

Local Binary Patterns and Superpixel-Based Multiple Kernels for Hyperspectral Image Classification

Wei Huang , Member, IEEE, Yao Huang, Hua Wang, Yan Liu, and Hiuk Jae Shim

Abstract—The superpixel-based multiple kernels model uses the average value of all pixels within superpixel as the spatial feature, which results in inaccurate extraction of edge pixels. To solve this problem, a local binary patterns and superpixel-based multiple kernels method is proposed for hyperspectral image (HSI) classification. First, the original HSI is segmented into multiple superpixels by using the entropy rate superpixel segmentation algorithm. On the HSI with superpixel index, the spectral kernel is second obtained by combining the spectral feature map with the radial basis kernel (RBF). By introducing local binary pattern (LBP) and weighted average filtering into RBF, the spatial kernels are obtained within and among superpixels. Finally, the combined kernel containing the abovementioned three kernels is inputted into the support vector machine classifier to generate a classification map. The experimental procedure in this article uses LBP to extract the information in superpixels, which effectively prevents the loss of edge features in superpixels. The experimental results show that the proposed method is superior to the state-of-the-art classifiers for HSI classification.

Index Terms—Hyperspectral image (HSI), local binary mode (LBP), multiple kernels (MK), superpixel, support vector machine (SVM).

I. INTRODUCTION

THE development of remote sensing technology has gone through the stages of panchromatic (black and white), color, and multispectral scanning imaging. The emergence of spectral imaging concept in the early 1980s led spectral remote sensing into a new stage—the hyperspectral remote sensing stage [1]. The emergence of hyperspectral remote sensing is a revolution in the field of remote sensing. Hyperspectral remote sensing can detect substances in wideband remote sensing. Hyperspectral image (HSI) can obtain rich spectral information from hundreds of identical narrow band spectral channels, which

supports the fine identification of various land-cover materials. Thus, HSI has drawn more and more attention and it has been widely utilized for classification [2], [3], target detection [4], [5], anomaly detection [6], [7], and spectral unmixing [8], [9]. The HSI classification task plays a substantial role in geological exploration [10], [11], crop detection [12], [13], national defense [14], [15], and other fields, it indicates that further research of this technique is warranted.

While HSI classification is widely used, it also faces considerable challenges, one of which is the Hughes phenomenon [16]. The Hughes phenomenon means that, with the increase of the number of operational bands, the classification accuracy will “increase at first and then decrease” [17] during the hyperspectral analysis. In HSI processing, the key step of solving this issue is feature extraction. In addition, because of the complexity and variability of spectral features, feature extraction in HSI is one of the most challenging tasks [18].

During the last decades, various spatial-spectral feature extraction methods have been proposed to address the classification problem of HSI. The initial method is the case of stacking method [19], i.e., the spectral features and spatial features are directly stacked in a vector to achieve the joint utilization of spatial and spectral information. Although, this method uses both spatial and spectral information, it directly increases the number of extracted feature dimensions. At the same time, simply stacking different types of features together is not conducive to the use of their respective advantages and cannot effectively achieve the joint use of different features.

The Kernel-based methods have gained general acceptance, and these methods [20], [21] use a simple linear weighting scheme to achieve joint learning and the utilization of spatial and spectral information. In order to complete the classification task, the kernel-based support vector machine (SVM) methods have been proven to effectively alleviate the well-known Hughes phenomenon. In addition, it has become an outstanding framework and core direction in the research of HSI classification. SVM not only overcomes the disadvantages of traditional statistical-based learning methods [22], [23], but also abandons the requirement for large numbers of training samples. Moreover, the new kernel functions, such as spectral weighted kernel, composite kernel (CK), spatial-spectral kernel (SSK), can flexibly utilize the spectral and spatial information by fusing different heterogeneous features in HSI, thereby improving the classification performance. Among them, the CKs [24], [25] and SSKs [26], [27] are the most commonly used kernels. In [24], multisource spectral and spatial information are effectively used

Manuscript received February 2, 2020; revised June 2, 2020, July 9, 2020, and July 28, 2020; accepted August 1, 2020. Date of publication August 6, 2020; date of current version August 24, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61602423 and Grant 61605175, in part by the Henan Province Science and Technology Breakthrough Project under Grant 182102210611, and in part by the Open Fund of Key Laboratory of Urban Land Resources Monitoring and Simulation Ministry of Land and Resources. (Corresponding author: Wei Huang.)

Wei Huang, Yao Huang, Hua Wang, and Yan Liu are with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450000, China (e-mail: hnhw235@163.com; 1924470370@qq.com; whuwanghua@163.com; lyanzju@163.com).

Hiuk Jae Shim is with the College of Electronic and Electrical Engineering, Sungkyunkwan University, Suwon 440746, South Korea, and also with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: waitnual@nuist.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2020.3014492

by applying CKs in SVM. G. Camps-Valls [25] captured spatial information from the square neighborhood of a target pixel and used the characteristics of the Mercer kernel to build a CK that contains the combined spatial and spectral information. Compared with the spectral-weighted kernel using only spectral information [28], CK took full advantage of HSI characteristics. Different from the way that the CK extracted spatial and spectral information separately, the SSK extracted spatial and spectral information directly from the spatial-spectral structure. In [26], a framework for a SSK-based sparse representation classification (KSRC) method was used to measure the similarity of kernel space and to extract spectral and spatial features directly from the spatial-spectral structure. However, SSK-based methods were time-consuming by directly computing the cluster feature in kernel space, because these kernels were obtained by pixel-wise calculation in the neighborhood.

In recent years, an increasing number of studies have focused on the use of spatial area information of HSI. In [29], the application of extended morphological profiles with partial reconstruction to high-resolution HSI of urban areas to mine spatial information could effectively improve classification accuracy. Tarabalka *et al.* [30] used the method of partition clustering to segment the spatial region. In [31], the spatial information could be sparsely represented by a linear combination of several training samples in a structured dictionary [32], [33]. The HSI classification based on edge retention filtering [34] and Gabor filter [35]–[37] has been used to extract spatial features. Although, the abovementioned methods may lead to good classification precision, the size and shape of the spatial area that used is fixed, resulting in the inability to make full use of the spatial texture of HSI.

Entropy rate superpixel segmentation (ERS) [38] can adaptively change the shape of the region according to the spatial structure of HSI. This method uses the similarity of features between pixels to group pixels and uses a small number of superpixels to replace a large number of pixels to express image features, which greatly reduces the complexity of image post-processing. In [39], [40], superpixels were regarded as a local neighborhood to obtain spatial information, thereby avoiding the choice of the best spatial neighborhood. In [41], multiple kernels (MK) were applied to effectively utilize the spectral-spatial information of the superpixel. Compared with [39], [40], it not only utilized the spatial information within the superpixel but also utilized the spatial information among superpixels, resulting in higher classification accuracy. Due to the powerful feature extraction performance of multiscale information [42], [43], the classification of multiscale superpixels for HSI is proposed in [44], [45], which avoids the selection of the optimal superpixels. Cui *et al.* [46] proposed a superpixel-based extended random walker classification method for HSI, which uses the principle of probability maximization to refine each superpixel label. Wang *et al.* [47] used superpixels for resolution enhancement and achieved good results. Sellars *et al.* [48] extracted spatial-spectral features from the superpixel region, and combined them with a semisupervised classifier to effectively improve the classification accuracy. Therefore, using superpixel segmentation in the processing of HSI can achieve good classification results.

However, the superpixel-based approach also has a drawback: the spatial information within the superpixel is represented by the mean value of the pixels, which leads to loss of the edge information of pixels within the superpixel.

In this article, we propose a classification method of local binary patterns and superpixel-based multiple kernels (SMK) for HSI. First, we use a superpixel segmentation algorithm to segment the image that is reduced by principal component analysis (PCA) [49] into a HSI with a superpixel index. Then, we use weighted average filtering and local binary patterns (LBP) to acquire the spatial characteristics within and among the superpixels, and the obtained spatial kernels among the superpixels, the spatial texture kernels within the superpixel, and the directly extracted spectral kernels were fused together. Finally, this combination kernel is input into the SVM classifier to generate a classification result map. The proposed method combines LBP with superpixels to effectively solve the problem of inaccurate edge information extraction within superpixels, thereby improving the classification accuracy.

The remainder of this article is organized as follows: Section II presents the superpixel segmentation algorithm, the SVM and the kernel function; Section III introduces the proposed method (a SMK method); Section IV presents the results of three famous experimental datasets; and Section V provides the conclusion of this article.

II. RELATED WORK

A. Kernel Function-Based SVM

SVM is a binary classification model, which is a linear classifier with the largest spacing defined on the feature space. However, true hyperspectral pixels are linearly inseparable, so SVM is often combined with kernel functions to solve the problem of linear inseparability. Kernel functions can map pixels to high-dimensional feature spaces.

Suppose that there is a set of labeled datasets $\{(x_1, y_1), \dots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^{B \times 1}$ and $y_i \in [-1, 1]$ ($i \in \{1, \dots, n\}$). A nonlinear mapping $\Phi: \mathbb{R}^{B \times 1} \rightarrow H$, which usually maps the data to a high-dimensional Hilbert space. SVM mainly solves the following dual problems:

$$\max \left\{ \sum_{i=0}^N \alpha_i - \frac{1}{2} \sum_{i=0}^N \sum_{j=0}^N \alpha_i \alpha_j y_i y_j \langle \varphi(x_i), \varphi(x_j) \rangle \right\} \quad (1)$$

where α_i and α_j are Lagrangian multipliers, $\varphi(\bullet)$ represents a mapping function, which converts pixels to a high-dimensional feature space. It is difficult to directly calculate the inner product $\langle \varphi(x_i), \varphi(x_j) \rangle$, so a kernel function is introduced to represent $\langle \varphi(x_i), \varphi(x_j) \rangle$

$$K(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle \quad (2)$$

The kernel function maps the samples of the original space into the feature space and performs the inner product operation, but not any function can be used as the kernel function. Usually, the kernel function is required to satisfy continuity, symmetry, and semipositive definiteness, which is Mercer's theorem [50];

the commonly used kernel functions are linear kernel, polynomial kernel, Sigmoid kernel, Gaussian radial basis kernel (RBF), and wavelet kernel. This article uses the RBF function, and its calculation formula is as follows:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (3)$$

If the spectral pixels are input into the RBF kernel in (3), the spectral kernel function can be obtained

$$K_{\text{Spec}}(x_i^{\text{Spec}}, x_j^{\text{Spec}}) = \exp\left(-\frac{\|x_i^{\text{Spec}} - x_j^{\text{Spec}}\|^2}{2\sigma^2}\right) \quad (4)$$

where x_i^{Spec} represents the i th original spectral pixel, and $K_{\text{Spec}}(x_i^{\text{Spec}}, x_j^{\text{Spec}})$ represents the spectral kernel.

Similarly, if x_i^{Spat} that represents the spectral mean or variance of the pixels in the neighborhood of the space of the i th pixel is input into (3), and the spatial kernel function $K_{\text{Spat}}(x_i^{\text{Spat}}, x_j^{\text{Spat}})$ can be obtained. The spatial-spectral CK function is defined as follows:

$$K_{\text{CK}}(x_i, x_j) = \omega_1 K_{\text{Spec}}(x_i^{\text{Spec}}, x_j^{\text{Spec}}) + \omega_2 K_{\text{Spat}}(x_i^{\text{Spat}}, x_j^{\text{Spat}}) \quad (5)$$

where $\omega_1 + \omega_2 = 1$, ω_1 represents the weight of the spectral kernel, and ω_2 represents the weight of the spatial kernel.

MK is an improvement based on CK. Unlike CK, MK is more flexible in the combination of different feature kernel functions. In this article, we use different techniques (i.e., LBP feature and weighted average filtering) to obtain the spatial features. After inputting these spatial features into the RBF kernel function, different spatial kernels are obtained. Then, these spatial kernels and spectral kernel are combined by a certain weight to obtain MK. It can be constructed as follows:

$$K_{\text{MK}}(x_i, x_j) = \omega_1 K_{\text{Spec}}(x_i^{\text{Spec}}, x_j^{\text{Spec}}) + \sum_{n=2}^N \omega_n K_{\text{Spat}_{n-1}}(x_i^{\text{Spat}_{n-1}}, x_j^{\text{Spat}_{n-1}}) \quad (6)$$

s.t. $\sum_{n=1}^N \omega_n = 1$

where $\omega_1, \omega_2, \dots, \omega_n$ are the balance parameters, which are determined by experimental verification in experiment section. N represents the number of kernel functions, including one spectral kernel function and $N - 1$ spatial kernel functions.

B. Superpixel Segmentation Algorithm

In recent years, superpixel segmentation algorithms have been widely used in the processing of HSI. These algorithms can segment HSI into many superpixels. A superpixel is a small area composed of a series of adjacent pixels with similar characteristics such as color, brightness, and texture. By abstracting basic information, a superpixel converts a pixel-level image into a region-level image. However, because the number of HSI bands is too large, which cannot be directly used for superpixel

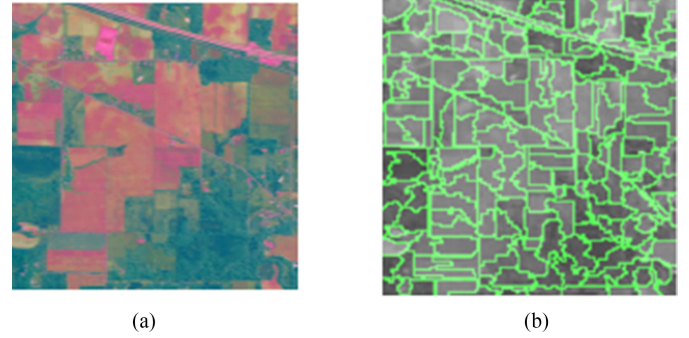


Fig. 1. False-color images. (a) Principal component extraction. (b) Superpixel segmentation map.

segmentation, so dimension reduction of HSI is required. There are many methods available for HSI dimension reduction. In this article, we choose PCA to reduce the dimension of the original HSI. PCA is a technique used for analyzing and simplifying datasets. In this method, linear projection is used to project data into a new coordinate space in which the first principal component contains the largest amount of information, followed by the second principal component and the other principal components, decreasing successively, and each principal component is independent of the others. It can be concluded that PCA [51] maximizes the retention of data information while reducing the data dimension. In this article, the first three principal components [as shown in Fig. 1(a)] are selected to segment the superpixel, and the superpixel segmentation map [as shown in Fig. 1(b)] is generated. The superpixel segmentation map is then combined with the original HSI to generate the HSI with superpixel index.

The most important aspect of superpixel segmentation is choosing the number of superpixels. The number of superpixels, L , is selected according to the complexity of the texture. Its calculation formula is as follows:

$$L = L_{\text{base}} \times R_{\text{texture}} \quad (7)$$

where L_{base} is the number of base superpixels, R_{texture} indicates that the texture ratio is mainly used to reflect the complexity of the texture in HSI, which is calculated as follows:

$$R_{\text{texture}} = \frac{t}{T} \quad (8)$$

where t denotes the number of nonzero elements in the filtered image (The image obtained after inputting the first three principle components to the Sobel filter [52]), and T denotes the total number of pixels in the first three principle components. Sobel filtering is a simple texture detector for detecting the texture structure of HSI. It is not very accurate for edge localization, so the extracted image contours are sometimes unsatisfactory. The third part of this article will introduce another texture detection method in detail, it will be combined with superpixels to fully mine the texture information of HSI.

Given the number of superpixels L , an ERS segmentation algorithm is used to generate a 2-D superpixel map on the first three PCs. ERS is a graph-based clustering algorithm, which can construct a graph $G = (V, E)$ on the basis image, where V

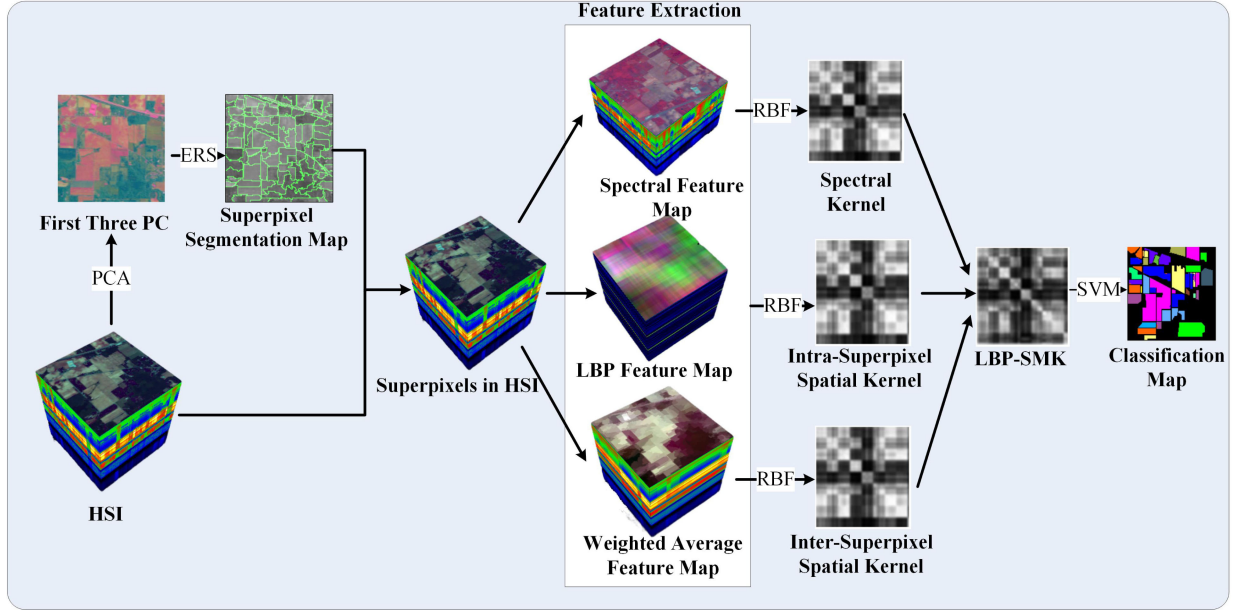


Fig. 2. Schematic diagram of the LBP-SMK method.

is the vertex set corresponding to the pixels of the basis image, and E is the edge set corresponding to the similarity among adjacent pixels. The goal of ERS is to find a subset of edges A (i.e., $A \subseteq E$) to make the segmentation map $G = (V, A)$ contain exactly L discrete subgraphs (each subgraph denotes a superpixel). The objective function of superpixel segmentation is defined as follows:

$$\max_A \{H(A) + \lambda B(A), \text{s.t. } A \subseteq E\} \quad (9)$$

where $H(A)$ is a randomly changing entropy rate, $B(A)$ is a balance term, it is mainly used to promote the clusters to have similar sizes and reduce the number of unbalanced superpixels, A denotes edge set, λ ($\lambda \geq 0$) is a weight of the balance term. The optimization problem of (9) can be solved by the greedy algorithm.

III. PROPOSED METHOD

This article proposes a local binary pattern and superpixel-based multiple kernel (LBP-SMK) method for HSI classification. The schematic diagram is shown in Fig. 2. It is divided into three main parts: first, generating a HSI with a superpixel index; second, using the HSI with a superpixel index to obtain three kernel functions; and third, fusing the obtained three kernel functions together for input into the SVM classifier to generate the classification result map.

A. Extraction of Spectral Features Based on Superpixels

Since each superpixel is composed of a set of adjacent spectral pixels, the spectral information can be directly composed of the spectral pixels in the superpixel. All the spectral pixels that are in the HSI make up the spectral characteristics of the HSI. Let

the input spectral pixels be $(x_1^{\text{Spec}}, \dots, x_n^{\text{Spec}})$. According to (4), the spectral kernel can be calculated.

B. Extraction of LBP Features Within Superpixels

As described in the second part of this article, the number of superpixels is partially determined by Sobel filtering, which is a simple texture detector, and it first roughly detects HSI texture features and then uses them for superpixel segmentation. Generally speaking, a superpixel seldom contains multiple objects, but special cases will occur. LBP can be used to improve the extraction performance of spatial texture features within superpixel. However, since each superpixel is an irregular region, LBP cannot directly be used to extract spatial texture features within it. Therefore, in order to obtain the spatial texture information within a superpixel, first, we utilize a circular LBP operator for each PCA component of HSI, to obtain a local texture feature image. Second, the pixel indexes of a superpixel on the local texture feature map are used to obtain LBP statistical histogram for this superpixel. Moreover, all the statistical histograms are concatenated to obtain spatial texture features. Finally, we use the obtained spatial texture features as internal features of a superpixel.

The traditional LBP [53] method uses thresholds to mark the differences among the central pixel and its neighboring pixels so as to analyze the local texture structure. The algorithm is gradual and unaffected by changes in lighting conditions. In order to obtain the local area of HSI, PCA is first used to reduce the dimension of the original HSI. Moreover, the first three PCs are retained, one of the principal components is divided into many image patches, and the histogram of each image patch is extracted using LBP, and then the extracted histograms are connected together to form the eigenvectors of principal components. Finally, the same operation is performed

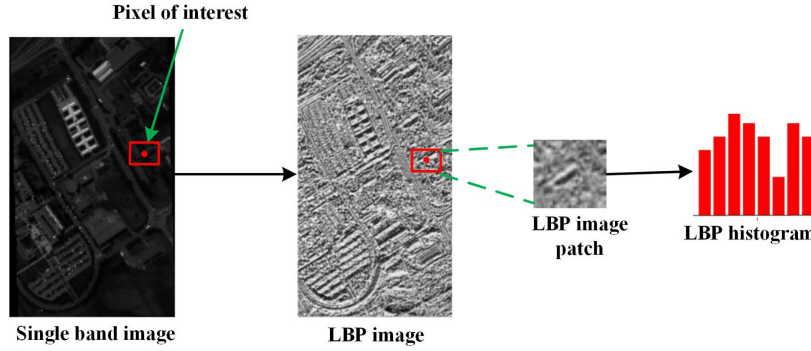


Fig. 3. Feature extraction of LBP.

on each principal component, and all bands of the obtained LBP histogram are connected in series to form a spatial feature vector. For an image patch, the center pixel is located at the center of the circle, and adjacent pixels of the center pixel are distributed on the circumference instead of rectangular. We compare these adjacent pixels with the center pixel. If the gray value of the pixel is greater than the gray value of the center pixel, the pixel is marked as 1. If the gray value of the pixel is less than the gray value of the center pixel, the pixel is marked as 0. For example, by comparing eight surrounding pixels in the neighborhood with the central pixel, a set of 8-b binary numbers can be generated (generally they will be converted to decimal numbers, and there are a total of 256 combinations). The expression is as follows:

$$\text{LBP}_{(c,P)} = \sum_{i=0}^{P-1} s(t_i - t_c) 2^i \quad (10)$$

where c denotes the center pixel, P denotes the number of surrounding pixels, t_i denotes the gray value of adjacent pixels, t_c denotes the gray value of the center pixel, and $s(\bullet)$ is the sign function defined as follows:

$$s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (11)$$

The direction and smoothness of the texture of the local area are reflected by the output of the LBP code. After obtaining the LBP codes of all pixels, a histogram is calculated for the local area centered on the pixels of interest (Fig. 3).

However, for the LBP operator with P sampling points in a circular area of radius R , 2^P patterns are generated, and as the number of pixels in the field set increases, the LBP pattern increases rapidly, which greatly increases the difficulty of classification.

Most binary numbers are transitions from 1 to 0 or from 0 to 1, and the number of transitions does not exceed 2, so the LBP equivalent mode uniform LBP (ULBP) is used to reduce the feature vector [54], which is defined as follows:

$$\text{ULBP}_{(c,P)} = \begin{cases} \sum_{i=0}^{P-1} s(t_i - t_c) 2^i & \text{U(LBP}_{(c,P)}) \leq 2 \\ P + 1 & \text{otherwise} \end{cases} \quad (12)$$

where, $\text{U(LBP}_{(c,P)})$ represents the number of transitions of the cyclic binary number corresponding to LBP from 1 to 0 or from 0 to 1. The number of LBP statistical histograms is reduced from

2^P to $P \times (P - 1) + 3$. In this article, all the LBP statistical histograms can be seen as the spatial texture features of HSI.

C. Extraction of Spatial Features Among Superpixels

In order to increase the weight of pixels of interest, a new weighted average filtering method is proposed for anomaly detection [55]. This method can assign different weights to different pixels, e.g., increase the weight of the background samples and reduce the weight of the anomalous pixels or noise signals, and achieve the better results. Inspired by this, the weighted average filtering method is used to extract spatial features among superpixels. Let x_i^{wao} be the weighted average pixel belonging to superpixel X_i , $\{X_{i1}, X_{i2}, \dots, X_{iJ}\}$ be the neighboring superpixel of superpixel X_i , and x_{ij}^{Mean} be the average of all pixels belonging to superpixel X_{ij} . x_i^{wao} can then be calculated in the following:

$$x_i^{\text{wao}} = \sum_{j=0}^J \omega_{ij} x_{ij}^{\text{Mean}} \quad (13)$$

where ω_{ij} is the weight of x_{ij}^{Mean} , and ω_{ij} can be calculated as the following:

$$\omega_{ij} = \frac{\exp(-\|x_{ij}^{\text{Mean}} - x_i\|_2^2 / h)}{\sum_j \exp(-\|x_{ij}^{\text{Mean}} - x_i\|_2^2 / h)}, j = 1, 2, \dots, J \quad (14)$$

where h is a predefined scalar.

All the pixels in the superpixel X_i are replaced by the calculated x_i^{wao} , and the same operation is performed in each superpixel. Finally, all superpixels obtained by the combination are combined to obtain a spatial feature based on the superpixels.

D. LBP-SMK

The proposed LBP-SMK method in this article uses three kernel functions based on the spectral kernel, the spatial kernel within the superpixel, and the spatial kernel among the superpixels. The kernel function in the training phase can be obtained from (6), which is expressed as follows:

$$K_{\text{SMK}}^{\text{train}}(x_i, x_j) = \omega_{\text{Spec}} K_{\text{Spec}}(x_i^{\text{Spec}}, x_j^{\text{Spec}}) + \omega_{\text{Intras}} \bullet K_{\text{Spat1}}^{\text{Intras}}(x_i^{\text{Spat1}}, x_j^{\text{Spat1}}) + \omega_{\text{Inters}} K_{\text{Spat2}}^{\text{Inters}}(x_i^{\text{Spat2}}, x_j^{\text{Spat2}}) \quad (15)$$

Algorithm 1: LBP-SMK for Classification of HSI.

- 1: **Input:** The original HSI R , the base superpixel number L_{base} ;
- 2: **Step 1:** PCA was used to reduce the dimension of R , and the first three PCs were selected for superpixel segmentation to obtain the superpixel segmentation image P ;
- 3: **Step 2:** Combine P with R to generate a HSI I with superpixel index;
- 4: **Step 3:** Do the following for I :
- 5: 1) Combine all the spectral pixels in I from a spectral feature image I_{Spec} ;
- 6: 2) Weighted average filtering on I generates weighted average feature image I_{WAF} ;
- 7: 3) LBP feature extraction is performed on the first three principal components after dimensionality reduction of I , and the extracted spatial features are concatenated to form a local binary feature image I_{LBP} .
- 8: **Step 4:** A set of training data is randomly selected in R , and the corresponding pixels are extracted in I_{Spec} , I_{WAF} and I_{LBP} to generate the corresponding three sets of training data according to the position of the training data;
- 9: **Step 5:** The three sets of training data in Step 4 are input into (3) to generate three kernel functions: Spectral kernel $K_{Spec}(x_i^{Spec}, x_j^{Spec})$ spatial kernel $K_{Spat1}^{Intras}(x_i^{Spat1}, x_j^{Spat1})$ among superpixels, and spatial kernel $K_{Spat2}^{Inters}(x_i^{Spat2}, x_j^{Spat2})$ within superpixels;
- 10: **Step 6:** The above three kernel functions are fused by (15) and input to the SVM classifier.
- 11: **Output:** Classification result map.

where $\omega_{Spec} + \omega_{Intra} + \omega_{Inters} = 1$, $K_{Spec}(x_i^{Spec}, x_j^{Spec})$ represents the spectral kernel, $K_{Spat1}^{Intras}(x_i^{Spat1}, x_j^{Spat1})$ is based on the spatial kernel among superpixels, and $K_{Spat2}^{Inters}(x_i^{Spat2}, x_j^{Spat2})$ is based on the spatial kernel within superpixels. All of these terms can be calculated by (3), and the kernel function in the test phase can be obtained in the same way.

The solution steps of the proposed LBP-SMK method are shown in Algorithm 1.



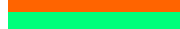











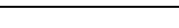

IV. EXPERIMENT

To verify the rationality of this algorithm, the Indian Pines, Pavia University, and Pavia Center datasets were used to verify it in the MATLAB2019 environment and compared the results with several state-of-the-art methods.

A. Dataset Introduction

1) *Indian Pines Dataset:* In 1992, the United States acquired the Indian Pines dataset by imaging Indian pine trees in Indiana. An airborne visible infrared imaging spectrometer with an

TABLE I
CLASS TYPES AND STATISTICS OF INDIAN PINES DATASET

Class name	Corresponding color	Number of test samples	Number of training samples
Alfalfa		5	49
Corn-no till		143	1291
Corn-min till		83	751
Corn		24	210
Grass/pasture		48	449
Grass/tree		73	674
Grass/pasture-mowed		3	23
Hay-windrowed		48	441
Oats		2	18
Soybeans-no till		97	871
Soybeans-min till		246	2222
Soybeans-clean till		59	555
Wheat		21	191
Woods		127	1167
Bldg-grass-tree-drives		39	341
Stone-steel towers		9	86
Total		1027	9339

imaging wavelength range of 0.4–2.5 μm was used to image ground features in continuous 220 bands. Because the 104–108, 150–163, and 220 bands cannot be reflected by water, in this experiment we removed these 20 bands and used the remaining 200 bands as the research object. The Indian Pines dataset contains 145×145 pixels and 16 types of ground objects. There are only 10 366 pixels in the dataset that contain ground objects, and the rest are background pixels. After removing the background pixels, we took 10% of the samples in each class as training samples, leaving 90% as test samples. Table I displays the feature categories and statistics of the Indian Pines dataset.

2) *Pavia University Dataset:* The dataset of Pavia University was acquired by Germany in 2003 in Pavia, Italy. An airborne reflection spectrometer was used to continuously image 115 bands with a wavelength range of 0.43–0.86 μm . Because 12 of these bands are affected by noise, we excluded the 12 bands and used the remaining 103 spectral bands. The Pavia University dataset contains a total of nine types of features. The size of the data is 610×340 , and it contains a total of $610 \times 340 = 220\ 7400$ pixels. However, only 42 776 pixels contain ground features in the dataset. The rest are all background pixels. After removing the background pixels, we choose 10% of the samples in each class to train, and the remaining samples were used for testing. Table II shows the features and statistics of the Pavia University dataset.

3) *Pavia Center Dataset:* Same as Pavia University dataset, the Pavia Center dataset is also obtained by the ROSIS sensor. It contains nine types of land cover, 102 spectral bands and 1096×715 pixels. But it contains only 7456 land cover pixels, and the rest are background pixels. After excluding background pixels, 10% of the samples in each class are selected for training, and the rest for testing. Table III displays the class types and statistics of the Pavia Center.

B. Parameter Settings

The LBP-SMK method proposed in this article is compared with several state-of-the-art methods, such as CKs for HSI

TABLE II
CLASS TYPES AND STATISTICS OF PAVIA UNIVERSITY DATASET



















Class name	Corresponding color	Number of test samples	Number of training samples
Asphalt		66	6565
Meadows		186	18463
Gravel		21	2078
Trees		31	3033
Metal sheets		13	1332
Bare soil		50	4979
Bitumen		13	1317
Bricks		37	3645
Shadows		9	938
Total		426	42350

TABLE III
CLASS TYPES AND STATISTICS OF PAVIA CENTER DATASET

Class name	Corresponding color	Number of test samples	Number of training samples
Water		82	742
Trees		82	738
Asphalt		81	735
Self-Blocking Bricks		80	728
Bitumen		80	728
Tiles		126	1134
Shadows		47	429
Meadows		82	742
Bare Soil		82	738
Total		742	6714

classification (SVMCK) [25], spectral-spatial HSI classification method using a spectral data regularization and a spatially adaptive TV constraint (SpATV) [2], HSI classification by exploiting spectral-spatial information of superpixel via multiple kernels (SC-MK) [40], region-based relaxed multiple kernel collaborative representation for HSI classification (RMK) [56], adjacent superpixel-based multiscale spatial-spectral kernel for hyperspectral classification (ASMGSSK) [44], local binary patterns and extreme learning machine for HSI classification (LBPELM) [57] and the deep learning method of random patch network (RPNNet) [58]. Refer to the literature [40], [44], [57] for specific parameter adjustment strategies, the method of autonomously selecting parameters within the threshold range is used to select the best parameters. The parameters used in the comparison experiments are their optimal values. The experiments were run on three real datasets, i.e., the Indian Pines dataset, Pavia University dataset, and Pavia Center dataset. The overall accuracy value (OA) of the experiment was the average value obtained after ten times of randomly selected samples.

1) *Effect of Sigma Parameter of RBF Kernel Function*: This article uses the RBF kernel function. The sigma parameter of RBF kernel function is set on the three datasets as shown in Fig. 4. According to experience, we set the sigma parameter from 2^{-10} to 2^{-1} . From Fig. 4, we can see that on the Indian Pines dataset, the classification accuracy is the best when the kernel bandwidth is 2^{-6} . On the University of Pavia dataset, as the sigma parameter changes, the classification accuracy increases gradually, and then decreases slightly. The best kernel bandwidth

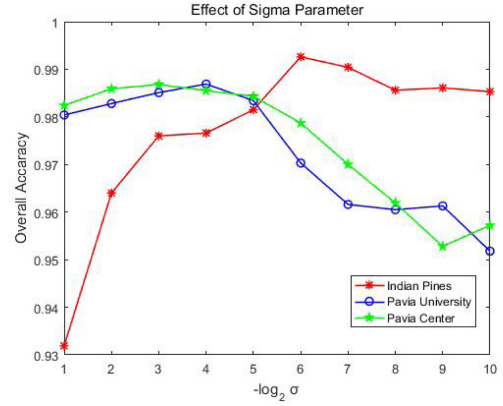


Fig. 4. Effect of σ parameter.

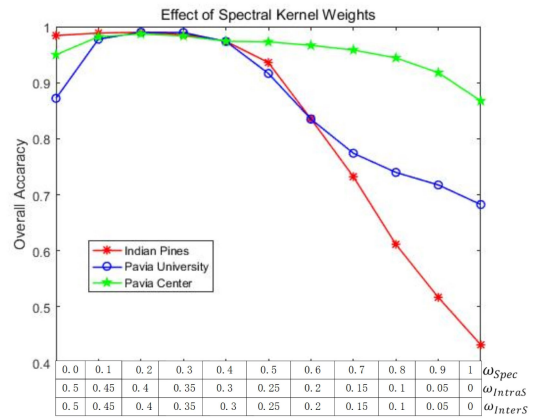


Fig. 5. Effect of spectral kernel weights.

on the University of Pavia dataset is 2^{-4} . On the Pavia Center dataset, the optimal kernel bandwidth is 2^{-3} .

2) *Effect of Kernel Weights*: This article uses three kernel functions. To verify the effect of different kernel functions on the experimental results, we set the spectral kernel weights from 0 to 1 and set the kernel weights within the superpixel and the kernel weights among the superpixels to the same value. The experimental results as shown in Fig. 5, where we can see that when the spectral kernel weight is 1, only the spectral information is used, and the classification result is very unsatisfactory. When the spectral weight is increased to 0.2, the classification accuracy is the highest on the three data sets, but as the spectral weight continues to increase, the classification accuracy shows a downward trend, indicating that the spatial kernel weights within and among superpixels need to account for a larger proportion. Therefore, the spectral kernel weight factor was fixed at 0.2 to observe the influence of the spatial kernel weight factor within and among superpixels in Fig. 6.

From Fig. 6, we can observe that the kernel weights within superpixel are between 0 and 0.5, and the classification accuracy increases. During the increase from 0.5 to 0.8, the classification accuracy decreases slightly, indicating the kernel weights of the superpixels need to account for a relatively large proportion, but larger is not better.

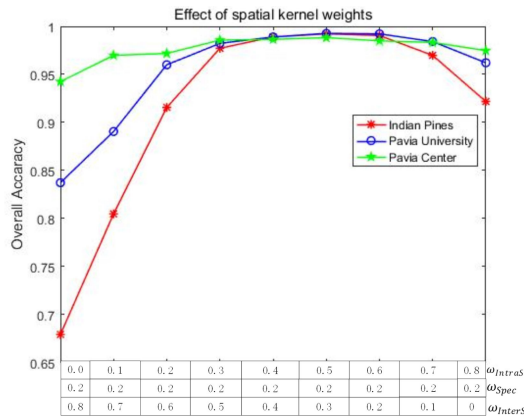


Fig. 6. Effect of spatial kernel weights.

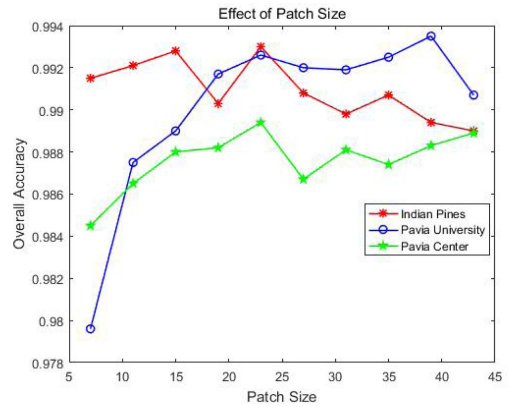


Fig. 8. Effect of patch size.

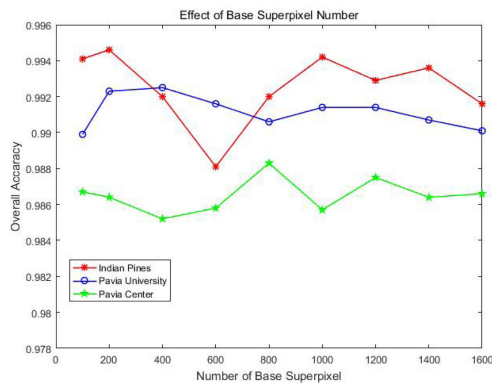


Fig. 7. Effect of base superpixel number.

The abovementioned experimental results show that spectral information, spatial information within superpixel, and spatial information among superpixels are all beneficial to HSI classification.

3) *Effect of the Base Superpixel Number:* To verify the impact of the base superpixel number on the classification results, we set the local areas ω of the LBP feature extraction of the Indian Pines dataset, Pavia University dataset, and Pavia Center dataset to 23×23 pixels, 19×19 pixels, and 39×39 pixels, respectively. Next, we changed the base superpixel number to see how it affects the classification results. In the experiment, we set the base superpixel number to 100, 200, 400, 600, 800, 1000, 1200, 1400, and 1600. The experimental results are shown in Fig. 7.

As can be seen from Fig. 7, with the increase of base superpixel number, the OA rate also increased. However, with the continuous increase of base superpixel number, OA did not continue to increase, and fluctuations occurred. Therefore, we can come to the conclusion that an increased base superpixel number is not always better. Choosing appropriate base superpixel values can lead to good classification results. It can also be obtained from Fig. 7 that for the Indian Pines dataset, the Pavia University dataset, and the Pavia Center dataset, the proposed method achieves the best classification accuracy when the base superpixel number are set to 200, 400, and 800, respectively.

4) *Effect of the Patch Size:* In this article, LBP is used to extract the spatial texture features of HSI images. In order to verify the effect of different patch sizes on the classification accuracy of HSI, we conducted experiments on three datasets. The patch size selection is shown in Fig. 8.

As can be seen from Fig. 8, the Pavia University dataset is sensitive to the transformation of patch size, and the overall classification accuracy fluctuates greatly. When the patch size is less than 39×39 pixels, the overall classification accuracy increases with an increase in patch size. When the patch size increases to 39×39 pixels, the overall classification accuracy was the best. On the Indian Pines dataset and the Pavia Center dataset, as the patch size continues to increase, the OA of the classification appears to float up and down, and the best classification accuracy is obtained when the patch size is 23×23 pixels.

In this article, for the Indian Pines dataset, the superpixel is set to 200 and the patch size is set to 23×23 pixels. For the Pavia University dataset, the superpixel is set to 400, and the patch size is set to 39×39 pixels. For the Pavia Center dataset, the base superpixel is set to 800, the patch size is set to 23×23 pixels, and the kernel bandwidth is set to 2^{-3} . The spectral weight factor of both datasets is set at 0.2, the kernel weight factor within the superpixel is set at 0.5, and the kernel weight among the superpixels is set to 0.3. The experimental results are shown in the next section.

C. Experimental Results and Analysis

In this article, the proposed LBP-SMK method is compared with other HSI classification methods, i.e., SVMCK [25], SpATV [2], LBPELM [57], SC-MK [40], RMK [56], RP-Net [58], and ASMGSSK [44]. Tables IV, V, and VI show the comparisons of classification accuracy, whereas Figs. 9, 10, and 11 show comparisons of the classification results by different methods.

In the experiments, we choose the overall classification accuracy (OA), kappa coefficient, and their standard deviations as the evaluation indicators. It can be seen from Tables IV, V, and VI that RMK and SC-MK use MK to explore spatial-spectral

TABLE IV
CLASSIFICATION RESULTS FOR THE INDIAN PINES DATASET (NUMBERS IN BOLD REPRESENT THE BEST CLASSIFICATION PERFORMANCE)

Class name	SVMCK	SpATV	LBPELM	SC-MK	RMK	RPNet	ASMGSSK	LBP-SMK
Alfalfa	0.5286	0.8510	0.9854	0.9610	0.9551	0.8049	0.9878	0.9756
Corn-no till	0.9352	0.9709	0.9703	0.9416	0.9660	0.9574	0.9755	0.9865
Corn-min till	0.9405	0.9839	0.9656	0.9734	0.9731	0.9620	0.9908	0.9969
Corn	0.8433	0.9948	0.9689	0.9704	0.9710	0.8798	0.9624	0.9948
Grass/pasture	0.9508	0.9581	0.9833	0.9667	0.9644	0.9476	0.9844	0.9837
Grass/tree	0.9812	0.9849	0.9807	0.9982	0.9898	0.9907	0.9911	0.9998
Grass/pasture-mowed	0.1696	0.5000	0.9364	0.9640	0.9652	0.9000	0.9652	0.9640
Hay-windrowed	0.9907	1.0000	0.9950	0.9833	0.9937	0.9942	0.9977	1.0000
Oats	0.0667	0.0556	0.9280	0.9111	0.9833	0.5944	0.9056	1.0000
Soybeans-no till	0.8939	0.9437	0.9227	0.9410	0.9662	0.9483	0.9777	0.9859
Soybeans-min till	0.9612	0.9904	0.9891	0.9698	0.9958	0.9833	0.9914	0.9999
Soybeans-clean till	0.9218	0.9724	0.9831	0.9453	0.9668	0.9444	0.9814	0.9919
Wheat	0.9901	0.9953	0.9901	0.9946	0.9963	0.9908	0.9901	0.9946
Woods	0.9744	0.9944	0.9940	0.9938	0.9968	0.9945	0.9938	0.9999
Bldg-grass-tree-drives	0.8185	0.9845	0.9952	0.9467	0.9912	0.9611	0.9909	1.0000
Stone-steel towers	0.8930	0.7826	0.9253	0.9714	0.9860	0.7893	0.9419	0.9774
mean_OA	0.9375	0.9751	0.9816	0.9670	0.9823	0.9665	0.9861	0.9946
STD_OA	0.0026	0.0069	0.0045	0.0052	0.0030	0.0032	0.0018	0.0017
mean_kappa	0.9286	0.8726	0.9789	0.9623	0.9799	0.9617	0.9842	0.9939
STD_kappa	0.0030	0.0135	0.0032	0.0060	0.0034	0.0037	0.0020	0.0019

TABLE V
CLASSIFICATION RESULTS FOR THE PAVIA UNIVERSITY DATASET (NUMBERS IN BOLD REPRESENT THE BEST CLASSIFICATION PERFORMANCE)

Class name	SVMCK	SpATV	LBPELM	SC-MK	RMK	RPNet	ASMGSSK	LBP-SMK
Asphalt	0.9414	0.9795	0.8815	0.8555	0.9963	0.9446	0.9937	0.9947
Meadows	0.9816	0.9969	0.9605	0.9374	0.9996	0.9948	0.9985	0.9959
Gravel	0.7129	0.7768	0.9343	0.9214	0.9712	0.8773	0.9952	0.9976
Trees	0.8971	0.8863	0.7643	0.9578	0.9014	0.9602	0.9087	0.9611
Metal sheets	0.9899	0.9998	0.8869	0.9963	0.9917	0.9664	0.9878	0.9999
Bare soil	0.8472	0.8409	0.9785	0.9662	0.9995	0.9254	0.9999	0.9973
Bitumen	0.8120	0.7471	0.9537	0.9794	0.9878	0.8087	0.9911	0.9969
Bricks	0.8814	0.8984	0.8988	0.9239	0.9689	0.9471	0.9827	0.9951
Shadows	0.9658	0.8401	0.6929	0.9734	0.5594	0.8439	0.8163	0.9956
mean_OA	0.9264	0.9191	0.9263	0.9315	0.9776	0.9565	0.9854	0.9935
STD_OA	0.0054	0.0095	0.0056	0.0169	0.0040	0.0102	0.0036	0.0017
mean_kappa	0.9017	0.7927	0.9047	0.9102	0.9702	0.9419	0.9805	0.9914
STD_kappa	0.0073	0.0112	0.0088	0.0212	0.0054	0.0138	0.0048	0.0022

TABLE VI
CLASSIFICATION RESULTS FOR THE PAVIA CENTER DATASET (NUMBERS IN BOLD REPRESENT THE BEST CLASSIFICATION PERFORMANCE)

Class name	SVMCK	SpATV	LBPELM	SC-MK	RMK	RPNet	ASMGSSK	LBP-SMK
Water	0.9971	0.9939	0.9932	0.9996	0.9921	0.9961	1.0000	0.9999
Trees	0.9640	0.9595	0.9601	0.9343	0.8506	0.9249	0.9041	0.9531
Asphalt	0.9756	0.9763	0.9850	0.9694	0.9321	0.9647	0.9248	0.9821
Blocking Bricks	0.9848	0.9903	0.9669	0.9995	1.0000	0.9971	0.9996	1.0000
Bitumen	0.9657	0.9514	0.9559	0.9883	0.9836	0.9957	0.9968	0.9985
Tiles	0.9837	0.9956	0.9961	0.9881	0.9768	0.9834	0.9801	0.9974
Shadows	0.9106	0.8890	0.9081	0.9652	0.9758	0.9371	0.9569	0.9863
Meadows	0.9952	0.9924	0.9995	0.9893	0.9387	0.9699	0.9964	0.9949
Bare Soil	0.9979	0.5113	0.4183	0.9980	0.8229	1.0000	0.9423	1.0000
mean_OA	0.9878	0.9773	0.9762	0.9897	0.9630	0.9806	0.9879	0.9948
STD_OA	0.0023	0.0047	0.0057	0.0018	0.0069	0.0031	0.0015	0.0014
mean_kappa	0.9826	0.9678	0.9662	0.9854	0.9479	0.9726	0.9829	0.9926
STD_kappa	0.0033	0.0067	0.0080	0.0025	0.0096	0.0044	0.0021	0.0019

information. Using MK can achieve better classification accuracy compared with SVMCK which uses two kernel functions to extract spatial-spectral information and SpATV which using a spectral data regularization and a SpATV constraint for HSI classification. RPNet is a deep learning classification method. It uses random patches as the inputs of the neural network

for HSI classification. As can be seen from the experimental results, for the classification problems with a small number of samples, the performance of the kernel function is better than RPNet. ASMGSSK directly extracts spatial-spectral information from the spatial-spectral structure and uses the superpixel segmentation algorithm to generate a multiscale SSK, which

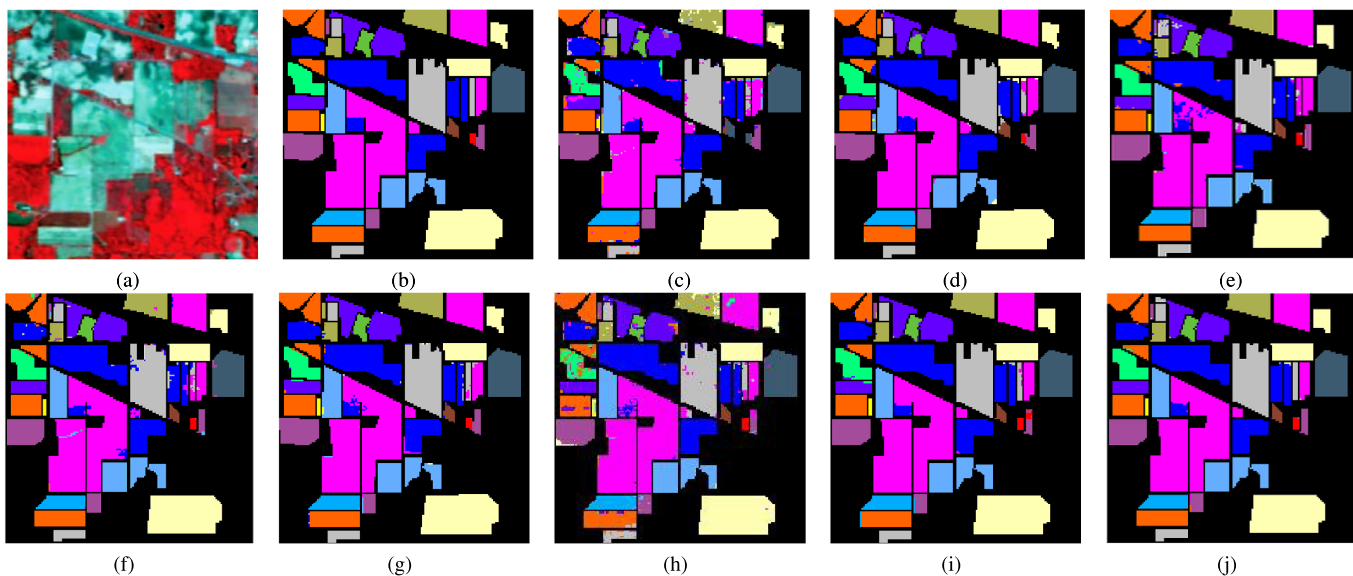


Fig. 9. Indian Pines data classification. (a) False-color maps. (b) Real feature marker. (c) SVMCK. (d) SpATV. (e) LBPELM. (f) SC-MK. (g) RMK. (h) RPNNet. (i) ASMGSSK. (j) LBP-SMK.

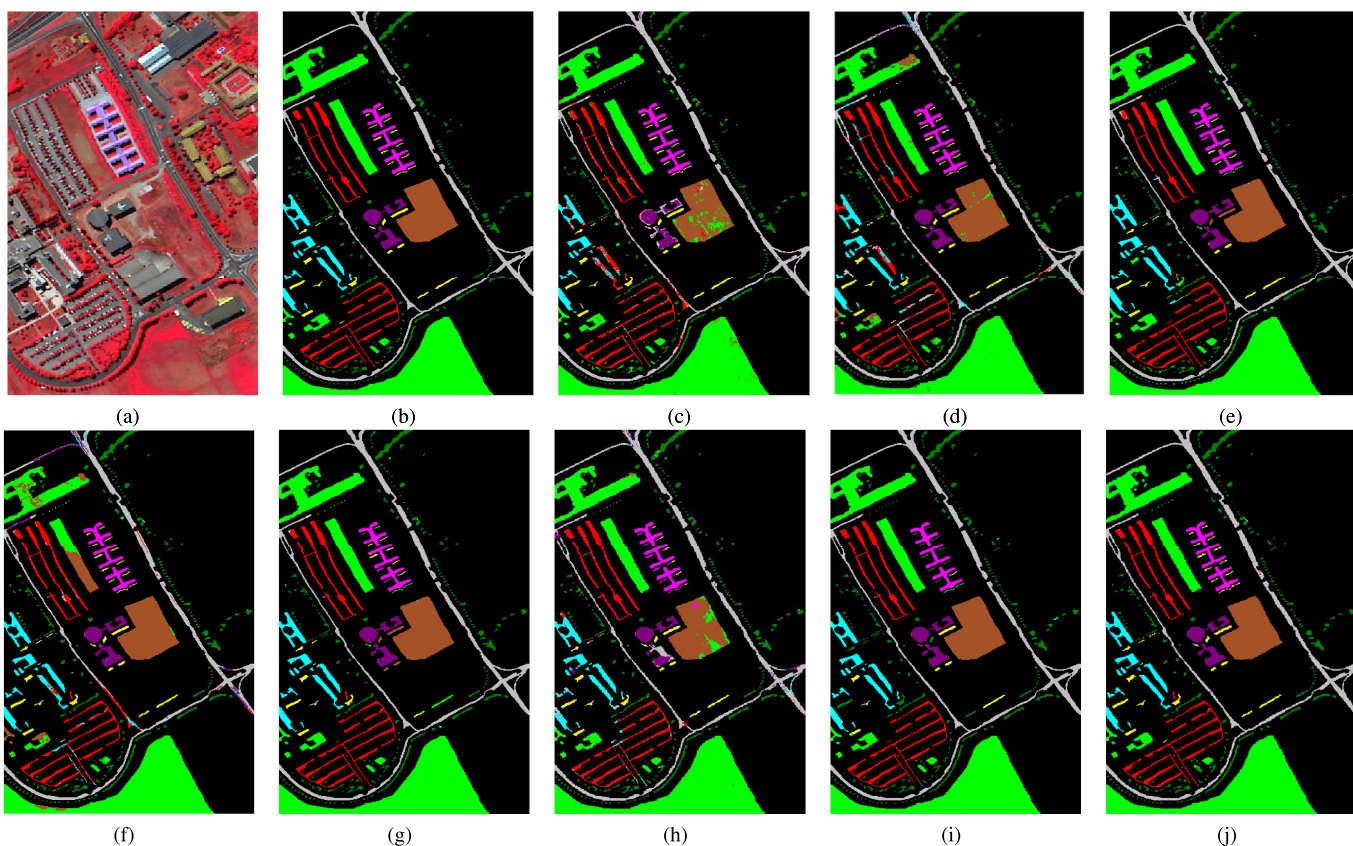


Fig. 10. Pavia University data classification. (a) False-color maps. (b) Real feature marker. (c) SVMCK. (d) SpATV. (e) LBPELM. (f) SC-MK. (g) RMK. (h) RPNNet. (i) ASMGSSK. (j) LBP-SMK.

has a better classification effect than RMK and SC-MK. The LBP-SMK method in this article is proposed on the basis of MK. The experimental results show that the accuracy is improved by 1%–6% compared with other classification methods, and the standard deviation of the algorithm is also the

smallest among the classification algorithms. This demonstrates that the algorithm proposed in this article not only has better classification accuracy but also has a more stable classification performance under the condition of limited training samples.

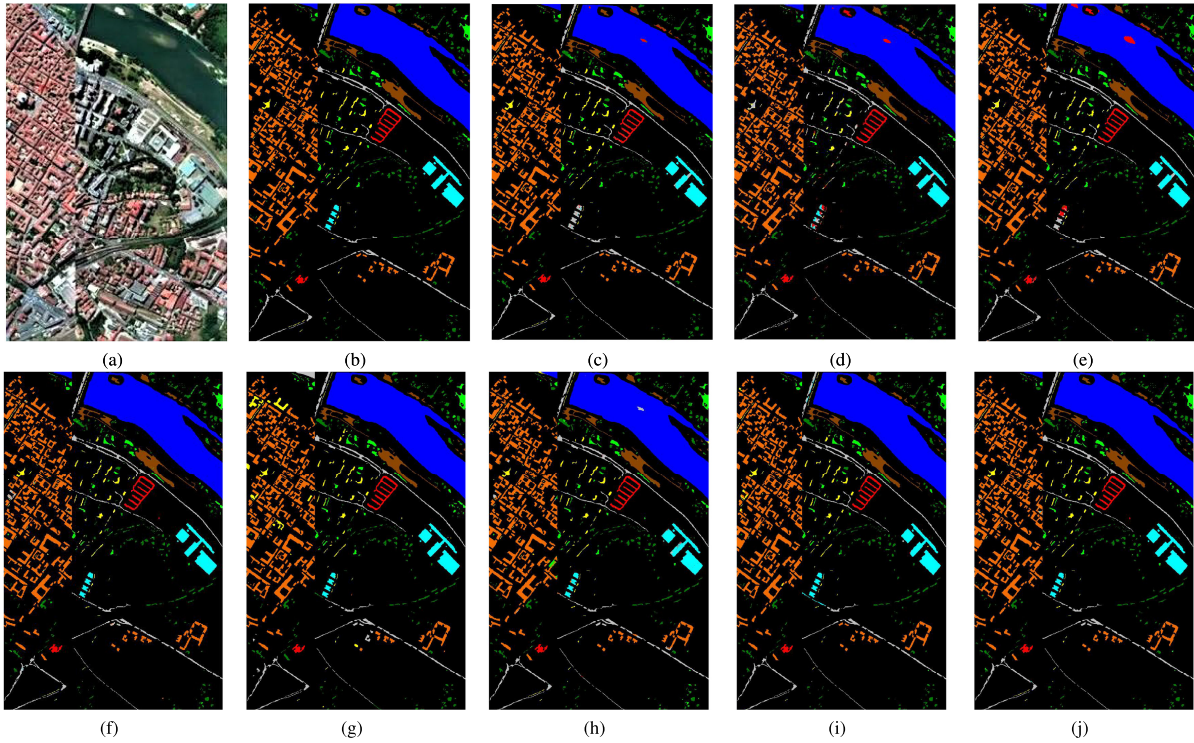


Fig. 11. Pavia Center data classification. (a) False color maps. (b) Real feature marker. (c) SVMCK. (d) SpATV. (e) LBPELM. (f) SC-MK. (g) RMK. (h) RPNNet. (i) ASMGSSK. (j) LBP-SMK.

TABLE VII

COMPARISON WITH A STATE-OF-THE-ART SUPERPIXEL-BASED CLASSIFIER ON INDIAN PINES AND PAVIA UNIVERSITY DATASETS (NUMBERS IN BOLD REPRESENT THE BEST CLASSIFICATION PERFORMANCE)

Indian Pines			Pavia University		
Class name	SuWLP	LBP-SMK	Class name	SuWLP	LBP-SMK
Alfalfa	0.9728	0.9930	Asphalt	0.4956	0.6993
Corn-no till	0.9856	0.5079	Meadows	0.8472	0.7069
Corn-min till	0.5373	0.4855	Gravel	0.8501	0.8068
Corn	0.7341	0.9667	Trees	0.3503	0.8590
Grass/pasture	0.9294	0.7454	Metal sheets	0.8600	0.9492
Grass/tree	0.5571	0.7955	Bare soil	0.9078	0.8842
Grass/pasture-mowed	0.7352	0.9880	Bitumen	0.8995	0.9169
Hay-windrowed	0.8626	0.9998	Bricks	0.6418	0.9093
Oats	0.8172	1.0000	Shadows	0.1749	0.8256
Soybeans-no till	0.9348	0.6617			
Soybeans-min till	0.6931	0.6673			
Soybeans-clean till	0.9294	0.6553			
Wheat	0.3835	0.9960			
Woods	0.7654	0.8502			
Bldg-grass-tree-drives	0.5413	0.8890			
Stone-steel towers	0.9478	0.9933			
mean_OA	0.6917	0.7073	mean_OA	0.7029	0.7765
STD_OA	0.0623	0.0321	STD_OA	0.0173	0.0723
mean_kappa	0.6500	0.6716	mean_kappa	0.6100	0.7221
STD_kappa	0.0700	0.0360	STD_kappa	0.0200	0.0831

Figs. 9, 10, and 11 illustrate the classification maps of the comparison algorithms on the three datasets, respectively. The classification map of the SVMCK method has a lot of noise when the spatial information is not fully utilized. Although, the six methods (SpATV, LBPELM, SC-MK, RMK, RPNNet, and ASMGSSK) make full use of spectral-spatial information, they do not maintain the detailed structure of

the original HSI. Many mismatched regions appear in the classification results. LBP-SMK makes full use of the spectral and spatial information, especially the spatial information. Combining LBP with superpixels results in good performance for extracting the edge features of the spatial information. Therefore, LBP can well maintain the detailed structure of HSI.

Recently, some scholars have made many new achievements in the HSI classification based on superpixel. These results have been published, but the authors have not published their codes. In these newly released hyperspectral classification methods based on superpixel, we choose the superpixel-level weighted label propagation method [59] as the comparison experiment. The experimental results are shown in Table VII (Since the common data sets used in these two articles are Indian Pines and Pavia University, we conduct comparative experiments on the two datasets). It can be seen from Table VII that even if each class only trains three labeled samples, LBP-SMK obtains a good classification result. In addition, it is noticed from Tables IV, V, and VI that RMK and SC-MK use the MK to explore spatial-spectral information well.

V. CONCLUSION

This article proposes a LBP-SMK HSI classification method. Unlike MK, the framework uses LBP to directly extract texture features of HSI with superpixel indexes. LBP filtering has powerful edge extraction capabilities, and the edge pixel information in the superpixel can be extracted to improve the classification accuracy of such pixels. Through experimental application, we have demonstrated that the method proposed in this article is not only better than MK, it is also better than other classification methods, which proves that the improvement is convincing.

In the future, the proposed method is planned to be implemented on the parallel computing platform, such as cloud computing and GPU [60], to accelerate the running time for more real-time applications.

REFERENCES

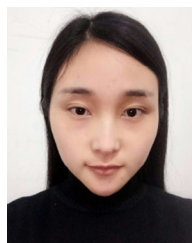
- [1] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.
- [2] L. Sun, Z. Wu, J. Liu, L. Xiao, and Z. Wei, "Supervised spectral-spatial hyperspectral image classification with weighted Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 3, pp. 1490–1503, Mar. 2015.
- [3] L. He, J. Li, and C. Liu, "Recent advances on spectral-spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, Mar. 2018.
- [4] O. Ozdil, A. Gunes, and Y. E. Esin, "4-stage target detection approach in hyperspectral images," in *Proc. IEEE 9th Workshop Hyperspectral Image Signal Process., Evol Remote Sens.*, 2018, pp. 1–5.
- [5] J. Lei, L. Wu, and Y. Li, "A novel FPGA-based architecture for fast automatic target detection in hyperspectral images," *Remote Sens.*, vol. 11, no. 2, pp. 146–165, 2019.
- [6] L. Ren, L. Zhao, and X. Li, "Anomaly detection-oriented band selection for hyperspectral image," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 3788–3791.
- [7] Y. Zhang, M. Xu, and Y. Fan, "A kernel background purification based anomaly target detection algorithm for hyperspectral imagery," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 441–444.
- [8] X. Weng, W. Lei, and X. Ren, "Kernel sparse representation for hyperspectral unmixing based on high mutual coherence spectral library," *Int. J. Remote Sens.*, vol. 41, no. 4, pp. 1286–1301, 2020.
- [9] L. Sun, F. Wu, T. Zhan, W. Liu, J. Wang, and B. Jeon, "Weighted nonlocal low-rank tensor decomposition method for sparse unmixing of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1174–1188, 2020.
- [10] X. Dong, B. Yan, and F. Gan, "Progress and prospectives on engineering application of hyperspectral remote sensing for geology and mineral resources," in *Proc. 5th Symp. Novel Optoelectronic Detection Technol. Appl. Int. Soc. Opt. Photon.*, 2019, pp. 11023–110232.
- [11] Y. Sun, Y. Zhao, and K. Qin, "Alteration hydrothermal stage division and its application in geological prospecting using casi-sasi airborne hyperspectral data: Taken the Baixianishan area in Liuyuan town of Gansu province as an example," in *Proc. IOP Conf. Series, Earth Environmental Sci.*, 2019, vol. 384, no. 1, Art. no. 012216.
- [12] B. J. Scherrer, "Weed and crop discrimination with hyperspectral imaging and machine learning," College Eng., Montana State Univ., Bozeman, MT, USA, 2019.
- [13] J. Zhang, Y. Huang, and K. N. Reddy, "Assessing crop damage from dicamba on non-dicamba-tolerant soybean by hyperspectral imaging through machine learning," *Pest Manage. Sci.*, vol. 75, no. 12, pp. 3260–3272, 2019.
- [14] G. Melillos *et al.*, "Field spectroscopy for the detection of underground military structures," *Eur. J. Remote Sens.*, vol. 52, no. 3, pp. 385–399, 2019.
- [15] M. Shimoni, R. Haelterman, and C. Perneel, "Hyperspectral imaging for military and security applications: Combining myriad processing and sensing techniques," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 101–117, Jun. 2019.
- [16] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 55–63, Jan. 1968.
- [17] H. Xu, H. Zhang, W. He, and L. Zhang, "Superpixel-based spatial-spectral dimension reduction for hyperspectral imagery classification," *Neurocomputing*, vol. 360, pp. 138–150, 2019.
- [18] A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1349–1362, Mar. 2016.
- [19] F. Gao, Q. Wang, and J. Dong, "Spectral and spatial classification of hyperspectral images based on random multi-graphs," *Remote Sens.*, vol. 10, no. 8, pp. 1271–1290, 2018.
- [20] W. Li, H. Liu, J. Wang, L. Xiang, and Y. Yang, "An improved linear kernel for complementary maximal strip recovery: Simpler and smaller," *Theor. Comput. Sci.*, vol. 786, pp. 55–66, 2019.
- [21] Y. Chen, J. Wang, R. Xia, Q. Zhang, Z. Cao, and K. Yang, "The visual object tracking algorithm research based on adaptive combination kernel," *J. Ambient Intell. Humanized Comput.*, vol. 10, pp. 4855–4867, 2019.
- [22] Y. Chen, W. Xu, J. Zuo, and K. Yang, "The fire recognition algorithm using dynamic feature fusion and IV-SVM classifier," *Cluster Comput.*, vol. 22, pp. 7665–7675, 2019.
- [23] Y. Chen, J. Xiong, W. Xu, and J. Zuo, "A novel online incremental and decremental learning algorithm based on variable support vector machine," *Cluster Comput.*, vol. 22, pp. 7435–7445, 2019.
- [24] D. Tuia, F. Ratle, A. Pozdnoukhov, and G. Camps-Valls, "Multisource composite kernels for urban-image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 88–92, Jan. 2010.
- [25] G. Camps-Valls, L. Gomez-Chova, and J. Muñoz-Mar, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.
- [26] J. Liu, Z. Wu, and Z. Wei, "Spatial-spectral kernel sparse representation for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 6, pp. 2462–2471, Dec. 2013.
- [27] L. Sun, C. Ma, and Y. Chen, "Low rank component Induced spatial-spectral kernel method for hyperspectral image classification," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–14, 2019, doi: 10.1109/TCSVT.2019.2946723.
- [28] B. Qi, C. Zhao, and E. Youn, "Use of weighting algorithms to improve traditional support vector machine based classifications of reflectance data," *Opt. Express*, vol. 19, pp. 26816–26826, 2011.
- [29] W. Liao, R. Bellens, and A. Piurica, "Classification of hyperspectral data over urban areas based on extended morphological profile with partial reconstruction," in *Proc. Int. Conf. Adv. Concepts Intell. Vis. Syst.*, 2012, pp. 278–289.
- [30] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2973–2987, Aug. 2009.
- [31] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.

- [32] Y. Song, G. Yang, H. Xie, D. Zhang, and X. Sun, "Residual domain dictionary learning for compressed sensing video recovery," *Multimedia Tools Appl.*, vol. 76, no. 7, pp. 10083–10096, 2017.
- [33] Y. Song, W. Cao, Y. Shen, and G. Yang, "Compressed sensing image reconstruction using intra prediction," *Neurocomputing*, vol. 151, no. 3, pp. 1171–1179, 2015.
- [34] 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.
- [35] X. Zhang, F. Peng, and M. Long, "Robust coverless image steganography based on DCT and LDA topic classification," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3223–3238, Dec. 2018.
- [36] M. Darvishnezhad, H. Ghassemian, and M. Imani, "Local binary graph feature reduction for three-dimensional Gabor filter based hyperspectral image," *Int. Archives Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. XLII-4/W18, pp. 285–291, Oct. 2019.
- [37] S. Jia *et al.*, "3-D Gaussian–gabor feature extraction and selection for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8813–8826, Nov. 2019.
- [38] M. Y. Liu, O. Tuzel, and S. Ramalingam, "Entropy rate superpixel segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 2097–2104.
- [39] W. Duan, S. Li, and L. Fang, "Superpixel-based composite kernel for hyperspectral image classification," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2015, pp. 1698–1701.
- [40] T. Zhan, L. Sun, and Y. Xu, "Hyperspectral classification via superpixel kernel learning-based low rank representation," *Remote Sens.*, vol. 10, no. 10, pp. 1639–1655, 2018.
- [41] L. Fang, S. Li, and W. Duan, "Classification of hyperspectral images by exploiting spectral–spatial information of superpixel via multiple kernels," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6663–6674, Dec. 2015.
- [42] J. Liu, C. Gu, J. Wang, G. Youn, and J. Kim, "Multi-scale multi-class conditional generative adversarial network for handwritten character generation," *J. Supercomputing*, vol. 75, pp. 1922–1940, 2019.
- [43] D. Zhang, T. Yin, G. Yang, M. Xia, L. Li, and X. Sun, "Detecting image seam carving with low scaling ratio using multi-scale spatial and spectral entropies," *J. Vis. Commun. Image Representation*, vol. 48, pp. 281–291, Oct. 2017.
- [44] L. Sun, C. Ma, Y. Chen, H.J. Shim, Z. Wu, and B. Jeon, "Adjacent superpixel-based multiscale spatial-spectral kernel for hyperspectral classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1905–1919, Jun. 2019.
- [45] S. Jia, X. Deng, and J. Zhu, "Collaborative representation-based multiscale superpixel fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7770–7784, Oct. 2019.
- [46] B. Cui, X. Xie, X. Ma, G. Ren, and Y. Ma, "Superpixel-based extended random walker for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3233–3243, Jun. 2018.
- [47] P. Wang, L. Zhang, G. Zhang, H. Bi, M. D. Mura, and J. Chanussot, "Super-resolution land cover mapping based on pixel- subpixel- and superpixel-scale spatial dependence with pansharpening technique," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 10, pp. 4082–4098, Oct. 2019.
- [48] P. Sellars, A. I. Aviles-Rivero, and C.-B. Schnlieb, "Superpixel contracted graph-based learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4180–4193, Jun. 2020.
- [49] S. M. Holland, "Principal components analysis (PCA)," Dept. Geology, Univ. Georgia, Athens, GA, USA, 2008.
- [50] C. Carmeli, E. Vito, and A. Toigo, "Vector valued reproducing kernel Hilbert spaces of integrable functions and Mercer theorem," *Anal. Appl.*, vol. 10, no. 4, pp. 377–408, 2006.
- [51] Y. Chen *et al.*, "Saliency detection via improved hierarchical principle component analysis method," *Wireless Commun. Mobile Comput.*, vol. 6, pp. 1–12, 2020.
- [52] S. Gupta, and S. G. Mazumdar, "Sobel edge detection algorithm," *Int. J. Comput. Sci. Manage. Res.*, vol. 2, pp. 1578–1583, 2013.
- [53] M. Pietikinen, A. Hadid, and G. Zhao, "Local binary patterns for still images," *Computer Vision Using Local Binary Patterns*. Berlin, Germany: Springer, 2011, pp. 13–47.
- [54] J. E. Tapia, C. A. Perez, and K. W. Bowyer, "Gender classification from iris images using fusion of uniform local binary patterns," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 751–763.
- [55] Q. Guo, B. Zhang, Q. Ran, L. Gao, J. Li, and A. Plaza, "Weighted-RXD and linear filter-based RXD: Improving background statistics estimation for anomaly detection in hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2351–2366, Jun. 2014.
- [56] J. Liu, Z. Wu, and Z. Xiao, "Region-based relaxed multiple kernel collaborative representation for hyperspectral image classification," *IEEE Access*, vol. 5, pp. 20921–20933, 2017.
- [57] 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.
- [58] Y. Xu *et al.*, "Hyperspectral image classification via a random patches network," *ISPRS J. Photogrammetry Remote Sens.*, vol. 142, pp. 344–357, 2018.
- [59] S. Jia, X. Deng, M. Xu, J. Zhou, and X. Jia, "Superpixel-level weighted label propagation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 5077–5091, Jul. 2020.
- [60] Z. Wu *et al.*, "GPU parallel implementation of spatially adaptive hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1131–1143, Apr. 2018.



Wei Huang (Member, IEEE) was born in Henan, China, in 1982. He received the Ph.D. degree in pattern recognition and intelligence systems from the Nanjing University of Science and Technology, Nanjing, China, in 2015.

He is currently a Lecturer with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China. His current research interests include pan-sharpening, hyperspectral classification, image processing, and machine learning.



Yao Huang was born in Henan, China, in 1995. She received the B.S. degree in physics from Zhengzhou University of Light Industry (ZZULI), Zhengzhou, China, in 2018. She is currently working toward the master's degree with School of Computer and Communication Engineering, ZZULI.

Her research interests covers hyperspectral image classification, kernel function, and machine learning.



Hua Wang was born in Henan, China, in 1986. He received the Ph.D. degree in cartography and geographical information system from the Wuhan University, Wuhan, China, in 2013.

He is currently an Associate Professor with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China. His current research interests include artificial intelligence spatial optimization decision, and land use planning.



Yan Liu was born in Henan, China, in 1988. She received the B.S. degree in physics from Henan University, Kaifeng, China, in 2010, and the doctoral degree in optics from the State Key Laboratory of Modern Optical Instruments, Zhejiang University, Hangzhou, China, in 2015.

She is currently a Lecturer with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China. Her current research interests include image processing, camera algorithm design, and computer vision.



Hiuk Jae Shim received the B.S., M.S., and Ph.D. degrees in electronics electrical engineering from Sungkyunkwan University, Suwon, South Korea in 2000, 2002, and 2013, respectively.

From 2013 to 2016, he was a Postdoctoral Researcher with the College of Information and Communication Engineering, Sungkyunkwan University, Suwon, South Korea. From 2016 to 2017, he was an Exchange Scholar with the School of Computer Science and Engineering, Nanjing University of Science and Technology (NUIST), Nanjing, China. He is currently an Associate Professor with the School of Computer and Software, NUIST, Nanjing, China. His main research interests include image/video compression, distributed video coding, compressed sensing, and information hiding.