Band Weighting via Maximizing Inter-class Distance for Hyperspectral Image Classification

Cheng Yan, Xiao Bai, Peng Ren, Lu Bai, Jun Zhou *Senior Member, IEEE*

***Abstract*—We present a novel band weighting strategy that exploits multiple binary support vector machines (SVMs) to maximize inter-class spectral distances for multi-class hyperspectral remote image classification. Specifically, we commence by training binary SVMs based on the original training samples. We then balance the bands of training samples by maximizing the modified classification scores for SVMs. This balance scheme enlarges the distances between individual training samples and the SVM hyperplane. For each class, we reformulate the binary SVM objective function based on the balanced training samples, resulting in a weighting vector that associates a weight to each spectral band for the class. For a testing sample, we weight it and then classify it by using the binary SVM, both with respect to every individual class. The classification result is obtained from the classifier with the greatest score. Experiments on two benchmark datasets show the effectiveness of the proposed strategy.**

***Index Terms*—Band weighting, support vector machines(SVMs), hyperspectral image classification.**

I. Introduction

Hyperspectral remote sensing images provide rich information of land-covers. Every pixel in a hyperspectral image can be represented as a high-dimensional vector across spectral dimensions. Different substances in a hyperspectral image exhibit distinctive spectral signatures. Therefore, most hyperspectral imagery classification approaches[1][2] extensively exploit the inter-class discriminative information extracted from high dimensional spectral space for classifying different objects. However, the high dimensionality of the spectral data leads to the expensive costs for computation and storage and also results in Hughes phenomenon[3] in signal processing. To address the challenges of high-dimensionality arising in processing the hyperspectral data, a widely accepted solution is to transform the original data in the spectral space to a target space of lower dimensions. In this regard, representative methods include projection pursuit (PP)[4], principal component analysis (PCA)[5] and independent component analysis (ICA)[6]. One drawback of these transformation based methods is that they tend to modify the original imagery data in a heuristic manner such that certain useful spectral characteristics are not preserved. Another family of approaches are based on band selection which keeps a subset of the original bands[7][8]. Though reducing the redundancy of the spectral data, the band-selection methods may suffer from the possibility of discarding bands that are useful for classification.

C. Yan and X. Bai are with School of Computer Science and Engineering, Beihang University, Beijing, China (e-mail: beihangyc@buaa.edu.cn; baixiao.buaa@gmail.com).

P. Ren is with College of Information and Control Engineering, China University of Petroleum (Huadong), Qingdao, China (e-mail: pengren@upc.edu.cn).

L. Bai is with School of Information, Central University of Finance and Economics, Beijing, China (e-mail: bailu69@hotmail.com).

J. Zhou is with School of Information and Communication Technology, Griffith University, Nathan, QLD, Australia (e-mail:jun.zhou@griffith.edu.au).

To render a more effective band selection strategy, band weighting methods[9][10][11][12] have recently been broadly studied. The purpose of band weighting is to assign each band of the hyperspectral data a weight which is supposed to reflect the importance of the band. In the literature, Qi et al. [11] proposed a spectral weighting algorithm by using similarity entropy and separation coefficients computed from the hyperspectral data. Imani et al. [12] presented a supervised band weighting method by discovering the relative importance of each band from the training set. These band weighting methods avoid bruteforcely discarding the seemingly useless bands and hence are believed to be capable of preserving all spectral characteristics. However, these methods are not capable of providing multiple weight configurations for different classes in terms of improving inter-class discriminability.

We observe that most state of the art band weighting methods do not consider the class specific information residing in different spectra. To address this shortcoming, we develop a new band weighting method based on SVMs. The contributions of our paper are two-folds. Firstly, instead of computing the spectral weights for the whole data, we develop a class specific band weighting scheme which not only preserve all spectral characteristics but also encodes the importance of each band. Specifically, we develop a band weight vector for each class via maximizing class distance based on a SVM scheme. Secondly, we design classifiers based on the resulted class-specific spectral weight vectors and hence improve the classification performance over alternative band weighting methods.

Fig. 1 illustrates the main steps for implementing our band weighting method. Accordingly, the organization of this paper is presented as follows. In Section II-A, we use a set of training samples to train a family of traditional binary SVM classifiers for each class of the hyperspectral data (Fig. 1A). In Section II-B, we compute a modified SVM classification score for each sample, which reflects the distance between the sample and the hyperplane. We balance the spectral bands for each training sample by maximizing the classification score sum (Fig. 1B). In Section II-C, we develop a new class-specific SVM objective function based on the balanced training samples and compute a class specific spectral weight vector via maximizing class distance (Fig. 1C). Finally, we retrain each binary SVM classifier by using the weighted training samples (Fig. 1D). For a test sample, we weight it by using the spectral weight vector of one class and compute the corresponding classification score with respect to the class. We categorize it into the class whose SVM classifier results in the highest classification score. In Section III, we empirically evaluate our method and the experimental results validate the effectiveness of our proposed method on two public benchmark hyperspectral datasets.

II. Band Weighting

A. Initial SVM Training

For each class, we use training data for this class as positive samples and the rest as negative samples to train a binary linear SVM classifier. The objective function is

$$L(w) = \frac{1}{2}\|w\|^2 + C\sum_i max(0, 1 - y_i f(x_i, w)) \quad (1)$$

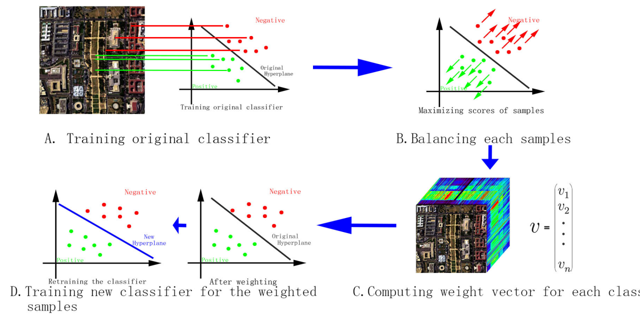

Fig. 1. Key steps of the proposed method. A. Training the initial classifier for each class by using traditional SVM. B. Balancing the spectral bands for each sample by maximizing the modified SVM classification scores. C. Computing the spectral weight vector for each class. D. Training new classifiers by using weighted samples.

where w represents the SVM classification hyperplane and x_i represents a hyperspectral pixel training sample with n bands. C is a penalty parameter that controls the model complexity, y_i is the class label, and $f(\cdot)$ is the classification score for the initial SVM training. The implementation details of SVM can be found in [13]. Here $w^T x_i$ is a measure to evaluate the distance between the sample and the hyperplane.

B. Balancing Spectral Bands for Training Samples

After the initial training for each class, we balance the spectral bands for each sample in the training set. To this end, we define a modified classification score for each sample as follows

$$g(x_i, w, b_i, v) = w^\top T(b_i \circ x_i) - \gamma ||b_i - v||^2 \quad (2)$$

Here b_i is an n-dimensional vector representing the balance vector for the sample x_i. We assign a balance vector b_i to each sample x_i by the entry-wise product $b_i \circ x_i$, resulting in a balanced sample of n dimensions. We aim to maximize the classification score (2) for obtaining optimal balance vectors. In this case, the balance vector b_i that enlarges the the distance $w^T(b_i \circ x_i)$ between the balanced sample $b_i \circ x_i$ and the hyperplane is favored. On the other hand, the term $\gamma||b_i - v||^2$ in (2) avoids over-fitting problems. Here v is a class weight vector and all its element are set to be 1 for the time being in the process of computing the optimal balance vector b_i^*. γ is a parameter that leverages the contribution of the two terms in (2). A large γ forces b_i to be close to v, and thus results in more uniform distributed balanced training samples. We maximize the sum of the classification scores of the training samples to obtain the optimal balance vector b_i for each sample. Note that, the classification score in Eq.2 is based on original hyperplane which is just for training. The gradient descent algorithm[14] is used to perform the numerical computation for maximization.

C. Band Weighting Vector for Each Class

In the previous subsection, we have obtained the spectral balanced training samples that improve classification scores over the original training samples. However, the balanced trained samples are computed based on the assumption that all entries of the class-specific weight vector v are 1. To enhance the effectiveness for classification one step further, we reformulate the objective function of the binary class-specific SVM to derive an optimal weight vector v for each class. The new SVM objective function is

$$\begin{aligned} L^{\text{new}}(w, v) &= \frac{1}{2}\|w\|^2 + \frac{1}{2}\ \theta||v - \mathbf{1}||^2 \\ &+C\sum_i max(0, 1 - y_i g(x_i, w, b_i^*, v)) \end{aligned} \tag{3}$$

In (3), the contribution of the regularization term $||v - \mathbf{1}||^2$ is controlled by θ. When θ is large, the regularization term tends to force the entries of the weighting vector to be the same for all bands, and the original spectral data is thus mostly maintained. On the contrary, when θ is small, the resulting weights may differ significantly for different bands. The score $g(x_i, w, b_i^*, v)$ computed based on balanced training samples is involved in the the objective function (3). In the previous section, v is defaulted as an all 1 vector. Here we consider v as a parameter vector to be optimized.

To minimize the objective function (3), the gradient descent algorithm[14] is used again to compute the class weight vector v. Finally, we weight each training sample with the weight vector of the class that the sample belongs to and then retrain the classifiers according to Equation (1). For a test sample, the test classification score is

$$Score = w^{\top} v \circ x \tag{4}$$

we process test sample x by using each classifier after weighting it with the corresponding class weight vector v. We obtain the classification result from the retrained SVM classifier with the highest score. This algorithm for computing the weight vector for each hyperspectral class is summarized in Algorithm 1.

Algorithm 1: The training framework.

Data: Training data $X = \{x_i\}$ and labels $Y = \{y_i\}$
for Each class $c = 1 \cdots M$(number of classes) **do**
  Train an initial SVM classifier using Equation (1).
  for Each sample $i = 1 \cdots N$ **do**
    Optimize b_i using Equation (2)
  end
  Optimize v for this class using Equation (3)
  Weight training samples and retrain the SVM classifier
  Output v and the corresponding w
end

The major operations involved in our method is to assign each sample a weight vector and then calculate a weight vector for each class. The time complexity is $O(MN)$ where M is the number of iterations of gradient descent and N is the number of training samples.

III. Experimental Evaluation

In this section, we present the experimental results on two datasets, i.e., Indian Pines AVIRIS and Salinas AVIRIS. We use average classification accuracy to evaluate alternative methods.

A. Dataset Description

The Indian Pines dataset was acquired by AVIRIS sensors over the Indian Pines test site in Northwestern Indiana. It consists of $145 * 145$ pixels and 224 spectral bands across the wavelength range of 0.4 to $2.5\ m$, with 16 classes. The second dataset is Salinas scene which was collected by the AVIRIS sensor over Salinas Valley, California. It has 224 bands over 0.4 to $2.5\ m$ with 16 classes. In our experiment, the number of bands is reduced to 200 with low-SNR and water-absorption bands removed.

B. Performance Analysis

We randomly selected a part of samples in each class from both datasets as the training sets and the remaining data as the testing sets. We compare our method with several alternative band weighting methods in the literature, i.e. Customizing Kernel Function (CKF) [9], Bacterial Foraging Optimization (BFO) [10], SVM-based Feature Weighting (CSC-SVM and SE-SVM) [11] and Feature Extraction using Weighted Training samples (FEWT) [12]. We also compare our method with the original linear SVM [13]. Note that the FEWT scheme and our method exploit weighted samples at different stages. The FEWT scheme extracts features based on weighted training samples; in contrast, our framework performs classification using weighted samples. To make fair comparisons, we follow the optimal strategies suggested by the authors in their original papers to estimate the parameter values, e.g. those for cross validations, for the comparison methods. Alternatively, we set the parameter values to be those achieving best performance on the dataset. The main difference between our method and these methods is that we expanding the distance between classes by weighting samples.

At the training stage, a large γ forces b_i to be close to v, and thus leads to more uniform expanding of training samples. Table I shows the average classification results on Salinas dataset with varied parameter values. The results imply that our method is not very sensitive to the parameters. We empirically set γ to be 1 and θ to be 10 as they result in best performance in experiments. All experiments are conducted by using MATLAB on a $3.4GHz$ machine with 20GB RAM.

TABLE I
ACCURACIES BASED ON DIFFERENT PARAMETER VALUES.

θ \ γ	0.1	1	5	10	100
0.1	95.1	97.7	97.6	97.6	97.0
1	96.4	98.5	98.3	97.9	97.5
5	97.6	98.6	98.9	98.0	97.5
10	98.0	99.0	98.5	98.1	97.9
100	97.6	97.5	97.9	98.9	96.5

The classification results are given in Tables II and III. The top 5 bands with the largest weight for each class are also given. Our method is a band weighting strategy which keeps all bands but with different weights in the classification task. Furthermore, the class weight is also developed for each class based on the dataset. We process the test sample by using each classifier after weighting it with the corresponding class level weight vector.

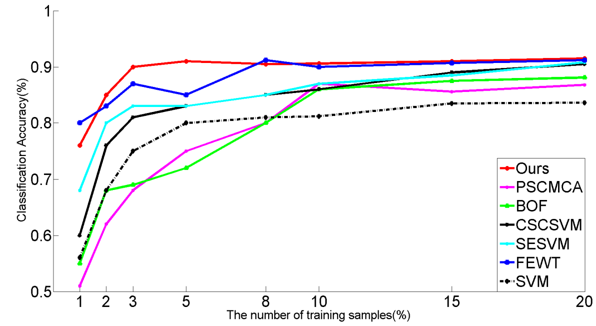

Fig. 2. The results for different percent of training samples.

Since the training samples are randomly selected, we show the average accuracy of performing experiments for ten times for each class from two datasets. To show the performance in small training samples, we represent the experimental results for different percent of training samples in Fig.2.

We observe that our method performs better than the alternatives in terms of classification accuracy. We also evaluate the execution time for different methods. From the result table, we observe that the CSCSVM and SESVM have better classification performance than the CKF and BFO with more expensive time costs. Even though our method is not the fastest, the execution time is comparable to alternatives and acceptable for applications.

IV. Conclusion

In this paper, we have introduced a new hyperspectral band weighting method based on the optimization of the classification results of SVM. We have balanced the spectral bands of individual samples based on initial SVM training. Furthermore, we have developed a new SVM objective function that results in a band weight vector for each class. Finally, we have trained new classifiers with weighted training samples. Our strategy is effective because it not only preserves spectral properties but also characterizes band importance for classification. Experimental evaluations have shown that the proposed method has better performance than several alternative state-of-the-arts methods.

References

[1] W. Li, S. Prasad, and J. E. Fowler, "Hyperspectral image classification using gaussian mixture models and markov random fields," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 1, pp. 153–157, 2014.

TABLE II
NUMBER OF TRAINING AND TESTING SAMPLES AND AVERAGE CLASSIFICATION ACCURACY ON THE INDIAN PINES DATA.

Class	Train	Test	Ours Ac (%)	PSCMCA Ac (%)	BFO Ac (%)	CSCSVM Ac (%)	SESVM Ac (%)	FEWT Ac (%)	SVM Ac (%)	Bands with largest weight
C1 - Alfalfa	5	50	94.0	90.0	82.0	86.0	90.0	98.0	54.3	6,27,28,85,94
C2 - Corn-N	142	1420	93.0	94.4	91.0	95.0	94.1	69.2	73.3	2,4,86,91,101
C3 - Corn-M	83	830	89.0	90.0	87.0	88.1	88.4	75.6	65.0	28,32,53,56,67
C4 - Corn	23	230	80.0	70.0	75.0	90.4	91.2	0.79	79.2	13,117,119,125,184
C5 - Grass-P	48	480	91.0	86.0	87.5	89.3	89.3	90.0	80.1	31,43,44,45,89
C6 - Grass-T	73	730	95.1	89.6	91.1	93.3	94.5	89.0	65.2	21,22,115,116,118
C7 - Grass-P-M	3	30	80.0	83.3	86.6	90.0	90.0	96.0	89.3	13,17,96,97,99
C8 - Hay-W	47	470	97.0	92.2	93.4	95.3	95.3	84.0	90.0	35,64,69,73,75
C9 - Oats	2	20	80.0	75.0	70.0	85.0	85.0	100	85.0	2,3,23,24,71
C10 - Soybean-N	97	970	90.1	88.1	89.6	90.1	89.9	80.5	73.5	38,43,44,56,57
C11 - Soybean-M	245	2450	90.1	76.5	90.0	87.6	88.9	85.0	84.6	3,87,94,96,97
C12 - Soybean-C	59	590	87.0	90.0	85.7	86.5	88.5	85.2	83.7	40,55,56,72,73
C13 - Wheat	20	200	98.0	89.0	88.0	91.0	91.0	91.0	90.1	19,20,34,38,75
C14 - Woods	126	1260	98.0	96.0	97.0	97.5	98.2	97.5	95.5	14,31,61,84,88
C15 - Buildings-G	38	380	87.1	90.5	87.1	91.6	90.5	91.6	94.3	115,117,123,164,165
C16 - Stone-S	9	90	96.7	76.6	83.0	90.0	92.2	100	90.3	61,90,128,134,161
TOTAL	1020	10200	91.5	87.0	88.1	90.5	90.8	91.2	83.6	
Time(s)			20.24	9.00	31.60	16.50	18.00	15.00	5.00	

TABLE III
NUMBER OF TRAINING AND TESTING SAMPLES AND ACCURACY FOR EACH CLASS ON THE SALINAS DATA.

Class	Train	Test	Ours Ac (%)	PSCMCA Ac (%)	BFO Ac (%)	CSCSVM Ac (%)	SESVM Ac (%)	FEWT Ac (%)	SVM Ac (%)	Bands with largest weight
C1 - Broccoli-1	12	2000	99.6	94.5	97.9	96.3	96.3	97.8	93.5	20,37,39,100,101
C2 - Broccoli-2	15	3500	99.4	90.4	96.8	99.2	99.4	99.4	91.2	36,37,70,93,94,
C3 - Fallow	11	1700	96.4	90.0	99.6	98.0	98.5	96.6	90.5	32,33,35,36,37
C4 - Fallow-R	10	1100	99.7	91.6	96.4	99.2	99.0	99.5	91.8	5,7,10,11,12
C5 - Fallow-S	15	2500	98.8	96.8	98.8	98.0	96.6	98.6	97.0	6,19,21,92,93
C6 - Stubble	15	3900	99.6	92.0	98.9	98.6	98.6	99.0	92.4	27,28,31,36,37
C7 - Celery	15	3500	98.9	97.2	96.9	97.5	97.6	99.1	98.0	64,148,149,154,196
C8 - Grapes-U	15	10000	97.0	94.4	97.8	98.8	98.8	97.5	95.6	19,34,42,45,48
C9 - Sell-V	15	5500	98.4	98.3	95.4	95.2	96.0	98.0	98.3	43,52,53,55,56
C10 - Corn-S	15	3000	97.8	94.5	95.3	99.0	98.8	99.3	95.0	32,69,70,76,92
C11 - L-romain-4	10	900	99.3	91.0	97.7	97.0	97.4	99.2	92.5	29,34,71,101,152
C12 - L-romain-5	12	1800	99.6	91.5	98.9	95.9	96.1	98.7	93.1	13,31,33,120,121
C13 - L-romain-6	10	800	99.4	95.1	99.7	96.0	96.0	99.0	95.0	19,41,47,164,167
C14 - L-romain-7	10	800	99.4	93.7	95.6	96.8	96.8	95.2	94.0	14,122,135,137,168
C15 - Vineyard-U	15	6500	99.7	95.0	97.8	99.0	98.9	98.7	96.1	2,5,107,108,148
C16 - Vineyard-V	10	1500	100.0	97.3	97.3	99.6	99.6	98.8	99.0	9,166,172,185
TOTAL	205	49000	99.0	94.5	97.4	97.9	98.0	98.9	95.6	
Time(s)			6.08	3.57	7.31	5.10	5.50	5.00	3.00	

[2] X. Bai, H. Zhang, and J. Zhou, "Vhr object detection based on structural feature extraction and query expansion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 10, pp. 6508–6520, 2014.

[3] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.

[4] A. Ifarraguerri and C. Chang, "Unsupervised hyperspectral image analysis with projection pursuit," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 6, pp. 2529–2538, 2000.

[5] A. Agarwal, T. El-Ghazawi, H. El-Askary, and J. Le-Moigne, "Efficient hierarchical-PCA dimension reduction for hyperspectral imagery," *IEEE International Symposium on Signal Processing and Information Technology*, pp. 353–356, 2007.

[6] Q. Du and H. Yang, "Similarity-based unsupervised band selection for hyperspectral image analysis," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 564–568, 2008.

[7] X. Bai, Z. Guo, Y. Wang, Z. Zhang, and J. Zhou, "Semisupervised hyperspectral band selection via spectral–spatial hypergraph model," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2774–2783, 2015.

[8] N. H. Ly, Q. Du, and J. E. Fowler, "Sparse graph-based discriminant analysis for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 7, pp. 3872–3884, 2014.

[9] B. Guo, S. R. Gunn, R. I. Damper, and J. D. Nelson, "Customizing kernel functions for svm-based hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 622–629, 2008.

[10] A. Chakrabarty, O. Choudhury, P. Sarkar, A. Paul, and D. Sarkar, "Hyperspectral image classification incorporating bacterial foraging-optimized spectral weighting," *Artificial Intelligence Research*, vol. 1, no. 1, p. p63, 2012.

[11] B. Qi, C. Zhao, and G. Yin, "Feature weighting algorithms for classification of hyperspectral images using a support vector machine," *Applied optics*, vol. 53, no. 13, pp. 2839–2846, 2014.

[12] M. Imani and H. Ghassemian, "Feature extraction using weighted training samples," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 7, pp. 1387–1386, 2015.

[13] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, p. 27, 2011.

[14] J. Snyman, *Practical mathematical optimization: an introduction to basic optimization theory and classical and new gradient-based algorithms*. Springer Science& Business Media, 2005, vol. 97.