) when AUC and AP are employed as evaluation criterion. This is probably due to the fact that SVMMN obtains a large margin and simultaneously takes into account the information embedded in unlabeled data. SVMMN and KSVMMN are inferior to logistic when the labeled positive data is small in MIR Flickr and NUS-WIDE-Lite databases. This is probably because that the similarity between labels is not well estimated in this case. On the Wiki10 dataset, the performance of SVMMN and KSVMMN are better than other methods in the cases of high label proportions, while are worse when the label proportion is low. This is probably because when the label proportion is low, the computed class correlations are likely to be far from the ground-truth class correlations.

The comparisons among KSVMMN, SVMMN, SVMMN$_{\beta=0}$ and SVMMN$_{\gamma=0}$ are also presented. In the MIR Flickr database, KSVMMN and SVMMN overall have the best result with $\beta \neq 0$ and $\gamma \neq 0$. It indicates that both smoothness of samples and smoothness of classes are important for multi-label learning. However, in NUS-WIDE-Lite and ESP Game databases, KSVMMN and SVMMN have the best result with $\beta$=0 or $\gamma$=0. It indicates that if we do not well characterize the class smoothness or example smoothness, they will have a bad influence for multi-label learning. The unreliable computed class correlations while low label proportions may also lead to unsatisfactory influence of class smoothness.

Table 6 shows some annotation results in ESP Game and MIR Flickr datasets. These two examples demonstrate that the ground-truth label sets do not provide all semantic information related to the images. Specifically, label “eye” (exists in the prediction of SVM, MLML and SVMMN) on the first example and label “sky” and “clouds” (exist in the prediction of logistic regression, KSVM and SVMMN) on the second example also exist in the respective images. In addition, SVMMN can identify more correct labels than other methods, which further approves the validity and meaningfulness of our proposed method.

**Table 4**
AUC and AP results on the NUS-WIDE-Lite dataset (mean(std) %). The best result in each column is highlighted in bold.

| Algorithms | AUC | | | | AP | | | |
|---|---|---|---|---|---|---|---|---|
| | 20% | 40% | 80% | 100% | 20% | 40% | 80% | 100% |
| Logistic | 78.64 | 79.32 | 80.28 | 80.57 | 83.13 | 83.51 | 83.95 | 81.08 |
| | (0.10) | (0.12) | (0.06) | (0.01) | (0.03) | (0.10) | (0.02) | (0.03) |
| KSVM | 81.81 | 81.04 | 82.55 | 82.79 | 82.88 | 82.06 | 82.84 | 82.20 |
| | (2.45) | (2.57) | (3.36) | (2.59) | (2.21) | (2.58) | (3.36) | (2.96) |
| MLML | 78.57 | 79.54 | 82.47 | 82.53 | 83.03 | 83.41 | 84.04 | 83.46 |
| | (0.06) | (0.12) | (0.07) | (0.06) | (0.09) | (0.10) | (0.05) | (0.06) |
| SLEEC | 69.90 | 71.92 | 73.75 | 76.46 | 76.42 | 77.13 | 79.21 | 80.88 |
| | (0.09) | (0.47) | (0.99) | (1.32) | (3.03) | (1.05) | (0.50) | (3.01) |
| SVMMN$_{\beta=0}$ | 81.51 | 82.23 | 82.85 | 82.95 | 82.32 | 83.58 | 84.30 | 84.52 |
| | (0.11) | (0.09) | (0.06) | (0.01) | (0.09) | (0.06) | (0.05) | (0.02) |
| SVMMN$_{\gamma=0}$ | 81.70 | 82.61 | 81.92 | 82.10 | 82.49 | 83.94 | 83.54 | 83.91 |
| | (0.05) | (0.01) | (0.03) | (0.02) | (0.12) | (0.11) | (0.09) | (0.05) |
| SVMMN | 81.85 | 81.92 | 83.86 | 82.42 | 82.14 | 83.52 | 85.20 | 83.93 |
| | (0.05) | (0.08) | (0.07) | (0.03) | (0.11) | (0.07) | (0.03) | (0.05) |
| KSVMMN | **84.59** | **85.23** | **85.77** | **86.44** | **85.19** | **86.12** | **86.66** | **87.35** |
| | **(0.56)** | **0.27** | **(0.18)** | **(0.02)** | **(0.53)** | **(0.38)** | **(0.29)** | **(0.05)** |

**Table 5**
AUC and AP results on the Wiki10 dataset (mean(std) %). The best result in each column is highlighted in bold.

| Algorithms | AUC | | | | AP | | | |
|---|---|---|---|---|---|---|---|---|
| | 20% | 40% | 80% | 100% | 20% | 40% | 80% | 100% |
| Logistic | **75.24** | 76.25 | 76.30 | 75.93 | 26.31 | 26.50 | 27.85 | 28.55 |
| | **(1.28)** | (0.48) | (0.32) | (0.00) | (3.03) | (2.17) | (1.96) | (0.00) |
| KSVM | 68.73 | 69.20 | 69.02 | 68.74 | 25.76 | 26.76 | 27.02 | 27.03 |
| | (1.19) | (0.42) | (0.52) | (0.00) | (1.07) | (0.83) | (0.72) | (0.00) |
| MLML | 74.13 | 75.10 | 76.38 | 76.89 | 29.16 | 29.46 | 29.70 | 29.79 |
| | (1.20) | (1.40) | (0.08) | (0.00) | (1.10) | (0.60) | (0.02) | (0.00) |
| SLEEC | 62.80 | 64.23 | 67.44 | 70.88 | 19.63 | 20.31 | 22.66 | 25.17 |
| | (0.18) | (0.01) | (0.03) | (0.00) | (0.19) | (0.02) | (0.11) | (0.00) |
| SVMMN$_{\beta=0}$ | 73.68 | 76.25 | 78.76 | 79.79 | 28.72 | 30.87 | 32.55 | 33.19 |
| | (0.46) | (0.08) | (0.10) | (0.01) | (0.52) | (0.18) | (0.04) | (0.00) |
| SVMMN$_{\gamma=0}$ | 73.21 | 76.15 | 78.36 | 79.42 | 28.47 | 30.71 | 31.93 | 32.45 |
| | (1.00) | (0.15) | (0.06) | (0.00) | (1.27) | (0.27) | (0.05) | (0.00) |
| SVMMN | 73.65 | 75.81 | 78.75 | 79.80 | 28.82 | 30.51 | **32.58** | 33.20 |
| | (0.89) | (0.83) | (0.14) | (0.00) | (0.86) | (0.91) | **(0.07)** | (0.00) |
| KSVMMN | 74.25 | **76.99** | **78.92** | **79.80** | **29.33** | **30.94** | 32.33 | **33.33** |
| | (0.14) | **(0.32)** | **(0.07)** | **(0.00)** | **(0.25)** | **(0.13)** | (0.06) | **(0.00)** |

**Table 6**
The result of the predicted positive labels using different methods in ESP Game and MIR Flickr datasets. The correct labels are highlighted in bold.

| Images | Ground-truth | Logistic | KSVM | MLML | SVMMN |
|---|---|---|---|---|---|
| | **black,brown, girl,hair, smile,white, woman** | **black**,blue, man,people, red,**white, woman** | **black,eye**, glasses, man, **woman**, old,**smile** | **black,eye, girl,hair**, man,**white, woman** | **black, girl,hair, eye**,man, **smile, white,woman** |
| | **explore, blue,nikon, green,macro, flower** | **explore, sky,nikon, blue**,canon, water, clouds | **sky,nikon, blue**,night, **green**,clouds, **flower** | **explore**, portrait, sunset, **nikon,macro**, landscape | **explore, sky,nikon, blue**,clouds, **macro,flower** |

### 4.4. Annotation accuracy of each class

In the aforementioned experiments, evaluations are based on the ranking results, in other words, we only evaluate the whole performance of the proposed model. In the following experiments, we show the classification performance of each class via Top-5 precision, Top-5 $F_1$ and P@k measures, in the case of 100% label proportion. Specifically, the first 5 classes in the ranking list of each testing image are assigned to be positive while all others be negative. As a result, we obtain a discrete label matrix by the precision and $F_1$ measure. Precision at k (P@k) which considers only the results in the top $k$ positive predictions has been widely employed as evaluation criteria for multi-label ranking [15,42]. Given the ground truth label vector $\mathbf{y} \in \{+1, -1\}^m$ and a prediction $\widehat{\mathbf{y}} \in \mathbf{R}^m$, the precision at $k$ can be calculated as:

$$P@k(\widehat{\mathbf{y}}, \mathbf{y}) = \frac{1}{k} \sum_{i \in rank_k(\widehat{\mathbf{y}})} \mathbf{y}_i \tag{21}$$

According to Eq. (21), we can see that P@k is equal to Top-5 precision when $k = 5$. Repeating the aforementioned process two times, we output the mean as the final Top-5 precision, Top-5 $F_1$ and P@k ($k = 1,3$) measure results. The average Top-5 precision and

**Table 7**
Comparisons on the average Top-5 precision (%) and Top-5 $F_1$ measure (%) of each method on four datasets. The best result in each column is highlighted in bold.

| Algorithms | Top-5 precision | | | | Top-5 $F_1$ measure | | | |
|---|---|---|---|---|---|---|---|---|
| | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 |
| Logistic | 25.54 | 9.45 | 80.92 | 44.82 | 28.11 | 12.94 | 68.57 | 24.60 |
| KSVM | 26.28 | 7.98 | 82.48 | 45.63 | 28.57 | 10.76 | 70.05 | 25.18 |
| MLML | 28.56 | 9.25 | 78.04 | 29.38 | 30.88 | 12.73 | 66.29 | 25.79 |
| SLEEC | 27.32 | 8.25 | 81.44 | **50.44** | 29.75 | 11.13 | 69.49 | **27.81** |
| SVMMN$_{\beta=0}$ | 28.98 | 9.60 | 82.35 | 50.17 | 31.50 | 12.94 | 69.97 | 27.44 |
| SVMMN$_{\gamma=0}$ | 28.92 | 9.14 | 82.27 | 49.26 | 31.41 | 12.37 | 69.86 | 26.91 |
| SVMMN | 28.95 | 9.72 | 82.48 | 50.18 | 31.46 | 13.11 | 70.11 | 27.45 |
| KSVMMN | **29.36** | **10.08** | **84.20** | 50.08 | **31.89** | **13.68** | **70.75** | 27.35 |

**Table 8**
P@1 and P@3 results of each method on four datasets. The best result in each column is highlighted in bold.

| Algorithms | P@1 | | | | P@3 | | | |
|---|---|---|---|---|---|---|---|---|
| | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 |
| Logistic | 34.22 | **14.55** | **96.73** | 90.86 | 28.00 | 10.32 | 89.71 | 57.07 |
| KSVM | 35.85 | 8.58 | 84.31 | **91.24** | 29.73 | 8.61 | 85.19 | 54.28 |
| MLML | 34.96 | 10.62 | 90.85 | 88.89 | 30.57 | 9.72 | 84.75 | 46.14 |
| SLEEC | 35.85 | 9.37 | 92.16 | 91.13 | 29.58 | 8.51 | 89.32 | 62.60 |
| SVMMN$_{\beta=0}$ | 36.30 | 12.91 | 96.27 | 89.66 | 32.94 | 10.61 | 89.56 | 63.27 |
| SVMMN$_{\gamma=0}$ | 36.00 | 11.78 | 96.08 | 89.00 | 33.23 | 9.76 | 89.76 | 62.67 |
| SVMMN | 36.19 | 12.71 | 96.21 | 89.69 | 32.96 | 10.60 | 89.79 | **63.31** |
| KSVMMN | **36.74** | 13.82 | 95.03 | 89.67 | **33.48** | **11.24** | **90.24** | 62.49 |

**Table 9**
Actual run times of each method on four datasets (measured in seconds).

| Algorithms | Times (seconds) | | | |
|---|---|---|---|---|
| | ESP Game | MIR Flickr | NUS-WIDE-Lite | Wiki10 |
| Logistic | **22.04** | **9.28** | 3.33 | 1622.26 |
| KSVM | 70.65 | 14.41 | **2.41** | 387.80 |
| MLML | 95.73 | 55.51 | 4.01 | 7068.33 |
| SLEEC | 46.40 | 26.55 | 10.78 | **61.30** |
| SVMMN | 94.48 | 54.12 | 3.60 | 7001.00 |
| KSVMMN | 96.92 | 46.87 | 29.46 | 6890.60 |

Top-5 $F_1$ measure in the case of 100% label proportion are shown in Table 7. P@1 and P@3 results of each method on four datasets are shown in Table 8.

Comparing with the aforementioned experiments, we can see that the Top-5 precision, Top-5 $F_1$, P@1 and P@3 measures on both ESP Game and MIR Flickr datesets are low. The possible reason may be the average numbers of labels per sample, respectively 11.21% and 6.64% (shown in Table 1). On ESP Game, MIR Flickr and NUS-WIDE-Lite datesets, SVMMN's performance is better than other algorithms in terms of Top-5 precision, Top-5 $F_1$ and P@3 measures, which means that the recall of SVMMN or KSVMMN is higher than other methods. On the Wiki10 dataset, SLEEC does the best in terms of Top-5 precision and Top-5 $F_1$ measures, which means SLEEC is good method to deal with extreme multi-label classification.

### 4.5. Computational complexity and convergence

The main computational cost of the proposed algorithm is the computation of $\nabla \mathbf{w}_i$ (see Eq. (18)), whose time complexity is $O(mn(m+n+d))$. Table 9 shows actual run times of the different labeling algorithms on four datasets. According to the Table 9, we can see that Logistic and KSVM spend less time than our model and MLML on ESP Game, MIR Flicker and NUS-WIDE-Lite dataset. This is because our model and MLML can be seen as an extension of SVM and linear logistic regression model, respectively. Our method doer better than MLML on all datasets, however, our method and MLML perform not well on big dataset (i.e. Wiki10). The main reason is that when the number of candidate classes is large, the complexity of MLML and SVMMN will be high. In contrast, SLEEC performs much better than other methods on this big dataset, which proves that it is an efficient algorithm for handling extreme multi-label classification.

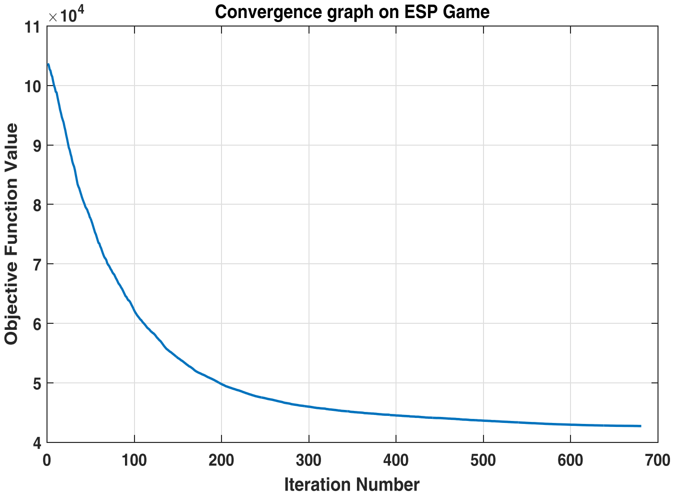

**Fig. 3.** Convergence curve of our method on the ESP Game dataset.

We show the convergence curve of our method on four databases from Figs. 3–6. Considering the number of classes $m$, we can find out that our proposed algorithm has good convergence with a small number of iterations. This indicates that our proposed algorithm has good computational efficiency.

### 4.6. Discussions

In order to estimate the effect of $\tau$ in Eq. (11) to our method, we construct the training data such that the portions of the labeled data vary from 20% (i.e., 80% missing labels) to 100% (i.e., no missing labels) in ESP Game dataset. We repeat this process 10 times

**Table 10**
AUC and AP results on the NUS-WIDE-Lite dataset (mean %). The best result in each column is highlighted in bold.

| The value of $\tau$ | AUC | | | | AP | | | |
|---|---|---|---|---|---|---|---|---|
| | 20% | 40% | 80% | 100% | 20% | 40% | 80% | 100% |
| 1 | **73.45** | **74.42** | **75.19** | **75.31** | **38.41** | 39.12 | 39.54 | **39.71** |
| 3 | 73.11 | 73.81 | 75.05 | 75.31 | 38.06 | **39.13** | 39.45 | 39.70 |
| 5 | 72.99 | 74.00 | 75.09 | 75.30 | 38.10 | 39.02 | **39.60** | 39.70 |
| 10 | 73.01 | 74.16 | 75.10 | 75.26 | 38.24 | 39.01 | 39.37 | 39.73 |
| 30 | 73.03 | 74.14 | 75.12 | 75.24 | 38.28 | 39.08 | 39.39 | 39.66 |

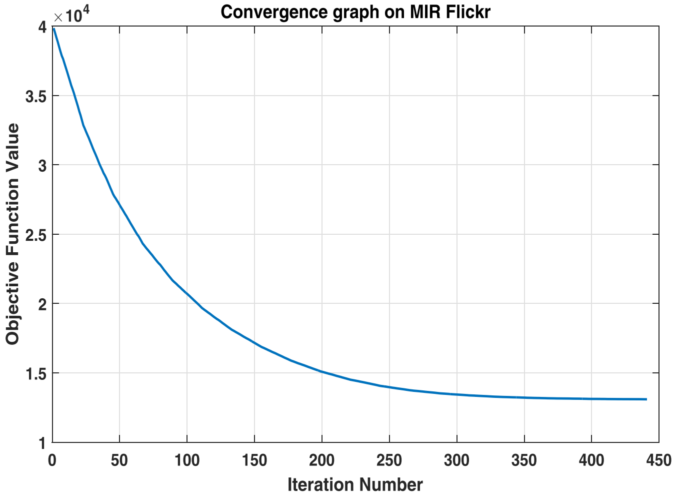

**Fig. 4.** Convergence curve of our method on the MIR Flickr dataset.

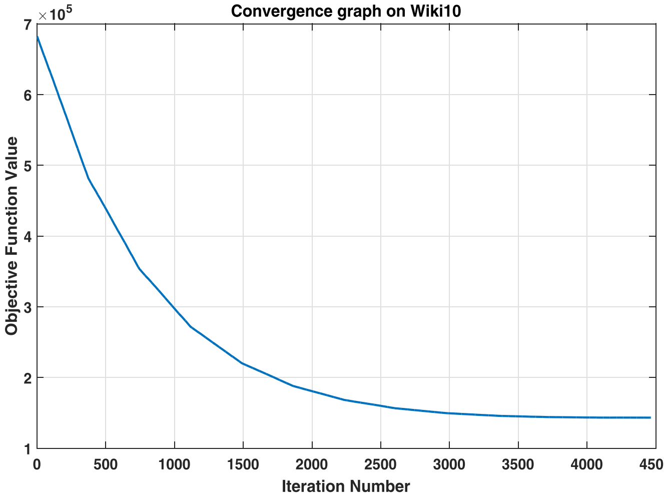

**Fig. 6.** Convergence curve of our method on the Wiki10 dataset.

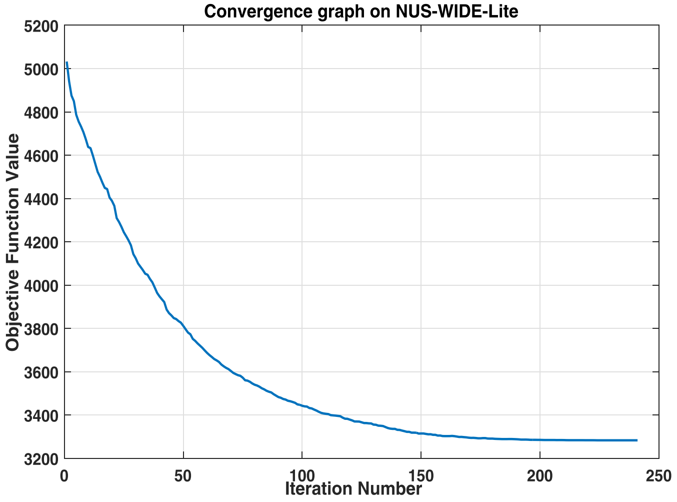

**Fig. 5.** Convergence curve of our method on the NUS-WIDE-Lite dataset.

to get different missing labels and show the mean value of AUC and AP in Table 10. As can be seen in Table 10, when $\tau$ is set as 1, our method has a good performance. The reason may be that we enhance the relationship between projection direction $\mathbf{w}_i$ and bias $b_i$ in Eq. (11). This may result in over-fitting.

## 5. Conclusions and future work

In this paper, we propose a novel algorithm named SVMMN for multi-label learning with missing labels. Different from most existing related algorithms, the proposed algorithm aims to learn a mapping function which guarantees not only the large margin with the projected data but also the minimum number of samples which live in the margin area. In addition, sample smoothness and class smoothness are also be considered in our proposed algorithm. Experiments on ESP Game, MIR Flickr, NUS-WIDE-Lite and Wiki10 benchmark datasets illustrate the superiority of our algorithm over the other related algorithms.

In future, there are many directions we need to further study. At first, the proposed SVMMN model suffers from the computational complexity. The optimization of solution will be explored in our future work. Secondly, the convergence of the model is only proved by some experiments in this paper, and the theoretical proof will be our next core task. Finally, we will also explore other applications of SVMMN, such as facial action unit recognition, scene classification and image denoising.

## Acknowledgment

The authors would like to thank the anonymous reviewers and AE for their constructive comments and suggestions, which improved the paper substantially. This work is supported by National Natural Science Foundation of China under Grant 61773302, China Postdoctoral Science Foundation (Grant 2012M521747) and the 111 Project of China (B08038).

## References

[1] S.H. Yang, H. Zha, B.G. Hu, Dirichlet–Bernoulli alignment: a generative model for multi-class multi-label multi-instance corpora., in: Proceedings of the Advances in Neural Information Processing Systems, 2009, pp. 2143–2150.

[2] T.N. Rubin, A. Chambers, P. Smyth, M. Steyvers, Statistical topic models for multi-label document classification, Mach. Learn. 88 (1–2) (2011) 157–208.

[3] G. Sohrab, M. Miwa, Y. Sasaki, In-deductive and DAG-tree approaches for large-scale extreme multi-label hierarchical text classification, in: Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, 2016.

[4] S.S. Bucak, R. Jin, A.K. Jain, Multi-label learning with incomplete class assignments, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 2801–2808.

[5] Z. Chen, M. Chen, K.Q. Weinberger, W. Zhang, Marginalized denoising for link prediction and multi-label learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2015.

[6] B. Wu, S. Lyu, B. Ghanem, Constrained submodular minimization for missing labels and class imbalance in multi-label learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2016.

[7] Y. Luo, T. Liu, D. Tao, C. Xu, Multiview matrix completion for multilabel image classification, IEEE Trans. Image Process. 24 (8) (2015) 2355–2368.

[8] Y. Li, B. Wu, B. Ghanem, Y. Zhao, H. Yao, Q. Ji, Facial action unit recognition under incomplete data based on multi-label learning with missing labels, Pattern Recognit. 60 (2016) 890–900.

[9] S. Eleftheriadis, O. Rudovic, M. Pantic, Multi-conditional latent variable model for joint facial action unit detection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 3792–3800.

[10] K. Zhao, W.S. Chu, F.D.L. Torre, J.F. Cohn, Joint patch and multi-label learning for facial action unit and holistic expression recognition, IEEE Trans. Image Process. 25 (8) (2016) 3931–3946.

[11] P. Heymann, G. Koutrika, H. Garcia-Molina, Can social bookmarking improve web search? in: Proceedings of the International Conference on Web Search and Data Mining, 2008, pp. 195–206.

[12] A. Vailaya, M.A.T. Figueiredo, A.K. Jain, H.J. Zhang, Image classification for content-based indexing, IEEE Trans. Image Process. 10 (1) (2001) 117–130.

[13] F. Wu, Y. Han, Q. Tian, Y. Zhuang, Multi-label boosting for image annotation by structural grouping sparsity, in: Proceedings of the Eighteenth ACM International Conference on Multimedia, ACM, 2010, pp. 15–24.

[14] X. Li, X. Zhao, Z. Zhang, F. Wu, Joint multilabel classification with community-aware label graph learning, IEEE Trans. Image Process. 25 (1) (2016) 484–493.

[15] K. Bhatia, H. Jain, P. Kar, M. Varma, P. Jain, Sparse local embeddings for extreme multi-label classification, in: Proceedings of the Advances in Neural Information Processing Systems, 2015, pp. 730–738.

[16] M.L. Zhang, L. Wu, Lift: multi-label learning with label-specific features, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 107–120.

[17] G.P.C. Fung, J.X. Yu, H. Lu, P.S. Yu, Text classification without negative examples revisit, IEEE Trans. Knowl. Data Eng. 18 (1) (2006) 6–20.

[18] A. Kapoor, R. Viswanathan, P. Jain, Multilabel classification using Bayesian compressed sensing, in: Proceedings of the Advances in Neural Information Processing Systems, 2012.

[19] G. Chen, Y. Song, F. Wang, C. Zhang, Semi-supervised multi-label learning by solving a Sylvester equation, in: Proceedings of the SIAM International Conference on Data Mining, Atlanta, Georgia, USA, 2008, pp. 410–419.

[20] Y.Y. Sun, Y. Zhang, Z.H. Zhou, Multi-label learning with weak label., in: Proceedings of the AAAI Conference on Artificial Intelligence, Atlanta, Georgia, USA, 2010.

[21] B. Wu, S. Lyu, B.G. Hu, Q. Ji, Multi-label learning with missing labels for image annotation and facial action unit recognition, Pattern Recognit. 48 (7) (2015) 2279–2289.

[22] M. Chen, A. Zheng, K. Weinberger, Fast image tagging, in: Proceedings of the International Conference on Machine Learning, 2013, pp. 1274–1282.

[23] Z. Lin, G. Ding, M. Hu, J. Wang, X. Ye, Image tag completion via image-specific and tag-specific linear sparse reconstructions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1618–1625.

[24] L. Wu, R. Jin, A.K. Jain, Tag completion for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 35 (3) (2013) 716–727.

[25] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, 2000.

[26] A.E. Elisseeff, J. Weston, A kernel method for multi-labelled classification, Adv. Neural Inf. Process. Syst. 14 (2002) 681–687.

[27] R.E. Schapire, Y. Singer, BoosTexter: a boosting-based system for text categorization, Mach. Learn. 39 (2–3) (2000) 135–168.

[28] X. Li, L. Wang, E. Sung, Multilabel SVM active learning for image classification, in: Proceedings of the International Conference on Image Processing, 2004, pp. 2207–2210Vol. 4.

[29] S.-J. Huang, Z.-H. Zhou, Z. Zhou, Multi-label learning by exploiting label correlations locally., in: Proceedings of the AAAI, 2012.

[30] M.-L. Zhang, Z.-H. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (8) (2014) 1819–1837.

[31] P.W. Holland, R.E. Welsch, Robust regression using iteratively reweighted least--squares, Commun. Stat. Theory Methods 6 (9) (1977) 813–827.

[32] Q. Wang, L. Ma, Q. Gao, Y. Li, Y. Huang, Y. Liu, Adaptive maximum margin analysis for image recognition, Pattern Recognit. 61 (2017) 339–347.

[33] S. Nikitidis, A. Tefas, I. Pitas, Maximum margin projection subspace learning for visual data analysis, IEEE Trans. Image Process. 23 (10) (2014) 4413–4425.

[34] L. Armijo, Minimization of functions having Lipschitz continuous first partial derivatives., Pac. J. Math. 16 (1) (1966) 1–3.

[35] L. Von Ahn, L. Dabbish, Labeling images with a computer game, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2009, pp. 319–326.

[36] M.J. Huiskes, M.S. Lew, The mir flickr retrieval evaluation, in: Proceedings of the ACM SIGMM International Conference on Multimedia Information Retrieval, Vancouver, British Columbia, Canada, 2008, pp. 39–43.

[37] X. Chen, Y. Mu, S. Yan, T.-S. Chua, Efficient large-scale image annotation by probabilistic collaborative multi-label propagation, in: Proceedings of the Eighteenth ACM international conference on Multimedia, ACM, 2010, pp. 35–44.

[38] A. Zubiaga, Enhancing Navigation on Wikipedia with Social Tags, in: Proceedings of the 4th Annual Conference of the Wikimedia Community, August, 2009.

[39] C. Liu, J. Yuen, A. Torralba, J. Sivic, W.T. Freeman, Sift flow: dense correspondence across different scenes, in: Proceedings of the European Conference on Computer Vision, Marseille, France, 2008, pp. 28–42.

[40] Y. Zhang, Z.H. Zhou, Multilabel dimensionality reduction via dependence maximization, in: Proceedings of the AAAI Conference on Artificial Intelligence, Chicago, Illinois, USA, 2008, pp. 1503–1505.

[41] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874.

[42] Y. Prabhu, M. Varma, FastXML: a fast, accurate and stable tree-classifier for extreme multi-label learning, in: Proceedings of the Twentieth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2014, pp. 263–272.

**Yang Liu**, received the B.Eng. degree from the Xidian University of Technology, Xi'an, China, in 2014. He is currently pursuing the Ph.D. degree with Xidian University, Xi'an. His research interests include dimensionality reduction, pattern recognition, and deep learning.

**Kaiwen Wen**, received the B.Eng. degree from Xidian University of Technology, Xi'an, China, in 2016. Now, she is working toward the M.S. degree at The Hong Kong University of Science and Technology, Hongkong, China. Her research interests include pattern recognition and deep learning.

**Quanxue Gao** received the B.Eng. degree from Xi'an Highway University, Xi'an, China, in 1998, the M.S. degree from the Gansu University of Technology, Lanzhou, China, in 2001, and the Ph.D. degree from Northwestern Polytechnical University, Xi'an China, in 2005. He was an associate research with the Biometrics Center, The Hong Kong Polytechnic University, Hong Kong from 2006 to 2007. From 2015 to 2016, he was a visiting scholar at the University of Texas at Arlington, Texas, USA. He is currently a professor with the School of Telecommunications Engineering, Xidian University, and also a key member of State Key Laboratory of Integrated Services Networks. He has authored 30 technical articles in refereed journals and proceedings, including the IEEE Transaction on Image Processing, IEEE Transaction on Neural Networks and Learning Systems, IEEE Transactions on Cybernetics, Pattern Recognition, CVPR, AAAI and IJCAI. His current research interests include pattern recognition and machine learning.

**Xinbo Gao** received the B.Eng., M.Sc., and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 1994, 1997, and 1999, respectively. He was a Research Fellow with the Department of Computer Science, Shizuoka University, Shizuoka, Japan, from 1997 to 1998. From 2000 to 2001,he was a Post-Doctoral Research Fellow with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong. Since 2001, he has been with the School of Electronic Engineering, Xidian University. He is currently a Professor of pattern recognition and intelligent systems, and the Director of the State Key Laboratory of Integrated Services Networks, Xidian University. He has authored five books and around 150 technical articles in refereed journals and proceedings, including IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Systems, Man and Cybernetics, and Pattern Recognition in his areas of expertise. His current research interests include computational intelligence, machine learning, computer vision, pattern recognition and wireless communications.

**Feiping Nie** received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2009. He was a Post-doctoral Research Associate, a Research Assistant Professor, and a Research Professor with The University of Texas at Arlington, Arlington, TX, USA, from 2009 to 2015. He is currently a Professor with Northwestern Polytechnical University, Xi'an China. He has authored 160 technical articles in refereed journals and proceedings, including the IEEE Transactions on pattern Analysis and Machine Intelligence, the International Journal of Computer Vision, the IEEE Transaction on Image Processing, the IEEE Transaction on Neural Networks and Learning Systems, the IEEE Transactions on Cybernetics, the IEEE Transactions on Knowledge and Data Engineering, ICCV, CVPR, ICML, AAAI, IJCAI, and NIPS. His current research interests include machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing, and information retrieval.