Semi-Supervised Boosting Using Similarity Learning Based on Modular Sparse Representation With Marginal Representation Learning of Graph Structure Self-Adaptive

SHU HUA XU\textsuperscript{1,2} AND FEI GAO\textsuperscript{1}

\textsuperscript{1}College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310000, China
\textsuperscript{2}School of Mathematics, Physics and Information Science, Shaoxing University, Shaoxing 312000, China

Corresponding author: Shu Hua Xu (xu_shuhua2001@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant C12412135, in part by the National Natural Science Youth Foundation of China under Grant 61602307, and in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LY19F020013 and Grant Q14A010009.

\textbf{ABSTRACT} The purpose of semi-supervised boosting strategy is to improve the classification performance of one given classifier for a large number of unlabeled data. In the semi-supervised boosting strategy, the unlabeled samples are assigned for pseudo labels according to similarities between the labeled samples and the unlabeled samples, and the unlabeled samples with high confidences of pseudo labels are selected as labeled samples at the same time. Good similarities help to assign more appropriate pseudo labels to the unlabeled samples. These selected samples with pseudo labels will be used as the labeled samples to train the new ensemble classifier. Therefore, good and distinguishable similarities learning between unlabeled samples and labeled samples has shown remarkable importance due to its promising performance for semi-supervised boosting strategy. This article presents semi-supervised boosting framework using similarity learning based on modular sparse representation by employing a marginal regression function with probabilistic graphical structure adaptation. In this article, distinguishable regression targets analysis, graph structure adaptation, robust modular sparse representation and semi-supervised boosting learning are seamlessly incorporated into a joint framework. This framework learns marginal regression targets from data rather than exploiting the conventional zero-one matrix that greatly hinders the freedom of regression fitness and degrades the performance of regression results to improve the interclass separation of the learned representation. Meanwhile, a regularization term based on probabilistic connection knowledge is used to construct a graph regularization with adaptive optimization, which improves the intra-class compactness of the learned representation. Additionally, modular sparse representation learning is used to improve the robustness of the learned representation. The experimental results on four datasets including face and object show that the recognition rates of the proposed method are significantly better than other state-of-the-art methods.

\textbf{INDEX TERMS} Low rank representation, margin representation, modular sparse representation, semi-supervised boosting, similarity learning.

\section{I. INTRODUCTION}

With the development of electronic equipment, more and more visual image data or non-visual text data are produced in the Internet and daily social communication. Most of the generated data is unclassified or unlabeled, so it is difficult to use supervised methods for image and document classification. Therefore, semi-supervised learning (SSL) \cite{1} has attracted more and more attention in machine learning and data mining. The core idea of SSL, especially semi-supervised learning...
classification, is to learn classification model by using labeled and unlabeled data. For a given classifier, the goal of semi-supervised boosting is to improve its classification performance by using the supervised information of labeled data and its relationship with unlabeled data. In particular, using unlabeled data in existing boosting methods [2]–[5] can obtain better performance of the boosted classifier.

In the semi-supervised boosting strategy, similarity is required, as shown in Figure 1. A new ensemble classifier is trained by selecting reliable unlabeled samples by similarity. Due to the simplicity of calculating Euclidean distance, using Euclidean distance measures the similarity between samples in most boosting strategies [3]. However, similarity (or distance measure) learning plays an important role in boosting tasks for two reasons: firstly, it is usually assumed that two samples with high similarity belong to the same class; secondly, due to the nonlinearity of data, Euclidean distance cannot represent the structural nonlinearity of data, especially in high-dimensional space. In the semi-supervised boosting strategy, Euclidean distance is used to calculate the Gaussian kernel similarity. However, how to set the kernel width in Gaussian kernel similarity is also a problem to be solved.

Recently, in the adaptive field of data similarity, representation based on sparse coding has received extensive attention [6]–[8]. Therefore, this article uses sparse representation as a measure of similarity.

In sparse representation learning, when a classifier or other prediction variables are constructed, the discriminative information of the data and the effective visualization of the data make it easier to extract information features from the data. As we all know, the traditional sparse representation and low rank representation (LRRs) cannot meet the requirements of real-time applications due to the large amount of computation [9], [10]. In addition, the learned data representation still failed to capture discriminable attributes of potential explanatory factors of observed inputs from different objects. Moreover, in many cases, the problem of image recognition will be complicated because the image is occluded, for example, the face image occluded with sunglasses, headdress, scarf, mask, facial hair or hand. In this case, the recognition method using local image information has advantages over the whole histogram feature. Features extracted from occluded areas will be lost, but those features extracted from no occluded areas will not be lost, and may be sufficient for accurate image classification. Classification decision is usually obtained by nearest neighbor algorithm, support vector machine or boosting strategy. In order to solve the above problems, this article proposes a modular sparse representation learning method (MSPASEMBIOOOST) with marginal structure representation to calculate the similarity between images efficiently and effectively for semi-supervised boosting strategy. Firstly, the image is divided into several smaller image blocks. Then, on each image block set, the similarity between image blocks is calculated based on a simple and effective sparse representation with margin regression target. Secondly, the weighted voting algorithm is used to combine the similarity results learned from each image block set into a whole. Instead of using the fixed 0-1 matrix as the regression target, the self-adjusting regression target with better approximate optimal margin constraint is constructed directly to measure the regression result more accurately. The purpose of probabilistic graph structure adaptation is to capture the potential structure with data connectivity, so as to guide the construction of margin regression target. In addition, the regression results are further predicted in the
We first give some symbols used in this article. The matrix conclusion.

The Frobenius norm of matrix $X$ is defined as $||X||_F^2 = tr(X^TX) = tr(XX^T)$, where $tr(\cdot)$ is the trace operator.

The goal of semi-supervised boosting [2]–[5] is to improve its classification performance by using the supervised information of labeled data and its relationship with unlabeled data. Its strategy is shown in the figure 1.

Given that one dataset $X = \{x_1, x_2, \cdots , x_n\} = [X_l, X_u]$ contains $n$ elements to represent the entire dataset, there are $n_l$ labeled data $X_l = \{x_1^l, x_2^l, \cdots , x_n^l\}$ and $n_u$ unlabeled data $X_u = \{x_1^u, x_2^u, \cdots , x_n^u\}$. The label subset of $n_l$ labeled data is $y_l = [y_1^l, y_2^l, \cdots , y_{n_l}^l]$, label set is $l = [1, 2, \cdots , C]$, $C$ is the total number of classes. $y_i^l$ is one vector. If $x_i^l$ belongs to the $k$th class, then the $k$th element of $y_i^l$ is 1, that is, $y_i^l(k) = 1$, and the other elements are 0, with $y_i^l = [0, \cdots , 0, 1, \cdots , 0]_C$. The label subset of unlabeled data is $y_u = [y_1^u, y_2^u, \cdots , y_{n_u}^u]$, where $n = n_l + n_u$. The label of the total dataset is $y = [y_l^u, y_u]$. $S = [S_{i,j}]_{m \times n}$ is one symmetric similarity matrix, where $S_{i,j} \geq 0$ represents the similarity between $x_i$ and $x_j$. Unlabeled data $x_u$ has the same label as labeled data with the highest similarity. The symmetric matrix $S^{iu}$ represents the similarity between labeled data and unlabeled data. $\Gamma_L(Y, S^{lu})$ represents the inconsistency between labeled data and unlabeled data. Two unlabeled data with the highest similarity are assumed to have the same label. The symmetric matrix $S^{uu}$ represents the similarity between unlabeled data. $\Gamma_U(Y_u, S^{uu})$ represents the inconsistency between unlabeled data. Therefore, the loss function $\Gamma(y, S)$ can be obtained from the above two terms. Our goal is to find the label $y_u$ that minimizes $\Gamma(y, S)$.

Specifically, the loss function is as follows:

$$\Gamma(y, S) = w_1\Gamma_L(y, S^{lu}) + w_2\Gamma_U(y_u, S^{uu}),$$

where $w_1$ and $w_2$ are weights, weighing the importance of labeled data and unlabeled data. According to [2], let $w_1 = \frac{1}{n_l^2}$, $w_2 = \frac{1}{n_u^2}$.

The two terms in (1) are as follows:

$$\Gamma_L(y, S^{lu}) = \sum_{i=1}^{n_l} \sum_{j=1}^{n_u} S_{i,j}^{lu} \exp(-2y_i^l y_j^u).$$

$$\Gamma_U(y_u, S^{uu}) = \sum_{i=1}^{n_u} \sum_{j=1}^{n_u} S_{i,j}^{uu} \exp(y_i^u - y_j^u).$$

Let $h_i(x)$ denote the classifier trained by the learning algorithm in the $i$th iteration, where $h(x_i) = y_i^l$, $i = 1, \cdots , n_l$ and $H(x)$ denotes the ensemble classifier, which has the following equation:

$$H(x) = \sum_{i=1}^{T} \alpha_i h_i(x),$$

where $\alpha_i$ is the weight of the weak classifier $h_i(x)$. $\Gamma(y, S)$ uses upper bound optimization method for iterative minimization, similar to [2]. The upper bound of (1) is shown in

II. RELATED WORK

We first give some symbols used in this article. The matrix is represented in capital letters, such as $X$. The $i$th row and $j$th column elements of matrix $X$ are represented by $X_{ij}$. Lowercase letters represent column vectors, such as $x$. The rest of this article is organized as follows. The second part briefly introduces the related work. Then, the proposed MPASEMIBOOST is described and the optimization algorithm is given in the third part. The fourth part gives a large number of experimental results, and the fifth part gives the conclusion.
In order to obtain flexible similarity matrix based on prior knowledge [11], [12]. In recent years, there are no reliable models to select, so it is difficult to determine the best parameters for measurement of similarity based on prior knowledge [11], [12]. In recent years, the representation based on sparse coding has attracted much attention [6]–[8]. In order to obtain flexible similarity matrix and get better performance, Wang et al. [13] utilized sparse representation to learn similarity, and proposed similarity learning based on sparse representation for semi-supervised boosting. The function of similarity learning based on sparse representation is

\[
(D, S, E_0) = \min_{D, S, E_0} ||S||_1 + \lambda ||E_0||_F
\]

s.t. \( X = DS + E_0, \text{diag}(S) = 0, \) (8)

where matrix \( D \in R^{d \times C} \) is obtained from data \( X = [x_1, \cdots, x_n] \in R^{d \times n}, d \) is the dimension of samples, \( n \) is the number of samples, \( C \) is the number of classes of samples, \( ||E_0||_F \) is \( l_{2,1} \)-norm, \( l_1 \)-norm or squared Frobenius norm \( || \cdot ||_F^2 \) of \( E_0 \).

However, the global sparse representation of the algorithm has high computational complexity and cannot capture the local potential distinguishable attributes of the data, especially when the image is occluded. The modular sparse representation can weaken the feature extraction of the occluded area, strengthen the feature extraction of the non-occluded area, and achieve more accurate classification for image recognition. In order to solve these problems, this article uses modular sparse representation to learn similarity in semi-supervised boosting learning, introduces and applies structural predictor learning to mine the inter-class separation of the learned representation by imposing the low-rank regularization, and introduces and applies adaptive graph structure learning to improve the intra-class compactness of the learned representation by constructing an adaptive probabilistic graph [14]. To mine the inter-class separation and improve the intra-class compactness of the learned representation, the function of similarity learning is

\[
(D, S) = \min_{D, S} \frac{1}{\sum_{i,j} P_{ij}^2} + \frac{\gamma}{\lambda_1} ||D - FZ||_F^2
\]

\[
+ ||D||_F^2 \quad \text{s.t.} \quad S_{il} - \max_{j \neq l} S_{ij} \geq J, \quad F^T F = I, \quad 0 < P_{ij} < 1, \quad \text{Pe}_n = e_n. \quad (9)
\]

where \( S = [S_1, \cdots, S_n] \), \( S_{il} \) is the value at the \( l \) position in \( S_i \), and \( J \) is the margin constant, the rank of matrix \( D \) is \( s(s < \min(C, n)), F \in R^{d \times s}, Z \in R^{s \times C}, \gamma, \lambda_1 \) and \( \beta \) are non-negative parameters, the constraint \( 0 < P_{ij} < 1, \text{Pe}_n = e_n \) guarantees that \( P \) represents the transition probability matrix, and each row of which is one probability distribution.

The decision combination problem of different image blocks based on modular sparse representation has been widely concerned [15]. The main idea is to combine the decisions of different sub-image blocks using voting scheme. The existing typical voting scheme is the maximum number of votes [15]. Majority voting requires that each classifier has a vote to vote for the class with the lowest representation error, while all other classes will not get a vote. The decision center summarizes the number of votes obtained by each class, and classifies the image into the class that get the most votes. On the basis of the maximum number of votes, the Borda counting method was proposed in [16], in which each class is given a weight, that is, ranking representation errors of all classifiers from low to high, and the number of classes representation errors of which are higher than representation error of this class is the weight of this class [16]. If we consider a sub-image \( y_k(k = 1, \cdots, M) \), the weight of the class that produces the minimum representation error for
\( y_k(1, \ldots, M) \) is \((C - 1)\), and the weight of the next class with the smallest representation error is \((C - 2)\). The rest can be done in the same way. For each class \( j (j = 1, 2, \ldots, C) \), we denote \( N_j^{(i)} \) as the number of times that the \( j \)th class is ranked in the \( i \)th position on all sub-images \( y_m (m = 1, \ldots, M) \). The total number of votes obtained by the \( j \)th class can be expressed as follows:

\[
N_j = \lambda_1 N_j^{(1)} + \lambda_2 N_j^{(2)} + \cdots + \lambda_C N_j^{(C)},
\]

where \( \lambda_i (i = 1, \ldots, C) \) is the weight of the \( j \)th class in different ranking positions. In the case of Borda counting, the weight \( \lambda_j \) is \( C - i \). The class that gets the most votes is used as the output of the classifier, which is called Borda counting vote SRC:

\[
\hat{c}_{\text{Borda}} = \arg\max_{j \in \{1, \ldots, C\}} N_j.
\]

The Borda counting method is easy to implement, but the difference between classifiers is not considered in the weight calculation of each sub image in the Borda counting method [16]. Although it is well known that some parts of the image have more information for expression recognition, the classifier treats each sub-image equally. In addition, the relative difference in the representation error of different classes to a region is not reflected in the votes allocated to each class, that is, no matter how close the representation error of two classes to the region, the number of votes obtained by two classes is a difference of 1. The original calculation model of voting weight cannot be directly applied to semi-supervised boosting framework. In order to solve this problem, sigmoid function is used to calculate the credibility weight of each sub image in this article, so as to reduce the impact of information of these occluded and polluted modules on recognition.

For each sub image \( y_k (k = 1, \ldots, M) \) of the sample \( y \), sparse representation is used to solve the sparse coefficient \( O_k^y \), and the coefficients corresponding to training samples of the \( i \)th class are summed by absolute value, and the coefficient of the \((i = 1, 2, \ldots, C)\)th class in the \( k \)th sub image of the sample \( y \) is constructed.

\[
b_y = \{b_{y,1}, b_{y,2}, \ldots, b_{y,C}\}. \tag{13}
\]

Use (14) to calculate the sparsity \( s_{y,k} \) of each sub image \( y_k (k = 1, \ldots, M) \) of the sample \( y \).

\[
s_{y,k} = \frac{\sqrt{C}}{\sqrt{C} - 1} \left( \frac{||b_{y,k}||_2}{||b_{y,y}||} - \frac{1}{\sqrt{C}} \right). \tag{14}
\]

When there is only one non-zero element in \( b_{y,k} \), \( s_{y,k} \) is the maximum value 1, and when all the elements have the same non-zero value in \( b_{y,k} \), \( s_{y,k} \) is the minimum value 0.

The residual error \( r_{y,k} \) of each sub image \( y_k (k = 1, \ldots, M) \) of the sample \( y \) is calculated by using the \( l_2 \) normal form.

\[
r_{y,k} = ||y_k - D_k O_k^y||_2. \tag{15}
\]

The sigmoid function is used to calculate the confidence level \( w_{y,k} \) of sparsity of each sub image \( y_k (k = 1, \ldots, M) \) in the sample \( y \).

\[
w_{y,k} = \begin{cases} 0, & s_{y,k} \leq s_1 \\ \frac{1}{1 + \exp \left(-20 \left( s_{y,k} - \frac{s_1 + s_2}{2} \right) \right)}, & s_1 < s_{y,k} < s_2 \\ 1, & s_{y,k} \geq s_2, \end{cases} \tag{16}
\]

where \( s_1 \) and \( s_2 \) are two security thresholds of sparsity.

The sigmoid function is used to calculate the confidence level \( w_{y,k} \) of residual of each sub image \( y_k (k = 1, \ldots, M) \) in the sample \( y \).

\[
\hat{w}_{y,k} = \begin{cases} 1, & r_{y,k} \leq r_1 \\ \frac{1}{1 + \exp \left(20 (r_{y,k} - \frac{r_1 + r_2}{2}) \right)}, & r_1 < r_{y,k} < r_2 \\ 0, & r_{y,k} \geq r_2, \end{cases} \tag{17}
\]

where \( r_1 \) and \( r_2 \) are two security thresholds of residual.

Figure 2 shows that changes of the credibility weights \( w_{y,k} \) and \( w_{y,k} \) with \( s_{y,k} \) and \( r_{y,k} \), respectively.

According to \((s_{y,k} \geq s_2) \land (r_{y,k} \leq r_1)\), the sub images which are not occluded or polluted are distinguished. According to \((s_{y,k} \leq s_1) \lor (r_{y,k} \geq r_2)\), the sub images with occlusion or pollution are identified. Therefore, the credibility weight value of sub image can be obtained by the following equation (18).

\[
w_{y,k} = w_{y,k} \times \hat{w}_{y,k}. \tag{18}
\]

III. PROPOSED MODULAR SPARSE SEMI-SUPERVISED BOOSTING LEARNING

A. BASIC IDEA

Combined with related work, the idea of this article is based on the following considerations:

(1) In [16], the weights of sparse representation of image blocks were calculated by Borda voting weighting algorithm. Although Borda counting is easy to implement, the difference between classifiers corresponding to image blocks is not considered in the classifier of Borda counting method [16]. In addition, the relative differences in the representation errors between different classes within the same region are not reflected in the votes allocated to each class. We use the sigmoid function to calculate the credibility weight of each sub image to improve the performance of Borda voting weighting algorithm, and use it in semi-supervised boosting learning.

(2) In semi-supervised boosting learning, structure prediction learning is introduced and applied to mine the interclass separation of the learned representation by imposing the
low-rank regularization. Adaptive graph structure learning is introduced and applied to improve the intra-class compactness of the learned representation by constructing an adaptive probabilistic graph.

Based on the above analysis and the existing related work, we apply the low-rank regularized structure prediction learning, sparse representation with adaptive graph structure learning, and the improved model of sparse weight calculation of related modules to improve similarity learning model of semi-supervised boosting, and propose an adaptive modular sparse representation (MSPASEMIBOOST) learning algorithm.

On the basis of [14], MSPASEMIBOOST learning model can be expressed as follows:

$$\min_{F,D,S} \Gamma(Y, S) + \lambda_1 \Psi(D) + \lambda_2 \Phi(D)$$

$$\Psi(D) = \left| |D| \right|^2_F + \frac{\gamma}{\lambda_1} \left| |D - FZ| \right|^2_F$$

$$\Phi(D) = \sum_{i,j=1}^n \left( | \left| D^T x_i - D^T x_j \right|_2^2 P_{ij} + \beta P_{ij}^2 \right)$$

s.t. \( F^T F = I, \ 0 < P_{ij} < 1, \ Pe_n = e_n \) \( (19) \)

where \( \Gamma(\cdot) \) is the discriminant loss function to measure the error between the predetermined target and the predicted result, \( \Psi(D) \) is the structural prediction learning for controlling the complexity of matrix \( D \), and \( \Phi(D) \) is the graph adaptive learning for controlling the smoothness of matrix \( D \). In addition, \( \lambda_1 \) and \( \lambda_2 \) are two parameters balancing the importance of the three terms.

### B. ALGORITHM STEPS

Therefore, an iterative optimization algorithm is proposed. In order to clearly express the main algorithm steps, the key algorithms are defined and described first.

1) Key algorithm 1: calculate \( D \) and its two decomposition matrices \( F \) and \( Z \), labeled data sample connection probability matrix \( P \) and matrix \( S \). The specific formula is given in [14].

$$\Gamma(D, F, Z, S, P) = \| D^T X - S \|_F^2 + \gamma \| D - FZ \|_F^2 + \lambda_1 \| D \|_F^2$$

$$+ \lambda_2 \sum_{i,j=1}^n \left( | \left| D^T x_i - D^T x_j \right|_2^2 P_{ij} + \beta P_{ij}^2 \right)$$

s.t. \( F^T F = I, \ 0 < P_{ij} < 1, \ Pe_n = e_n \) \( (20) \)

where \( \gamma, \lambda_1, \lambda_2 \) and \( \beta \) are regularization parameters.

1) \( D \) and its two decomposition matrices \( F \) and \( Z \) calculated by iterative process.

The specific calculation formula is obtained by referring to [14].

$$\Gamma(D, F, Z) = \| D^T X - S \|_F^2 + \gamma \| D - FZ \|_F^2 + \lambda_1 \| D \|_F^2$$

$$+ \lambda_2 \sum_{i,j=1}^n \left( | \left| D^T x_i - D^T x_j \right|_2^2 P_{ij} \right)$$

s.t. \( F^T F = I \). \( (21) \)

According to [14], (21) can be equivalently transformed into formula (22).

$$\Gamma(D, F, Z) = \| D^T X - S \|_F^2 + \gamma \| D - FZ \|_F^2 + \lambda_1 \| D \|_F^2$$

$$+ \lambda_2 \text{tr}(D^T X E X^T D)$$

s.t. \( F^T F = I \). \( (22) \)

where \( E \) is the graph Laplacian matrix of \( P \), which is defined as \( E = T - P \), and \( T \) is a diagonal matrix whose main diagonal elements are the column sum of matrix \( P \), that is, \( T = \sum_{j=1}^n P_{ij} \).

Update \( Z \): According to the constraint term \( F^T F = I \), we can deduce the solution of \( Z \) by setting the first derivative \( \frac{\partial\Gamma}{\partial Z} = 0 \) of \( \Gamma \) versus \( Z \). Namely:

$$2\gamma(F^T FZ - F^T D) = 0 \iff Z = F^T D. \ (23)$$
Update $D$: we plug $Z$ into the objective function $\Gamma$, the objective function $\Gamma$ is re-expressed as:

$$
\Gamma(D, F, Z) = ||D^TX - S||_F^2 + \lambda_2 tr(D^TXEX^TD)
+ \lambda_1 ||D||_F^2 + \gamma tr(D(I-F^TD)(I-F^TD)).
$$

(24)

By taking the first derivative $\frac{\partial \Gamma}{\partial D} = 0$ of (24), we can get the following equation (25).

$$
D = (K - \gamma FF^T)^{-1}XS^T,
$$

(25)

where $K = XX^T + \lambda_2 XX^T + (\lambda_1 + \gamma)I$.

Update $F$: the following equation (26) is obtained by minimizing (22).

$$
\min_F ||D - FZ||_F^2 \quad \text{s.t.} \quad FT^TF = I.
$$

(26)

2) According to [14], $S$ was updated by $P$, $D$, $F$, $Z$. Removing the constant term independent of $S$, the objective function of (20) can be rewritten as:

$$
\min_S ||D^TX - S||_F^2 \quad \text{s.t.} \quad S_{ij} - \max_{j \neq l} S_{il} \geq J,
$$

(27)

Get:

$$
S_j = \begin{cases} W_j + \xi, & j = c \\ W_j + \min(\xi - g_j, 0), & \text{otherwise,} 
\end{cases}
$$

(28)

where $W = D^TX$, $\xi = \frac{\sum_{g \in \mathcal{G}} \Omega(\psi(\xi_g) > 0)}{1 + \sum_{g \in \mathcal{G}} \Omega(\psi(\xi_g) > 0)}$, $g \in \mathcal{G}$, $g_j = W_j + 1 - W_c$.

3) Fix $S, D, F$ and $Z$, connection probability matrix $P$ of the labeled data samples is updated iteratively.

$$
p_i = \max(t_i + g, 0), \quad g = \frac{1}{k} \left(1 - \sum_{j=1}^{k} t_{ij}\right),
$$

(29)

where $p_i$ and $t_i$ are the $i$th row of $P$ and $T$, respectively, $t_{ij} = n_{ij}/[w_{ij} - w_{ij}^2]^{1/2}$, $k$ is the number of nearest neighbors, and $\tilde{t}_i$ is $t_i$’s vector in ascending order.

(2) Key algorithm 2: iterative computation of modular sparse dictionary for labeled data samples

The convergence criterion used in key algorithm 2 is that the number of iterations reaches 30 or $|\Gamma_{t+1} - \Gamma_t|/\Gamma_t < 0.001$, where $\Gamma_t$ is the value of the objective function in the $t$th iteration. The main steps of algorithm 2 are shown in Table 1.

(3) Key algorithm 3: Calculate the similarity of modular sparse representation of unlabeled test samples according to the obtained modular sparse dictionaries $D_1, D_2, \ldots, D_M$.

The main steps of algorithm 3 are shown in Table 2.

Based on the above main algorithms. The main steps of MSPASEMIBOOST are shown in Table 3.

### TABLE 1. Solution of modular sparse dictionary for labeled data samples.

| Algorithm 2 | Solution of modular sparse dictionary for labeled data samples |
|-------------|--------------------------------------------------|
| **Input:** | dataset $X$ divided into $M$ sub-modules, where labeled data set $X_L = [X_{11}, X_{12}, \ldots, X_{1M}] = [X_{11}, X_{12}, \ldots, X_{1M}]; X_{21}, X_{22}, \ldots, X_{2M}]; \ldots; X_{n1}, X_{n2}, \ldots, X_{nM}];$ unlabeled dataset $X_U = [X_{n+1,1}, X_{n+1,2}, \ldots, X_{n+1,M}] + [X_{n+2,1}, X_{n+2,2}, \ldots, X_{n+2,M}] + \ldots [X_{n+\lambda,1}, X_{n+\lambda,2}, \ldots, X_{n+\lambda,M}]$; label matrix $L = [I_{11}, I_{12}, \ldots, I_{1M}] + [I_{21}, I_{22}, \ldots, I_{2M}] + \ldots [I_{\lambda1}, I_{\lambda2}, \ldots, I_{\lambdaM}]$; $y_t$ represents the label set of unlabeled data, $y_u$ represents the label set of unlabeled data, parameters $\lambda_1, \lambda_2, \gamma, C, n_1, n_2, k, \eta$. |
| **Step 1:** | iter = 0, Use KSVT to initialize the dictionary $D_1, D_2, \ldots, D_M$ on the dataset $X, S = L, s, P, Z, P$ as the identity matrix respectively, and use $X$ to initialize the graph Laplacian matrix $E$. |
| **Step 2:** | Loop |
| **Step 3:** | $m = 1$ |
| **Step 4:** | Loop |
| **Step 5:** | $Z = FT D_m$ |
| **Step 6:** | $K = X_{li,m} X_{il,m}^T + \lambda_2 X_{li,m} X_{il,m}^T + (\lambda_1 + \gamma)I$, where $lu = 1, 2, \ldots, n_1, n_2, \ldots, n_M$. |
| **Step 7:** | $D_m = (K - \gamma FF^T)^{-1}X_{li,m}Z^T$ |
| **Step 8:** | $[U, \Sigma, V^T] = SVD(D_mZ^T)$ |
| **Step 9:** | $F = UV^T$ |
| **Step 10:** | Use the equation (28) to update the margin target vector $S$ column by column. |
| **Step 11:** | Use the equation (29) to update the $i$th row of $P_m$ row by row, i.e. $P_{mi}$. |
| **Step 12:** | update $E = \text{diag}(\text{sum}(P_m)) - P_m$. |
| **Step 13:** | $m = m + 1$ |
| **Step 14:** | until $m > M$ |
| **Step 15:** | iter = ite + 1 |
| **Step 16:** | until iter > 30 or satisfy the convergence condition. |
| **Output:** | modular sparse dictionaries $D_1, D_2, \ldots, D_M$. |

### IV. EXPERIMENTAL RESULTS AND ANALYSIS

In order to evaluate the performance of proposed MSPASEMIBOOST algorithm, we compare it with 12 mainstream data representation learning methods. In order to compare with the mainstream algorithms, three real face databases (EXTENDED YALEB, CMU PIE and AR) and COIL-100 object recognition database are selected as datasets.

#### A. EXPERIMENTAL DATASET

1) EXTENDED YALEB FACE DATABASE

EXTENDED YALEB face database contains 2414 frontal face images from 38 subjects, each of whom took about 64 images under different lighting conditions. The main challenge of this database is to deal with different lighting conditions and facial expressions. Figure 3 shows a set of images from EXTENDED YALEB face database.

2) CMU PIE FACE DATABASE

CMU PIE face database includes 41368 face images of 68 subjects. The experiment was conducted on images of five postures (C05, C07, C09, C27 and C29), with 170 images per subject. Figure 4 shows a set of images from CMU PIE face database.
TABLE 2. Similarity algorithm using modular sparse representation.

| Algorithm 3: Similarity algorithm using modular sparse representation based on voting weight |
|---------------------------------------------------------------|
| **Input:** modular sparse dictionaries $D_1, D_2, \ldots, D_M \in R^{d \times n}$, $d < n$, sparsity $s = 8$, number of classes $C$, unlabeled test data $y$, $M$ sub modules $y_1, y_2, \ldots, y_M$, maximum number of votes $m$. |
| **Step 1:** Use the orthogonal matching pursuit (OMP) algorithm to solve $M$ modular sparse coefficients of $y$, that is $G_k^y = \text{OMP}(D_k^e, y_k, D_k^e + D_k, \text{sparsity}(y_k), k = 1, 2, \ldots, M)$. |
| **Step 2:** Use (12) to construct the sub-module class coefficient $s_k y_k$ of the $i$th ($i = 1, 2, \ldots, C$) class in the $k$th sub-module $y_k (k = 1, \ldots, M)$ of test data $y$. |
| **Step 3:** Use (13) to construct the sub-module class vector $b_k = [b_{yk1}, b_{yk2}, \ldots, b_{ykC}]$ of $y_k$. |
| **Step 4:** Use (14) to calculate sparsity $s_k y_k$ for each sub-module $y_k (k = 1, \ldots, M)$ of $y$. |
| **Step 5:** Use (15) to calculate residual error $r_i y_k$ of $y_k$. |
| **Step 6:** Use (16) to calculate confidence level $w_k y_k$ of sparsity for each sub-module $y_k (k = 1, \ldots, M)$ of $y$. |
| **Step 7:** Use (17) to calculate confidence level $w_k y_k$ of residual error for each sub-module $y_k (k = 1, \ldots, M)$ of $y$. |
| **Step 8:** Use (18) to calculate confidence level $w_k y_k$ of $y_k$ if the confidence level of all sub-modules of $y$ is 0, then $w_k y_k$ is 1. |
| **Step 9:** Calculate the residual error $r_i y_k$ of $y_k$ if the confidence level of all sub-modules of $y$ is 0, then $r_i y_k$ is 1. |
| **Step 10:** Sort the residual errors $r_1 y_k, r_2 y_k, \ldots, r_M y_k$ of the $k$th sub-module of $y$ in ascending order. The class with the smallest residual error gets $m$ votes, the class with the minor residual error gets $m - 1$ votes, and so on. If a certain class gets 1 vote, all the remaining classes get 1 vote. $N_k$ denotes the number of votes obtained by the $i$th ($i = 1, 2, \ldots, C$) class in the $k$th sub-module $y_k (k = 1, 2, \ldots, M)$. |
| **Step 11:** By weighted sum of the total votes of each class on all modules of test sample $y$, the total number of votes obtained for each class is calculated as $N_i = w_1 N_i^1 + w_2 N_i^2 + \ldots + w_M N_i^M$. |
| **Step 12:** Calculate $i = \arg \max_i N_i$. |
| **Output:** the coefficients $O^y = [\phi_1(O^y_1), \phi_2(O^y_2), \ldots, \phi_i(O^y_i)]$ of sparse representation for test data $y$. |

3) AR FACE DATABASE

AR face database contains 4000 color face images of 126 subjects with different illumination, disguise and facial expression. Each subject provided 26 face images which were taken in two stages under different conditions. In this experiment, a subset of 2600 images were selected from 50 female and 50 male subjects. Similar to the implementation in [17], the image is projected to 1540 dimensions using the randomly generated matrix in normal distribution with zero mean. Figure 5 shows a set of images from AR face database.

4) COIL-100 DATABASE

COIL-100 database consists of 100 objects with different illumination conditions and different angle in 360 degree rotation. The size of each image is adjusted to a 32 by 32 matrix. The challenge of this database is that the images were taken from different angles. Figure 6 shows a set of images from COIL-100 database.

B. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

For EXTENDED YALEB and CMU PIE, 10, 15, 20 and 25 images were randomly selected from each subject as training samples, and the rest were used as test samples. For AR dataset, 8, 11, 14 and 17 images were randomly selected from each subject as training samples, and the rest were used as test samples. For COIL-100 data set, we randomly selected 10, 15, 20, 25 images from each object as training samples, and the rest were used as test samples. We selected CAPSVM [18], PROCRSC [19], DSRM [20], RLSL [21], DLSR [22], RPCA [23], LLRR [24], SLRM [25], PCE [26], MSRL [14], SEMIBOOST [2], XGBOOST [27] to
TABLE 4. Experimental comparison results on EXTENDED YALEB face database.

|    | 10  | 15  | 20  | 25  |
|----|-----|-----|-----|-----|
| CAPSVM | 88.98 | 94.34 | 95.88 | 97.13 |
| SEMiboost | 86.12 | 94.02 | 95.38 | 96.68 |
| PROCR | 86.88 | 93.12 | 95.86 | 98.01 |
| DSRM | 89.88 | 93.56 | 95.12 | 96.80 |
| RLSL | 88.12 | 93.06 | 94.43 | 95.22 |
| DLSE | 87.12 | 94.11 | 94.20 | 95.42 |
| RPCA | 86.10 | 90.12 | 93.02 | 95.12 |
| LR | 83.07 | 90.78 | 92.12 | 94.99 |
| SLRM | 82.87 | 88.86 | 92.23 | 94.45 |
| PCE | 83.28 | 89.45 | 92.70 | 94.75 |
| XGBoost | 87.12 | 94.22 | 96.08 | 97.60 |
| MSRL | 90.94 | 94.92 | 96.40 | 97.60 |
| MSSESEMIBOOST | 94.11 | 96.81 | 98.29 | 98.99 |

TABLE 5. Experimental comparison results on CMU PIE face database.

|    | 10  | 15  | 20  | 25  |
|----|-----|-----|-----|-----|
| CAPSVM | 85.32 | 90.77 | 93.11 | 94.50 |
| SEMiboost | 85.02 | 90.72 | 92.12 | 93.88 |
| PROCR | 86.99 | 91.62 | 93.40 | 94.50 |
| DSRM | 85.02 | 90.50 | 92.75 | 93.99 |
| RLSL | 86.45 | 91.02 | 92.96 | 94.05 |
| DLSE | 84.45 | 90.56 | 92.08 | 93.40 |
| RPCA | 81.52 | 83.80 | 87.92 | 90.72 |
| LR | 80.78 | 83.82 | 87.89 | 91.43 |
| SLRM | 83.52 | 87.91 | 91.07 | 92.72 |
| PCE | 83.33 | 87.26 | 88.17 | 88.65 |
| XGBoost | 85.82 | 90.02 | 92.98 | 93.68 |
| MSRL | 88.92 | 91.92 | 94.74 | 95.74 |
| MSSESEMIBOOST | 90.97 | 94.25 | 95.36 | 97.12 |

TABLE 6. Experimental comparison results on AR face database.

|    | 8  | 11 | 14 | 17 |
|----|----|----|----|----|
| CAPSVM | 88.55 | 93.62 | 95.71 | 97.07 |
| SEMiboost | 86.94 | 90.12 | 92.98 | 95.23 |
| PROCR | 88.70 | 93.02 | 94.99 | 96.01 |
| DSRM | 87.85 | 92.21 | 93.58 | 94.34 |
| RLSL | 89.19 | 92.20 | 95.71 | 97.38 |
| DLSE | 85.34 | 92.80 | 93.74 | 94.62 |
| RPCA | 75.90 | 83.06 | 87.98 | 91.95 |
| LR | 86.49 | 92.84 | 95.02 | 95.63 |
| SLRM | 84.83 | 91.86 | 95.54 | 96.21 |
| PCE | 86.74 | 90.87 | 93.42 | 95.42 |
| XGBoost | 87.87 | 91.86 | 94.56 | 96.13 |
| MSRL | 91.16 | 94.69 | 96.37 | 97.24 |
| MSSESEMIBOOST | 95.46 | 97.64 | 98.84 | 98.92 |

2) EXPERIMENTAL RESULTS ON CMU PIE FACE DATABASE

Table 5 lists the experimental results of 13 comparison methods on CMU PIE face database. The results of the first 12 methods were copied from the respective papers.

Fig. 8 shows the results of some typical algorithms.

As can be seen from table 5 and Figure 8, the proposed MSSESEMIBOOST method achieves the highest average recognition rate, which also verifies the effectiveness of our method. Compared with other algorithms, MSSESEMIBOOST achieves about 1.6% improvement on CMU PIE face database.

3) EXPERIMENTAL RESULTS ON AR FACE DATABASE

Table 6 lists the experimental results of 13 comparison methods on AR face database. The results of the first 12 methods were copied from the respective papers.

Fig. 9 shows the results of some typical algorithms.

As can be seen from table 6 and Figure 9, the proposed MSSESEMIBOOST method achieves the highest average recognition rate, which also verifies the effectiveness of our method. Compared with other algorithms, MSSESEMIBOOST achieves about 3% improvement on AR face database.

4) EXPERIMENTAL RESULTS ON COIL-100 DATABASE

Table 7 lists the experimental results of 13 comparison methods on COIL-100 database. The results of the first 12 methods were copied from the respective papers.

Fig. 10 shows the results of some typical algorithms.

compare with the proposed MSSESEMIBOOST method in this article, so as to evaluate the classification performance of MSSESEMIBOOST method. All experiments were repeated 20 times, and then the average recognition rates were calculated. The average recognition results are shown in table 4-7, and relations of the number of training samples from each subject to average recognition accuracy on four datasets are given in Fig7-10.

1) EXPERIMENTAL RESULTS ON EXTENDED YALEB FACE DATABASE

Table 4 lists the experimental results of 13 comparison methods on EXTENDED YALEB face database. The results of the first 12 methods were copied from the respective papers.

Fig. 7 shows the results of some typical algorithms.

As can be seen from table 4 and Figure 7, the proposed MSSESEMIBOOST method achieves the highest average recognition rate, which also verifies the effectiveness of our method. Compared with other algorithms, MSSESEMIBOOST achieves about 2% improvement on EXTENDED YALEB face database. The recognition rate increases with the increase of training samples.
As can be seen from table 7 and Figure 10, the MSPASEMIBOOST method proposed in this article achieves the highest average recognition rate, which also verifies the effectiveness of our method. Compared with other algorithms, MSPASEMIBOOST achieves about 4% improvement on COIL-100 database.

From table 4 to 7 and figure 7 to 10, we can see that the proposed MSPASEMIBOOST method achieves better classification performance in two different application scenarios. According to these experimental results, it can be concluded that:

1. Compared with the linear regression methods such as SLRM, the proposed MSPASEMIBOOST method has a significant improvement. This shows the necessity and superiority of learning prediction target from robust potential subspace and adaptive probability graph structure.

2. SLRM and the proposed MSPASEMIBOOST method are better than those based on sparse, low rank representation and dictionary learning. The main reason is that they focus on exploring the best reconstruction of the original data, which does not mean the best discrimination. This result clearly proves that the discriminant information encoded by the margin constraint and the graph regularization play a positive role in improving the intra-class compactness.

3. DLSR, MSRL and the proposed MSPASEMIBOOST method achieves good results due to the relaxation of regression objectives. However, the margin targets of MSRL and MSPASEMIBOOST are directly learned from the data rather than binary regression target. The adaptive graph structure provides a better solution to the over fitting problem. This proves that these two properties are important for learning margin visual representation.

4. SLRM, DSRM, MSRL, the proposed MSPASEMIBOOST method are generally better than other methods, which shows the effectiveness of graph structure learning in predicting targets, and makes the scattered visually similar data share the common target implicitly together.

5. The classification recognition rate of the proposed MSPASEMIBOOST method based on modular sparse representation is significantly higher than that using global information. MSPASEMIBOOST achieves about 3%
improvement on AR face database, which proves the effectiveness of the modular method.

(6) The classification performance of the proposed MSPASEMIBOOST method is superior to that of other non-BOOST methods, and it benefits from the BOOST method combining several weak classifiers to produce a very accurate classifier.

(7) The proposed MSPASEMIBOOST method has a higher recognition rate than SEMIBOOST and XGBOOST methods. It benefits from the use of sparse representation to measure the similarity between unlabeled data and labeled data in the MSPASEMIBOOST method, and the margin representation and adaptive graph structure learning are integrated in the sparse representation, which can correctly maximize the multiclass margin and realize efficient and effective visual representation learning.

V. CONCLUSION

When the image has occlusion or pollution, residual errors generated by all classes in the polluted sub image may be similar, and residual error generated by the correct class may not be minimum, which affects the final discrimination result. To solve this problem, a semi-supervised boosting framework based on modular sparse representation (MSPASEMIBOOST) is proposed. The similarity algorithm based on modular sparse representation is used in semi-supervised boosting framework to seamlessly combine the local consistency and global consistency of the regression target into a common framework for data representation. The margin target learned from the data provides enough flexibility for the fitting regression task. At the same time, the potential information of the data is used to predict the target. Compared with other representation methods, the learning method of data representation described in this article has stronger information and discrimination ability. The problem is solved by iterative optimization strategy. In addition, compared with other algorithms, the proposed MSPASEMIBOOST method achieves about 2%, 1.6%, 3% and 4% improvement on EXTENDED YALEB, CMU PIE, AR face database and COIL-100 database respectively. Experimental results on four databases show that our method is superior to other data representation methods, which shows the effectiveness of the proposed MSPASEMIBOOST method.

REFERENCES

[1] O. Chapelle, B. Schölkopf, and A. Zien, “Adaptive computation and machine learning,” in Semi-Supervised Learning, Cambridge, MA, USA: MIT Press, 2006, pp. 5–10.

[2] C. Liu and P. C. Yuen, “A boosted co-training algorithm for human action recognition,” IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 9, pp. 1203–1213, Sep. 2011, doi: 10.1109/TCSVT.2011.2130270.

[3] P. K. Mallapragada, R. Jin, A. K. Jain, and Y. Liu, “SemiBoost: Boosting for semi-supervised learning,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 11, pp. 2000–2014, Nov. 2009, doi: 10.1109/TPAMI.2008.235.

[4] J. Tanha, “MSSBoost: A new multiclass boosting to semi-supervised learning,” Neurocomputing, vol. 314, pp. 251–266, Nov. 2018, doi: 10.1016/j.neucom.2018.06.047.

[5] J. Tanha, M. van Someren, and H. Afsarmanesh, “Boosting for multiclass semi-supervised learning,” Pattern Recognit. Lett., vol. 37, pp. 63–77, Feb. 2014, doi: 10.1016/j.patrec.2013.10.008.

[6] N. Heidari, Z. Moslehi, A. Mirzaei, and M. Safayani, “Bayesian distance metric learning for discriminative fuzzy c-means clustering,” Neurocomputing, vol. 319, pp. 21–33, Nov. 2018, doi: 10.1016/j.neucom.2018.08.071.

[7] F. Zang and J.-S. Zhang, “Label propagation through sparse neighborhood and its applications,” Neurocomputing, vol. 97, pp. 267–277, Nov. 2012, doi: 10.1016/j.neucom.2012.03.017.

[8] H. Cheng, Z. Liu, L. Hou, and J. Yang, “Sparsity-induced similarity measure and its applications,” IEEE Trans. Circuits Syst. Video Technol., vol. 26, no. 4, pp. 613–626, Apr. 2016, doi: 10.1109/TCSTV.2012.2225911.

[9] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, “A survey of sparse representation: Algorithms and applications,” IEEE Access, vol. 3, pp. 490–530, 2015, doi: 10.1109/ACCESS.2015.2430359.

[10] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, “Robust recovery of subspace structures by low-rank representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 1, pp. 171–184, Jan. 2013, doi: 10.1109/TPAMI.2012.88.

[11] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, “Learning with local and global consistency,” in Proc. Adv. Neural Inf. Process. Syst., vol. 16, no. 16, Sep. 2004, pp. 321–328, doi: 10.1007/s11222-007-9033-z.

[12] S. Li and Y. Fu, “Low-rank coding with b-matching constraint for semi-supervised classification,” in Proc. Int. Joint Conf. Artif. Intell., Beijing, China, 2013, pp. 1472–1478.

[13] Q. Wang, M. Lu, and J. Li, “Similarity learning based on sparse representation for semi-supervised boosting,” Int. J. Comput. Intell. Appl., vol. 17, no. 2, pp. 1–9, Jun. 2018, doi: 10.1142/S1469026818500116.

[14] Z. Zhang, L. Shao, Y. Xu, L. Liu, and J. Yang, “Marginal representation learning with graph structure self-adaptation,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 10, pp. 4645–4659, Oct. 2018, doi: 10.1109/TNNLS.2017.2772264.

[15] L. I. Kuncheva, Combining Pattern Classifiers: Methods Algorithms, Hoboken, NJ, USA: Wiley, 2004, pp. 21–35.

[16] T. Kam Ho, J. J. Hull, and S. N. Srihari, “Decision combination in multiple classifier systems,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 16, no. 1, pp. 66–75, Jan. 1994, doi: 10.1109/34.273716.

[17] Z. Jiang, Z. Lin, and L. S. Davis, “Label consistent K-SVD: Learning a discriminative dictionary for recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 11, pp. 2651–2664, Nov. 2013, doi: 10.1109/TPAMI.2013.88.

[18] F. Nie, X. Wang, and H. Huang, “Multiclass capped lp-norm SVM for robust classifications,” in Proc. AAAI Conf. Artif. Intell., San Francisco, CA, USA, 2017, pp. 2415–2421.

[19] S. Cui, L. Zhang, W. Zuo, and X. Feng, “A probabilistic collaborative representation based approach for pattern classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 2950–2959.

[20] Y. Xu, Z. Zhong, J. Yang, J. You, and D. Zhang, “A new discriminative sparse representation method for robust face recognition via l2 regularization,” IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 10, pp. 2233–2242, Oct. 2017, doi: 10.1109/TNNLS.2016.2580572.

[21] X. Fang, S. Teng, Z. Lai, Z. He, S. Xie, and W. K. Wong, “Robust latent subspace learning for image classification,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 6, pp. 2502–2515, Jun. 2018, doi: 10.1109/TNNLS.2017.2693221.

[22] S. Xiang, F.Nie, G. Meng, C. Pan, and C. Zhang, “Discriminative least squares regression for multiclass classification and feature selection,” IEEE Trans. Neural Netw. Learn. Syst., vol. 23, no. 11, pp. 1738–1745, Nov. 2012, doi: 10.1109/TNNLS.2012.2212721.

[23] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?” J. ACM, vol. 58, no. 3, pp. 1–31, 2011, doi: 10.1145/1970392.1970395.

[24] G. Liu and S. Yan, “Latent low-rank representation for subspace segmentation and feature extraction,” in Proc. Int. Conf. Comput. Vis., Nov. 2011, pp. 1615–1622.

[25] J. Tang, L. Yang, J. Yu, and M. K. Ng, “Semi-supervised low-rank mapping learning for multi-label classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Boston, MA, USA, Jun. 2015, pp. 1483–1491.

[26] X. Peng, J. Lu, Z. Yi, and R. Yan, “Automatic subspace learning via principal coefficients embedding,” IEEE Trans. Cybern., vol. 47, no. 11, pp. 3583–3596, Nov. 2017, doi: 10.1109/TCYB.2016.2572306.

[27] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 2016, pp. 785–794.
SHU HUA XU received the B.S. degree in computer science and technology from the East China University of Technology, Nanchang, Jiangxi, China, in 1998, and the M.S. degree in computer science and technology from Guizhou University, Guiyang, Guizhou, China, in 2004. She is currently pursuing the Ph.D. degree in computer science and technology with the Zhejiang University of Technology, Hangzhou, Zhejiang, China.

Since 2004, she has been a Lecturer with the School of Mathematics, Physics and Information Science, Shaoxing University, Shaoxing, Zhejiang. Her major research interests include image processing, pattern recognition, and machine learning.

FEI GAO was born in April 1974. He received the Ph.D. degree in mechanical engineering from Zhejiang University, Hangzhou, Zhejiang, China. He was a Visiting Scholar of Industrial Engineering with Pennsylvania State University, from November 1, 2007 to April 30, 2008. He is currently a Full Professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou. His current research interests include pattern recognition, image processing, and machine learning. His awards and honors include the 9th “My Favorite Teacher” Award of the Zhejiang University of Technology, the young and middle-aged academic leaders in colleges and universities of Zhejiang, and the third level of the “New Century 151 Talent Project” in Zhejiang.