STRUCTURED BINARY FEATURE EXTRACTION FOR HYPERSPECTRAL IMAGERY CLASSIFICATION

Zisha Zhong, Bin Fan, Jun Bai, Shiming Xiang, Chunhong Pan

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences

ABSTRACT

In this paper, we propose a novel structured binary feature extraction method for hyperspectral image classification. To pursuit high discriminative ability and low memory cost, we resort to applying the learning to hash technique to the traditional spectral-spatial hyperspectral features. We show how the structured information among different kinds of features and different feature groups can be used to learn discriminative binary features for classification. Experiments on two standard benchmark hyperspectral data sets demonstrate the effectiveness of the proposed method.

Index Terms— binary feature extraction, structured regularization, hyperspectral image classification, learning to hash

1. INTRODUCTION

Hyperspectral imaging sensors can provide us with images of hundreds of spectral bands at each pixel [1]. A vital application of hyperspectral images is land-cover classification, which classifies pixels into multiple predefined categories [2].

Although it has been studied over a decade, it is still an active research topic due to its difficulties and importance. The existing methods can be categorized into three classes [3]: (1) spectral-spatial feature extraction, (2) spatial-spectral image segmentation or post-processing, and (3) other methods, e.g. multiple kernel learning, etc. In terms of spectral-spatial feature extraction, representative techniques include Gabor filtering, gray-level concurrence matrices, extended morphological profiles (EMP, [4]), extended attribute profiles (EAP, [5]), etc. These methods can obtain satisfactory performances on hyperspectral imagery classification. However, these single kind of features can only describe some characteristics of the considered pixel. Some researchers integrated multiple types of features in order to further improve classification.

Previous researches demonstrate that high feature redundancy requires dimension reduction for decreasing the computational cost. In general, dimension reduction can be achieved by feature extraction or feature selection. Representative feature extraction technologies include principal component analysis (PCA), nonparametric weighted feature extraction (NWFE, [6]), kernel linear discriminant analysis (KLDA, [7]), etc.. All these methods consider features using float-vector representation, few works are focused on obtaining more compact features in binary space.

In order to get a compact binary feature representation (i.e., binary codes), learning to hash technique has been well studied in recent year [8]. Very recently, Demir and Bruzzone [9] have introduced the hashing technique to scalable image retrieval in large remote sensing archives. More recently, Zhong et al. [10] have also conducted a comparative study on hashing based multiple feature fusion in hyperspectral imagery classification, demonstrating the promising potential of using hashing in remote sensing community.

In this paper, we propose a learning-to-hash based structured binary feature extraction method on multiple features for hyperspectral image classification. Our method is inspired by the supervised discrete hashing (SDH, [11]), which aims to learn hash functions to maximize the classification accuracy and simultaneously minimize the quantization errors. However, for hyperspectral images, it is usual to have some redundant elements in the extracted features. Meanwhile, when it comes to combine multiple features for better performance, some modalities may be redundant too. These factors should be considered in learning hashing functions for hyperspectral image classification. To this end, we extend SDH with two regularized terms to better handle the problem of hyperspectral classification. Since our formulation takes structural information into consideration when learning hashing functions, we call our method structured SDH (SSDH).

2. THE PROPOSED METHOD

In this section, we first briefly introduce the learning to hashing technique and the supervised discrete hashing. Then we give details about our model and its optimization algorithm.

2.1. Supervised Discrete Hashing

Given N training samples $\{\mathbf{x}_i, y_i\}_{i=1}^N$ where $\mathbf{x}_i \in \mathbb{R}^{1\times d}$ is the i-th training point, $y_i \in \{1, \cdots, c\}$ is the class label of i-th training point and c is the number of labeled classes. The learning-to-hash is to learn a set of hash

This work was supported by the National Natural Science Foundation of China under Grants No. 61573352, 61472119, 91646207, 61403377, 91438105, and 61375024.

functions $\{h_l(\mathbf{x})\}_{l=1}^{r}$ to map the original high-dimensional float-type feature $\mathbf{x}_i$ into a low-dimensional binary code $\mathbf{z}_i = [h_1(\mathbf{x}_i), h_2(\mathbf{x}_i), \cdots, h_r(\mathbf{x}_i)] \in \{-1,1\}^{1\times r}$. To obtain discriminative and compact binary codes, a lot of research endeavors have been devoted on the design of hash functions based on different strategies [8]. Recently, the supervised discrete hashing (SDH, [11]) is proposed to maximize classification accuracy and minimize quantization error, which can be formulated as:

$$\begin{aligned}&\min_{\mathbf{B},\mathbf{W},\mathbf{P}} ||\mathbf{Y}-\mathbf{B}\mathbf{W}||_F^2 + \lambda_1 ||\mathbf{W}||_F^2 + \lambda_2 ||\mathbf{B}-F(\mathbf{X})||_F^2\\ &s.t.\ \mathbf{B}\in\{-1,1\}^{N\times r},\end{aligned}$$

where $\mathbf{X}\in\mathbb{R}^{N\times d}$ is the set of N training samples with d-dimensional features. $F(\mathbf{X})$ is a linear or nonlinear embedding (e.g., kernel mapping) of $\mathbf{X}$. $\mathbf{Y}\in\mathbb{R}^{N\times c}$ is the label matrix and the i-th row $\mathbf{y}_i\in\mathbb{R}^{1\times c}$ is the one-hot based label vector. $\mathbf{W}\in\mathbb{R}^{r\times c}$ is the classification coefficients based on the learned $\mathbf{B}$. The first two terms formulate the learning-to-hash as a linear classification problem and the third term models the fitting error of the binary codes by the embedding $F(\mathbf{X})$. λ_1 and λ_2 are regularization parameters.

2.2. The Proposed Model

In our method, we propose to learn compact binary features from multiple spatial-spectral features. Therefore, each sample x_i in our formulation is a concatenated vector of V groups of features. For the v-th feature group, supposing its dimension is d_v $(v=1,\cdots,V)$, its corresponding data matrix can be denoted as $\mathbf{X}^{(v)}\in\mathbb{R}^{N\times d_v}$. Therefore, the whole data matrix is $\mathbf{X}\in\mathbb{R}^{N\times d}\left(d=\sum_{v=1}^{V}d_v\right)$. Meanwhile, we choose to use the linear embedding as our hash function for its simplicity, i.e., $F(\mathbf{X};\mathbf{P},\mathbf{t}) = \mathbf{X}\mathbf{P}+\mathbf{1}_N\mathbf{t}^T$, $\mathbf{P}\in\mathbb{R}^{d\times r}$ is a linear transformation matrix, $\mathbf{t}\in\mathbb{R}^{1\times r}$ is a bias term and $\mathbf{1}_N\in\mathbb{R}^{N\times 1}$ is an all-one vector.

As we explained before, SDH does not take the intrinsic relationship among different features in the original feature space. **On one hand**, as some features in the original space may be not useful for generating good binary features (e.g., due to noisy or irrelevant features), it is desirable to have some rows of the projection matrix $\mathbf{P}$ be all zeros. This motivates us to adopt the $L_{2,1}$-norm regularizer [12]

$$||\mathbf{P}||_{2,1} = \sum_{i=1}^{d}||\mathbf{p}_{i,:}||_2 = \sum_{i=1}^{d}\sqrt{\sum_{j=1}^{r}P_{ij}^2},$$

where $\mathbf{p}_{i,:}$ is the i-th row of $\mathbf{P}$. **On the other hand**, to model the structured information in different feature modalities, we introduce the group structured regularization, i.e., the $G_{2,1}$-norm , which is defined as

$$||\mathbf{P}||_{G_{2,1}} = \sum_{v=1}^{V}\sqrt{\sum_{i\in g_v}\sum_{j=1}^{r}P_{ij}^2} = \sum_{v=1}^{V}\left|\left|\mathbf{P}^{(v)}\right|\right|_F,$$

where g_v is a set of indices belonging to the v-th feature, $\mathbf{P}^{(v)}\in\mathbb{R}^{d_v\times r}$ is the sub-matrix related to v-th feature. With the above two structured terms, the proposed SSDH model is formulated as follows:

$$\begin{aligned}&\min_{\mathbf{B},\mathbf{W},\mathbf{P},\mathbf{t}} ||\mathbf{Y}-\mathbf{B}\mathbf{W}||_F^2 + \lambda_1||\mathbf{W}||_F^2\\ &+\lambda_2\left(\left|\left|\mathbf{B}-\mathbf{X}\mathbf{P}-\mathbf{1}_N\mathbf{t}^T\right|\right|_F^2 + \beta_1||\mathbf{P}||_{2,1} + \beta_2||\mathbf{P}||_{G_{2,1}}\right)\\ &s.t.\ \mathbf{B}\in\{-1,1\}^{N\times r},\end{aligned}$$

where λ_1, λ_2, β_1 and β_2 are regularization parameters.

2.3. Optimization

In general, the problem of SSDH is NP hard and difficult to solve due to the binary constraint on $\mathbf{B}$. One common method is to adopt the alternative optimization technique.

(1) $\mathbf{P},\mathbf{t}$**-subproblem**. Fix $\mathbf{B},\mathbf{W}$, update $\mathbf{P},\mathbf{t}$. From the derivation in [12], we can simply reformulate the $\mathbf{P},\mathbf{t}$-subproblem into a standard least squares problem, for which we can easily derive the closed solutions.

(2) $\mathbf{W}$**-subproblem**. Fix $\mathbf{P},\mathbf{t}$ and $\mathbf{B}$, we solve $\mathbf{W}$. It can be easily derived that the solution is $\mathbf{W} = \left(\mathbf{B}^T\mathbf{B}+\lambda_1\mathbf{I}_r\right)^{-1}\mathbf{B}^T\mathbf{Y}$.

(3) $\mathbf{B}$**-subproblem**. Fix $\mathbf{P}$, $\mathbf{t}$ and $\mathbf{W}$, we solve $\mathbf{B}$. Let $\mathbf{R} = \mathbf{X}\mathbf{P}+\mathbf{1}_N\mathbf{t}^T$, the subproblem can be written as

$$\begin{aligned}&\min_{\mathbf{B}} \quad ||\mathbf{Y}-\mathbf{B}\mathbf{W}||_F^2 + \lambda_2||\mathbf{B}-\mathbf{R}||_F^2\\ \Leftrightarrow\ &\min_{\mathbf{B}} \quad tr\left(\mathbf{Y}^T\mathbf{Y}-2\mathbf{Y}^T\mathbf{B}\mathbf{W}+\mathbf{B}\mathbf{W}\mathbf{W}^T\mathbf{B}^T\right)\\ &\qquad +\lambda_2 tr\left(\mathbf{B}^T\mathbf{B}-2\mathbf{B}^T\mathbf{R}+\mathbf{R}^T\mathbf{R}\right)\\ &s.t. \quad \mathbf{B}\in\{-1,1\}^{N\times r}\end{aligned}$$

Since $\mathbf{B}\in\{-1,1\}^{N\times r}$, $tr\left(\mathbf{B}^T\mathbf{B}\right)$ is a constant. By denoting $\mathbf{Q}\leftarrow\mathbf{Y}\mathbf{W}^T+\lambda_2\mathbf{R}$, the subproblem is equivalent to the following form:

$$\begin{aligned}&\min_{\mathbf{B}} \quad tr\left(\mathbf{B}\mathbf{W}\mathbf{W}^T\mathbf{B}^T\right) - 2tr\left(\mathbf{B}^T\mathbf{Q}\right)\\ &s.t. \quad \mathbf{B}\in\{-1,1\}^{N\times r}\end{aligned}$$

Similar to SDH [11], we adopt discrete cyclic coordinate descent method to solve the subproblem. For complement, we briefly describe it here. In each iteration, we learn one column of $\mathbf{B}$ with others fixed. Without loss of generality, suppose we learn the l-th column, $l=1,\cdots,r$. Based on MATLAB expression, let $\mathbf{z} = \mathbf{B}(:,l)\in\{-1,1\}^{N\times 1}$ being the l-th column of $\mathbf{B}$, $\mathbf{B}_1 = \mathbf{B}(:,[1,\cdots,l-1,l+1,\cdots,r])\in\{-1,1\}^{N\times(r-1)}$, $\mathbf{v} = \mathbf{W}(l,:)^T\in\mathbb{R}^{c\times 1}$, $\mathbf{q} = \mathbf{Q}(:,l)\in\mathbb{R}^{N\times 1}$, $\mathbf{W}_1 = \mathbf{W}([1,\cdots,l-1,l+1,\cdots,r],:)\in\mathbb{R}^{(r-1)\times c}$, $\mathbf{Q}_1 = \mathbf{Q}(:,[1,\cdots,l-1,l+1,\cdots,r])\in\mathbb{R}^{N\times(r-1)}$, then we have

$$\begin{aligned}&tr\left(\mathbf{B}\mathbf{W}\mathbf{W}^T\mathbf{B}^T\right)\\ &= tr\left(\mathbf{z}\mathbf{v}^T\mathbf{v}\mathbf{z}^T\right) + tr\left(\mathbf{B}_1\mathbf{W}_1\mathbf{W}_1^T\mathbf{B}_1^T\right) + 2tr\left(\mathbf{v}^T\mathbf{W}_1^T\mathbf{B}_1^T\mathbf{z}\right)\\ &= const + 2tr\left(\mathbf{v}^T\mathbf{W}_1^T\mathbf{B}_1^T\mathbf{z}\right)\\ &tr\left(\mathbf{B}^T\mathbf{Q}\right) = tr\left(\mathbf{B}_1^T\mathbf{Q}_1\right) + tr\left(\mathbf{q}^T\mathbf{z}\right) = const + tr\left(\mathbf{q}^T\mathbf{z}\right)\end{aligned}$$

Thus, for solving the l-column of $\mathbf{B}$, the related $\mathbf{z}$-subproblem can be rewritten as

$$\begin{aligned} \min_{\mathbf{z}} \quad & \left(\mathbf{v}^T\mathbf{W}_1^T\mathbf{B}_1^T - \mathbf{q}^T\right)\mathbf{z} \\ s.t. \quad & \mathbf{z} \in \{-1, 1\}^{N\times 1}, \end{aligned}$$

where the solution can be derived as $\mathbf{z} = sign\left(\mathbf{q} - \mathbf{B}_1\mathbf{W}_1\mathbf{v}\right)$. We iteratively update each column one by one until the obtained $\mathbf{B}$ converges.

With the learnt linear hash transformation $\mathbf{P}$ and $\mathbf{t}$, we can reduce a test sample $\mathbf{x}$ into binary codes using $\mathbf{b} = sign\left(\mathbf{x}\mathbf{P} + \mathbf{t}\right)$. The proposed method is summarized in Algorithm 1.

Algorithm 1 SSDH algorithm.

Input: Training data $\mathbf{X}, \mathbf{Y}$; number of bits r; number of maximum iterations T; converge precision ϵ; parameters $\lambda_1, \lambda_2, \beta_1, \beta_2$.

Output: Binary codes $\mathbf{B}$; Hash transformation $\mathbf{P}$ and $\mathbf{t}$.

Initialize $\mathbf{B}$ and $\mathbf{P}$ randomly.
for $t = 1, \cdots, T$ **do**
    1) Solve $\mathbf{P}, \mathbf{t}$-subproblem;
    2) Solve $\mathbf{W}$-subproblem;
    3) Solve $\mathbf{B}$-subproblem as follows.
    while not converged **do**
        Set $\mathbf{B}_0 = \mathbf{B}$.
        for $l = 1, \cdots, r$ **do**
            Update l-th column of $\mathbf{B}$.
        end for
        if $\|\mathbf{B} - \mathbf{B}_0\|_F \leq \epsilon$ **then**
            break
        end if
    end while
end for

3. EXPERIMENTS

3.1. Data Sets

Indian Pines [13] (denoted as D1). The image has a size of 145×145 pixels and 220 spectral bands with a spatial resolution of 20 m/pixel. It has 16 land-cover classes. For each class, we randomly select 50 samples as training set and the remaining as testing set. For those classes with less than 50 samples, 15 samples are selected as training set. In experiments, we removed 20 noisy bands. The false color image and its ground truth are shown in Fig. 1.

University of Pavia [13] (denoted as D2). The size of the image is 610×340 pixels with a spatial resolution of 1.3 m/pixel. It has 103 bands. Here, 30 samples per class are randomly selected as training set and the rest as test set.

3.2. Experimental Setup

In experiments, three kinds of features ($V = 3$) are extracted for each pixel: (1) the original spectral feature, (2) EMP [4], and (3) EAP [5]. For EMP and EAP, we use five and three principal components on the two data sets, respectively. For EMP, 9 MPs are computed for each component with a disk-shaped structural elements, whose radius is increased from 1 with a step size of 2. For EAP, four APs are computed for each component with the same parameters as [5].

Four representative methods are compared, which include two subspace-based dimension reduction methods: (**NWFE**[6], **KLDA**) and two learning to hash methods: **FastHash**[14], **SDH**[11]. For NWFE, KLDA, FastHash and SDH, we select their parameters by grid search. Additionally, concatenating the original float multiple features (denoted as **MultiFeature**) is served as a baseline. For features represented by float vectors, we adopt nearest neighbor classifier with Euclidean distance and output the classification results. While for binary features, Hamming distance is used.

For the two data sets, we conduct ten independent trials and report the **best** average results with standard deviation of two common scores: overall accuracy (OA) and kappa (κ).

3.3. Experimental Results

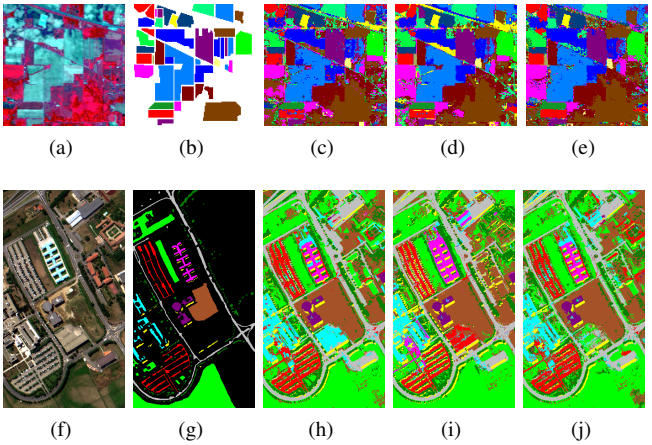
Table 1 shows classification accuracies of the compared methods on Indian Pines data set. From this table, we have the following observations. **First**, all the evaluated methods can achieve better performance than the original MultiFeature method without dimension reduction. This indicates that dimension reduction can preserve the important discriminative information. **Second**, compared to the traditional subspace-based dimension reduction methods, the obtained binary features by hashing methods require much less storage cost. **Third**, the proposed SSDH can achieve comparable or competitive results among all compared methods. Based on the fact that SSDH outperforms SDH, the effectiveness of the used structured regularization terms can be validated. Noted that since SSDH adopts linear hash functions, it is much simpler in the procedure of model learning, compared to the FastHash that utilizes the boosted decision trees as hash functions. **Fourth**, from the results on the University of Pavia data set, SSDH can still achieve the competitive results among all compared methods on this data set, and also achieves higher accuracy than original SDH, with a relative improvement of 63.7% on classification error measured by OA.

Fig. 1 visualizes the obtained classification maps of different methods. Generally speaking, the supervised methods can generate better classification maps with better visual quality. Compared with the SDH method, the proposed SSDH has better classification maps. Especially, in some left-bottom areas, SSDH achieves smoother prediction outputs.

The classification OAs obtained by the compared methods with varying number of reduced dimensions in the range of

Table 1. Performance of different methods on the two data sets. The number in each bracket is the number of bytes used. $Time(ns)$ is the averaged time for distance computation between two feature vectors measured by nanoseconds.

	Classes	MultiFeature(1700)	NWFE(80)	KLDA(60)	FastHash(8)	SDH(6)	SSDH(8)
D1	OA	85.65 ± 1.27	93.85 ± 0.98	94.76 ± 0.66	95.16 ± 0.52	91.51 ± 1.40	$\mathbf{95.28 \pm 0.28}$
	κ	0.8363 ± 0.0141	0.9295 ± 0.0111	0.9398 ± 0.0076	0.9444 ± 0.0060	0.9028 ± 0.0160	$\mathbf{0.9457 \pm 0.0032}$
	$Time(ns)$	622.5365	23.7454	17.9966	7.9272	6.4014	7.9272
	Classes	MultiFeature(952)	NWFE(56)	KLDA(32)	FastHash(4)	SDH(6)	SSDH(8)
D2	OA	89.28 ± 0.79	97.62 ± 0.52	98.47 ± 0.58	97.20 ± 0.77	96.14 ± 1.45	$\mathbf{98.60 \pm 0.28}$
	κ	0.8623 ± 0.0100	0.9688 ± 0.0067	0.9799 ± 0.0076	0.9634 ± 0.0100	0.9497 ± 0.0185	$\mathbf{0.9816 \pm 0.0036}$
	$Time(ns)$	333.7593	16.7077	10.4053	8.0419	**6.3868**	7.9240



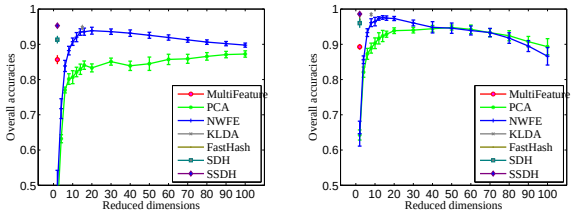
(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 1. Classification maps of different methods on two data sets. (a, f) False color image, (b, g) ground truth image and classification maps obtained by (c) MultiFeature, (d) SDH, (e) SSDH, (f) KLDA, (g) FastHash (h) SSDH.

$[2, 4, \cdots, 16, 20, 30, \cdots, 100]$ are shown in Fig. 2. For hashing methods, we just show the results of 64 bits (i.e., 2 float values). From these two figures, we can see that SSDH performs significantly better than SDH. Second, both SSDH and FastHash achieve very high classification accuracies, which indicates that we can use only 8 bytes to obtain better classification performance than that with 1700 bytes of the original MultiFeature method. As storing binary features is much more economical than the original float-type features, it can potentially facilitate subsequent processing for large scale hyperspectral data analysis (e.g., similar hyperspectral objects retrieval in large archives).

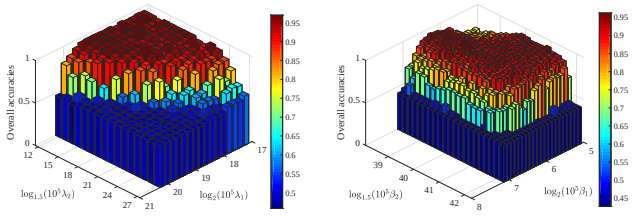
To check the parameter sensitivities of SSDH, extensive experiments are conducted. Fig. 3 shows the classification OAs of SSDH with two variable parameter values while keeping the other two fixed on University of Pavia data set. Note that the x,y-axis are log-scaled. From these two figures, we can see that a proper selection of these regularization parameters is important for a good classification performance. Even



(a) Indian Pines data set. (b) University of Pavia data set.

Fig. 2. Classification OAs with different reduced dimensions.

though, it is not difficult task since SSDH performs well in a relatively large range of these parameters. Note that when both β_1 and β_2 become zeros, it is equivalent to SDH with linear embedding. The better performance of SSDH with appropriate values of β_1 and β_2 demonstrates the effectiveness of incorporating structural information in hashing learning.



(a) Fix β_1,β_2. (b) Fix λ_1, λ_2.

Fig. 3. Parameter sensitivities of SSDH on University of Pavia data set.

4. CONCLUSION

In this paper, we have proposed a novel structural regularized binary feature extraction method for hyperspectral image classification. The introduced structured terms can handle feature redundancy and help generate more discriminative binary features. Comparative experiments on two data sets have shown the effectiveness of the proposed method.

5. REFERENCES

[1] Antonio Plaza, Jon Atli Benediktsson, Joseph W Boardman, Jason Brazile, Lorenzo Bruzzone, Gustavo Camps-Valls, Jocelyn Chanussot, Mathieu Fauvel, Paolo Gamba, Anthony Gualtieri, et al., "Recent advances in techniques for hyperspectral image processing," *Remote sensing of environment*, vol. 113, pp. S110–S122, 2009.

[2] J Bioucas-Dias, Antonio Plaza, G Camps-Valls, PAUL Scheunders, N Nasrabadi, and Jocelyn Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geoscience and Remote Sensing Magazine*, vol. 1, no. 2, pp. 6–36, June 2013.

[3] Gustavo Camps-Valls, Devis Tuia, Lorenzo Bruzzone, and Jn Atli Benediktsson, "Advances in hyperspectral image classification: Earth monitoring with statistical learning methods," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 45–54, Jan. 2014.

[4] Jón Atli Benediktsson, Jón Aevar Palmason, and Johannes R Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 480–491, Mar. 2005.

[5] Mauro Dalla Mura, Jon Atli Benediktsson, B. Waske, and Lorenzo Bruzzone, "Extended profiles with morphological attribute filters for the analysis of hyperspectral data," *International Journal of Remote Sensing*, vol. 31, no. 22, pp. 5975–5991, 2010.

[6] Bor-Chen Kuo and David A Landgrebe, "Nonparametric weighted feature extraction for classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 5, pp. 1096–1105, 2004.

[7] Deng Cai, Xiaofei He, and Jiawei Han, "Efficient kernel discriminant analysis via spectral regression," in *IEEE International Conference on Data Mining*, 2007, pp. 427–432.

[8] Jingdong Wang, Heng Tao Shen, Jingkuan Song, and Jianqiu Ji, "Hashing for similarity search: A survey," *arXiv preprint arXiv:1408.2927*, 2014.

[9] B. Demir and L. Bruzzone, "Hashing-based scalable remote sensing image search and retrieval in large archives," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 892–904, Feb. 2016.

[10] Z. Zhong, B. Fan, K. Ding, H. Li, S. Xiang, and C. Pan, "Efficient multiple feature fusion with hashing for hyperspectral imagery classification: A comparative study," *IEEE Trans. Geosci. Remote Sens.*, vol. PP, no. 99, pp. 1–18, 2016.

[11] Fumin Shen, Chunhua Shen, Wei Liu, and Heng Tao Shen, "Supervised discrete hashing," *arXiv preprint arXiv:1503.01557*, vol. 1, no. 1, pp. 1–8, 2015.

[12] Shiming Xiang, Feiping Nie, Gaofeng Meng, Chunhong Pan, and Changshui Zhang, "Discriminative least squares regression for multiclass classification and feature selection," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 11, pp. 1738–1754, Nov 2012.

[13] "Hyperspectral remote sensing scenes," http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes, Accessed: 2017-05-20.

[14] G. Lin, C. Shen, and A. van den Hengel, "Supervised hashing using graph cuts and boosted decision trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PP, no. 99, pp. 1–1, 2015.