Received July 31, 2019, accepted August 20, 2019, date of publication August 29, 2019, date of current version October 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2938397

SuperBF: Superpixel-Based Bilateral Filtering Algorithm and Its Application in Feature Extraction of Hyperspectral Images

ZHIKUN CHEN[1,4], JUNJUN JIANG[2], (Member, IEEE), CHONG ZHOU[3], (Member, IEEE), SHAOYUAN FU[1], AND ZHIHUA CAI[1,4], (Senior Member, IEEE)

[1]Key Laboratory of Marine Geographic Information Resources Development and Utilization in the Beibu Gulf, Beibu Gulf University, Qinzhou 535011, China
[2]School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China
[3]School of Information Engineering, Hebei GEO University, Shijiazhuang 050031, China
[4]School of Computer Science, China University of Geosciences, Wuhan 430074, China

Corresponding author: Zhihua Cai (zhcai@cug.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61773355, Grant 61971165, and Grant 41666003, in part by the Qinzhou Scientific Research and Technology Development Plan Project under Grant 201714322, in part by the High-Level Talents Research Projects of Beibu Gulf University under Grant 2019KYQD27, and in part by the Construction Project of Characteristic Specialty and Experimental Training Teaching Base (Center) in Guangxi Undergraduate Universities.

ABSTRACT Bilateral filtering (BF), which is an edge-preserving filtering (EPF) method, has been widely recognized as a simple and efficient approach for hyperspectral image (HSI) feature extraction. However, due to the limitation of spatial resolution and the influence of the complexity of land feature distribution in HSIs, updating the target pixel through weighted averaging of neighbouring pixels is prone to generating mixed pixels, i.e., the updated target pixel is mixed with the feature of other land objects in addition to that of the target object, decreasing the quality of the image feature extraction. To address this problem, in this study, we propose a superpixel-based BF algorithm, SuperBF. This algorithm divides a HSI into many homogeneous regions via superpixel segmentation and then separately filters each homogenous region via BF; this approach ensures that the pixel structure in the template after BF is similar to that in the filtering process, reduces the probability of generating mixed pixels, and thus improves the quality of the image feature extraction. To verify the effectiveness of this proposed method, a support vector machine (SVM) classifier is used to classify the extracted SuperBF features. Experiments on three commonly employed HSI datasets demonstrated that SuperBF is significantly superior to the traditional BF-based hyperspectral feature extraction method and some new feature extraction methods.

INDEX TERMS Superpixel, bilateral filtering, feature extraction, hyperspectral images.

I. INTRODUCTION

Ahyperspectral image is a digital image of hundreds of narrow spectral bands and visible infrared spectral bands acquired by satellite sensors [1]–[4]. It can not only provide spatial characteristic information about ground objects [5]–[8] but also contain rich spectral characteristic information that reflects the unique physical properties of the ground objects [9]–[12], which enables accurate detection and recognition [13], [14] and attribute analysis of the ground objects, even when the label information is contaminated by noise [15]–[17]. HSI has an active role due to its unique advantages in the fields of precision agriculture, forest protection, marine monitoring, and military reconnaissance [18]–[21].

Feature extraction of HSIs is a key technology in remote sensing science; numerous studies in this area have been reported [22]. Chen *et al.* [23] used propagation filtering to extract HSI features and improve the performance of a classifier. Jiang and Ma [24] proposed a superpixel principal component analysis (SuperPCA) approach to integrate spatial context information about a HSI into unsupervised dimensionality reduction via superpixel segmentation and extract the discriminative, compact, and noise-resistant features. SuperPCA is a simple but very effective method. Just like PCA, it can be easily added to the pre-processing of existing methods. Li *et al.* [25] proposed a classification paradigm that utilized the texture features of HSIs and used a local

The associate editor coordinating the review of this manuscript and approving it for publication was Guitao Cao.

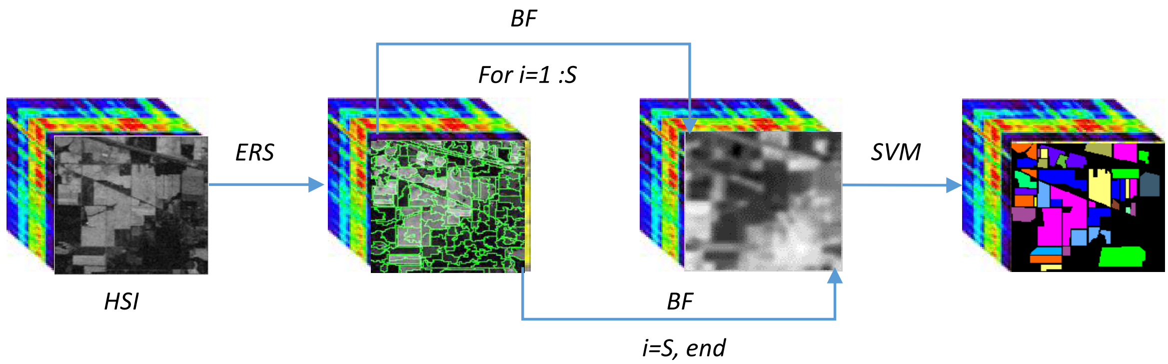


FIGURE 1. Schematic of SuperBF-based Classification for HSIs (S is the number of regions after ERS segmentation).

binary pattern (LBP) to extract local image texture features, obtaining excellent classification results. Zhou and Wei [26] proposed a deep hierarchical model of a spectral space network (SSN) and extracted spatial and spectral features of HSIs. The results showed that the SNN has excellent robustness and accuracy. Pan *et al.* [27] proposed the R-VCANet deep learning method, which can combine spectral and spatial features using a rolling guide filter (RGF) and extract the depth features of HSIs using the new vertex component analysis network (VCANet). The obtained features had a more powerful expression ability.

Recently, bilateral filtering (BF) was used to update the target pixel by weighted averaging of the neighbouring pixels through the spatial distance and the pixel value distance within the template, which has been demonstrated to be effective for feature extraction of HSIs [29]. Kang and Li [30] proposed a spectral-spatial feature extraction classification method that is based on edge-preserving filtering (EPF) and employs BF and guided filtering to ensure that a smooth probability is aligned with the edge of the real object; the method obtains reasonable results. Shen *et al.* [31] proposed a spectral-spatial feature extraction method for extreme learning machine (ELM) classifiers, which can improve the accuracy of the kernel-based ELM classifier by extracting spectral-spatial features via BF. Wang *et al.* [32] applied a combination of BF and graphic cutting technology to extract spectral-spatial features and improve the classification performance. Soomro *et al.* [33] combined elastic net regression and BF to extract spectral-spatial features, which improved the accuracy of the classifier.

However, due to the limitation of spatial resolution and the influence of the complexity of land feature distribution in hyperspectral remote sensing images, updating the target pixel through the weighted averaging of neighbouring pixels is prone to generating mixed pixels, i.e., the updated target pixel is mixed with the feature of other land objects in addition to that of the target object, decreasing the quality of image feature extraction. A superpixel BF algorithm (SuperBF) was proposed to extract HSI features. The specific framework is shown in Fig. 1 schematic of SuperBF-based Classification for HSIs. First, the HSI is segmented into many different regions via superpixel segmentation, and each region is considered to be a homogeneous region with a similar structure [34]. Second, each of the segmented homogeneous regions is filtered using BF. As the structural similarity of the pixels in the segmented homogeneous regions is extremely high, the possibility that the updated target pixel contains the features of other categories decreases, thus lowering the probability of mixed pixel generation. To verify the validity of the extracted features, the extracted features are classified using a common support vector machine (SVM) classifier.

The remainder of the article is organized as follows. The second section briefly introduces the entropy rate superpixel segmentation (ERS) algorithm and the related topic of BF; it also describes the feature extraction algorithm for HSIs based on SuperBF. The third section shows the experimental results and analysis. The fourth section presents the conclusion.

II. SUPERBF-BASED FEATURE EXTRACTION ALGORITHM FOR HYPERSPECTRAL IMAGES

A. ERS METHOD

In reference [35], the source image is replaced with a weighted undirected graph. Each pixel of the source image is treated as a node of the undirected graph. The similarity between the two nodes is employed as the weight between the two nodes. An objective function that combines the entropy rate of a random walk on a graph and a balancing term is employed. The segmentation result is obtained by iteratively maximizing this objective function. This method projects the image to an undirected graph $G = (V, E)$, where V is the set of vertices of the graph, E is the set of edges of the graph, and the weights of the edges represent the similarity among the vertices, which is quantified by the weight function $\omega : E \rightarrow R^{+} \cup \{0\}$. The graph is divided into connected subsets by selecting a subset of $A \subseteq E$, and the undirected graph $G = (V, E)$ is composed of smaller connected components / subgraphs. In the objective function of ERS, the superpixel segmentation is optimized by combining the entropy rate term

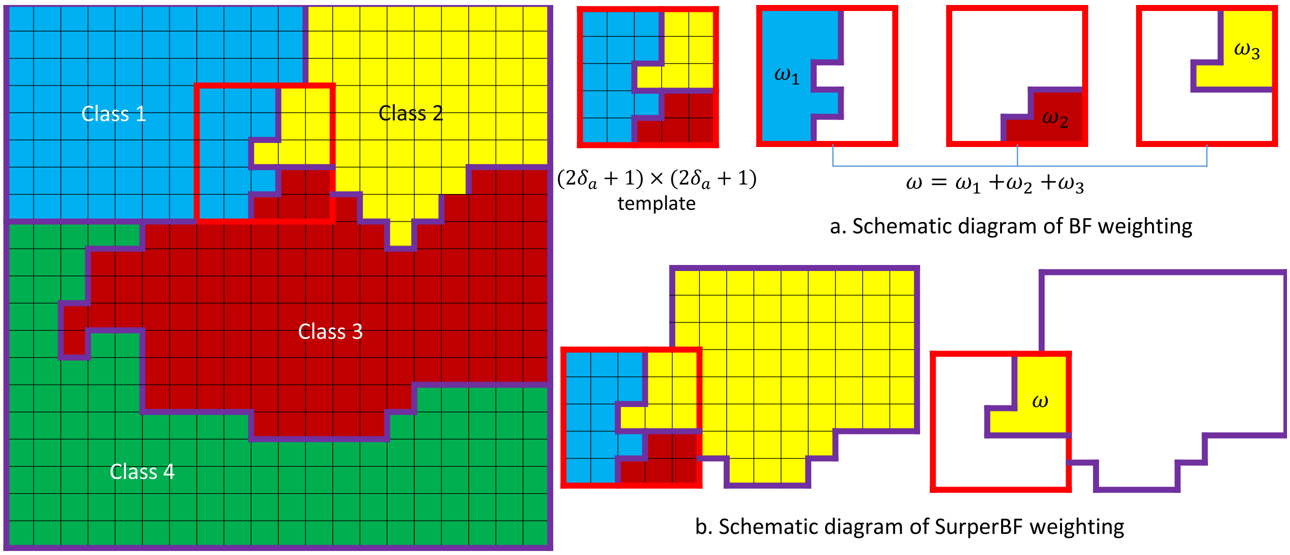


FIGURE 2. Schematic of BF and SuperBF weighting (The red box is $(2\delta_\alpha+1)\times(2\delta_\alpha+1)$ BF template, the area within the purple boundary is a homogeneous region, and different colours denote different categories.).

$H(A)$ and the balancing term $B(A)$.

$$A^* = argmaxTr\{H(A)+\alpha B(A)\}, \quad s.t. A \subseteq E. \tag{1}$$

where α is used to balance the contribution of the entropy rate term $H(A)$ and the balancing term $B(A)$. This function ensures a higher degree of similarity and homogeneity among the pixels within the segmentation regions. The first term can help form a uniform and compact cluster, while the second term can be used to encourage the clusters of similar size.

B. PRINCIPLE OF BF

BF is a type of nonlinear filter. The weighting coefficient is a nonlinear combination of a spatial distance measurement function and a grey value distance measurement function. The specific equations are expressed as follows:

$$O_s = \frac{1}{Z_s}\sum_{t\in N_s}\omega_{s,t}I_t \tag{2}$$

$$O_s = \frac{1}{Z_s}\sum_{t\in N_s}G_{\delta_\alpha}(\|s-t\|)\,G_{\delta_\gamma}(\|I_s-I_t\|)\,I_t \tag{3}$$

$$\omega_{s,t} = G_{\delta_\alpha}(\|s-t\|)\,G_{\delta_\gamma}(\|I_s-I_t\|) \tag{4}$$

$$Z_s = \sum_{t\in N_s}G_{\delta_\alpha}(\|s-t\|)\,G_{\delta_\gamma}(\|I_s-I_t\|) \tag{5}$$

where O is the output pixel grey value after filtering; I is the input pixel grey value; I_s and I_t represent the grey values of the pixels s and t, respectively; $\omega_{s,t}$ is the weight of the pixel t; Z_s is the filter normalization factor; δ_α is the filter radius; δ_γ is the filter ambiguity; N_s is the template with the centre as pixel s and the window size as $(2\delta_\alpha+1)\times(2\delta_\alpha+1)$; and pixel t represents a pixel at any position in the template. $G_{\delta_\alpha}(\|s-t\|)$ is a spatial proximity measurement function, $\|s-t\|$ is the Euclidean distance between any pixel t and the target pixel s in the template, $G_{\delta_\gamma}(\|I_s-I_t\|)$ is a pixel grey scale similarity measurement function, and $\|I_s-I_t\|$ is the pixel value distance between any pixel t and the target pixel s in the template. These two measurement functions are defined by Gaussian function:

$$G_{\delta_\alpha}(\|s-t\|) = exp\left(\frac{-\|s-t\|^2}{2\delta_\alpha^2}\right) \tag{6}$$

$$G_{\delta_\gamma}(\|I_s-I_t\|) = exp\left(\frac{-\|I_s-I_t\|^2}{2\delta_\gamma^2}\right) \tag{7}$$

C. SUPERBF-BASED FEATURE EXTRACTION ALGORITHM FOR HSIS

According to Eq. 2 through Eq. 7, when BF is performed for HSIs, if the distance between the non-structural similar pixels and the target pixel is relatively small, i.e., $\|s-t\|$ is small, its influence on the output value may be greater than that of the pixel points with a similar structure and large distance; accordingly, the proportion of non-structurally similar pixels in the updated target pixels will increase, making the method more prone to generating mixed pixels. As shown in Fig. 2a, BF assigns the weights to all non-structural similar pixels (such as the blue and the brown areas), which has a large negative impact on the output value and increases the abundance of non-structurally similar pixels, inevitably resulting in mixed pixels. In addition, the features of HSIs differ from those of general images. HSIs have many homogeneous regions, and the pixels in each homogeneous region are more likely to be structurally similar [36]. As shown in Fig. 2b, due to the limitations of BF and the characteristics of HSIs, homogenous regions can be reasonably segmented based on the homogeneity characteristics of HSIs, the area within thepurple boundary is a homogeneous region; then the homogenous regions can be separately filtered by BF, the homogeneous region of the yellow part is filtered, which

Algorithm 1 Algorithm of SuperBF-Based HSI Feature Extraction

Data: HSI $I = (I_1, I_2, \cdots, I_n) \in R^{d \times n}$, d is the dimension, n is the number of pixels, is the filter radius, δ_α is the filter radius, δ_γ is the filter ambiguity, and S is the number of regions after ERS segmentation.

Result: $O = (O_1, O_2, \cdots, O_n) \in R^{d \times n}$.

```
Segment the HSI into S homogeneous regions using Eq. 8;
for i = 1:S do
    Input the i-th homogeneous region;
    Count the number m of pixels in the i-th homogeneous region;
    for s = 1:m do
        Calculate the weight coefficients of any pixel t in the BF template of the i-th homogeneous region using Eq. 6 and Eq. 7 and Eq. 4;
        Calculate the pixel value O_s of the pixel s output by the BF filter operation using Eq. 2;
    end
end
Output O = (O_1, O_2, ..., O_n) ∈ R^{d×n}.
```

substantially enhances the restriction of BF for non-structural similar pixels and thus greatly decreases the abundance of non-structurally similar pixels in the update target pixels; and this process effectively avoids the generation of mixed pixels and renders BF-extracted HSI features more significant and distinguishable.

According to these ideas, a SuperBF algorithm was proposed in this study; the algorithm reasonably divides an image into homogeneous regions via superpixel segmentation. As ERS has excellent performance in many HSI superpixel segmentation methods, this study applied ERS to perform hyperpixel segmentation on HSIs. The specific equation is expressed as

$$I = U_k^S \kappa_k \quad s.t. \quad \kappa_k \cap \kappa_g = \emptyset, \ \ (k \neq g) \tag{8}$$

where S represents the number of superpixels, and κ_k is the k^{th} superpixel.

As shown in Fig. 2b, the superpixel segmentation uses the spatial continuity of the physical features to segment the HSI into different spectrally similar homogenous regions. This approach can considerably reduce the possibility of occurrence of pixels with large differences in non-structural similarity in the BF template, enhance the influence of structurally similar pixels in the BF template on the output value, and solve the problem of the large negative impact of weighting of non-structural similar pixels on the output value, effectively avoiding the generation of mixed pixels.

After ERS is performed on the HSI to obtain S segmented homogeneous regions, BF is used to filter each segmented homogeneous region. Algorithm 1 describes the specific process of the SuperBF-based HSI feature extraction. The algorithm is divided into two steps. In the first step, ERS is employed to segment the HSI and divide the pixels with similar structure into the same region to segment the image into multiple homogeneous regions. In the second step, the BF algorithm is applied to filter the pixels in each homogeneous region and extract the HSI features.

III. EXPERIMENTAL RESULTS AND ANALYSIS

This study compared the proposed SuperBF-SVM classification method with several currently popular classification methods, including SVM [37], BF-SVM [30], EPF-SVM [30], LBP-ELM [25], HiFi [38], and R-VCANet-SVM [27]. The SVM algorithm was implemented in the libsvm [39] library with five-fold cross-validation, and the default parameters in the references were employed in other algorithms. PAN et al. constructed a hierarchical guidance filtering (HiFi) and a matrix of spectral angle distance and iteratively trained classifiers using the integrated learning spatial and spectral information from different scales to achieve good generalization performance. Similar to many previous studies, the performance of different HSI classifications was evaluated using overall accuracy (OA), average accuracy (AA), and kappa coefficients. The OA indicates the probability that the classification results are consistent with the reference classification results. The AA refers to the mean of the percentage of correctly classified pixels for each class. The kappa coefficient is used for consistency check.

A. DATA SET DESCRIPTION

To verify the effectiveness of the proposed method, three real HSIs of Indian Pines, Salinas, and University of Pavia were employed in the experiments.

The image of Indian Pine was acquired by an airborne visible/infrared imaging spectrometer (AVIRIS) sensor. The image shows an agricultural pine test site in northwestern Indiana. The size of the image is 145×145, the spatial resolution is 20 m, and the spectral range extends from 0.4 to 2.45 μm. The image contains 224 bands, of which 24 bands were removed due to water vapour absorption; 200 bands remain.

The image of Salinas was acquired by an AVIRIS sensor. The image shows Salinas Valley, California, USA. The size of the image is 512×217, and the spatial resolution is 3.7 m. The image contains 224 bands, of which 24 bands were removed; 200 bands remain.

The image of University of Pavia was acquired by the reflective optical system imaging spectrometer (ROSIS) sensor. The image shows the urban area around the University of Pavia. The size of the image is 610×340, the spatial resolution is 1.3 m, and the spectral range extends from 0.43 to 0.86 μm. The image contains 115 bands, of which 12 bands of the noise channels were removed; 103 bands remain.

To ensure the objectivity of the experiment, the experiment was repeated 10 times, and the average value was used as

TABLE 1. Classification accuracy of different methods for Indian Pines.

Class	Train	Test	SVM	BF	EPF	LBP-ELM	HiFi	R-VCANet	SuperBF
aifalfa	20	26	55.00	54.17	57.78	**100**	**100**	**100**	**100**
corn_n	20	1408	52.16	82.97	85.80	78.85	84.94	65.41	**93.13**
corn_m	20	810	63.35	62.12	89.35	92.14	**93.09**	85.31	91.27
corn	20	217	53.33	67.48	43.06	**100**	87.10	97.24	91.81
grass_m	20	463	82.80	79.92	92.93	96.02	92.01	91.36	**97.58**
grass_t	20	710	85.91	93.35	91.93	**99.59**	97.61	96.48	99.44
grass_p	14	14	37.14	32.56	82.35	**100**	**100**	**100**	18.42
hay_w	20	458	97.89	**100**	**100**	**100**	99.78	99.13	**100**
oats	10	10	27.27	32.00	**100**	**100**	**100**	**100**	90.91
soybean_n	20	952	57.38	56.25	66.32	89.87	**93.70**	83.61	81.61
soybean_m	20	2435	71.57	89.12	92.13	74.96	78.52	71.79	**95.92**
soybean_c	20	573	37.88	72.47	52.77	85.86	94.24	87.43	**96.00**
wheat	20	185	88.14	92.86	**100**	**100**	99.46	99.46	**100**
woods	20	1245	92.55	98.03	96.94	99.45	98.23	95.74	**100**
buildings	20	366	39.31	78.71	88.99	**100**	93.99	95.36	86.29
stone	20	73	95.77	87.80	87.95	94.12	**100**	**100**	97.33
OA			66.27±2.46	79.57±1.65	83.03±1.85	88.19±1.77	89.82±2.01	83.23±1.75	**93.69±1.12**
AA			64.84±2.28	78.32±2.55	83.02±3.19	94.43±1.01	**94.54±0.97**	91.77±0.82	89.98±2.09
kappa			0.62±0.03	0.77±0.01	0.81±0.02	0.87±0.02	0.87±0.02	0.81±0.01	**0.93±0.01**

TABLE 2. Classification accuracy of different methods for Salinas.

Class	Train	Test	SVM	BF	EPF	LBP-ELM	HiFi	R-VCANet	SuperBF
weeds_1	20	1989	98.05	**100**	**100**	99.90	98.49	99.90	**100**
weeds_2	20	3706	99.37	**100**	99.89	97.38	98.70	99.84	**100**
fallow	20	1956	91.22	96.54	94.91	**100**	99.80	99.39	**100**
fallow_p	20	1374	97.68	91.30	97.86	99.42	97.45	**99.56**	90.03
fallow_s	20	2658	97.00	99.10	**99.96**	96.88	88.75	99.62	98.71
stubble	20	3939	**100**	**100**	99.92	91.77	99.59	99.97	**100**
celery	20	3559	99.94	99.44	**100**	98.90	96.60	98.17	99.92
grapes	20	11251	72.98	87.53	82.04	90.00	82.13	78.54	**99.91**
soil	20	6183	98.59	98.75	99.48	99.13	99.97	99.26	99
corn	20	3258	79.39	91.66	85.06	94.81	87.97	94.69	**96.15**
lettuce_4	20	1048	93.65	94.08	98.21	99.62	96.18	98.76	**99.81**
lettuce_5	20	1907	94.34	99.74	**100**	93.55	99.48	**100**	98.96
lettuce_6	20	896	93.37	96.61	96.10	91.74	97.21	94.31	**100**
lettuce_7	20	1050	92.29	89.02	**99.20**	94.48	92.67	96.86	93.79
vinyard_U	20	7248	54.30	77.53	73.97	91.67	73.17	85.32	**99.99**
vinyard_T	20	1787	94.44	97.15	99.49	**100**	96.75	99.27	95.00
OA			84.96±1.17	92.76±0.95	91.41±2.29	94.86±1.13	90.50±1.32	91.58±1.09	**98.98±0.56**
AA			91.04±0.53	94.90±0.63	95.38±0.85	96.20±0.48	94.06±0.68	96.05±0.40	**98.2±0.72**
kappa			0.83±0.01	0.91±0.01	0.90±0.03	0.94±0.01	0.89±0.01	0.91±0.01	**0.99±0.01**

TABLE 3. Classification accuracy of different methods for the University of Pavia.

Class	Train	Test	SVM	BF	EPF	LBP-ELM	HiFi	R-VCANet	SuperBF
asphalt	20	18629	87.52	95.46	**98.05**	68.90	80.40	79.96	98.03
meadows	20	2079	91.00	**98.03**	97.40	84.14	89.74	83.39	95.34
gravel	20	3044	61.72	81.61	**89.16**	83.55	82.92	88.12	67.42
trees	20	1325	70.10	75.43	96.20	76.84	83.64	96.75	**98.38**
sheets	20	5009	98.42	93.01	95.05	88.68	99.17	**100**	99.10
soil	20	1310	46.04	71.57	64.27	96.57	89.72	93.57	90.06
bitumen	20	3662	54.64	84.60	58.20	90.53	96.79	99.01	91.45
bricks	20	927	80.23	76.97	76.20	91.29	92.55	88.39	**95.21**
shadows	20	170	**100**	**100**	99.89	68.18	99.46	**100**	99.89
OA			75.73±1.64	88.05±2.86	87.00±2.43	83.29±1.68	88.48±1.90	87.03±1.19	**93.30±1.87**
AA			76.63±1.43	88.93±2.65	86.05±2.39	83.19±1.27	90.49±0.97	91.17±0.89	**92.78±1.89**
kappa			0.69±0.02	0.85±0.03	0.83±0.03	0.79±0.02	0.83±0.02	0.83±0.01	**0.91±0.02**

the result. 20 training samples were randomly selected in each of the three data sets, and the remaining samples were used as test samples to test the effectiveness of the proposed method, as indicated in Tables 1 to 3. To test the stability of the algorithm, 10-50 samples were randomly selected from the three data sets to use as training samples, and the

TABLE 4. **Classification accuracy of different training samples for Indian Pines.**

perclass	SVM			BF			EPF			LBP-ELM		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	57.43	55.87	0.52	67.96	66.48	0.64	69.32	72.06	0.66	80.89	89.16	0.79
20	66.27	64.84	0.62	79.57	73.74	0.77	83.03	83.02	0.81	88.19	94.43	0.87
30	73.31	69.84	0.70	85.00	79.69	0.83	87.41	87.6	0.86	92.57	96.09	0.92
40	75.94	72.67	0.73	87.42	83.42	0.86	89.63	89.74	0.88	94.42	96.76	0.94
50	78.66	75.86	0.76	90.15	87.06	0.89	92.41	92.02	0.91	95.76	97.77	0.95

perclass	HiFi			R-VCANet			SuperBF		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	81.08	**89.44**	0.79	75.40	85.82	0.72	**87.32**	87.43	**0.86**
20	89.82	**94.54**	0.88	83.23	91.77	0.81	**93.69**	89.98	**0.93**
30	91.65	**95.74**	0.91	87.56	94.00	0.86	**95.21**	93.81	**0.95**
40	93.63	**96.36**	0.93	89.66	95.05	0.87	**96.31**	94.84	**0.96**
50	93.44	**96.72**	0.93	91.33	95.88	0.90	**97.23**	95.52	**0.97**

TABLE 5. **Classification accuracy of different training samples for Salinas.**

perclass	SVM			BF			EPF			LBP-ELM		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	82.64	88.87	0.81	89.44	92.61	0.88	87.71	93.80	0.86	90.41	92.92	0.89
20	84.96	91.04	0.83	92.76	94.90	0.91	91.41	95.38	0.90	94.86	96.20	0.94
30	86.42	91.38	0.85	93.71	95.89	0.93	92.70	95.96	0.92	96.45	96.81	0.97
40	86.20	91.77	0.85	94.02	96.10	0.93	92.73	96.12	0.92	97.69	98.38	0.97
50	87.70	92.75	0.86	95.04	96.63	0.94	94.15	96.85	0.93	98.02	98.67	0.98

perclass	HiFi			R-VCANet			SuperBF		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	86.53	92.08	0.85	87.96	94.32	0.87	**97.88**	**96.62**	**0.98**
20	90.50	94.06	0.89	91.58	96.05	0.91	**98.98**	**98.20**	**0.99**
30	92.08	95.47	0.91	92.93	96.68	0.92	**99.03**	**98.47**	**0.99**
40	92.67	96.20	0.92	93.29	96.91	0.93	**99.05**	**98.50**	**0.99**
50	93.59	96.76	0.93	94.21	97.34	0.94	**99.28**	**98.77**	**0.99**

TABLE 6. **Classification accuracy of different training samples for University of Pavia.**

perclass	SVM			BF			EPF			LBP-ELM		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	67.02	69.12	0.59	76.44	76.79	0.70	73.76	76.21	0.67	73.98	76.15	0.67
20	75.73	76.63	0.69	88.05	86.3	0.85	87.00	86.05	0.83	83.29	83.19	0.79
30	78.95	77.69	0.73	89.37	87.5	0.86	88.97	88.56	0.86	86.52	86.42	0.83
40	82.30	80.23	0.77	92.41	89.88	0.90	92.19	90.89	0.90	88.83	87.93	0.85
50	83.78	81.36	0.79	93.9	91.81	0.92	93.57	92.66	0.92	90.77	90.36	0.88

perclass	HiFi			R-VCANet			SuperBF		
	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa	OA (%)	AA (%)	kappa
10	81.83	85.40	0.77	81.47	**87.21**	0.76	**82.14**	82.14	**0.77**
20	88.48	90.49	0.83	87.03	92.13	0.83	**93.30**	**92.78**	**0.91**
30	88.64	91.91	0.85	90.95	93.51	0.88	**94.67**	**94.11**	**0.93**
40	90.22	92.99	0.87	92.18	94.48	0.90	**95.54**	**94.62**	**0.94**
50	90.94	93.58	0.88	93.46	**95.51**	0.91	**96.03**	94.92	**0.95**

remaining samples were used as test samples, as indicated in Table 4 to 6.

B. PARAMETER ANALYSIS

The algorithm proposed in this study involves three important parameters: the number of super-pixels S, the size of the filter δ_α and the degree of ambiguity δ_γ. As shown in Fig. 3, the influence of these three parameters on the OA of SVM classifier in the three images was analysed. When one parameter was analysed, the other two parameters were fixed. When the numbers of super-pixels S in the three scenarios of Indian Pine, Salinas, and University of Pavia were 30, 10, and 110,

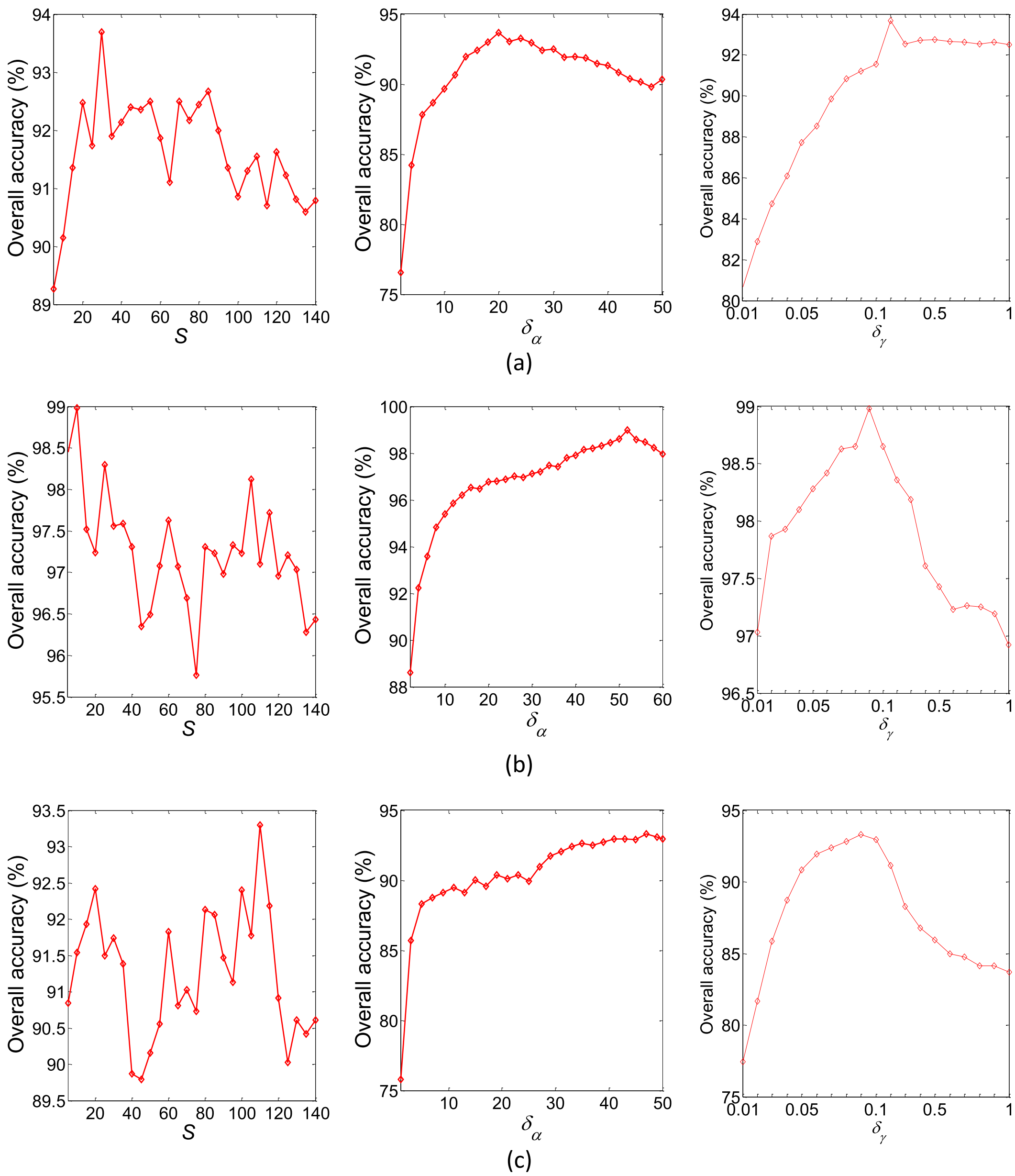


FIGURE 3. Influence of three parameters of S, δ_α and δ_γ on the three data sets: (a) Indian Pine, (b) Salinas, and (c) University of Pavia.

respectively, the proposed method obtained the highest OA. As the number of super-pixels S was increased, the experimental results showed that the total performance initially increased and then decreased. The superpixels with an excessively small or large S can cause performance degradation of the proposed SuperBF method because too many superpixels can cause excessive concentration and all samples belonging to a uniform region would not be fully utilized, while too few superpixels can cause excessive decomposition and introduce some non-homogeneous samples from different uniform regions. The ideal effects were obtained when the δ_α of the three scenarios of Indian Pine, Salinas, and University of Pavia were 20, 52, and 47, respectively. If the δ_α is too small, some useful spatial information will be disregarded; if the δ_α is too large, an excessive amount of useless information will be acquired. The classification performance was the best when the δ_γ values of the three scenarios of Indian Pine, Salinas, and University of Pavia were 0.2, 0.09, and 0.09, respectively. If the δ_γ is too small, the result will not be sufficiently smooth; if it is too large, the result will be too smooth. Therefore, the parameters of the three scenarios in this study were set as follows: Indian Pine: $S = 30$, $\delta_\alpha = 20$, $\delta_\gamma = 0.2$; Salinas: $S = 10$, $\delta_\alpha = 52$, $\delta_\gamma = 0.09$; University of Pavia: $S = 110$, $\delta_\alpha = 47$, $\delta_\gamma = 0.09$.

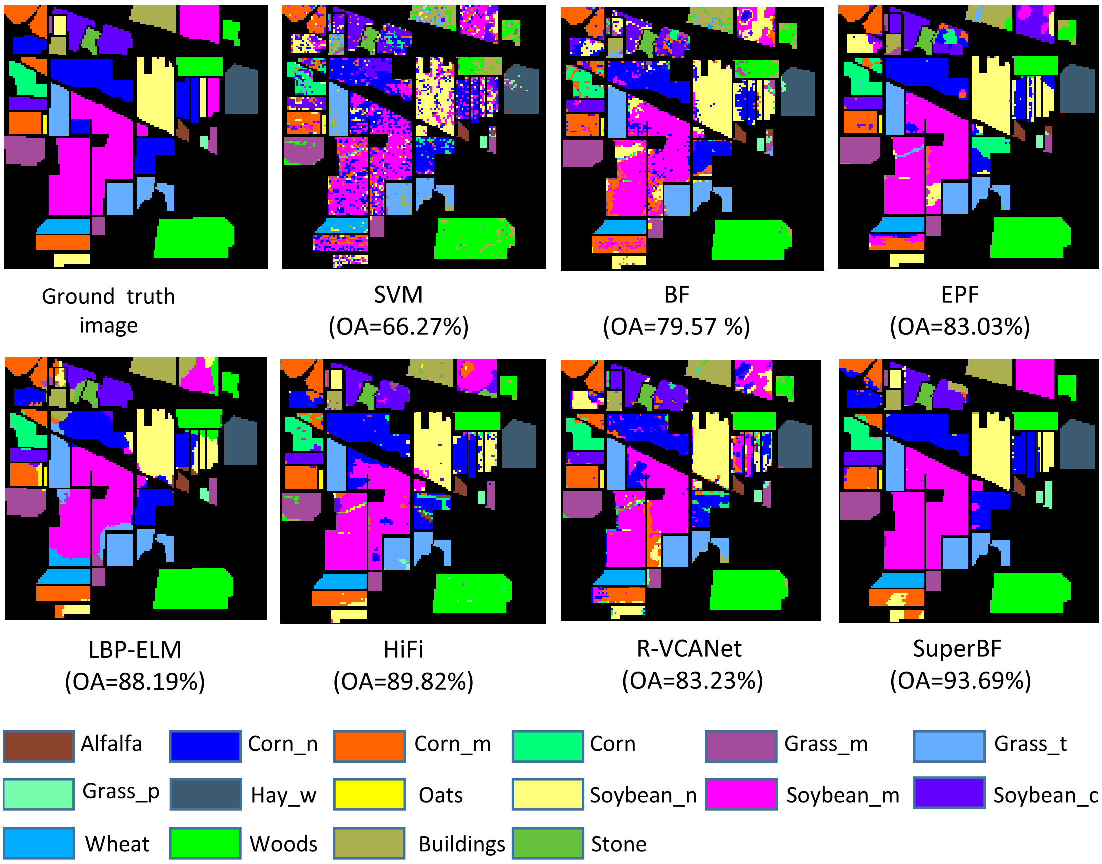


FIGURE 4. **Classification results of the Indian Pines image.**

C. EXPERIMENTAL RESULTS

The improvement for BF in SuperBF is effective. In image segmentation, an image is segmented into many different regions, and each region is considered to be homogenous [35]. The regions form a segmentation map of the spatial structure that can be employed for spectral-spatial classification. As BF can filter in these segmentation ranges, the extracted features are more effective, and the classification accuracy is higher. As shown in Figs. 4 through 6 and Tables 1 through 3, in the three scenarios of Indian Pine, Salinas and University of Pavia, the OA, AA and kappa of SurperBF were greater than those of BF. When the number of training samples was 20, the OA was greater by 14.12%, 6.22%, and 5.25%. Compared with the improved EPF algorithm based on BF, the OA, AA, and kappa of SurperBF were also greater than those of EPF. When the number of training samples was 20, the OA was greater by 10.66%, 7.57%, and 6.30%.

The SuperBF classification method is superior to some advanced methods. As shown in Figs. 4 through 6 and Tables 1 through 3, with the exception of the AA of Indian Pine, the SuperBF method obtained the best OA, AA and kappa. Compared with the three advanced methods of LBP-ELM, HiFi, and R-VCANet methods of deep learning, the OA values of the SuperBF classification method was greater by 5.5%, 3.87%, and 10.46%, respectively, in the Indian Pine scenario; greater by 4.12%, 8.48%, and 7.40%, in the Salinas scenario; and greater by 10.01%, 4.82%, and 6.27%, in the University of Pavia scenario. AA was not the best in Indian Pine as the classification accuracy of grass_p was only 18.42%, which may be related to the small number of grass_p; it was similar to grass_m, which causes misclassification.

The SuperBF classification method has strong robustness. As shown in Tables 4 through 6 and Figs. 7 through 9, when the number of training samples was increased from 10 to 50, the OA, AA and kappa also increased, and the highest OA and kappa were obtained by SuperBF. Compared with other classification methods, the OA was greater by a minimum of 3.87%, and the OA in the Indian Pines scenario was the highest, which was even greater than that of the SVM method by 27.42%. Especially in the Salinas scenario, for the condition in which the OA was greater by more than 90%, with the exception of the SVM method, the OA of the proposed method exceeded that of other classification methods by 4.12% to 14.01%.

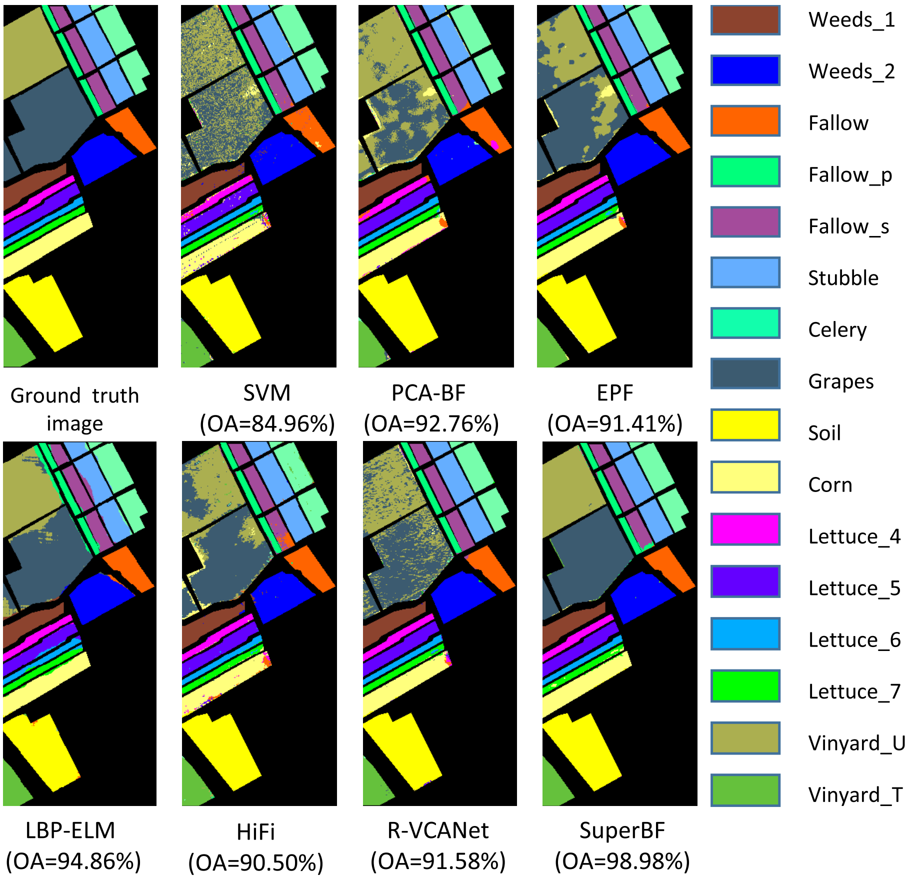


FIGURE 5. Classification results of the Salinas image.

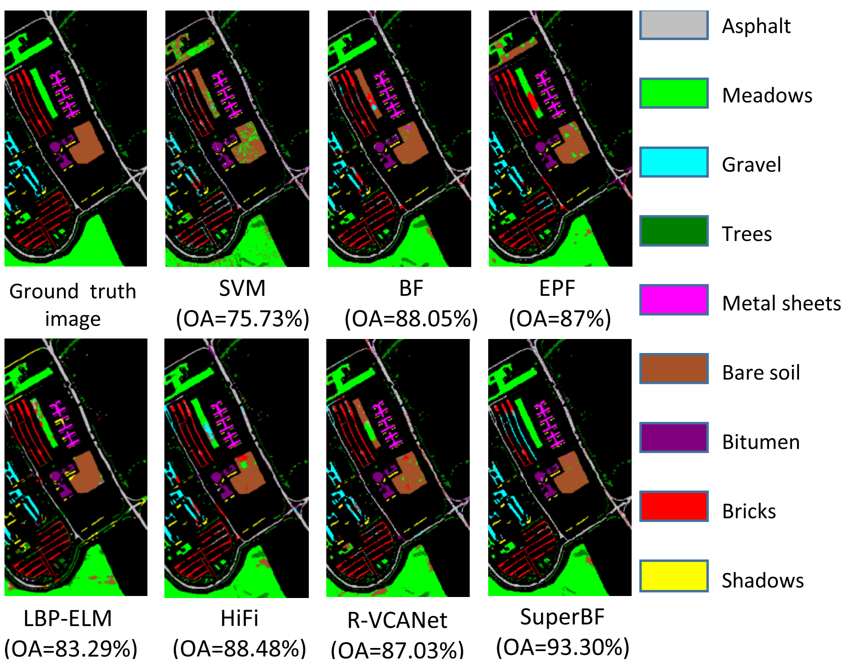


FIGURE 6. Classification results of the University of Pavia image.

The robust expression of the SuperBF classification method is effective for the problem of images with a small sample size. Achieving a fine classification of HSIs is challenging in the case of a small number of samples. As reported in Tables 4 through 6 and Figs. 7 through 9, when the number of training samples was small (for example, 10),

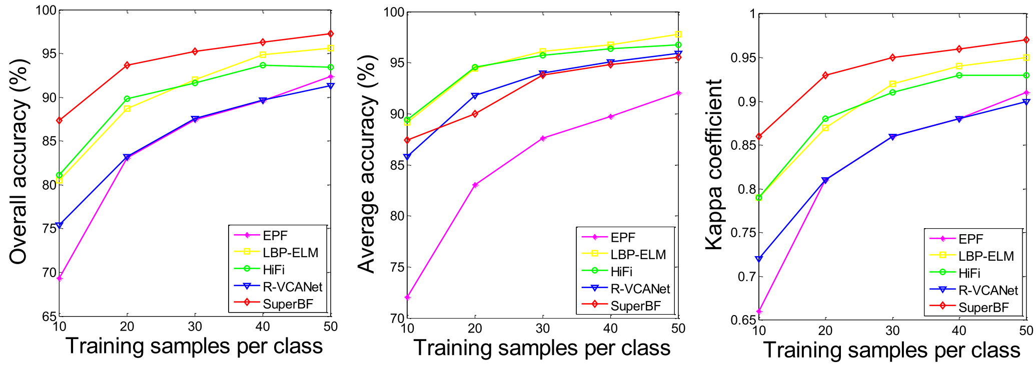


FIGURE 7. Influence of the training samples on the Indian Pines dataset.

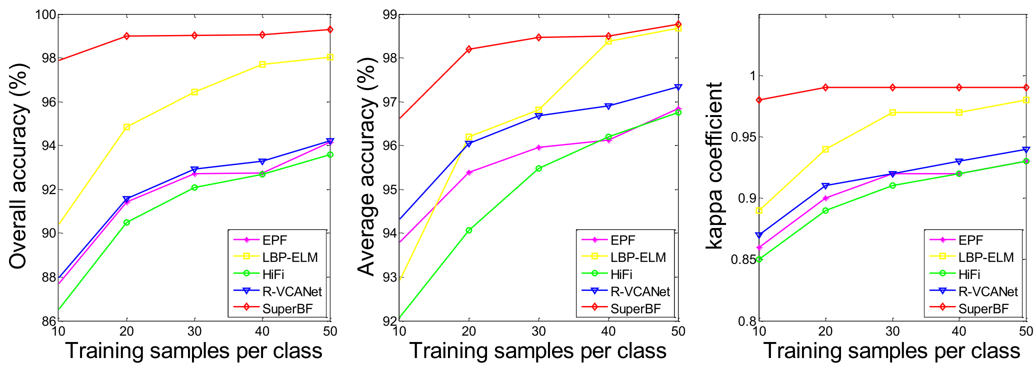


FIGURE 8. Influence of the training samples on the Salinas dataset.

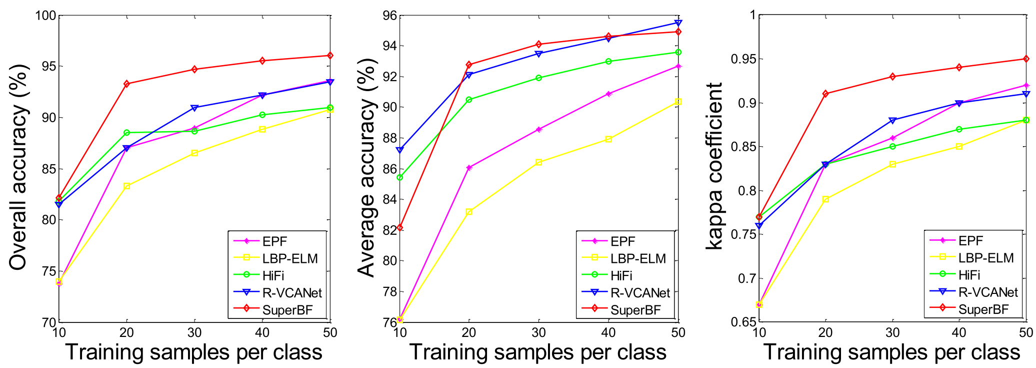
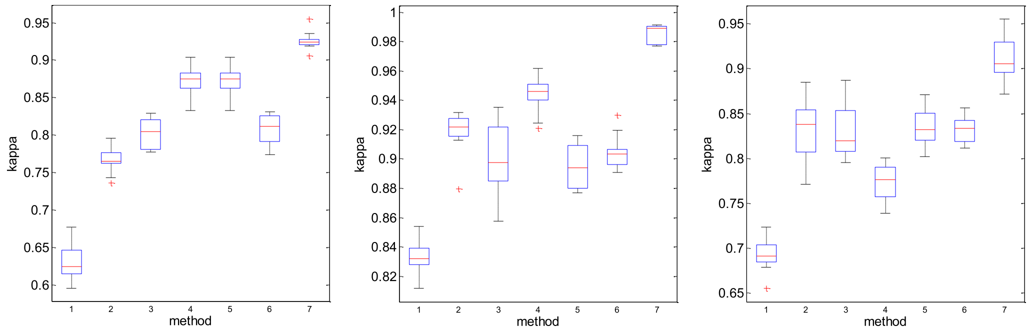


FIGURE 9. Influence of the training samples on the University of Pavia dataset.

the OA of many classification methods was not high. In the Indian Pines scenario, the OA of the SVM classification method was only 57.43%; in the University of Pavia scenario, the OA of the SVM classification method was only 67.02%. The OA of many methods ranged from 70% to 79%. In this case, the method proposed in this study was effectively improved for a small sample size. For example, in the Indian Pines scenario, when the number of the training samples was 10, compared with other methods, the OA increased by 6.24-29.89%; in the Salinas scenario, the OA increased by 7.47%-15.24%; and in the University of Pavia scenario, the OA increased by 0.31%-15.12%. In the Salinas scenario, when the number of the training samples was 10, the OA of the SuperBF classification method was 97.88%, and the category of the real objects was almost completely and correctly identified. Therefore, the results of SuperBF are very competitive when solving the problem of images with a small sample size.

Statistical evaluation about the results: To further validate whether the observed gains in kappa is statistically significant, we use paired t-test to show the statistical evaluation about the results. T-test is popular in many related works. We accept the hypothesis that the mean kappa of SuperBF-SVM is larger than a compared method only if Eq. 9



(a) Indian Pine (b) Salinas (c) University of pavia

FIGURE 10. **Box plot of kappa of different methods on three datasets. (a) Indian Pine (b) University of pavia (c) Salinas 1. SVM 2. BF-SVM 3. EPF-SVM 4. LBP-ELM 5. HiFi 6. R-VCANet-SVM 7. SuperBF-SVM. The center line is the median value, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme points, and the abnormal outliers are plotted by '+'.**

is valid:

$$\frac{(\bar{a}_1 - \bar{a}_2)\sqrt{n_1 + n_2 - 2}}{\sqrt{(\frac{1}{n_1} + \frac{1}{n_2})(n_1 s_1^2) + n_2 s_2^2}} > t_{1-\alpha}[n_1 + n_2 - 2] \tag{9}$$

where $\bar{a}_1$ and $\bar{a}_2$ are the means of kappa of SuperBF-SVM and a compared method, s_1 and s_2 are the corresponding standard deviations, n_1 and n_2 are the number of realizations of experiments reported which is set as 10 in this paper. Paired t-test shows that the increases on kappa are statistically significant in all the three datasets (at the level of 95%), and it can be also observed in Figure 10.

IV. CONCLUSION AND FUTURE WORK

This study proposed a simple and effective SuperBF based algorithm for the feature extraction of HSIs. In this study, a HSI is divided into multiple homogeneous regions with a similar structure. The BF can effectively limit the influence of non-structurally similar pixels on the target pixel during the filtering process, which improves the effect of BF filtering and more effectively extracts the HSI features. The experimental results show that the proposed method is superior to existing advanced feature extraction methods, especially when solving the problem of images with a small sample size.

Our future work is data imbalance. By convention, in a sample-size-related imbalanced data set, the classes with small size are named minority classes, and the ones with large size are named majority classes. The common situation in performance assessment is that the correct classification of large-size classes contributes more than that of small-size classes. In SuperBF-SVM, AA was not the best in Indian Pine as the classification accuracy of grass_p was only 18.42%, which may be related to the small number of grass_p i.e. small class. Therefore, we will propose a novel solution to solve the sample-size-related imbalanced data problem more effectively. The new solution consists of two parts: one for large-size and the other for small-size.

REFERENCES

[1] M. J. Khan, H. S. Khan, A. Yousaf, K. Khurshid, and A. Abbas, "Modern trends in hyperspectral image analysis: A review," *IEEE Access*, vol. 6, pp. 14118–14129, 2018.

[2] Y. Zhou, J. Peng, and C. L. P. Chen, "Extreme learning machine with composite kernels for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2351–2360, Jun. 2015.

[3] N. H. Ly, Q. Du, and J. E. Fowler, "Sparse graph-based discriminant analysis for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 3872–3884, Jul. 2014.

[4] J. Peng, W. Sun, and Q. Du, "Self-paced joint sparse representation for the classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1183–1194, Feb. 2019.

[5] J. Jiang, C. Chen, Y. Yu, X. Jiang, and J. Ma, "Spatial-aware collaborative representation for hyperspectral remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 3, pp. 404–408, Mar. 2017.

[6] Y. Yuan, D. Ma, and Q. Wang, "Hyperspectral anomaly detection via sparse dictionary learning method of capped norm," *IEEE Access*, vol. 7, pp. 16132–16144, 2019.

[7] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, Mar. 2013.

[8] J. Jiang, J. Ma, Z. Wang, C. Chen, and X. Liu, "Hyperspectral image classification in the presence of noisy labels," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 851–865, Feb. 2018.

[9] J. Li, H. Zhang, L. Zhang, and L. Ma, "Hyperspectral anomaly detection by the use of background joint sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2523–2533, Jun. 2015.

[10] J. Ma, J. Jiang, H. Zhou, J. Zhao, and X. Guo, "Guided locality preserving feature matching for remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4435–4447, Aug. 2018.

[11] W. Sun, G. Yang, B. Du, L. Zhang, and L. Zhang, "A sparse and low-rank near-isometric linear embedding method for feature extraction in hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 4032–4046, Jul. 2017.

[12] L. Xu, A. Wong, F. Li, and D. A. Clausi, "Intrinsic representation of hyperspectral imagery for unsupervised feature extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 1118–1130, Feb. 2016.

[13] B. Tu, C. Zhou, W. Kuang, L. Guo, and X. Ou, "Hyperspectral imagery noisy label detection by spectral angle local outlier factor," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 9, pp. 1417–1421, Sep. 2018.

[14] W. Li, J. Liu, and Q. Du, "Sparse and low-rank graph for discriminant analysis of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4094–4105, Jul. 2016.

[15] H. Pu, Z. Chen, B. Wang, and G.-M. Jiang, "A novel spatial–spectral similarity measure for dimensionality reduction and classification of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7008–7022, Nov. 2014.

[16] B. Tu, X. Zhang, X. Kang, G. Zhang, and S. Li, "Density peak-based noisy label detection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1573–1584, Mar. 2019.

[17] B. Tu, X. Zhang, X. Kang, J. Wang, and J. A. Benediktsson, "Spatial density peak clustering for hyperspectral image classification with noisy labels," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5085–5097, Jul. 2019.

[18] Q. Xie, J. Huang, D. Liang, P. Chen, C. Wu, G. Yang, J. Zhang, L. Huang, and D. Zhang, "Leaf area index estimation using vegetation indices derived from airborne hyperspectral images in winter wheat," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 8, pp. 3586–3594, Aug. 2014.

[19] A. Banskota, S. P. Serbin, R. H. Wynne, V. A. Thomas, M. J. Falkowski, N. Kayastha, J.-P. Gastellu-Etchegorry, and P. A. Townsend, "An LUT-based inversion of DART model to estimate forest LAI from hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observat. Remote Sens.*, vol. 8, no. 6, pp. 3147–3160, Jun. 2015.

[20] M. Montes, J. Ryan, C. Davis, N. Tufillaro, R. Kudela, and J. Ryan, "Application of the hyperspectral imager for the coastal ocean to phytoplankton ecology studies in monterey Bay, CA, USA," *Remote Sens.*, vol. 6, no. 2, pp. 1007–1025, Dec. 2013.

[21] T.-M. Tu, C.-H. Chen, and C.-I. Chang, "A noise subspace projection approach to target signature detection and extraction in an unknown background for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 1, pp. 171–181, Jan. 1998.

[22] X. Jiang, X. Fang, Z. Chen, J. Gao, J. Jiang, and Z. Cai, "Supervised Gaussian process latent variable model for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1760–1764, Oct. 2017.

[23] Z. Chen, J. Jiang, X. Jiang, X. Fang, and Z. Cai, "Spectral-spatial feature extraction of hyperspectral images based on propagation filter," *Sensors*, vol. 18, no. 6, pp. 1978–1993, Jun. 2018.

[24] J. Jiang, J. Ma, C. Chen, Z. Wang, Z. Cai, and L. Wang, "SuperPCA: A superpixelwise PCA approach for unsupervised feature extraction of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4581–4593, Aug. 2018.

[25] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3681–3693, Jul. 2015.

[26] Y. Zhou and Y. Wei, "Learning hierarchical spectral–spatial features for hyperspectral image classification," *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1667–1678, Jul. 2016.

[27] B. Pan, Z. Shi, and X. Xu, "Hierarchical guidance filtering-based ensemble classification for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 4177–4189, Jul. 2017.

[28] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int. Conf. Comput. Vis.*, Jan. 1998, pp. 839–846.

[29] Z. Chen, J. Jiang, C. Zhou, X. Jiang, S. Fu, and C. Zhihua, "Trilateral smooth filtering for hyperspectral image feature extraction," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 781–785, May 2019. doi: 10.1109/LGRS.2018.2881704.

[30] X. Kang, S. Li, and J. A. Benediktsson, "Spectral–spatial hyperspectral image classification with edge-preserving filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2666–2677, May 2014.

[31] Y. Shen, J. Xu, H. Li, and L. Xiao, "ELM-based spectral-spatial classification of hyperspectral images using bilateral filtering information on spectral band-subsets," in *Proc. IEEE Int. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 497–500.

[32] Y. Wang, H. Song, and Y. Zhang, "Spectral-spatial classification of hyperspectral images using joint bilateral filter and graph cut based model," *Remote Sens.*, vol. 8, no. 9, p. 748. Sep. 2016.

[33] S. Soomro, L. Xiao, and B. N. Soomro, "Hyperspectral image classification via Elastic Net Regression and bilateral filtering," in *Proc. PIC*, Dec. 2015, pp. 56–60.

[34] C. Qin, G. Zhang, Y. Zhou, W. Tao, and Z. Cao, "Integration of the saliency-based seed extraction and random walks for image segmentation," *Neurocomputing*, vol. 129, no. 4, pp. 378–391, Apr. 2014.

[35] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 2097–2104.

[36] J. Peng, L. Li, and Y. Y. Tang, "Maximum likelihood estimation-based joint sparse representation for the classification of hyperspectral remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1790–1802, Jun. 2019.

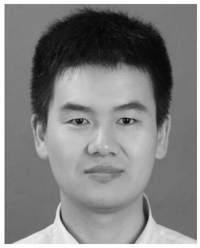
[37] M. Pal and G. M. Foody, "Feature selection for classification of hyperspectral data by SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 5, pp. 2297–2307, May 2010.

[38] B. Pan, Z. Shi, and X. Xu, "R-VCANet: A new deep-learning-based hyperspectral image classification method," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 5, pp. 1975–1986, May 2017.

[39] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

ZHIKUN CHEN received the B.Sc. degree from Nanning Normal University, Nanning, Guangxi, China, in 2003, and the M.Sc. and Ph.D. degrees from the China University of Geosciences, Wuhan, in 2008 and 2018, respectively.

He is currently an Associate Professor with the Key Laboratory of Marine Geographic Information Resources Development and Utilization in the Beibu Gulf, College of Resources and Environment, Beibu Gulf University, Qinzhou, Guangxi, China.



JUNJUN JIANG (M'15) received the B.S. degree from the Department of Mathematics, Huaqiao University, Quanzhou, China, in 2009, and the Ph.D. degree from the School of Computer, Wuhan University, Wuhan, China, in 2014.

From 2015 to 2018, he was an Associate Professor with the School of Computer Science, China University of Geosciences, Wuhan. Since 2016, he has been a Project Researcher with the National Institute of Informatics, Tokyo, Japan. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China.

CHONG ZHOU received the Ph.D. degree from the China University of Geosciences, Wuhan, in 2018.

He is currently a Lecturer with the School of Information Engineering, Hebei GEO University, Shijiazhuang, China. His current research interests include multiobjective optimization and hyperspectral image classification.

SHAOYUAN FU is currently an Intern-Researcher with the College of Resources and Environment, Beibu Gulf University, Qinzhou, Guangxi, China. Her current research interests include filtering algorithm and hyperspectral image classification.

ZHIHUA CAI received the B.Sc. degree from Wuhan University, Wuhan, China, in 1986, the M.Sc. degree from the Beijing University of Technology, Beijing, China, in 1992, and the Ph.D. degree from the China University of Geosciences, Wuhan, in 2003.

He is currently a Professor with the School of Computer Science, China University of Geosciences. He is also a Qinzhou Seagull Scholar with the Key Laboratory of Marine Geographic Information Resources Development and Utilization in the Beibu Gulf, Beibu Gulf University, Qinzhou, China.