Unsupervised iterative CEM-clustering based multiple Gaussian feature extraction for hyperspectral image classification

Bai Xue, Shengwei Zhong, Xiaodi Shang, Peter F. Hu, Chein-I Chang

Bai Xue, Shengwei Zhong, Xiaodi Shang, Peter F. Hu, Chein-I Chang, "Unsupervised iterative CEM-clustering based multiple Gaussian feature extraction for hyperspectral image classification," Proc. SPIE 10986, Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery XXV, 109861M (14 May 2019); doi: 10.1117/12.2519160

Event: SPIE Defense + Commercial Sensing, 2019, Baltimore, Maryland, United States

Unsupervised Iterative CEM-Clustering Based Multiple Gaussian Feature Extraction for Hyperspectral Image Classification

Bai Xue*[a], Shengwei Zhong[b], Xiaodi Shang[c], Peter F. Hu[d], and Chein-I Chang[a,b]
[a]Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County (UMBC), 1000 Hilltop Circle, Baltimore, MD 21250; [b]Department of Information, Engineering, Harbin Institute of Technology, Harbin 150001, China; [c]Center of Hyperspectral Imaging in Remote Sensing (CHIRS), Information and Technology College, Dalian Maritime University, Dalian 116026, China; [d]Department of Anesthesiology, University of Maryland School of Medicine, Baltimore, MD 21201

ABSTRACT

Recently, many spectral-spatial hyperspectral image classification techniques have been developed, such as widely used EPF-based and composite kernel-based approaches. However, the performance of these types of spectral-spatial approaches are generally depends on both techniques and its guided spatial feature information. To address this issue, an unsupervised subpixel detection based hyperspectral feature extraction for classification approach is proposed in this paper. Harsany-Farrand-Chang (HFC) method is utilized to estimate the number of distinct features of hyperspectral image can be decomposed into, and simplex growing algorithm (SGA) is utilized to generate endmembers as initial condition for K-means clustering. Subpixel detection maps are generated by constrained energy minimization (CEM) using centroid of K-means clusters. To capture spatial information, multiple Gaussian feature maps are generated by applying Gaussian spatial filters with different σ on CEM detection maps, and PCA is used to reduce the dimension of multiple Gaussian feature maps, and feedback it into hyperspectral band images to reprocess K-means in an iteration manner. The proposed unsupervised approach is evaluated by supervised approaches such as iterative CEM (ICEM), EPF-based, and composite kernel-based methods, and results shows that most classification performance is improved.

Keywords: unsupervised method, hyperspectral image classification, feature extraction, multiple Gaussian features

1. INTRODUCTION

A major advantage of hyperspectral imaging technique is that the captured image scene consists of spectral signatures together with spatial information of material substances. Therefore, hyperspectral image processing techniques, such as spectral-spatial based [1], composite kernel-based [2], and subpixel detection-based [3] classification approaches are received considerable interests in recent years. The classification performance of such approaches is generally based on two major procedures, which are how to effectively perform spectral signature classification, and how to find spatial features and add those features into classification procedure. Especially, the founded spatial features are important to classification approaches. For instance, in EPF-based approaches, SVM performance can be improved by using hyperspectral bands selected by some band selection for classification techniques, however, the performance of PCA results used by following edge preserving filters might be reduced, then the total performance of EPF-based approaches is reduced. This paper focus on extract joint spectral and spatial features used for hyperspectral image classification. An unsupervised iterative CEM-clustering based multiple Gaussian feature extraction approach is proposed. First, Harsany-Farrand-Chang (HFC) [4] and simplex growing algorithm (SGA) [5] are utilized to generate initial state for K-means [6], since hyperspectral images usually have sufficient spectral bands for HFC and SGA to work effectively. It is worth to mention that SGA is primarily designed for finding endmembers, and since endmembers are usually referred to distinct signatures of material substances, therefore, it is utilized for finding initial centroids for K-means clustering. Then, the subpixel detection approach constrained energy minimization (CEM) [7] is implemented with k-means obtained centroids, and multiple Gaussian filters are performed on the results of CEM following with PCA dimension reduction to generate multi-Gaussian spatial feature maps. Finally, the PCA results are feed back into hyperspectral band images to reprocess K-means in an iteration manner.

Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery XXV, edited by Miguel Velez-Reyes, David W. Messinger, Proc. of SPIE Vol. 10986, 109861M
 · doi: 10.1117/12.2519160

2. UNSUPERVISED ITERATIVE CEM-BASED CLASSIFICATION APPROACH

This section presents an iterative CEM-based with multiple Gaussian spatial feature approach for unsupervised classification feature extraction. This approach uses HFC [4] method to determine number of classes would be classify and uses K-means clustering method to generate CEM subpixel detection class desired targets, then followed a spatial information capture stage, and feeds this information as new band images back to original hyperspectral bands.

In addition, in order to obtain initialization centroids for K-means clustering method, simplex volume growing (SGA) [5] method is utilized, in which SGA is an endmember finding algorithm used in hyperspectral image unmixing applications, since the endmembers can be seen as a set of distinct spectral signatures. Then CEM is utilized to perform subpixel detection with K-means generated centroids.

To capture spatial information from CEM subpixel detection maps, a spatial capturing procedure is designed and implemented in this paper, which uses several Gaussian image filters with different σ parameters to blur CEM subpixel detection results, then followed by PCA dimension reduction stage, and finally feedback with PCA results, and it is named multi-Gaussian feature (MGF) procedure.

Finally, an iterative stage, which iteratively feeds back MGF generated spatial features maps as a new set of band images into original band images. The feedback loop is continuous until a stopping condition is satisfied. Figure 1 describes a flowchart of unsupervised iterative CEM-based approach to hyperspectral unsupervised classification.

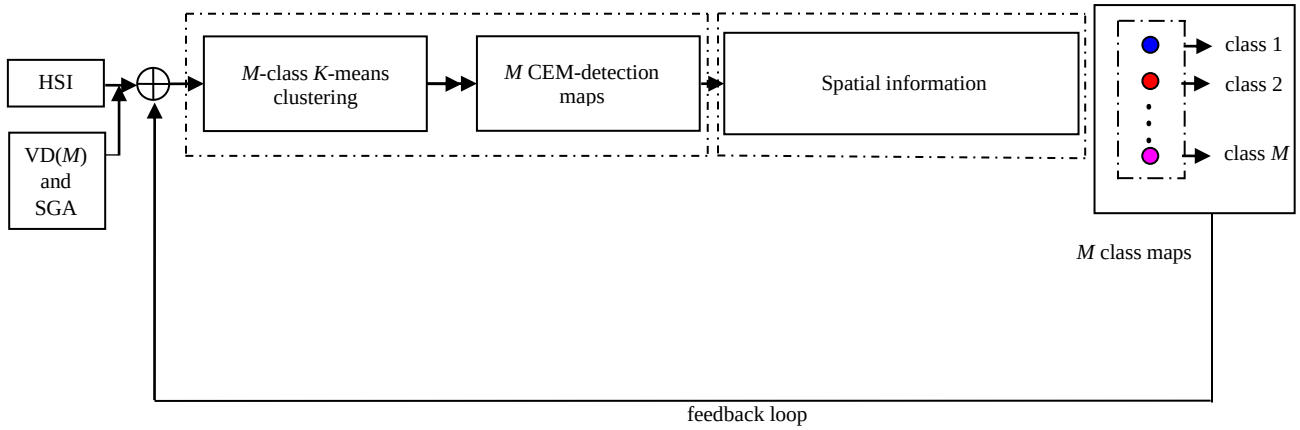


Figure 1. A diagram of unsupervised iterative CEM-based classification framework.

The detail implementation steps of unsupervised classification and feature extraction are shown in algorithm 1 as follow.

Algorithm 1 Unsupervised iterative CEM-based classification method I(KM-CEM-MGF)

1) *Initial Condition*: Let $\mathbf{\Omega}^{(0)}$ be the original set of band images, which contains N pixels. Normalize all pixels in hyperspectral image data into range 0 to 1 as

$$r = \frac{r - \min(r)}{\max(r) - \min(r)} \tag{1}$$

2) Implement VD algorithm to decide number of classes N_{class} and implement SGA algorithm to generate N_{class} endmembers $\mathbf{M}^{(1)} = \left[\mathbf{m}_1, \ldots, \mathbf{m}_{N_{class}}\right]$.

3) Implement *K*-means clustering algorithm, which using matrix $\mathbf{M}^{(k)} = \left[\mathbf{m}_1, \ldots, \mathbf{m}_{N_{class}}\right]$ as initial cluster centroid positions, to obtain cluster indices matrix $\mathbf{I}^{(k)}_{N\times N_{class}}$ and distance matrix $\mathbf{D}^{(k)}_{N\times N_{class}}$, where each row of $\mathbf{I}^{(k)}_{N\times N_{class}}$ indicates the cluster assignment of corresponding observation, and distance matrix $\mathbf{D}^{(k)}_{N\times N_{class}}$ contains distances of each pixels to every clusters centroid.

4) Let $\left\{\mathbf{d}_m^{(k)}\right\}_{m=1}^{N_{class}} = \left\{\left(d_{m,1}, \cdots, d_{m,N_{class}}\right)\right\}_{m=1}^{N_{class}}$ be the desired target pixels of N_{class} class, which is formed by taking sample mean of each cluster pixels on $\mathbf{D}^{(k)}_{N\times N_{class}}$ generate by *K*-means in previous step. Let δ_k^{CEM} be CEM subpixel detector on

$\mathbf{D}^{(k)}_{N\times N_{class}}$ using $\left\{\mathbf{d}^{(k)}_m\right\}^{M}_{m=1}$ and $\mathbf{R}^{(k)}$ to produce $\left\{\mathbf{B}^{\text{CEM}}_k\right\}^{N_{class}}_{m=1}$,which is N_{class} class detection map, and using Otsu's method to generate binary N_{class} class detection map $\left\{\mathbf{B}^{\text{BCEM}}_k\right\}^{N_{class}}_{m=1}$.

5) Implement spatial processing procedure to obtain $\left\{\mathbf{B}^{\text{spatial}}_k\right\}^{N_{class}}_{m=1}$ from $\left\{\mathbf{B}^{\text{CEM}}_k\right\}^{N_{class}}_{m=1}$ and $\left\{\mathbf{B}^{\text{BCEM}}_k\right\}^{N_{class}}_{m=1}$. Use N_{G}-multiple Gaussian filters to blur $\left\{\left|\mathbf{B}^{\text{CEM}}_k\right|\right\}^{N_{class}}_{m=1}$, where $\left\{\left|\mathbf{B}^{\text{CEM}}_k\right|\right\}^{N_{class}}_{m=1}$ is the absolute value of $\left\{\mathbf{B}^{\text{CEM}}_k\right\}^{N_{class}}_{m=1}$, and generate $\left(N_G\times M\right)$-Gaussian spatial feature maps. The resulting image is denoted by Gaussian-filtered $\left(N_G\times M\right)$-class CEM detection map $\left\{\left|\mathbf{B}\right|^{(k)}_{\text{GCEM}}\right\}^{M,N_G}_{m=1,n_G=1}$, and using PCA to reduce it into $\left\{\mathbf{B}^{\text{spatial}}_k\right\}^{N_{class}}_{m=1}$.

6) Check if $\mathbf{I}^{(k)}_{N\times N_{class}}$ satisfies a given stopping rule. If no, continue. Otherwise, algorithm is terminated.

7) Form $\mathbf{\Omega}^{(k+1)}=\mathbf{\Omega}^{(k)}\cup\left\{\mathbf{B}^{(k)}_{\text{GP}}\right\}^{M}_{m=1}$. Let $k\leftarrow k+1$ and go to step 3).

The final spatial feature $\left\{\mathbf{B}^{\text{spatial}}_k\right\}^{N_{class}}_{m=1}$ maps are feeding as new generated bands back to original processed dataset in each feedback loop of proposed unsupervised approach, and those feature maps are generated from subpixel detection results, and it jointly contain spectral and spatial information of hyperspectral dataset.

In order to effectively terminate algorithm, Tanimoto index (TI) is utilized [8] [3] by

$$\text{TI}^{(k)}=\frac{|S_k\bigcap S_{k-1}|}{|S_k\bigcup S_{k-1}|} \tag{4}$$

TI is used as a stopping criterion where $|S|$ is size of a set S, S_k and $S_{k\text{-}1}$ are the k^{th} thresholded binary image of the k^{th} CEM detection abundance fractional map, $\left\{\left|\mathbf{B}^{\text{CEM}}_k\right|\right\}^{N_{class}}_{m=1}$ and k-1$^{\text{st}}$ thresholded binary image of the k-1$^{\text{st}}$ CEM detection abundance fractional map, $\left\{\left|\mathbf{B}^{\text{CEM}}_{k-1}\right|\right\}^{N_{class}}_{m=1}$.

3. UNSUPERVISED CLASSIFICATION EVALUATION

The performance evaluation of unsupervised hyperspectral image classification is usually hard to implement, since the groundtruth is manually defined so that background is unlabeled and in one class might exist different spectral signature. In order to address this issue, in this paper, we performed several supervised classification approaches, which utilized 5% of labeled samples as training samples, obtained from unsupervised classification approach generated feature maps, as well as original band images, then evaluated the performance of the supervised classification.

In order to evaluate classification performance of the approaches in this experiment, the following experiments performance are evaluated with several parameters. More specifically, we can define variety factors

C_i = the ith class

n_{ij} = the number of samples in the jth class to be classified into ith class

n_{jj} = the number of samples in the jth class correctly classified into the jth class

n_j = the number of samples in the jth class, i.e., $n_j=\sum_{j=1}^{M}n_{ij}$

$\hat{n}_{ij}$ = the number of data samples classified in the ith class, which are supposed to in the jth class C_j

$\hat{n}_i$ = the number of data samples classified in the ith class, i.e., $\hat{n}_i=\sum_{j=1}^{M}\hat{n}_{ij}$

$\hat{n}_{\text{BKG}}$ = the number of data samples classified as background

N = total number of data samples, i.e., $N=\sum_{j=1}^{M}n_j$

$$p(C_j)=\frac{n_j}{N}$$

Further, according to the definition above, various quantitative measures can be defined as follows.

$$\mathrm{P_A}(C_j) = \mathrm{P}(C_{jj} \mid C_j) = \frac{n_{jj}}{n_j} \tag{5}$$

$$\mathrm{P_{PR}}(\hat{C}_i) = \frac{\hat{n}_{ii}}{\sum_{j=1}^{M} \hat{n}_{ij}} \tag{6}$$

$$\mathrm{P_A} = \frac{\hat{n}_{\mathrm{BKG}}}{N} + \sum_{i=1}^{M} \frac{n_{ii}}{N} \tag{7}$$

$$\mathrm{P_{AA}} = \frac{1}{M} \sum_{j=1}^{M} \mathrm{P}_A(C_j) = \frac{1}{M} \sum_{i=1}^{M} \frac{n_{ii}}{n_{ii}} \tag{8}$$

$$\mathrm{P_{OA}} = \frac{1}{N} \sum_{j=1}^{M} n_j \left(\frac{n_{jj}}{\sum_{i=1}^{p} n_{ij}} \right) \tag{9}$$

$$\mathrm{P_{PR}} = \frac{1}{N} \sum_{i=1}^{M} \hat{n}_i p_{PR}(\hat{C}_i) \tag{10}$$

4. EXPERIMENTS

In this section, three popular real hyperspectral images the Indiana Indian Pines, Salinas, and University of Pavia hyperspectral images are utilized on proposed method, and the detail description of the dataset can be found in [3]. In the following experiments, several supervised classification approaches are performed on the features of proposed unsupervised approach, which are BSNE-ICEM, linear SVM, RBF-kernel SVM [9], and stacked feature composite kernel SVM [2]. The proposed method is indicated as I(KM-CEM-MGF).

In the following experiments, the BSNE-ICEM is implemented with BSNE generated band images together with I(KM-CEM-MGF) generated feature maps, and it is denoted as BSNE-ICEM(Image-Feature). The SVM approaches are implemented with linear and RBF kernel on only I(KM-CEM-MGF) generated feature maps and are denoted as Linear SVM (Feature) and KSVM (Feature). The stacked feature composite kernel SVM with original band images formed spectral kernel, together with I(KM-CEM-MGF) generated feature maps formed spatial kernel, and it is denoted as SFCK-SVM (Image-Feature).

In order to comparatively analysis the proposed method, several classification approaches are utilized: i) RBF kernel SVM, which is denoted as KSVM(Image), ii) BSNE-ICEM which is denoted as BSNE-ICEM(Image), and iii) stacked feature composite kernel SVM, which using original band images to form spectral kernel and neighbor mean and standard deviation to form spatial kernel, and it is denoted as SFCK-SVM (image, μ-σ).

Table 1 tabulates the parameters used in proposed unsupervised iterative CEM-based classification approach and the detail structure of spatial information capture procedure.

Table 2, 3, and 4 tabulates the performance measures produced by several supervised classification approaches to evaluate proposed unsupervised classification method for the Indiana Indian Pines, Salina, and University of Pavia hyperspectral image scenes. Fig 2, 4 and 6 illustrate the proposed I(KM-CEM-MGF) generated PCA spatial feature maps for the Indiana Indian Pines, Salinas, and University of Pavia hyperspectral image scenes.

The I(KM-CEM-MGF) unsupervised classification results are shown in Fig 2(a) for Indiana Indian Pines scene, Fig 4(a) for Salinas scene, and Fig 6(a) for University of Pavia scene. The KSVM classification results using I(KM-CEM-MGF) obtained feature maps are shown in Fig 2(b) for Indiana Indian Pines scene, Fig 4(b) for Salinas scene, and Fig6(b) for University of Pavia scene. The stacked feature composite kernel SVM classification results jointly using I(KM-CEM-MGF) obtained feature maps and original band images are shown in Fig 2(c) for Indiana Indian Pines scene, Fig 4(c) for Salinas scene, and Fig6(c) for University of Pavia scene.

Table 1. Specifications of parameters used by unsupervised iterative CEM-based classification approach.

BSNE	None
d	Class train samples mean
VD	27(Indian Pines) 21(Salinas) 14(University of Pavia)

I(KM-CEM-MGF)	
size of Gaussian filter	11×11 window
σ used in Gaussian filter	$0.1\sqrt{2}, 0.15\sqrt{2}, \ldots, 2\sqrt{2}$
Thresholding method	Maximum likelihood
stopping threshold (TI)	0.99

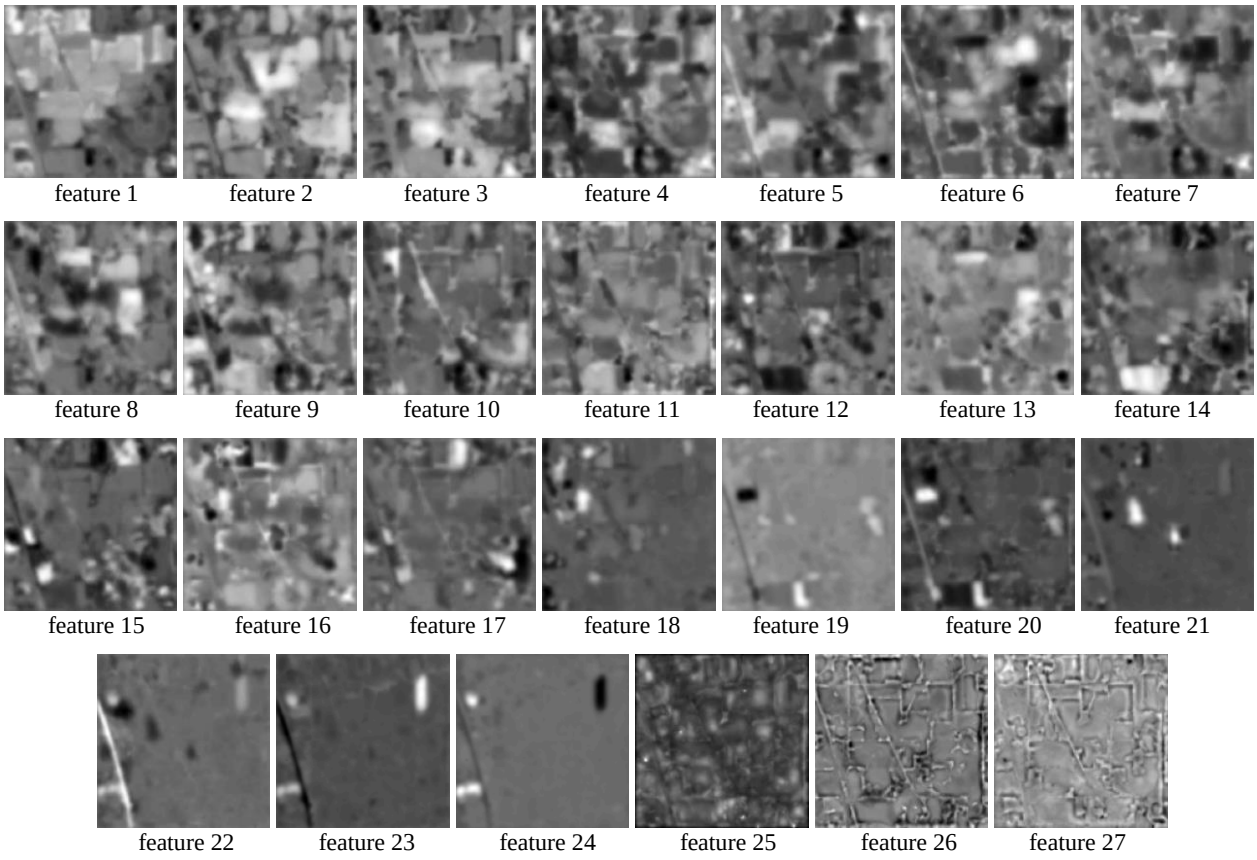


Figure 2. 27 I(KM-CEM-MGF) PCA spatial feature maps of Indiana Indian Pines scene.

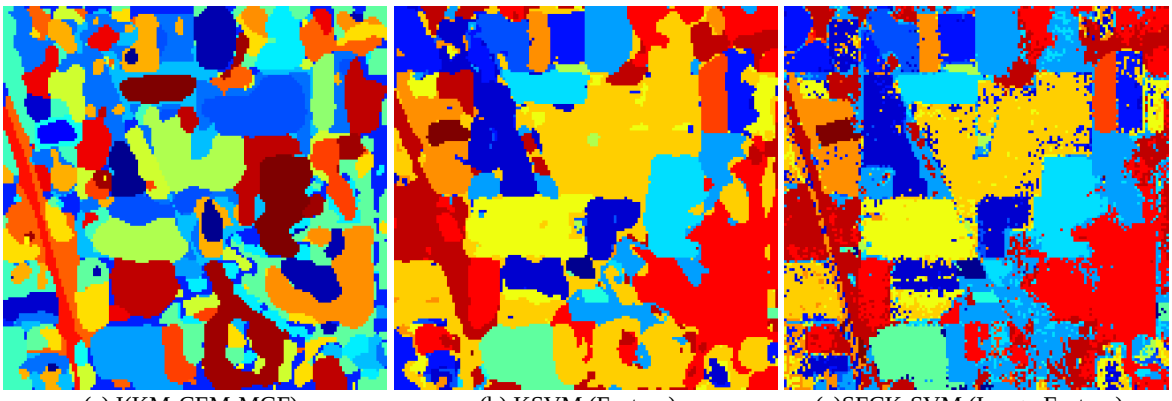
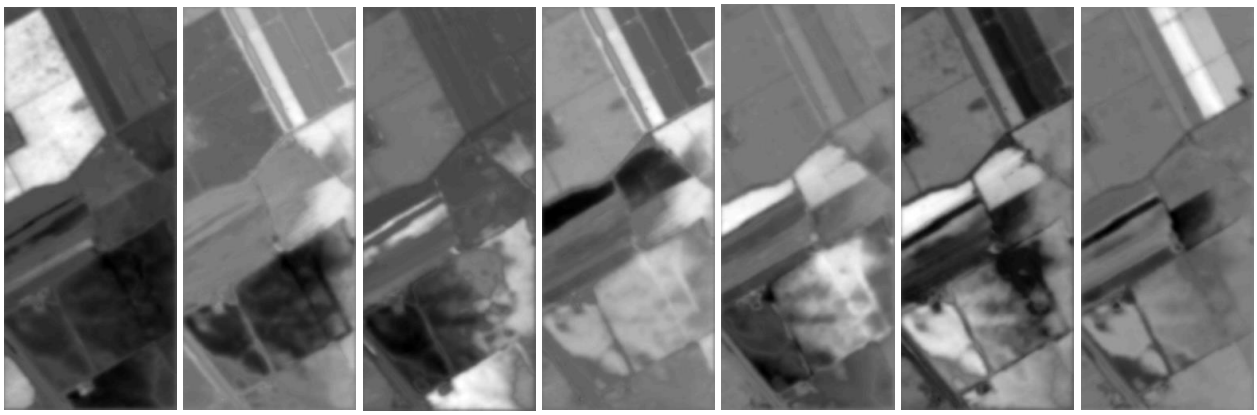


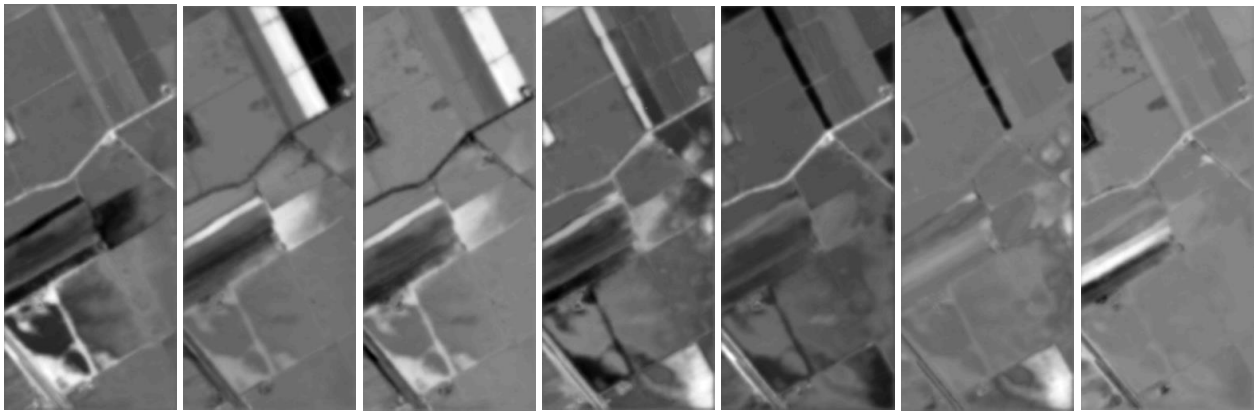
Figure 3. (a) 27 I(KM-CEM-MGF) unsupervised classification results, (b) 16 KSVM classification results using I(KM-CEM-MGF) extracted feature, (c) 16 stacked feature composite kernel SVM classification results jointly using I(KM-CEM-MGF) extracted feature and original scene band images.

Table 2. Performance evaluation comparison among kernel SVM (KSVM) and stacked feature composite kernel SVM with original hyperspectral image data, BSNE-ICEM, linear SVM and KSVM with I(KM-CEM-MGF) generated feature, and BSNE-ICEM, stacked feature composite kernel SVM with jointly using both for the Indiana Indian Pines scene.

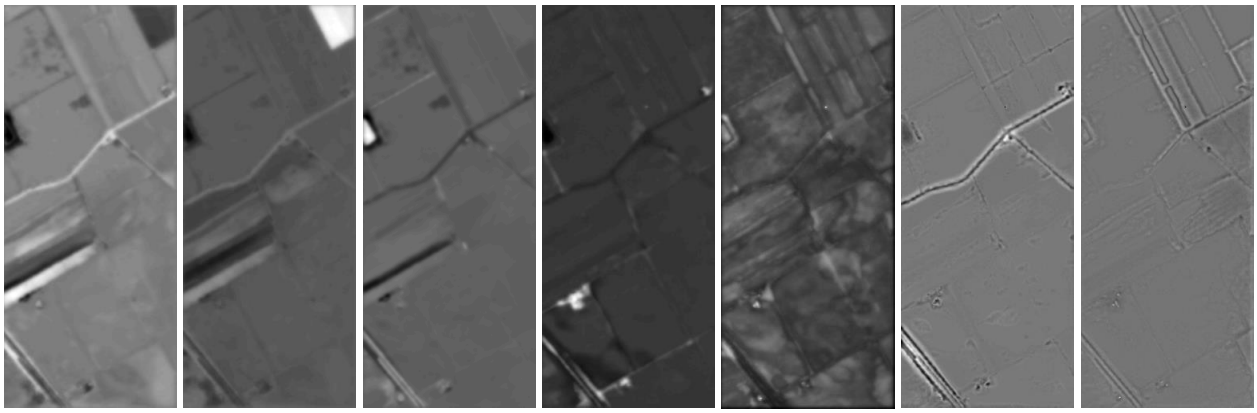
	KSVM (Image)		BSNE-ICEM (Image)		BSNE-ICEM (Image-Feature)		Linear SVM (Feature)		KSVM (Feature)		SFCK-SVM (Image-Feature)		SFCK-SVM (image, μ-σ)	
Class(C_i)	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$
1	0.8696	0.4545	0.7455	0.8913	0.8200	0.8913	0.9783	0.3879	0.9783	0.6081	0.9783	0.3750	0.9348	0.4574
2	0.6765	0.5305	0.7971	0.8348	0.8214	0.9130	0.7227	0.6128	0.8817	0.7137	0.8347	0.6416	0.8445	0.5906
3	0.6795	0.5306	0.6453	0.9576	0.8326	0.9225	0.8410	0.5479	0.9386	0.5969	0.8916	0.6554	0.8530	0.5276
4	0.8819	0.3628	0.9580	0.7972	0.8819	0.9952	0.9747	0.3690	1.0000	0.5823	0.9536	0.4603	0.9536	0.4475
5	0.9255	0.2407	0.8675	0.9209	0.7205	0.9405	0.9669	0.1526	0.9772	0.3026	0.9710	0.1366	0.9586	0.2223
6	0.9575	0.3132	0.9041	0.9218	0.9616	0.8720	1.0000	0.5130	1.0000	0.6553	0.9986	0.4389	0.9849	0.4651
7	0.8929	0.4808	0.9655	0.7778	1.0000	0.8235	1.0000	0.0798	1.0000	0.4912	1.0000	0.4590	0.9286	0.6341
8	0.9874	0.7410	0.9937	0.8636	0.9833	0.9162	1.0000	0.6704	0.9958	0.7449	0.9937	0.7457	0.9937	0.7930
9	0.8500	0.2237	0.9000	0.5294	1.0000	0.6897	1.0000	0.1205	1.0000	0.2985	1.0000	0.1709	1.0000	0.0995
10	0.8107	0.5212	0.7500	0.9306	0.8746	0.9020	0.7695	0.4533	0.9588	0.6495	0.9167	0.5976	0.8395	0.4880
11	0.7605	0.6892	0.7589	0.8748	0.8043	0.8860	0.8399	0.6774	0.9393	0.4027	0.8542	0.7798	0.7760	0.7605
12	0.8297	0.4995	0.9224	0.9041	0.9430	0.8949	0.9427	0.6309	0.9663	0.6235	0.9562	0.4655	0.9376	0.6805
13	0.9951	0.4163	0.9902	0.9022	0.9854	0.9395	1.0000	0.7218	1.0000	0.7270	1.0000	0.6193	1.0000	0.5325
14	0.9375	0.3388	0.8324	0.9419	0.9005	0.8321	0.9455	0.3874	0.9715	0.3550	0.9257	0.3846	0.9542	0.4013
15	0.7358	0.0880	0.7720	0.9835	0.7923	0.9748	0.9404	0.1473	0.9767	0.1802	0.8964	0.1334	0.8679	0.0834
16	0.9785	0.4596	0.9892	0.9485	0.9892	0.9583	0.9892	0.4718	0.9892	0.7541	0.9892	0.6133	1.0000	0.5536
P_A	0.3972		0.8754		**0.8923**		0.4259		0.4642		0.4420		0.4282	
P_{AA}	0.8605		0.8620		0.8944		0.9319		**0.9733**		0.9475		0.9267	
P_{OA}	0.8148		0.8089		0.8584		0.8736		**0.9523**		0.9067		0.8784	
P_{PR}	0.4983		**0.8989**		0.8981		0.5293		0.5277		0.5697		0.5561	



feature 1 feature 2 feature 3 feature 4 feature 5 feature 6 feature 7

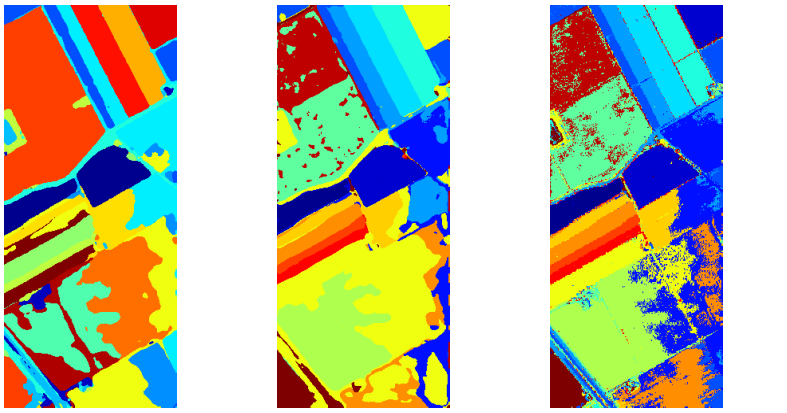


feature 8 feature 9 feature 10 feature 11 feature 12 feature 13 feature 14



feature 15 feature 16 feature 17 feature 18 feature 19 feature 20 feature 21

Figure 4. 21 I(KM-CEM-MGF) PCA spatial feature maps of Salinas scene.



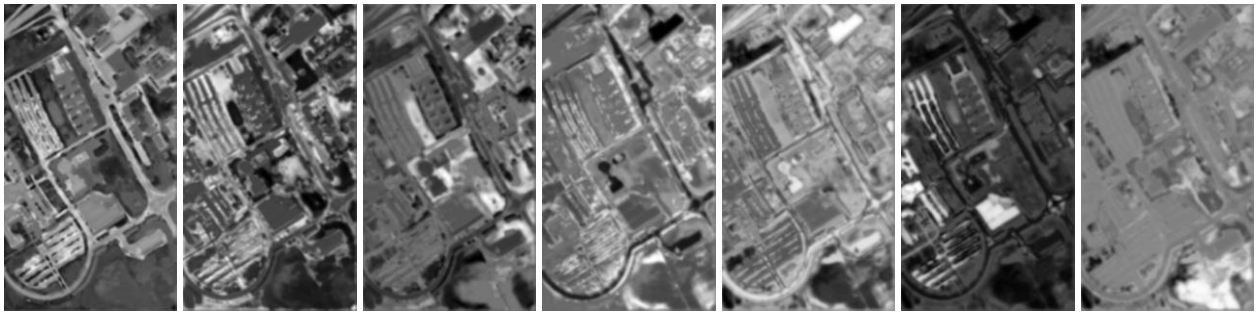
(a) I(KM-CEM-MGF) (b) KSVM (Feature) (c)SFCK-SVM (Image-Feature)

Figure 5. (a) 21 I(KM-CEM-MGF) unsupervised classification results, (b) 16 KSVM classification results using I(KM-CEM-MGF) extracted feature, (c) 16 stacked feature composite kernel SVM classification results jointly using I(KM-CEM-MGF) extracted feature and original scene image.

Table 3. Performance evaluation comparison among kernel SVM (KSVM) and stacked feature composite kernel SVM with original hyperspectral image data, BSNE-ICEM, linear SVM and KSVM with I(KM-CEM-MGF) generated feature, and BSNE-ICEM, stacked feature composite kernel SVM with jointly using both for the Salinas scene.

	KSVM (Image)		BSNE-ICEM (Image)		BSNE-ICEM (Image-Feature)		Linear SVM (Feature)		KSVM (Feature)		SFCK-SVM (Image-Feature)		SFCK-SVM (image, μ-σ)	
Class(C_i)	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$
1	0.9861	0.8234	0.9990	0.8551	0.9447	0.9694	0.9567	0.8120	0.9856	0.8915	0.9950	0.8264	0.9995	0.7735
2	0.9922	0.5814	0.9750	0.9787	0.9332	0.9989	0.9970	0.5607	0.9941	0.7773	0.9938	0.5466	0.9968	0.5615
3	0.9873	0.1134	0.9782	0.4909	0.9843	0.5589	0.9995	0.1387	0.9995	0.2478	0.9919	0.1173	0.9954	0.1202
4	0.9928	0.2898	0.8343	0.9417	0.7590	0.9113	0.9835	0.3664	0.9885	0.4019	0.9921	0.2511	0.9921	0.2751
5	0.9866	0.4125	0.9784	0.7777	0.8730	0.8975	0.9264	0.4315	0.9966	0.3875	0.9944	0.4046	0.9981	0.4645
6	0.9955	0.8434	0.9808	0.9186	0.9624	0.9426	1.0000	0.8345	1.0000	0.8459	0.9955	0.8376	0.9919	0.8398
7	0.9983	0.8507	0.9718	0.9508	0.9531	0.9627	1.0000	0.7705	1.0000	0.7890	0.9989	0.7606	0.9953	0.7930
8	0.7262	0.6961	0.8454	0.8514	0.8632	0.8617	0.7637	0.7958	0.9099	0.8672	0.8547	0.7517	0.8489	0.7228
9	0.9937	0.3749	0.9578	0.7349	0.9184	0.9855	0.9997	0.3781	0.9992	0.5356	0.9994	0.4477	0.9961	0.3273
10	0.9240	0.3209	0.8802	0.9747	0.8948	0.9816	0.9863	0.3361	0.9847	0.1151	0.9671	0.4046	0.9591	0.6289
11	0.9794	0.1655	0.9251	0.8949	0.9139	0.8841	0.9972	0.3022	0.9934	0.2954	0.9897	0.2463	0.9878	0.2725
12	0.9943	0.3453	0.9632	0.7339	0.9678	0.9339	0.9917	0.2096	0.9938	0.2858	0.9979	0.2077	0.9995	0.2262
13	0.9869	0.6226	0.9160	0.7749	0.9771	0.7546	0.9913	0.7012	0.9989	0.6778	0.9902	0.6059	1.0000	0.4372
14	0.9439	0.4574	0.8824	0.7627	0.8843	0.8020	0.9963	0.6364	0.9972	0.6779	0.9738	0.4155	0.9925	0.4728
15	0.6878	0.5845	0.6737	0.9454	0.7305	0.9396	0.9595	0.6681	0.8941	0.7664	0.8762	0.7316	0.8199	0.6912
16	0.9950	0.5362	0.9640	0.9197	0.9690	0.9250	1.0000	0.2961	0.9994	0.5191	0.9950	0.6091	0.9950	0.5781
P_A	0.4340		0.8777		**0.9113**		0.4570		0.4697		0.4619		0.4576	
P_{AA}	0.9481		0.9203		0.9080		0.9718		**0.9834**		0.9753		0.9730	

P_{OA}	0.8908	0.8893	0.8845	0.9380	**0.9641**	0.9481	0.9393
P_{PR}	0.5594	0.8378	**0.9011**	0.5834	0.6492	0.5922	0.5840



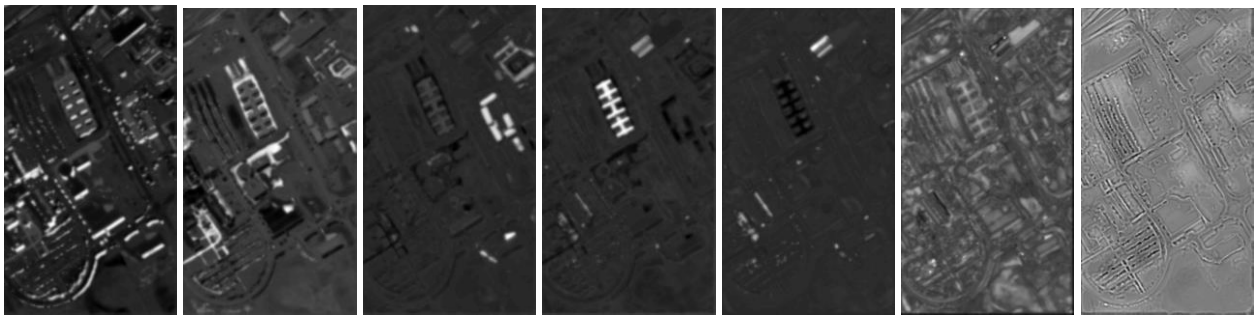
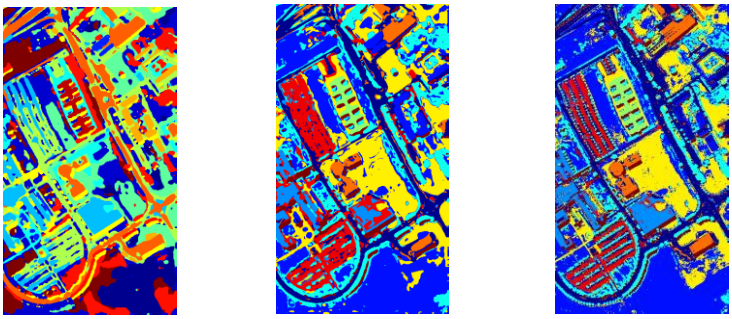


Figure 6. 14 I(KM-CEM-MGF) PCA spatial feature maps of University of Pavia scene.



(a) I(KM-CEM-MGF) (b) KSVM (Feature) (c)SFCK-SVM (Image-Feature)

Figure 7. (a) 14 I(KM-CEM-MGF) unsupervised classification results, (b) 9 KSVM classification results using I(KM-CEM-MGF) extracted feature, (c) 9 stacked feature composite kernel SVM classification results jointly using I(KM-CEM-MGF) extracted feature and original scene image.

Table 4. Performance evaluation comparison among kernel SVM (KSVM) and stacked feature composite kernel SVM with original hyperspectral image data, BSNE-ICEM, linear SVM and KSVM with I(KM-CEM-MGF) generated feature, and BSNE-ICEM, stacked feature composite kernel SVM with jointly using both for the University of Pavia scene.

	KSVM (Image)		BSNE-ICEM (Image)		BSNE-ICEM (Image-Feature)		Linear SVM (Feature)		KSVM (Feature)		SFCK-SVM (Image-Feature)		SFCK-SVM (image, μ-σ)	
Class(C_i)	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$	$P_A(C_i)$	$P_{PR}(C_i)$
1	0.8869	0.1700	0.7821	0.3209	0.8208	0.2716	0.9139	0.2301	0.9548	0.2212	0.9637	0.1771	0.9573	0.1931
2	0.9721	0.3075	0.7840	0.8749	0.8253	0.8561	0.9357	0.2760	0.9773	0.3284	0.9881	0.3209	0.9901	0.3369
3	0.8275	0.3191	0.5441	0.2464	0.6280	0.3115	0.8990	0.1968	0.9414	0.2703	0.9295	0.2063	0.9042	0.2491
4	0.9553	0.1132	0.7102	0.3056	0.7167	0.3077	0.8006	0.0688	0.9256	0.0795	0.9847	0.0959	0.9775	0.0979
5	0.9970	0.3814	0.9477	0.7987	0.9394	0.8031	1.0000	0.5197	1.0000	0.5483	1.0000	0.3313	1.0000	0.2396
6	0.9089	0.0989	0.6945	0.7461	0.7007	0.8536	0.9342	0.1561	0.9592	0.1202	0.9857	0.1304	0.9926	0.1097
7	0.9346	0.2691	0.5298	0.2305	0.5112	0.2234	0.8053	0.1071	0.9586	0.1881	0.9835	0.1887	0.9857	0.2234
8	0.8881	0.2053	0.7288	0.2286	0.7602	0.2897	0.9082	0.1584	0.9449	0.1506	0.9375	0.2851	0.9283	0.2234
9	1.0000	0.0773	0.7814	0.3152	0.8756	0.2912	0.9979	0.1074	0.9979	0.1203	1.0000	0.0799	1.0000	0.1024

P_A	0.1931	0.7638	**0.7687**	0.1893	0.1988	0.2015	0.2011
P_{AA}	0.9301	0.7225	0.7531	0.9105	0.9622	**0.9747**	0.9706
P_{OA}	0.9362	0.7381	0.7731	0.9176	0.9640	**0.9771**	0.9753
P_{PR}	0.2355	0.6118	**0.6166**	0.2246	0.2493	0.2423	0.2449

According to Table 2, for the Indiana Indian Pines hyperspectral image scene, the best P_A was produced by BSNE-ICEM(Image-Feature), and the best P_{AA} and P_{OA} were produced by KSVM(Feature). The best P_{PR} was produced by BSNE-ICEM(Image). In addition, KSVM (Feature), which used unsupervised features, can generate significant improved classification performance than KSVM (Image), which only used original hyperspectral image, and even linear SVM using unsupervised features can perform better than KSVM using original dataset. SFCK-SVM used unsupervised feature maps can also perform better than SFCK-SVM using spatial mean and standard deviation on original dataset. And BSNE-ICEM (Image-Feature) jointly using feature and original image has significant improvement compare to BSNE-ICEM (Image) on P_A, P_{AA}, and P_{OA}, and also generated lower but a close performance in terms of P_{PR}.

According to Table 4, for the Salinas hyperspectral image scene, the best P_A and P_{PR} were produced by BSNE-ICEM(Image-Feature), and the best P_{AA} and P_{OA} were produced by KSVM(Feature). The KSVM (Feature) can generate better classification performance than KSVM (Image) as shown in the experiment of the Indian Pines scene. And SFCK-SVM used unsupervised feature maps can also perform better than SFCK-SVM using spatial mean and standard deviation on original dataset. And BSNE-ICEM (Image-Feature) jointly using feature and original image can produce better classification performance compared to BSNE-ICEM (Image) on P_A, P_{OA}, and P_{PR}, and also generated a lower but a closer performance in terms of P_{AA}.

According to Table 6, for the University of Pavia hyperspectral image scene, the best P_A and P_{PR} were also produced by BSNE-ICEM(Image-Feature), and the best P_{AA} and P_{OA} were produced by SFCK-SVM (Image-Feature). The KSVM (Feature) can generate better classification performance than KSVM (Image). And SFCK-SVM used unsupervised feature maps can also perform better than SFCK-SVM using spatial mean and standard deviation on original dataset in terms of P_A, P_{OA}, and P_{AA}. And BSNE-ICEM (Image-Feature) jointly using feature and original image has significant improvement compare to BSNE-ICEM (Image) on P_A, P_{OA}, P_{AA}, and P_{PR}.

5. CONCLUSION

This paper designs an unsupervised hyperspectral imagery feature extraction approach for classification from hyperspectral subpixel detection aspect. In order to enhance hyperspectral classification methods or provided spectral-spatial approaches an efficient guided image with joint spectral and spatial information. HFC and SGA are utilized to generate initial state for K-means, subpixel detection results are obtained by CEM, and PCA is utilized to reduce dimensionality of multi-Gaussian spatial feature maps from Gaussian filter blurred CEM results, and then feedback PCA results into hyperspectral band images to reprocess K-means in an iteration manner. The experiments results illustrated that extracted feature can improve the performance of the supervised classification methods.

REFERENCES

[1] Kang, X., Li, S., and Benediktsson, J. A, "Spectral–spatial hyperspectral image classification with edge-preserving filtering.", IEEE Transactions on Geoscience and Remote Sensing, 52(5), 2666-2677 (2014).

[2] Camps-Valls, G., Gomez-Chova, L., Muñoz-Marí, J., Vila-Francés, J., and Calpe-Maravilla, J., "Composite kernels for hyperspectral image classification.", IEEE Geoscience and Remote Sensing Letters, 3(1), 93-97 (2006).

[3] Xue, B., Yu, C., Wang, Y., Song, M., Li, S., Wang, L., Chen, H.-M., and Chang, C.-I, "A subpixel target detection approach to hyperspectral image classification.", IEEE Transactions on Geoscience and Remote Sensing, 55(9), 5093-5114 (2017).

[4] Chang, C. I., and Du, Q., "Estimation of number of spectrally distinct signal sources in hyperspectral imagery.", IEEE Transactions on Geoscience and Remote Sensing, 42(3), 608-619 (2004).

[5] Chang, C. I., Wu, C. C., Liu, W., and Ouyang, Y. C., "A new growing method for simplex-based endmember extraction algorithm.", IEEE Transactions on Geoscience and Remote Sensing, 44(10), 2804-2819 (2006).

[6] Bishop, C. M., [Pattern recognition and machine learning], Springer. (2006)

[7] Chang, C.-I, [Hyperspectral imaging: techniques for spectral detection and classification], Springer, 54-55 (2003).

[8] Theodoridis, S., & Koutroumbas, K., [Pattern Recognition], Academic Press., New York, 366 (1999).

[9] Chang, C. C., and Lin, C. J., "LIBSVM: a library for support vector machines.", ACM Transactions on Intelligent Systems and Technology (TIST), 2(3), 27 (2011).