Small sample hyperspectral image classification combining the black-winged kite algorithm enfolded bi-exponential edge-preserving smoother

Abstract. The ability of deep learning models to capture long-range sequence information or spectral-spatial contextual relationships from hyperspectral images has been noteworthy. Unfortunately, under conditions where training samples are extremely rare, these designs often suffer from overfitting and struggle to reduce the computational complexity. To tackle the small sample issue, this paper provides a unique spectral-spatial hyperspectral image classification architecture that combines a black-winged kite algorithm enfolded bi-exponential edge-preserving smoother, named BKAB. The framework comprises three main steps: Initially, the black-winged kite algorithm, an original nature-inspired optimizer that operates without gradient, is initially brought to the bi-exponential edge-preserving smoother to find optimal spatial and range parameters, which are then used to reduce noise and improve land class separability in hyperspectral images. Furthermore, a broad extension strategy is used to extend the spatial filtering features to enhancement features, and then the obtained BKAB features and enhancement features are input into the support vector machine classifier to generate soft-classified probability distributions. Finally, BKAB-based post-processing filtering is applied to adjust the probability maps, incorporating the spatial context of each pixel for further refinement. Experimental results on several public datasets demonstrate that the proposed method is superior to state-of-the-art classification methods, especially in the presence of a very small training size. The code of this work is available at https://github.com/2572158784/BKB-_BES-_EPF.

1. Introduction

Sensors can now record hyperspectral images (HSIs) in hundreds of bands, thanks to the previous several decades' fast advancements in hyperspectral imaging technology. In real-world applications, HSIs have garnered significant interest and have been extensively employed in various domains, including urban planning [1], food safety [2], precision agriculture [3, 4], mineral exploration [5], and military reconnaissance [6]. Moreover, HSI classification is a fundamental activity in remote sensing, with the purpose of attempting to distribute precise labels to various classes using a multidimensional feature space. However, efficient HSI classification algorithms face

many challenges under noise interference, intra-class spectral variability, and small sample conditions [7, 8].

To address the challenge of HSI classification, several spectral classification approaches have been developed, including random forest, k-nearest neighbor, multinomial logistic regression, and support vector machine (SVM) classifiers [9–12]. In particular, SVM-based methods have drawn more attention for HSI classification due to their great performance, even with small training sizes and high-dimensional spectral bands. Additionally, a variety of approaches, including principal component analysis (PCA) [13], independent component analysis (ICA) [14], partitioned maximum margin criterion [15], and linear discriminant analysis (LDA) [16], have been studied for dimension reduction. Generally speaking, these spectral classification techniques treat HSI as a set of spectral vectors and only extract features from each pixel's spectral information. To address the challenges of spectral variability and confusion, various approaches integrate complementary spatial contextual information with spectral features to enhance HSI classification accuracy [17]. According to recent research, utilizing spatial cues to their fullest can significantly enhance HSI classification performance. For instance, for HSI classification, a multi-kernel classifier based on domain adaptation with active learning (AL) was created, which effectively leverages labeled information from auxiliary domains, compensating for domain distribution shifts through sequential active learning, thereby enhancing classification accuracy while reducing computational costs [18]. Moreover, several mathematical morphological operators, such as morphological profile (MP) and extended multi-attribute profile (EMAP), have also been created for the best possible extraction of spatial information from images. In [19], the tensor morphological profile (TMP), a unique multi-dimensional morphology operator that extends mathematical morphology to high-dimensional data, was introduced for HSI classification. In [20], a novel HSI classification fusion technique is described that integrates the information from extended multi-attribute morphological profiles (EMAPs) with the SVM.

Currently, edge-preserving filters have been a hot topic in image processing research. It has been shown that these filters perform smoothing operations to minimize noise and extract edge-preserving features (EPFs) that accurately capture the significant spectral-spatial structures of HSIs. In [21], a 3-D Gabor feature-based collaborative representation (3GCR) technique was described, which significantly improves the classification accuracy of HSI by using all acquired Gabor features to reinforce the L2 norm collaborative representation against noise. In [22], a unique PCA-based edge-preserving features (EPFs) extraction technique was provided for the classification of HSIs with the goals of completing the characterization of spatial information and improving class separability. In [23], this HSI classification method combined PCA, recursive filtering, and random patches network (RPNet) to obtain instructive deep information, thereby reducing the complexity of deep learning frameworks and offering good classification outcomes when few-shot learning was employed. Moreover, numerous studies have proven that edge-preserving filtering techniques work well for

adding geometric information to the classifier during the post-processing stage [24, 25]. Nonetheless, a drawback of the conventional edge-preserving smoothing procedure is that the classification performance is also significantly impacted by the filter's smoothing degree. In this case, it is challenging for the EPFs produced by manually choosing the ideal parameters to accurately depict the intricate spatial structures in the HSIs.

Singular spectrum analysis (SSA) has recently been applied to hyperspectral imaging for feature extraction, improving classification accuracy by enhancing feature discriminability and removing noise. In [26], SSA has shown strong efficacy in HSI classification by improving feature extraction and removing noise, thereby enhancing discriminative power and outperforming empirical mode decomposition. In [27], a novel multiscale 2-D SSA fusion method, integrating PCA and segmented-PCA, has been proposed for spectral-spatial feature extraction. Experimental results on four benchmark datasets demonstrate its superiority over state-of-the-art methods, particularly with limited training samples. Despite its advantages in HSI classification, conventional SSA suffers from high computational complexity due to pixel-wise singular value decomposition (SVD). To address this, a fast SSA (F-SSA) method is proposed, which applies a single SVD to a representative pixel instead of individual pixel-based SVDs. Experiments on two publicly available datasets demonstrate that F-SSA maintains classification accuracy with the SVM classifier while significantly reducing computational complexity [28].

Markov random field (MRF) is another model that leverages spatial-context information, which has been effectively implemented in HSI classification. In [29], a new supervised HSI classification method that simultaneously used low-rank matrix factorization (LRMF) and MRF was developed. This clearly improves the classifier's classification performance by simultaneously integrating spatial-spectral features and removing the complex noise that exists in the HSIs. In [30], a cascaded MRF architecture was created for HSI classification by concatenating the original spectral feature with the projected probability vector produced by the SVM classifier to exploit more discriminating features. Additionally, because of the benefits of its straightforward structure, small number of training parameters, and quick training procedure, researchers have focused more and more on developing BLS classifiers to increase the classification accuracy of HSIs [31, 32].

Despite the fact that hand-crafted feature extractors increase classification accuracy, it is difficult to extract features with a high degree of discrimination flexibility because they rely on manually produced information. Deep learning (DL) algorithms, on the other hand, can effectively address the limitations of limited generalization and characterization by automatically describing abstract and complicated structure information [33–35]. In [36], an active DL method for HSI classification was proposed to enhance the classification performance while lowering the labelling cost. This approach combines AL, convolutional neural networks (CNN), and MRF. In [37], a hybrid spectral CNN architecture for HSI classification was proposed This enables the model to acquire more abstract spatial representations while minimizing the 3-D CNN's complexity.

In [38], a unique dual-channel residual network (DCRN) that resolves HSI classification with noisy labels was created by combining a new dual-channel structure and a noise-robust loss function.

Because of their exceptional capacity to adaptively represent high-level features, the deep learning algorithms based on the HSI classification techniques mentioned above, in general, significantly enhance performance.Nevertheless, these methods generally overemphasize local content information, which may distort the sequence information and increase the difficulty of spectral attribute mining [39, 40]. Furthermore, they struggle to extract implicitly significant semantic information. To address these problems, recent work has advocated using Transformers in HSI to increase spectral feature characterization while also representing high-level semantic characteristics [41–43]. For example, a highly flexible Spectral Former network was incorporated into the HSI classification in 2021 and could be applied to both pixel and patch-wise inputs, outperforming state-of-the-art backbone networks [44]. In [17], a spectral-spatial feature tokenization transformer (SSFTT) method was built to increase classification performance effectively and efficiently, capturing high-level semantic features by organically merging a backbone CNN and transformer structure. In [45], a new Swin Transformer network was created for HSI classification, which uses group attention and sliding window attention computation to increase feature representation and classification outcomes. More recently, LCTCS, a better low-memory and less-parametric network technique for HSI classification, was developed in [46] to further increase the efficiency of computing resources used with outstanding classification performance and reduced computational resource levels. During the same year, an efficient Spectral Mamba [47] model for HSI classification was created, which is a novel state space model that combines a channel focus mechanism with efficacious grouped convolution and dynamic convolution. As expected, the suggested Spectral Mamba produces a new insight into the HSI classification, which achieves remarkable performance and efficiency gains.

In contrast, it has been extensively demonstrated that the previously mentioned DL models are more adept at managing non-linear data dynamics in intricate HSIs and produce cutting-edge classification accuracy when the training sample is adequate, particularly for the Transformers models. Notwithstanding these advantages, they are intrinsically hampered by difficulties in parallel training and the laborious pairwise multiplicative computations [47, 48]. Particularly, it seems that the majority of these versions have reached a bottleneck period where a trade-off between computational cost and accuracy is inevitable. In other words, these DL methods in question are not eco-friendly due to their high computational demands, noisy interference problems, and deficient labelled samples [31]. Particularly for Transformer-based classification architectures, achieving high accuracy faces significant challenges due to their quadratic computational complexity and memory-intensive self-attention mechanisms, especially in large-scale HSI processing and small-sample scenarios with hardware constraints. To address these issues, strategies such as model pruning and quantization may be

useful to reduce the number of parameters, thereby lowering storage and computational demands. Additionally, a sparse attention mechanism can be introduced to focus on the most relevant data, reducing computational complexity while maintaining classification performance.

With the aforementioned reasons in mind, we present a unique small sample HSI classification architecture that combines BKAB, a broad expansion strategy (BES), and a post-processing edge-preserving filtering (EPF) technique, namely BKB_BES_EPF. Specifically, BKAB stands for the black-winged kite algorithm (BKA) [49] enfolded with the bi-exponential edge-preserving smoother (BEEPS). In general, the suggested approach combines three essential elements: In the preprocessing stage, BKA is integrated with the BEEPS to adaptively select the best smoothing and range parameters. This procedure can efficiently extract the discriminant EPF features and eliminate the noise from the original HSIs at the same time. As far as we are aware, BKA is the first effort to optimize the BEEPS for the purpose of addressing the most suitable filtering parameters. Additionally, the BKAB features are mapped to the enhancement features using BES, and the resulting enhancement features and the original BKAB features are then cascaded and fed into the SVM to create the soft-classified probability maps. In the post-processing stage, the suggested BKAB is further utilized to modify the maps with probabilistic soft classification by incorporating context-sensitive details for every pixel, where revised probability maps can be generated via the main voting mechanism.

The primary innovations in this work can be summarized as follows:

- We construct a new edge-preserving smoother, named BKAB. As the name suggests, BKA is first utilized to optimize the BEEPS and is not only capable of adaptively determining the most appropriate range and smoothing parameters, but it can also effectively enhance the separability between land cover categories.

- BES is employed to characterize the enhancement features, and then the compact enhancement signatures and original BKAB features are effectively merged, which is our goal in merging them to improve the classification capacity of the classifier.

- At the stage of post-processing, BKAB is employed to make additional corrections to the incorrectly classified hyperspectral pixels that are included in the initial probability map, improving the classification accuracy even more.

The rest of this paper is organized as follows: Section 2 provides an in-depth exposition of BKA, BEEPS, BES, and the suggested classification methodology. Section 3 describes the datasets and experimental setup. In Section 4, experimental results conducted on three HSI datasets and related discussions are analyzed in detail. Section 5 reports the conclusions.

2. Related methodologies

Figure 1 shows a process flow illustrating the BKB_BES_EPF, which chiefly includes the subsequent phases: (1) BKAB is applied to the HSIs at the pre-processing stage in order to enhance image quality and choose the appropriate filtering settings in an adaptive manner. By improving the separability of land cover categories, it is then capable of fully exploiting the highly discriminatory EPF features. (2) The resulting BKAB features are extended to the enhancement layers using BES. The initial BKAB features and the mapped broad extension features are then combined and fed into the SVM, which creates the soft-classified probability distributions. (3) Leveraging the spatially informed data at each pixel, BKAB is further employed in the post-processing stage to correct the original probability maps. Subsequently, we shall systematically present the associated techniques.
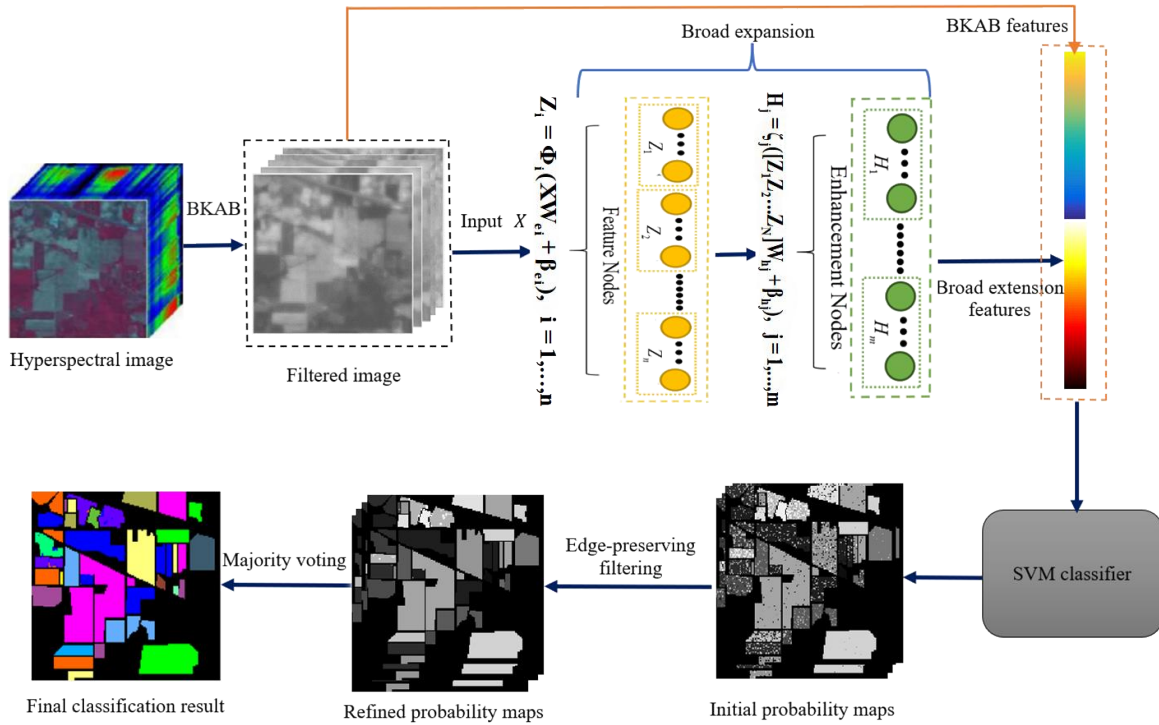

Figure 1. The structure of the proposed BKB_BES_EPF framework.

2.1. Black-winged kite algorithm

The black-winged kite algorithm (BKA) is a straightforward and effective meta-heuristic optimization technique created in 2024 to imitate the predatory skills and migratory patterns of black-winged kites [49]. The BKA can handle specific optimization problems by iterating and adjusting in two stages: attacking and migration behavior. The

mathematical model goes as follows:

2.1.1. Initialization phase In BKA, the first stage in population initialization is to generate a set of random solutions. To indicate the position of each black-winged kite (BK), use the following matrix:

$$BK = \begin{bmatrix} BK_{1,1} & BK_{1,2} & \cdots & \cdots & BK_{1,dim} \\ BK_{2,1} & BK_{2,2} & \cdots & \cdots & BK_{2,dim} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ BK_{pop,1} & BK_{pop,2} & \cdots & \cdots & BK_{pop,dim} \end{bmatrix} \tag{1}$$

where pop represents the number of possible solutions, dim denotes the dimension of the provided problem, and BK_{ij} indicates the jth dimension of the ith black-winged kite. Here, the position of each black-winged kite is spread evenly as follows:

$$X_i = BK_{lb} + rand(BK_{ub} - BK_{lb}) \tag{2}$$

where represents an integer ranging from 1 to pop, and BK_{lb} and BK_{ub} indicate the lower and upper limits of the ith BK in the jth dimension, respectively, and rand denotes a random value from $[0, 1]$.

The leader X_L in the initial population, which is thought to be the best place for black-winged kites, is chosen by BKA during the initialization phase. Here's a mathematical depiction of the original leader X_L, and the smallest value is considered an example.

$$f_{best} = min(f(X_i)) \tag{3}$$

$$X_L = X(find(f_{best} == f(X_i))) \tag{4}$$

2.1.2. Attacking behavior Black-winged kites are predators on tiny grassland mammals and insects. During a fight, they alter the angles of their wings and tails to suit the wind speed. They hover silently to study their prey, then swiftly plunge and attack. Several assault behaviors are included in this technique for global search and exploration. A mathematical model of the assault behavior of black-winged kites is presented below.

$$y_{t+1}^{i,j} \begin{cases} y_t^{i,j} + n(1 + sin(r)) \times y_t^{i,j} & p < r \\ y_{t+1}^{i,j} + n \times (2r - 1) \times y_t^{i,j} & else \end{cases} \tag{5}$$

$$n = 0.05 \times e^{-2 \times \left(\frac{t}{T}\right)^2} \tag{6}$$

where $y_t^{i,j}$ and $y_{t+1}^{i,j}$ denote the locations of the ith black-winged kites in the jth dimension in the t and $(t + 1)^{th}$ iteration steps, respectively. Here, r is a random number ranging from 0 to 1, whereas p is a constant value of 0.9, which controls the switching between two attack modes and is a crucial parameter that influences the algorithm's stability. T represents the total number of iterations, while t indicates the number of completed iterations.

2.1.3. Migration behavior The complicated behavior of birds during migration is controlled by various environmental conditions, such as climate and food availability. A hypothesis regarding bird migration is proposed, that is, the leader will resign from leadership and join the migrating population if the fitness value of the present population is lower than that of the random population, suggesting that it is unfit to lead the population forward. Conversely, the population will be guided until it achieves its target if the present population's fitness value is higher than that of the random population. With this approach, great leaders can be dynamically chosen to guarantee a smooth changeover. The migratory behavior of black-winged kites can be represented mathematically as follows:

$$y_{t+1}^{i,j} \begin{cases} y_t^{i,j} + C(0,1) \times \left(y_t^{i,j} - L_t^j\right) & F_i < F_{ri} \\ y_t^{i,j} + C(0,1) \times \left(L_t^j - m \times y_t^{i,j}\right) & else \end{cases} \tag{7}$$

$$m = 2 \times sin(r + \pi/2) \tag{8}$$

where L_t^j denotes the top scorer of the black-winged kites in the jth dimension of the tth iteration thus far. $y_t^{i,j}$ and $y_{t+1}^{i,j}$ denote the locations of the ith black-winged kites in the jth dimension in the t and $(t+1)^{th}$ iteration procedures, respectively. F_i depicts a black-winged kite's current position in the jth dimension gained in the t iteration. F_{ri} denotes the fitness value of the random position in the jth dimension acquired from any black-winged kites in the t iteration. Here, $C(0,1)$ symbolizes the Cauchy mutation.

The probability density function of the one-dimensional Cauchy distribution is represented as:

$$f(x, \delta, \mu) = \frac{1}{\pi} \frac{\delta}{\delta^2 + (x - \mu)^2}, -\infty < x < \infty \tag{9}$$

Its probability density function will take on the conventional form for $\delta = 1$, $\mu = 0$. This is the exact formula, as follows:

$$f(x, \delta, \mu) = \frac{1}{\pi} \frac{1}{x^2 + 1}, -\infty < x < \infty \tag{10}$$

2.2. BEEPS

The bi-exponential edge-preserving smoother (BEEPS), introduced by Thevenaz in 2012, is a highly efficient method that demands minimal coding effort, along with low memory and computational resources [50]. In essence, the BEEPS operates through two one-tap recursive processes: the regressive and progressive recursions. This approach has proven useful in high hyperspectral image classification tasks due to its ability to preserve edges while smoothing the data.

The first recursion is carried out in a progressive manner. Let $x[n]$ denote the current sample of the input sequence x at position $x \in \mathbb{Z}$, and the items of an auxiliary sequence φ are successively computed as:

$$\phi[n] = (1 - X[n]\rho)x[n] + X[n]\rho\phi[n-1] \tag{11}$$

$$X[n] = r(x[n], \phi[n-1]) \tag{12}$$

where $\lambda \in [0, 1)$ controls the degree of smoothing applied by a convolutional space filter with impulse response s. The variable r represents a nontrivial range filter, defined as:

$$r(u, v) = e^{\frac{-(u-v)^2}{2\sigma^2}} \tag{13}$$

where σ determines the spread of the bump, and r supplies a mechanism for progressively distinguishing between edge and non-edge regions.

The second recursion is carried out in a regressive manner and is quite similar to the first. Another auxiliary sequence, φ, is computed recursively as follows:

$$\vartheta[n] = (1 - \tau[n]\rho)x[n] + \tau[n]\rho\vartheta[n+1] \tag{14}$$

$$\tau[n] = r(x[n], \vartheta[n+1]) \tag{15}$$

To generate the samples of the output sequence y, the progressive sequence and regressive sequence are combined and computed as follows:

$$y[n] = \frac{\phi[n] - (1 - \rho)x[n] + \vartheta[n]}{1 + \rho} \tag{16}$$

2.3. BES

Unlike traditional deep learning architectures, which often require time-consuming training processes, the broad learning system [51] offers an alternative approach to learning within a deep structure. Studies have demonstrated that BLS possesses global general approximation capabilities, allowing it to achieve better generalization than many other machine learning models. In this section, we expand the original input by incorporating enhanced features through the use of the BES in BLS, aiming to leverage the highly separable spatial-spectral feature set for improved hyperspectral image classification.

Given the input $X \in R^{N \times d}$, where N denotes the number of training samples, and d represents the dimensionality of each sample, $Y \in R^{N \times c}$ represents the corresponding labels, with c indicating the total number of classes.

Initially, the input is projected into n groups using random weights and biases to form feature nodes $Z = [Z_1, Z_2, \ldots, Z_n]$. The feature mapping nodes for the ith group are represented as follows:

$$Z_i = \varphi(XW_{ei} + \beta_{ei}) \tag{17}$$

where Φ_i symbolizes the activation map, W_{ei} and β_{ei} signify the randomly generated weights and bias components.

Second, the extracted feature nodes are processed to generate enhancement nodes $H = [H_1, H_2, \ldots, H_m]$ in the enhancement layer. The jth group of enhancement nodes is expressed as follows:

$$H_j = \xi_j(ZW_{hj} + \beta_{hj}), j = 1, \ldots, m \tag{18}$$

where W_{hj} and β_{hj} represent the randomly initialized weight and bias terms, respectively, while ξ_j denotes a nonlinear activation function.

Additionally, the predicted output $\hat{Y}$ of the model can be expressed as:

$$\hat{Y} = AW = [Z_1, Z_2, \ldots, Z_n | H_1, H_2, \ldots, H_m]W \tag{19}$$

where $A = [Z|H]$ represents the combined features W denotes the weight matrix that connects these features. The optimal W is determined by minimizing the objective function:

$$J_{BLS} = \underbrace{argmin}_{W} : \|AW - Y\|_2^2 + \lambda\|W\|_2^2 \tag{20}$$

where λ represents the regularization parameter. The weight matrix W can be computed using the ridge regression method.

$$W = \left(A^T A + \lambda I\right)^{-1} A^T Y \tag{21}$$

where I represents the identity matrix, and the above formula gives the optimal output weight matrix W.

Algorithm 1 Pseudocode of BKB_BES_EPF

Input: I is the original HSI, X is the input, Y is the corresponding labels, N is the size of train samples, d is the sample dimensions, c is the number of classes, ρ is the smoothing parameter, σ is the range parameter, r is the regularization parameter, s is the shrinkage scale of the enhancement nodes, N_1 is the feature nodes per window, N_2 is the number of windows of feature nodes, N_3 is the number of enhancement nodes, γ is the kernel parameter, and C is the penalty parameter.

Output: Prediction labels $\hat{Y}$.

1: Select the optimal parameters ρ and σ, perform BKAB to smooth each spectral band of I and get $y[n]$ using equations (1)-(16), where the fitness value in BKA is determined by evaluating the MSE (mean-square error) between the relatively clean and the denoised images.
2: For the filtered image, randomly choose the training set X and the rest for testing.
3: Utilize the BES to transform X to create enhancement nodes $H = [H_1, H_2, \ldots, H_m]$ using equations (17)-(21), and then construct a spectral-spatial feature set S by combining the BKAB features and the mapped enhancement features H_j.
4: Set the proper C and γ, and then obtain the probabilistic classification maps I^{BKB_BES} through feeding the spectral and spatial features S into the SVM.
5: Adopt BKAB to obtain the refined probability maps $I^{BKB\_BES\_EPF}$ in the post-processing stage.
6: Classify the labels $\hat{Y}$ using the majority voting mechanism.

---

3. Data sets and experimental setup

3.1. Hyperspectral datasets

To evaluate the effectiveness of the proposed BKB_BES_EPF method, experiments are conducted on three genuine hyperspectral data sets, namely, the Indian Pines (IP), the University of Pavia (UP), and the Houston 2013 (HU) datasets.

The Indian Pines image is the first genuine data collection. It was taken in June 1992 over an agricultural region of Northwestern Indiana by the airborne visible infrared imaging spectrometer (AVIRIS). The image has a spatial resolution of 20 meters and 224 channels. After eliminating 24 channels affected by water and atmospheric absorption, 200 channels remain. Figure 2 presents a three-band false-color image along with a ground truth map. As shown in Figures 2(b) and (c), the reference includes 16 land cover classes, the majority of which are agricultural varieties.

The second dataset is from the University of Pavia. It was captured over the Italian city of Pavia in 2003 by the Reflective Optical Spectral Imaging System (ROSIS-3) sensor. The dataset contains nine real land cover classes with a pixel size of 610 × 340, a spatial resolution of 1.3 m/pixel. The remaining 103 bands are evaluated after 12 water absorption bands are removed. Figure 3 presents a three-band false color image as well as a ground truth map. As depicted in Figures 3(b) and (c), the reference only includes 9 land cover classes, and most of them are background pixels.

The third data set is the University of Houston 2013, which was acquired by the Compact Airborne Spectrographic Imager (CASI) sensor at Houston University. This image covers a spatial range of 349 by 1905, with 144 channels, spanning a wavelength range from 0.38 μm to 1.05 μm. Figure 4 presents a three-band false color image as well as a ground truth map. As depicted in Figures 4(b) and (c), the reference comprises 15 land cover categories.

Table 1 gives the related information of the three datasets. Table 2 lists the categories and the quantity of training and testing samples used in our experiments.
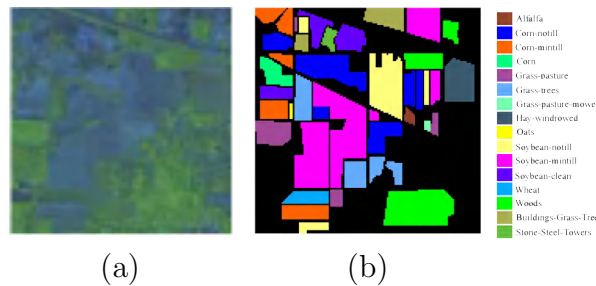

(a)   (b)

Figure 2. Indian Pines dataset. (a) Three-band color composite image. (b) Ground truth map
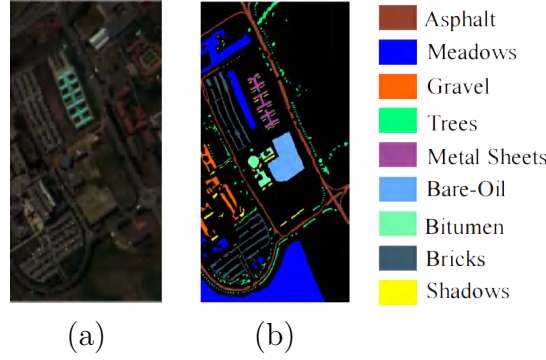
Figure 3. Pavia University dataset. (a) Three-band color composite image. (b) Ground truth map
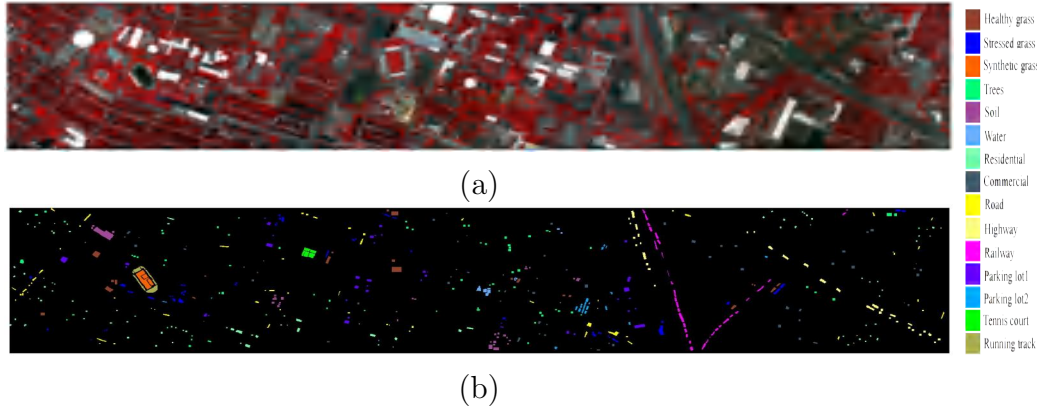

Figure 4. Houston 2013 dataset. (a) Three-band color composite image. (b) Ground truth map

Table 1. Related information of IP, UP, and HU datasets.

Dataset	Sensor	Spatial Resolution	Number of Spectral Bands	Number of Classes	Image size (Pixels)
Indian Pines (IP)	AVIRIS	20 m/pixel	200	16	145 × 145
University of Pavia (UP)	ROSIS-3	1.3 m/pixel	103	9	610 × 340
Houston 2013 (HU)	CASI	2.5 m/pixel	144	15	349 × 1905

3.2. Experimental Setup

In order to verify the efficacy and superiority of the suggested BKB_BES_EPF, we conducted a comparison between the BKB_BES_EPF and many sophisticated algorithms, including the SVM on the original spectral features [9], the BLS on the original spectral features [51], spatial-spectral classification based on locality-preserving projection, local binary pattern and broad learning system (LPP_LBP_BLS) [31], spectral-spatial shared kernel ridge regression (SSSKRR) [52], multiscale spatial features are fused with the global spectral features derived from PCA (MSF-PCs) [53], spectral-spatial feature tokenization transformer (SSFTT) [17], central vector oriented self-similarity network (CVSSN) [54], adaptive mask sampling and manifold to the Euclidean subspace learning (AMS-M2ESL) framework [55], SC_SS_MTr [56], SS_Mamba [57], SCTF [58], the BKA, encapsulated within BEEPS and succeeded by BLS (referred

Table 2. Land cover classes and training-test sets for IP, UP, and HU datasets.

No.	Indian Pines (1%) Class	Training	Test	Pavia University (1%) Class	Training	Test	Houston 2013 (1%) Class	Training	Test
1	Alfalfa	1	53	Asphalt	67	6564	Healthy grass	13	1238
2	Corn-notill	15	1419	Meadows	187	18462	Stressed grass	13	1241
3	Corn-mintill	9	825	Gravel	21	2078	Synthetic grass	7	690
4	Corn	3	231	Trees	31	3033	Trees	13	1231
5	Grass-pasture	5	492	Metal Sheets	14	1331	Soil	13	1229
6	Grass-trees	8	739	Bare-Oil	51	4978	Water	4	321
7	Grass-pasture-mowed	1	25	Bitumen	14	1316	Residential	13	1255
8	Hay-windrowed	5	484	Bricks	37	3645	Commercial	13	1231
9	Oats	1	19	Shadows	10	937	Road	13	1239
10	Soybean-notill	10	958				Highway	13	1214
11	Soybean-mintill	25	2443				Railway	13	1222
12	Soybean-clean	7	607				Parking lot1	13	1220
13	Wheat	3	209				Parking lot2	5	464
14	Woods	13	1281				Tennis court	5	423
15	Buildings-Grass-Trees	4	376				Running track	7	653
16	Stone-Steel-Towers	1	94						
	Total	111	10255	Total	432	42344	Total	158	14871

to as BKAB_BLS), broad expansion and spectral features followed by SVM (namely, BES_SVM).

Three assessment metrics for evaluating models' efficacy quantitatively are as follows: 1) Overall accuracy (OA) is the percentage of test samples that are properly identified out of all test samples; 2) Average accuracy (AA), which is the mean of each class's test accuracy; 3) Kappa coefficient (κ), which denotes the percentage of agreement adjusted by the mutually exclusive quantity of agreement. Meanwhile, to lessen the unpredictability introduced by training samples, the final trial result is the average of ten experimentals.

Before applying the suggested BKB_BES_EPF methods, parameter setup is required to select the optimal settings. For the three datasets, the population size is predetermined as $N = 10$, with a constant $p = 0.9$, and the total number of iterations is set to $T = 20$. The shrinkage factor for the enhancement nodes, denoted as s, is selected from the set $\{0.1, 0.2, \cdots, 0.9\}$, and the regularization parameter r is chosen from the range $\{2^{-45}, 2^{-40}, \cdots, 2^0\}$. Specifically, s is set to be $s = 0.8$ for the three datasets, r is set to be $r = 2^{-15}$, $r = 2^{-20}$, and $r = 2^{-20}$ for the IP, UP, and HU datasets, respectively. For the IP dataset, feature nodes per window N_1, is set to $N_1 = 24$, the number of windows of feature nodes N_2, is set to $N_2 = 15$, the number of enhancement nodes N_3, is set to $N_3 = 700$, the kernel parameter γ, is set to $\gamma = 2.^\wedge(2)$, the penalty parameter C, is set to $C = 2.^\wedge(10)$, the smoothing parameter λ, is set to $\lambda = 0.68$, the range parameter σ, is set to $\sigma = 3$. For the UP dataset, N_1 is set to be $N_1 = 18$, N_2 is set to be $N_2 = 6$, N_3 is set to be $N_3 = 400$, γ is set to $\gamma = 2.^\wedge(2)$, C is set to be $C = 2.^\wedge(8)$,
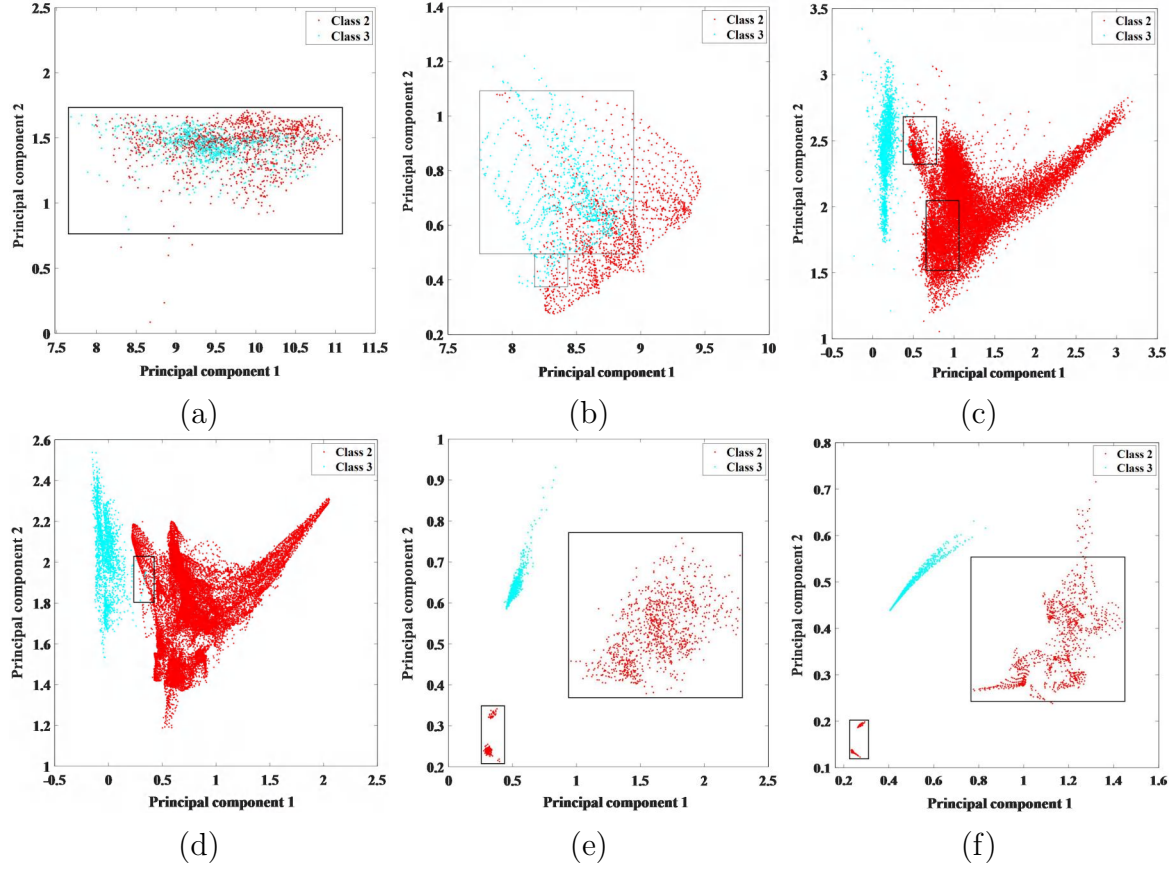
Figure 5. Distribution and spectral response of the second and third land cover types on IP, UP, and HU datasets. (a) original IP image. (b) BKAB-filtered image based on IP. (c) original UP image. (d) BKAB-filtered image based on UP. (e) original HU image. (f) BKAB-filtered image based on HU.

λ is set to be $\lambda = 0.68$, σ is set to be $\sigma = 5$. For the HU dataset, N_1 is set to be $N_1 = 21$, N_2 is set to be $N_2 = 21$, N_3 is set to be $N_3 = 700$, γ is set to be $\gamma = 2.^{\wedge}(2)$, C is set to be $C = 2.^{\wedge}(10)$, λ is set to be $\lambda = 0.53$, σ is set to be $\sigma = 4$. Due to space constraints, the optimal parameters for BKB_BES_EPF are only briefly discussed in the next section.

3.2.1. Evaluation of intra class similarity and inter class separability In general, sensors, the atmosphere, and quantization processing can all cause uncertain "noise interference" during signal collection or transmission, which degrades the quality of the resulting HSIs; furthermore, due to the complex spectral response mechanism and function of imaging spectrometers, which are affected by material dispersion or the surrounding environment, different forms of land cover in HSIs reflect comparable characteristics for electromagnetic waves in a certain band, while similar types of land cover display distinct characteristics in that same band. As a result, "same substance but different spectra" and "same spectral foreign objects" have become commonplace phenomena in HSIs. In this case, this work suggests an image denoising and spatial-

spectral feature extraction approach based on BKAB to address or at least partially mitigate the aforementioned issues.

In several HSIs, we assessed the similarity distribution of the same land cover class and the difference distribution of distinct land cover classes in order to confirm the efficacy of the suggested BKAB methodology. In particular, the spectral dimension is decreased while keeping the first two principal components using PCA for analytical convenience. Figure 5 displays the distribution and spectral response of the second and third land cover classes in three actual HSIs.
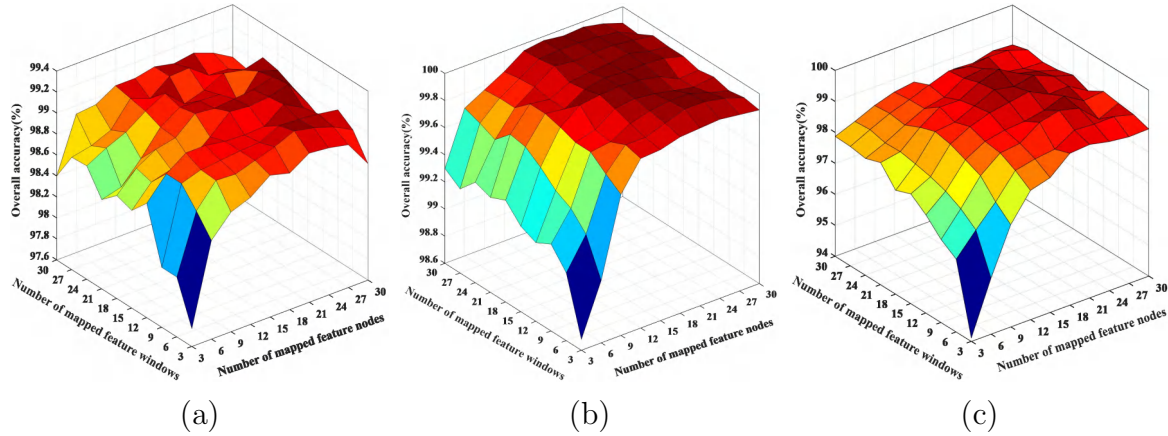

Figure 6. Classification performance of BKB_BES_EPF with the parameters N_1 and N_2 for (a) IP, (b) UP, and (c) HU datasets.

Figure 5(a) illustrates how the second and third land cover types in the IP dataset are typically entirely confounded, thus limiting the classifier's ability to recognize different land cover types based on the fact that the greater the difference in spectral response between different land cover classes, the more conducive it is to their correct recognition. Figure 5(b) shows that, although there are a few examples with confusion, the majority of the samples in the second and third types of classes have good separability. Figure 5(c) and 5(d) show that the second and third categories of features in the UP dataset exhibit high separability. In comparison to Figure 5(c), Figure 5(d) shows that the distribution of the third type of class is more concentrated, whereas the second type of class is mixed up with fewer pixels in the third type of class. Similarly, Figures 5(e) and 5(f) show that the second and third types of classes in the HU dataset exhibit high separability. Figure 5(f) shows a more clustered distribution of the third type of class than Figure 5(e).

3.2.2. Impact of the parameters N_1 and N_2 on OA This section explores how the number of feature nodes per window (N_1) and feature node windows (N_2) affect OA in three real HSIs. N_1 and N_2 are chosen from the interval $\{3, 6, 9, 12, 15, 18, 21, 24, 27, 30\}$, respectively. As seen in Figure 6, each parameter setting is repeated ten times to minimize random bias, and the average OA is computed while holding other parameters

constant. In particular, the training samples in each dataset are identical, as indicated by Table 2.

The mean OA of the BKB_BES_EPF is significantly impacted by N_1 and N_2, as Figure 6 illustrates. In general, since less feature information is recovered when N_1 and N_2 are small, BKB_BES_EPF typically obtains the lower mean OA. Furthermore, it is evident that as N_1 and N_2 rise, the performance of BKB_BES_EPF clearly improves, albeit at the expense of increased calculation and time consumption. It should be highlighted in particular that the OA does not precisely exhibit a rising trend as N_1 and N_2 grow; in this instance, N_1 and N_2 are roughly selected for three HSIs. That is, N_1 is approximately set to be 24, and N_2 is approximately set to be 15 for the IP dataset; for the UP dataset, N_1 is approximately set to be 18, and N_2 is approximately set to be 6; for the HU dataset, N_1 is approximately set to be 21 and N_2 is approximately set to be 21.
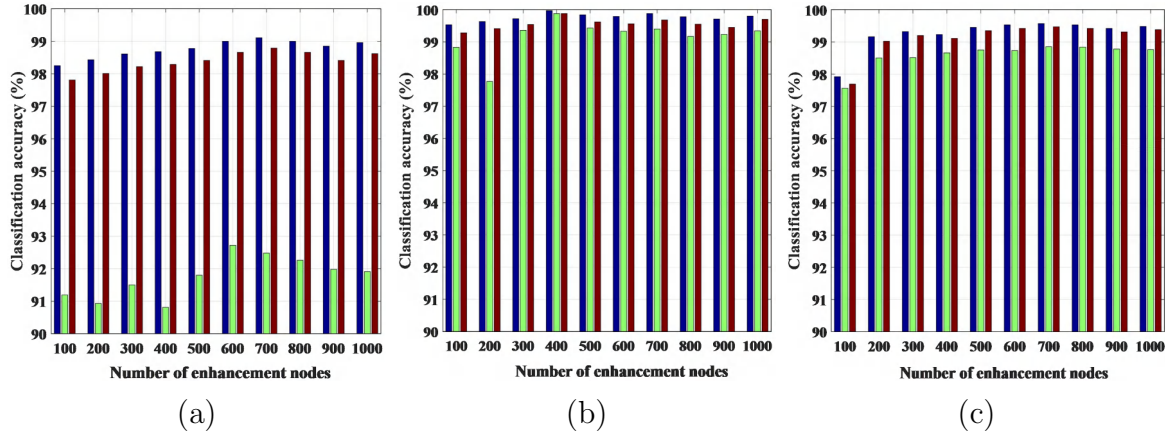

Figure 7. Examination of the influence of the enhancement node count N3 on OA using the BKB_BES_EPF method for (a) IP, (b) UP, and (c) HU datasets.

3.2.3. Impact of parameter N_3 on OA In this section, we confirm how the number of enhancement nodes N_3 affects OA. N_3 is selected from $[100, 200, 300, \ldots, 1000]$ in three datasets. Figure 7 displays the influence of the parameter on the average classification outcomes for the BKB_BES_EPF architecture following ten iterations of the experiment. In particular, the training samples in each dataset are identical, as indicated by Table 2. In all three datasets, the OA of BKB_BES_EPF increases approximately with the growth of N_3, as seen in Figure 7. This makes sense because the more features from enhancement nodes that may be retrieved, the greater N_3 is. It should be noted that, as N_3 continues to increase, the mean classification outcomes of BKB_BES_EPF do not continue to increase but slightly decrease. This suggests that training classifier models may not benefit from having too much information. In this instance, the numbers of enhanced nodes N_3 are roughly set to $N_3 = 700$ in both the IP and HU datasets, and $N_3 = 400$ in the UP dataset.

4. Classification results and analysis

4.1. Comparison of Classification Performance

To assess the efficacy of the proposed approach with a small training size, we arbitrarily choose 1% of the annotated pixels for training and use the leftover pixels in each class for testing. Tables 3-5 display the average classification outcomes (i.e., the classification accuracy of each class, OA, AA, and κ) for three datasets over ten runs using all of the aforementioned approaches, with bolded values denoting the best results. The three indications of SVM and BLS, OA, AA, and κ, are generally less than those of other techniques, as Tables 3-5 demonstrate. The reason behind the relatively low classification performance of SVM and BLS is that they solely utilize the spectral information of HSIs, disregarding the spatial information.

As mentioned previously, integrating the spatial and spectral signatures clearly improves performance. For instance, in Table 3, OA is over 11.14% higher for SSSKRR, MSF-PCs, CVSSN, SSFTT, and AMS-M2ESL than for spectral-based SVM, and 18.12% higher OA for LPP_LBP_BLS than for spectral-based BLS. In Table 4, SSSKRR, MSF-PCs, CVSSN, SSFTT, and AMS-M2ESL deliver over 5.37% greater OA than spectral-based SVM, while LPP_LBP_BLS offers over 12.46% higher OA than spectral-based BLS. This demonstrates how important spatial information is to improving classification performance. In particular, Table 5 shows that the classification performance of AMS-M2ESL, MSF-PCs, CVSSN, SSFTT, and SSSKRR is consistently significantly superior to that of spectral-based SVM. However, when measured against spectral-based BLS, the classification accuracy of LPP_LBP_BLS is considerably worse. This is most likely due to the tight spacing between land-cover categories and the scattered distribution of similar types in the HU image. This causes a lot of background and other land-cover pixels to be mixed together when extracting local texture information, making the classification task more difficult.

In contrast, for all three experimental datasets, the suggested BKAB_BLS provides over 12.57% higher accuracy than BLS, and the improvement is consistently quite substantial. This indicates that applying the EPF operator to HSIs at the pre-processing stage is a positive method; in other words, BKAB is a very discriminative spatial feature extractor. In particular, compared to employing spectral features alone (i.e., SVM and BLS), feature-level fusion of spectral characteristics and broad extension features greatly boosts classification accuracy. For instance, Table 3 shows that SVM's OA and AA for the IP dataset are 65.33% and 59.31%, but BES_SVM's accuracy rises to 73.87% and 75.42%, respectively. Likewise, BES_SVM provides 2.14% and 2.00% greater accuracy than SVM in Tables 4 and 5, respectively.

It is also significant to highlight that Table 3 has some imbalanced classification data, indicating that certain categories achieve relatively low classification accuracy. For instance, in the IP dataset, categories "Alfalfa", "Grass-pasture-mowed", "Oats", and "Stone-Steel-Towers" generate poor results on distinct classes in SVM and BLS; additionally, categories "Grass-pasture-mowed" and "Oats" underperform in SSFTT

Table 3. Accuracy for every class (%), OA (%), AA (%), and κ of different methods on the IP dataset.

Class	SVM	BLS	LPP_LBP_BLS	SSSKRR	MSF-PCs	SSFTT	CVSSN	AMS-M2ESL	SC_SS_MTr	SS_Mamba	SCTF	BKAB_BLS	BES_SVM	BKB_BES_EPF
1	35.10	28.78	92.08	45.28	92.45	77.33	**100.0**	**100.0**	40.44	41.77	60.44	79.25	63.40	73.33
2	60.52	59.39	82.18	81.96	86.04	81.61	84.66	83.36	81.74	72.63	73.71	83.58	72.70	**99.71**
3	42.01	45.32	72.53	72.61	53.69	76.33	68.83	96.08	82.89	75.98	69.67	84.32	56.43	**99.59**
4	36.54	41.13	90.04	69.27	90.47	85.70	89.36	**91.24**	67.17	41.88	71.45	70.63	62.45	88.72
5	61.06	62.56	68.74	80.12	69.30	75.75	89.56	**99.77**	66.69	75.48	72.76	79.64	90.05	99.00
6	88.69	75.08	84.33	89.76	86.33	94.77	47.44	97.30	88.78	97.06	82.82	94.81	88.46	**100.0**
7	53.60	25.67	**100.0**	91.33	**100.0**	0	14.77	73.53	39.25	55.55	85.18	99.33	61.55	**100.0**
8	91.65	83.13	90.79	90.88	**100.0**	91.71	96.92	**100.0**	98.43	**100.0**	96.82	97.97	91.93	**100.0**
9	54.74	36.41	66.32	82.46	94.73	0	18.6	85.00	25.26	36.84	63.15	**99.12**	65.36	30.00
10	52.65	51.39	79.69	75.02	92.58	78.82	85.97	90.36	83.26	81.62	82.39	82.13	56.88	**99.85**
11	66.81	70.86	83.89	84.55	94.55	80.15	84.47	78.43	93.81	93.34	89.63	87.09	68.38	**100.0**
12	52.13	55.84	72.52	72.54	78.08	80.13	62.89	90.94	72.36	55.02	65.89	78.99	71.53	96.71
13	97.51	95.94	77.99	95.30	99.04	91.39	97.25	**100.0**	79.80	95.44	77.12	99.28	92.59	**100.0**
14	91.43	85.65	95.16	93.00	85.48	92.53	100	90.92	96.54	97.26	90.99	92.13	91.92	**100.0**
15	28.14	35.13	81.81	75.80	97.60	77.97	63.79	94.91	70.15	57.12	73.45	86.62	73.03	**100.0**
16	36.38	0.36	50.42	64.72	80.85	55.14	95.77	98.90	12.39	81.52	45.43	77.66	**100.0**	96.63
OA(%)	65.33	64.33	82.45	82.17	86.24	82.71	76.47	88.18	84.88	82.88	81.09	86.57	73.87	**99.11**
AA(%)	59.31	53.28	80.54	78.99	87.57	67.80	75.02	91.92	68.69	72.41	75.06	87.03	75.42	**92.72**
κ(%)	60.38	58.92	80.09	79.68	84.22	80.17	73.24	86.40	82.64	80.31	78.44	84.71	70.00	**98.98**

Table 4. Accuracy for every class (%), OA (%), AA (%), and κ of different methods on the UP dataset.

Class	SVM	BLS	LPP_LBP_BLS	SSSKRR	MSF-PCs	SSFTT	CVSSN	AMS-M2ESL	SC_SS_MTr	SS_Mamba	SCTF	BKAB_BLS	BES_SVM	BKB_BES_EPF
1	89.83	74.76	81.66	94.24	99.02	96.57	92.99	95.83	99.36	99.41	96.94	98.32	92.77	**99.99**
2	94.00	95.56	98.61	99.42	**100.0**	99.35	99.09	99.11	99.90	99.97	99.58	99.92	95.38	**100.0**
3	65.31	51.74	92.02	81.55	98.02	97.89	78.91	95.71	98.02	98.49	93.66	91.34	73.56	**99.97**
4	96.21	81.01	53.67	86.59	93.01	97.25	97.49	99.79	97.13	97.77	75.27	90.98	95.86	**99.94**
5	99.41	98.02	88.36	79.60	**100.0**	95.47	87.45	99.85	98.61	**100.0**	93.16	85.29	99.65	**100.0**
6	82.77	33.87	99.23	96.23	**100.0**	99.76	97.68	99.38	99.73	99.91	99.72	99.34	85.34	**100.0**
7	77.52	44.56	93.91	93.76	99.92	90.70	94.71	99.84	99.90	99.84	98.64	99.34	80.57	**99.99**
8	74.51	67.22	89.34	89.08	97.17	84.67	91.12	96.01	98.44	98.55	95.86	95.79	77.94	**99.99**
9	**100.0**	79.58	44.40	68.77	74.49	82.06	92.00	99.40	85.31	99.89	64.69	87.27	99.84	99.77
OA(%)	88.71	77.59	90.05	94.08	98.44	96.29	95.45	98.28	99.02	99.52	95.87	97.43	90.85	**99.99**
AA(%)	86.62	69.59	82.36	87.69	95.73	94.18	92.38	98.32	97.38	99.31	90.89	94.17	88.99	**99.96**
κ(%)	85.00	69.25	86.73	92.11	97.93	95.08	93.96	97.71	98.70	99.36	94.52	96.57	87.87	**99.98**

Table 5. Accuracy for every class (%), OA (%), AA (%), and κ of different methods on the HU dataset.

Class	SVM	BLS	LPP_LBP_BLS	SSSKRR	MSF-PCs	SSFTT	CVSSN	AMS-M2ESL	SC_SS_MTr	SS_Mamba	SCTF	BKAB_BLS	BES_SVM	BKB_BES_EPF
1	85.70	94.10	62.88	92.65	84.49	94.94	93.36	88.19	91.90	93.53	80.14	96.09	90.74	**100.0**
2	93.17	86.14	59.30	92.73	93.07	98.16	91.13	98.25	93.65	95.32	83.57	94.40	94.90	**100.0**
3	99.89	99.35	87.80	96.48	**100.0**	98.36	92.25	**100.0**	97.68	**100.0**	85.44	97.93	98.40	99.98
4	97.90	94.24	60.05	87.30	96.91	89.37	99.02	92.97	97.05	98.08	66.02	96.53	95.49	**100.0**
5	89.44	75.72	85.46	96.89	**100.0**	92.58	83.58	92.14	98.04	99.13	96.24	99.71	99.77	**100.0**
6	99.64	91.00	81.27	47.70	88.78	98.26	85.71	93.33	82.30	87.47	80.31	64.22	**99.89**	83.13
7	77.81	81.56	59.70	76.90	97.92	91.58	90.42	93.21	87.07	90.62	80.60	79.01	83.43	**99.97**
8	68.37	58.77	51.00	54.21	56.70	96.45	76.49	89.88	74.94	71.77	66.75	58.76	85.59	**99.14**
9	65.03	64.15	47.04	83.89	83.77	87.81	83.70	72.24	89.73	89.84	70.10	81.75	62.55	**99.83**
10	62.01	66.03	95.67	91.53	85.74	83.85	78.49	80.89	93.77	98.51	95.46	92.43	61.65	**100.0**
11	56.82	63.86	89.62	91.39	90.34	79.30	84.02	82.05	89.77	90.45	82.48	84.26	58.35	**99.80**
12	59.49	54.01	79.67	77.53	86.31	84.58	71.66	84.73	85.49	89.88	86.16	87.23	50.75	**100.0**
13	24.42	14.31	80.15	84.53	70.68	89.20	83.38	**100.0**	74.78	50.90	82.37	74.18	52.34	99.96
14	89.43	79.01	**100.0**	91.79	99.52	97.51	64.20	93.60	99.71	99.52	93.80	95.00	93.90	**100.0**
15	98.32	87.61	73.49	88.19	**100.0**	91.21	91.93	**100.0**	98.52	**100.0**	85.08	96.86	99.64	**100.0**
OA(%)	76.73	74.35	71.51	84.62	88.51	90.21	84.89	88.84	90.48	91.31	81.55	87.28	78.73	**99.53**
AA(%)	77.83	73.99	74.21	83.59	88.95	90.84	84.62	90.76	90.29	90.34	82.30	86.56	81.76	**98.79**
κ(%)	74.82	72.22	69.19	83.37	87.57	89.41	83.65	87.93	89.70	90.60	80.06	86.24	76.97	**99.49**

and CVSSN, category "Alfalfa" performs poorly in SSSKRR, and category "Oats" conducts poorly in BKB_BES_EPF. This is most likely due to the percentage sampling used in the random sampling process, the fact that only one training sample was selected for training the classifier in the IP dataset for these categories, which caused a sample imbalance, and the highly significant deviation in sample size for some land cover categories. Particularly, the imbalanced classification outcomes are not reflected in the UP and HU datasets.

Remarkably, it is also discovered that the suggested BKB_BES_EPF approach achieves the highest values across the three metrics, for instance, OA = 99.53%, AA = 98.79%, and κ = 99.49% in the HU dataset, with over 90% accuracy in most classes. This can be attributed to three main processes: performing BKAB to effectively remove or reduce noise while enhancing separability between different land cover classes; fully integrating the initial filtering features and discriminative broad expansion features; and, finally, fully considering the context-aware information of each pixel.

Figures 8-10 show classification maps generated by related methodologies on the IP, UP, and HU datasets for qualitative assessment. These maps are consistent with the findings displayed in Tables 3-5. General speaking, pixel-wise classifiers (i.e., SVM, BLS) produce an excessive presence of salt-and-pepper noise for all three datasets [e.g., Figures 8(b), 8(c), Figures 9(b), 9(c)].

Comparatively, noise-filled maps produced by classification using spatial features (e.g., LBP, BEEPS, and BES) are somewhat better, but the visual results remain subpar in large heterogeneous regions and discontinuous homogeneous regions [e.g., Figures 8(d), 8(j), 8(k), 9(d), 9(k), 10(d), 10(k)]. To be more specific, the SSSKRR, MSF-PCs, SSFTT, CVSSN, and AMS-M2ESL frameworks generate smoother and clearer classification maps; however, some of these methods fail to maintain stability across all datasets. For example, in the IP dataset, SSSKRR and MSF-PCs also generate some salt and pepper noises in large homogeneous regions; additionally, the misclassification of CVSSN and SSFTT is clearly in the edge region.

As anticipated, under various challenging dataset situations, BKB_BES_EPF consistently produces more precise and fluid visual results with distinct land cover forms and bounds, mostly due to the adequate improvement of separability between various land classes. Visually speaking, this demonstrates the efficacy of both the BKAB, BES, and EPF modules [e.g., Figures 8(l), 9(l)].

4.2. Effect of the filtering scales

Multi-scale filtering features can, in actuality, influence the classifier's recognition performance to some extent. In this part, we will examine the effect of the filtering scales on the OA, AA, and κ of the proposed BKB_BES_EPF methodology, explaining why we only employ single-scale filtering features for experimental analysis in Tables 3-5. For each experiment dataset, 1% and 0.5% labeled samples are randomly chosen as training data, taking into account the stability and robustness of the suggested approach
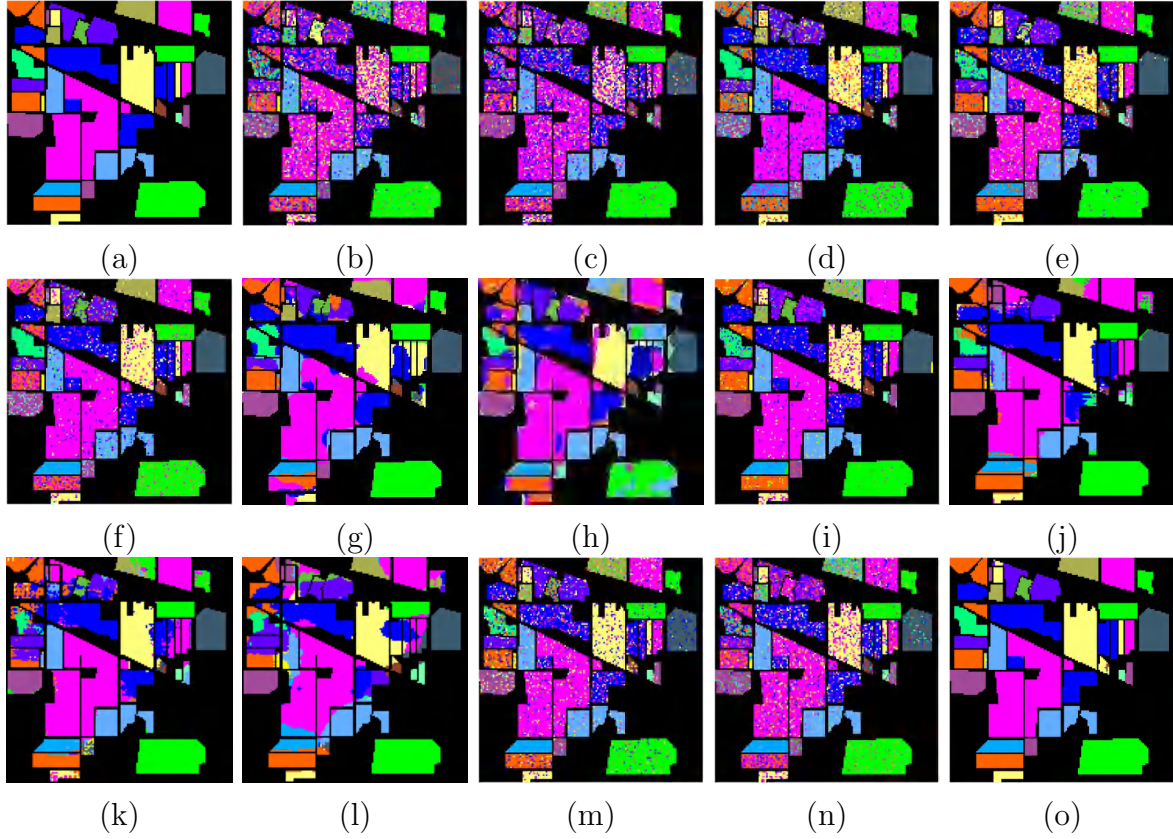
Figure 8. Classification maps for the IP dataset. (a) Ground truth map; and the classification outcomes produced by the (b) SVM, (c) BLS, (d) LPP_LBP_BLS, (e) SSSKRR, (f) MSF-PCs, (g) SSFTT, (h) CVSSN, (i) AMS-M2ESL, (j) SC_SS_MTr, (k) SS_Mamba, (l) SCTF, (m) BKAB_BLS, (n) BES_SVM, and (o) BKB_BES_EPF.

under various percentages of training samples. To lessen the bias resulting from random sampling, the experiment is conducted ten times using various randomly chosen samples, as indicated in Table 6.

It's evident that even when dealing with a tiny sample size, our approach produced good results. Furthermore, three objective metrics of BKB_BES_EPF increase as the number of filtering scales grows. However, the increased classification accuracy comes at the expense of time. Using the IP dataset as an example, BKB_BES_EPF with three filtering scales gives 0.5% greater OA than BKB_BES_EPF with a signal filtering scale when the training ratio is set to 1%; the execution time rises from 47.28 seconds to 140.44 seconds. In datasets from UP and HU, this phenomenon of sacrificing time to increase classifier recognition performance is particularly pronounced. This is mostly because adaptive filtering operations on the UP and HU datasets require longer execution times due to their larger size in comparison to the IP dataset. It is common knowledge that the two key metrics used to assess classifier performance in recognition are classification accuracy and execution time. Thus, this study uses features of a single filtering scale for experimental analysis to balance execution time and classification accuracy.
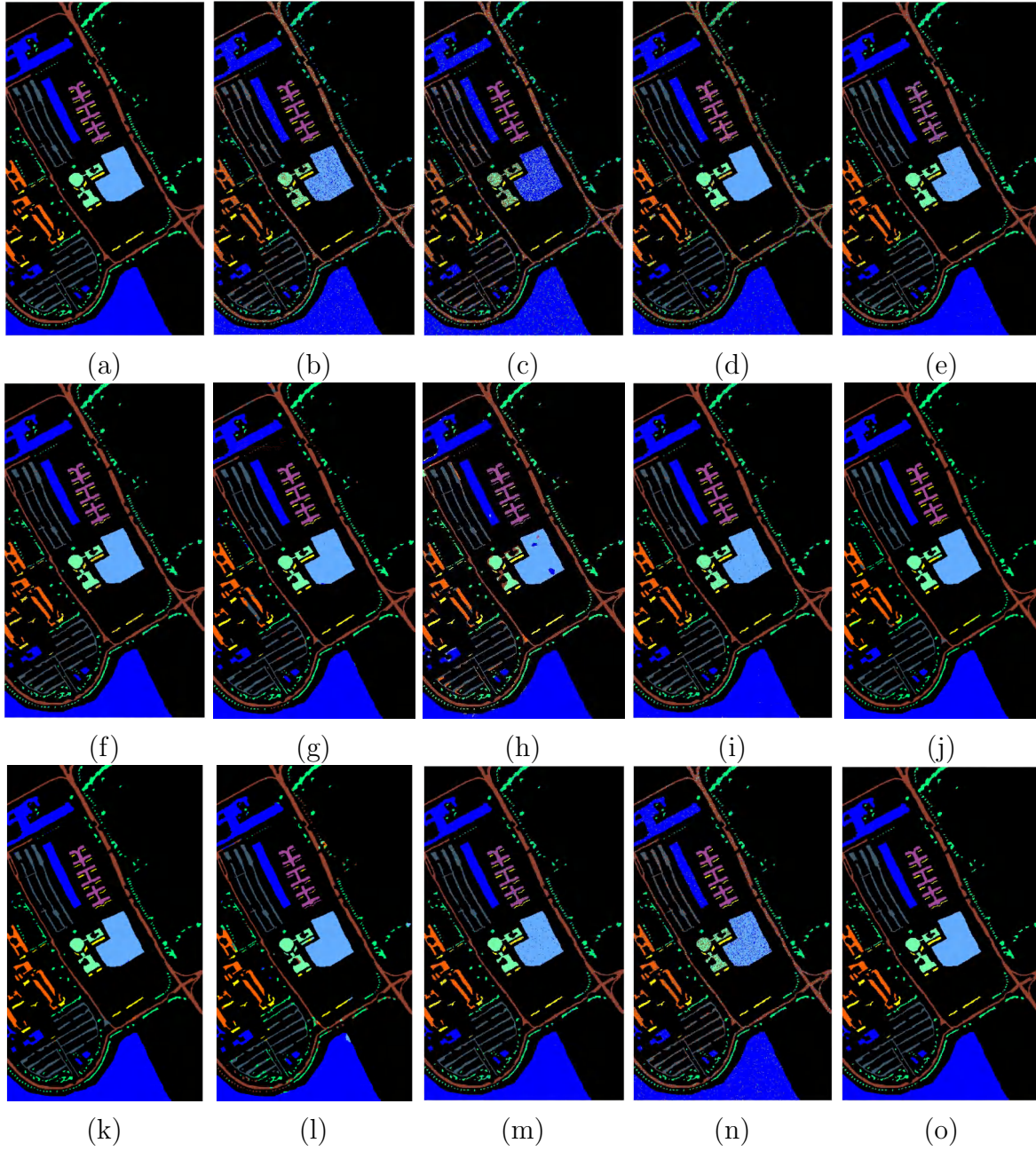
Figure 9. Classification maps for the UP dataset. (a) Ground truth map; and the classification outcomes produced by the (b) SVM, (c) BLS, (d) LPP_LBP_BLS, (e) SSSKRR, (f) MSF-PCs, (g) SSFTT, (h) CVSSN, (i) AMS-M2ESL, (j) SC_SS_MTr, (k) SS_Mamba, (l) SCTF, (m) BKAB_BLS, (n) BES_SVM, and (o) BKB_BES_EPF.
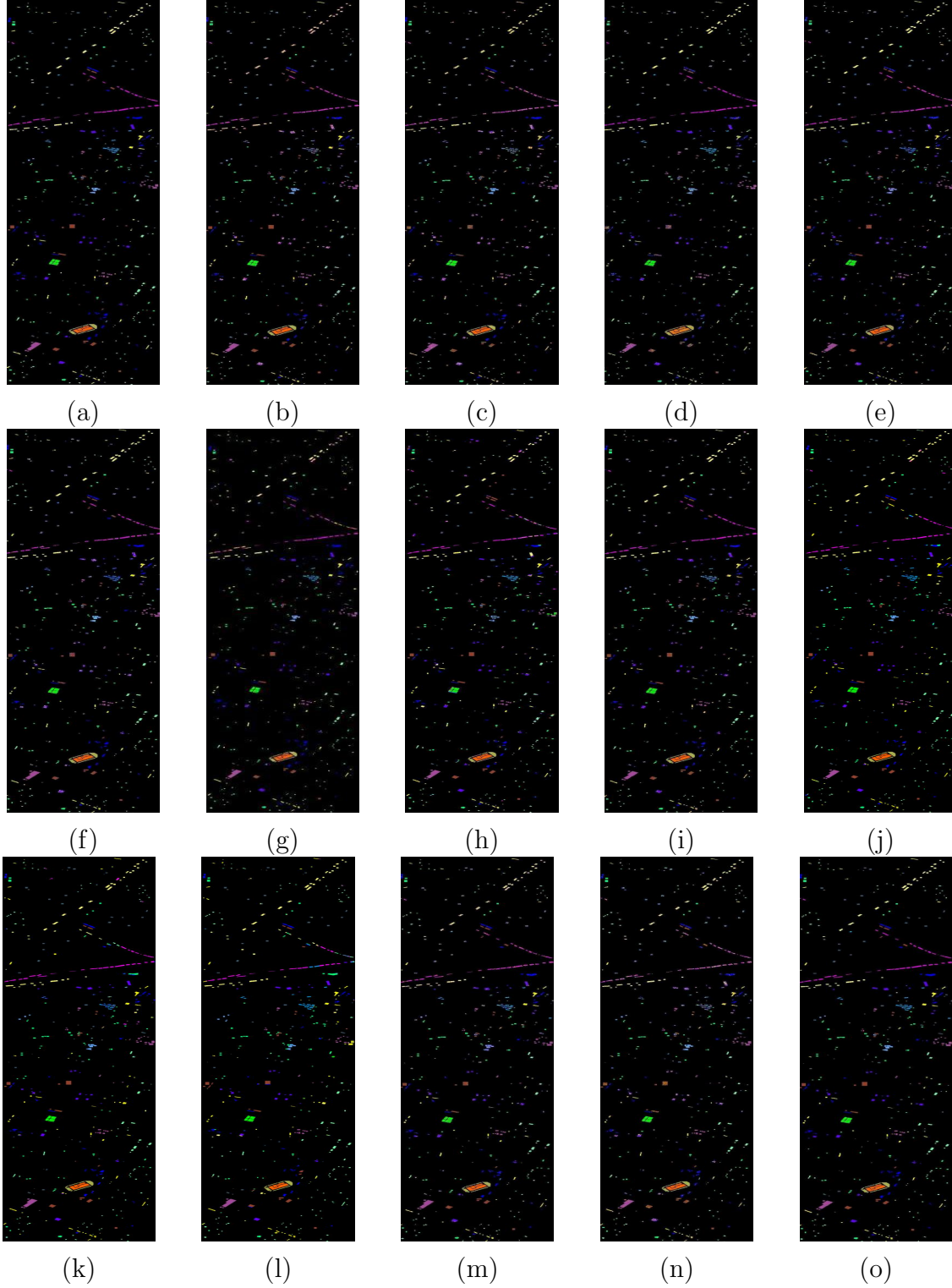
Figure 10. Classification maps for the HU dataset. (a) Ground truth map; and the classification outcomes produced by the (b) SVM, (c) BLS, (d) LPP_LBP_BLS, (e) SSSKRR, (f) MSF-PCs, (g) SSFTT, (h) CVSSN, (i) AMS-M2ESL, (j) SC_SS_MTr, (k) SS_Mamba, (l) SCTF, (m) BKAB_BLS, (n) BES_SVM, and (o) BKB_BES_EPF.

Table 6. Classification outcomes of BKB_BES_EPF with different filtering scales under limited training sizes on the three datasets.

Dataset	Metric	Training ratio (1%)			Training ratio (0.5%)		
		Scale Size (1)	Scale Size (3)	Scale Size (5)	Scale Size (1)	Scale Size (3)	Scale Size (5)
IP	OA(%)	99.11	99.61	99.70	98.50	99.14	99.30
	AA(%)	92.72	96.83	96.45	94.81	98.37	97.30
	$\kappa \times 100$	98.98	99.56	99.66	98.29	99.02	99.20
	Running time (s)	47.28	140.44	234.53	47.05	138.86	232.92
UP	OA(%)	99.99	100.0	100.0	99.77	99.99	100.0
	AA(%)	99.96	99.99	99.99	99.47	99.97	99.98
	$\kappa \times 100$	99.98	100.0	100.0	99.70	99.98	99.99
	Running time (s)	248.54	742.61	1236.82	247.23	739.46	1231.07
HU	OA(%)	99.53	99.90	99.94	98.21	99.04	99.23
	AA(%)	98.79	99.89	99.94	98.50	99.17	99.29
	$\kappa \times 100$	99.49	99.89	99.93	98.07	98.96	99.17
	Running time (s)	1080.93	3241.35	5406.49	1079.91	3235.06	5398.75

4.3. Ablation study

Ablation tests on IP, UP, and HU datasets are carried out in this section to further investigate and validate the contributions of the proposed BKB_BES_EPF framework's various modules, namely the BEEPS, BKA, spectral features (SF) extraction, broad extension features (BEF) extraction, and EPF modules. Table 7 lists all of the experimental results. In the first case, SVM with the original spectral characteristics exhibits the poorest classification performance across all three datasets. In the subsequent scenario, SVM with the BEEPS features clearly enhances classification performance to varied degrees in comparison to the first case; for example, 18.06%, 18.64%, and 20.71% on the IP dataset and 10.72%, 11.46%, and 11.61% on the HU dataset for OA, AA, and κ, respectively. The findings emphasize the importance of removing potential noise and improving the separability of different land cover categories.

In the third case, "BKA+BEEPS+SVM," which denotes the BKA, is first embedded with the BEEPS to adaptively pick the range and smoothing parameters, and the resulting features are then fed into the SVM. Table 7 shows that the third case's model outperforms the second; for example, the IP dataset shows approximately 0.61%, 1.69%, and 0.73% performance climbing for OA, AA, and κ, whereas the UP dataset shows approximately 1.48%, 2.16%, and 1.96% performance ascending for OA, AA, and κ. In the fourth case, the original spectral features and the broad extension features obtained from the BLS are put into the SVM in a combination known as "SF+BEF+SVM." The classification gains over 8% on the IP dataset and over 2% on the UP dataset relative to the initial scenario. This indicates unequivocally that, at the

Table 7. Ablation analysis of the proposed method on the three datasets (Optimal results in bold).

Datasets	Cases	BEEPS	BKA	SF	BEF	SVM	EPF	OA(%)	AA(%)	$\kappa \times 100$
IP	1	×	×	✓	×	✓	×	65.33	59.31	60.38
	2	✓	×	×	×	✓	×	83.39	77.95	81.09
	3	✓	✓	×	×	✓	×	84.00	79.64	81.82
	4	×	×	✓	✓	✓	×	73.87	75.42	70.00
	5	×	×	✓	✓	✓	✓	94.78	77.53	94.01
	6	✓	✓	×	✓	✓	✓	**99.11**	**92.72**	**98.98**
Datasets	Cases	BEEPS	BKA	SF	BEF	SVM	EPF	OA(%)	AA(%)	$\kappa \times 100$
UP	1	×	×	✓	×	✓	×	88.71	86.62	85.00
	2	✓	×	×	×	✓	×	96.01	94.73	94.71
	3	✓	✓	×	×	✓	×	97.49	96.89	96.67
	4	×	×	✓	✓	✓	×	90.85	88.99	87.87
	5	×	×	✓	✓	✓	✓	99.45	98.69	99.27
	6	✓	✓	×	✓	✓	✓	**99.99**	**99.96**	**99.98**
Datasets	Cases	BEEPS	BKA	SF	BEF	SVM	EPF	OA(%)	AA(%)	$\kappa \times 100$
HU	1	×	×	✓	×	✓	×	76.73	77.83	74.82
	2	✓	×	×	×	✓	×	87.45	89.29	86.43
	3	✓	✓	×	×	✓	×	89.26	90.93	88.39
	4	×	×	✓	✓	✓	×	78.73	81.76	76.97
	5	×	×	✓	✓	✓	✓	96.07	97.32	95.74
	6	✓	✓	×	✓	✓	✓	**99.53**	**98.79**	**99.49**

training classifier stage, the broad extension method is a highly discriminative spatial extractor. In the fifth case, the combinations, i.e., 'SF+ BEF +SVM+EPF', where EPF is adopted, further refine the classification maps produced from the fourth case. For the three datasets, the classification improves by over 8% in OA and κ compared to the fourth scenario. This indicates that performing EPF operations in the post-processing stage is quite advantageous to improve the recognition accuracy of the classifier.

Finally, we assess the classification performance of the proposed BKB_BES_EPF framework and examine the resulting improvements across all datasets. As shown in Table 7, performance benefits obtained by BKB_BES_EPF are noticeable in all of the discussed scenarios, particularly on the IP and HU scenes; for example, compared to the first case, the differences in terms of OA, AA, and κ are 33.78%, 33.41%, and 38.60%, as well as 22.80%, 20.96%, and 24.67%, respectively. Thus, it emphasizes the significance of improving separability between distinct land cover groups, incorporating broad expansion features, and revising classification probability maps.

4.4. Time cost comparison

This study utilizes deep learning on the PyTorch 1.9 framework with GPU support, while other experiments are conducted in MATLAB 2016a on a system with an AMD Ryzen 7 7745HX processor, Radeon Graphics @ 3.60 GHz, and 16 GB of RAM. Table 8 displays the processing times for the machine learning methodologies on the three datasets.

As illustrated, SVM and BLS have the quickest processing times, relying solely on spectral features, while models like LPP_LBP_ BLS, SSSKRR, and MSF-PCs require more time owing to the reduction of dimensionality and the extraction of spatial-spectral characteristics. The BES_SVM model takes slightly longer than BLS due to the calculation of broad expansion features combined with the spectral features. Similarly,BKAB_BLS consumes more time than BLS due to the additional BKAB pre-processing and BKA optimization step, where the parameter p is set to 0.9 to ensure an optimal solution, although this increases computational costs and execution time.

BKB_BES_EPF requires even more time than BKAB_BLS due to conduct the EPF in the post-processing stage, which extracts spatial contextual information without GPU acceleration. In classification tasks, accuracy and execution time are critical. A trade-off between these factors is acceptable if it significantly improves accuracy. For instance, as shown in Tables 3-4, BKAB_BLS achieves 99.11% OA on the IP dataset, whereas BLS achieves only 64.33% OA. Similarly, BKAB_BLS achieves 99.99% OA on the UP dataset, compared to BLS's 77.59% OA.

4.5. Impact of Training Sample Proportion

Figure 11 illustrates the impact of various training samples on all discussed techniques. Randomly selected training data for the IP dataset (2%, 4%, 6%, 8%, and 10% labeled samples) and the UP and HU datasets (1%, 2%, 3%, 4%, and 5%) are considered in light of the stability and robustness of various approaches. The averages of ten runs are used to report the classification accuracies. As seen in Figure 11, the general trend of the curve for each approach rises as the number of training samples increases. In detail, SVM and BLS continue to perform poorly with varying numbers of training samples on all three data sets. It is clear that the spatial-spectral-based technique consistently performs far better in classification than both SVM and BLS. As expected, the suggested BKB_BES_EPF technique outperforms previous approaches on all training samples, with particularly substantial differences under small sample conditions, demonstrating the proposed method's robustness.
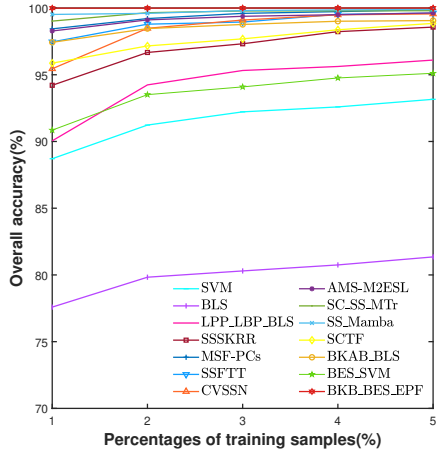
5. Conclusion

The HSI classification task faces two insurmountable challenges: the noise interference problem and the pixel-based classification phenomenon of diverse objects with the same spectrum. In this study, we present an HSI classification architecture based on the
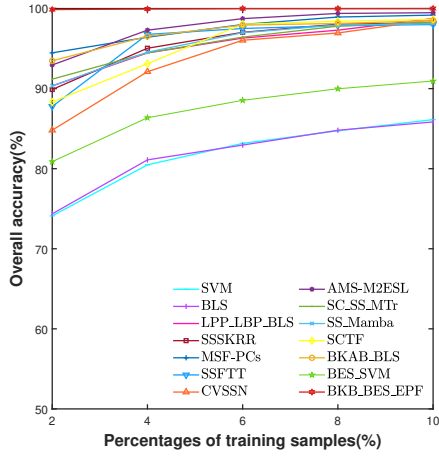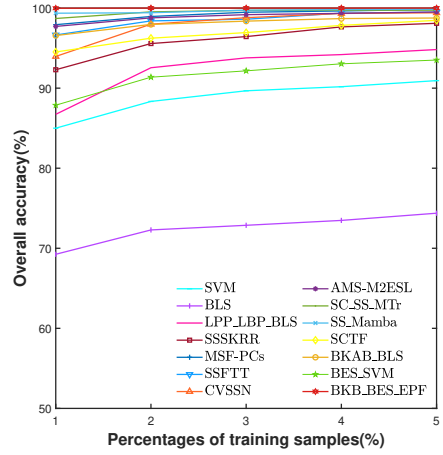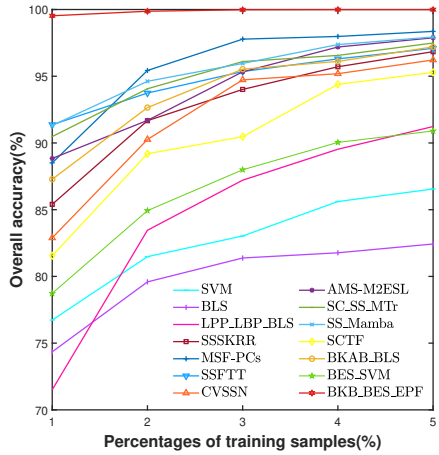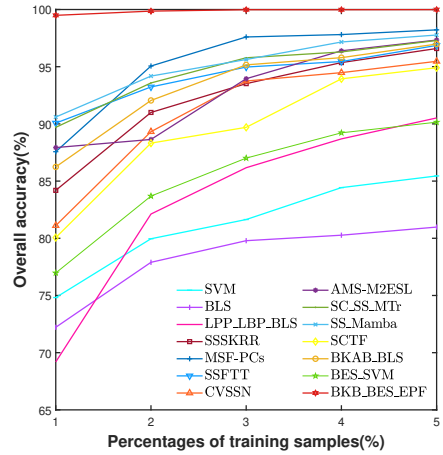
(a)
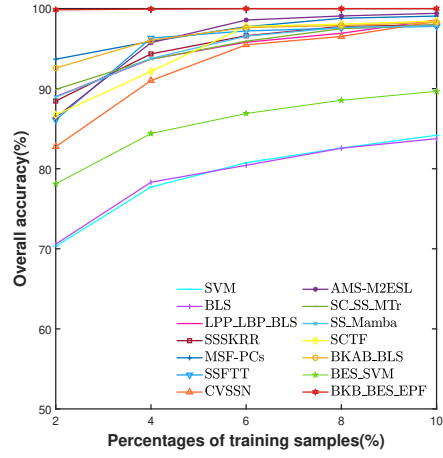
(b)

(c)

(d)

(e)

(f)

Figure 11. Classification results of different methods on IP, UP, and HU datasets with varying amounts of training samples. (a) OA on IP dataset. (b) κ on IP dataset. (c) OA on UP dataset. (d) κ on UP dataset. (e) OA on HU dataset. (f) κ on HU dataset.

Table 8. Execution time in minutes (min) for all approaches on three datasets.

Datasets	Methods							
	SVM	BLS	LPP_LBP_BLS	SSSKRR	MSF-PCs	BKAB_BLS	BES_SVM	BKB_BES_EPF
IP	0.03	0.01	1.06	0.02	0.23	0.74	0.04	2.35
UP	0.04	0.02	14.71	0.32	2.04	3.91	0.10	4.14
HU	0.07	0.08	2.99	0.14	7.52	17.78	0.11	18.01

BKB_BES_EPF with restricted training data. This method naturally incorporates a self-heuristic optimization algorithm, edge-preserving filtering technology, and a broad expansion strategy.

The BKAB module is originally presented to reduce noise and improve the separability of various land cover categories. The BKAB features are then passed into the BLS, which extracts the broad extension features. Furthermore, the original BKAB features and the derived broad extension features are combined to train the SVM, resulting in the probabilistic classification maps. Finally, the EPF approach is used to change the classification maps by integrating contextually pertinent features for each pixel. Comprehensive experimental results show that, when compared to other state-of-the-art techniques, the suggested BKB_BES_EPF significantly enhances the quality of the original HSIs and extracts discriminant spectral-spatial features to achieve exceptional classification outcomes with limited labeled samples on three well-known HSI datasets. In other words, only 1% of the dataset is utilized as the training set for the suggested method and other cutting-edge techniques, and obtaining comparable classification accuracy with fewer training samples is also well addressed.

Data Availability Statement

The data that support the findings of this study are openly available at the following URL:https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes.

Conflicts of interest

No potential conflict of interest was reported by the authors.

References

[1] Cody Weber, Rahim Aguejdad, Xavier Briottet, J. Avala, Sophie Fabre, Jean Demuynck, Emmanuel Zenou, Yannick Deville, Moussa Sofiane Karoui, Fatima Zohra Benhalouche, Sébastien Gadal, Walid Ouerghemmi, Clément Mallet, Arnaud Le Bris, and Nesrine Chehata. Hyperspectral imagery for environmental urban planning, 2018.

[2] Jun Sun, Yan Cao, Xin Zhou, Minmin Wu, Yidan Sun, and Yinghui Hu. Detection for lead pollution level of lettuce leaves based on deep belief network combined with hyperspectral image technology. *Journal of Food Safety*, 41(1):e12866, 2021.

[3] Corinna Vehlow, Thomas Reinhardt, and Daniel Weiskopf. Visualizing fuzzy overlapping communities in networks. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2486–2495, 2013.

[4] Xiaoyi Liao, Guiping Liao, and Linyu Xiao. Rapeseed storage quality detection using hyperspectral image technology-an application for future smart cities. *Journal of Testing and Evaluation*, 51(3):1740–1752, 2023.

[5] Sima Peyghambari and Yun Zhang. Hyperspectral remote sensing in lithological mapping, mineral exploration, and environmental geology: an updated review. *Journal of Applied Remote Sensing*, 15(3):031501–031501, 2021.

[6] Ahmad W Bitar, Loong-Fah Cheong, and Jean-Philippe Ovarlez. Sparse and low-rank matrix decomposition for automatic target detection in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5239–5251, 2019.

[7] Mathieu Fauvel, Yuliya Tarabalka, Jon Atli Benediktsson, Jocelyn Chanussot, and James C Tilton. Advances in spectral-spatial classification of hyperspectral images. *Proceedings of the IEEE*, 101(3):652–675, 2012.

[8] Dibyajyoti Chutia, D. K. Bhattacharyya, Kandarpa Kumar Sarma, Ranjan Kalita, and Singuluri Sudhakar. Hyperspectral remote sensing classifications: A perspective survey. *Trans. GIS*, 20(4):463–490, 2016.

[9] Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42(8):1778–1790, 2004.

[10] Jisoo Ham, Yangchi Chen, Melba M Crawford, and Joydeep Ghosh. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3):492–501, 2005.

[11] Jun Li, José M Bioucas-Dias, and Antonio Plaza. Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 48(11):4085–4098, 2010.

[12] Pavel V Kolesnichenko, Qianhui Zhang, Changxi Zheng, Michael S Fuhrer, and Jeffrey A Davis. Multidimensional analysis of excitonic spectra of monolayers of tungsten disulphide: toward computer-aided identification of structural and environmental perturbations of 2d materials. *Machine Learning: Science and Technology*, 2(2):025021, 2021.

[13] Giorgio Licciardi, Prashanth Reddy Marpu, Jocelyn Chanussot, and Jon Atli Benediktsson. Linear versus nonlinear pca for the classification of hyperspectral data based on the extended morphological profiles. *IEEE Geoscience and Remote Sensing Letters*, 9(3):447–451, 2011.

[14] Alberto Villa, Jón Atli Benediktsson, Jocelyn Chanussot, and Christian Jutten. Hyperspectral image classification with independent component discriminant analysis. *IEEE Trans. Geosci. Remote. Sens.*, 49(12):4865–4876, 2011.

[15] Aloke Datta, Susmita Ghosh, and Ashish Ghosh. Supervised feature extraction of hyperspectral images using partitioned maximum margin criterion. *IEEE Geosci. Remote. Sens. Lett.*, 14(1):82–86, 2017.

[16] Tatyana V Bandos, Lorenzo Bruzzone, and Gustavo Camps-Valls. Classification of hyperspectral images with regularized linear discriminant analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 47(3):862–873, 2009.

[17] Le Sun, Guangrui Zhao, Yuhui Zheng, and Zebin Wu. Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2022.

[18] Cheng Deng, Xianglong Liu, Chao Li, and Dacheng Tao. Active multi-kernel domain adaptation for hyperspectral image classification. *Pattern Recognition*, 77:306–315, 2018.

[19] Jie Liang, Jun Zhou, and Yongsheng Gao. Tensor morphological profile for hyperspectral image classification. In *2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016*, pages 2197–2201. IEEE, 2016.

[20] Benqin Song, Jun Li, Peijun Li, and Antonio Plaza. Decision fusion based on extended multi-

attribute profiles for hyperspectral image classification. In *5th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, WHISPERS 2013, Gainesville, FL, USA, June 26-28, 2013*, pages 1–4. IEEE, 2013.

[21] Sen Jia, Linlin Shen, and Qingquan Li. Gabor feature-based collaborative representation for hyperspectral imagery classification. *IEEE Transactions on Geoscience and Remote Sensing*, 53(2):1118–1129, 2014.

[22] Xudong Kang, Xuanlin Xiang, Shutao Li, and Jon Atli Benediktsson. Pca-based edge-preserving features for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12):7140–7151, 2017.

[23] Denis Uchaev and Dmitry Uchaev. Small sample hyperspectral image classification based on the random patches network and recursive filtering. *Sensors*, 23(5):2499, 2023.

[24] Bushra Naz Soomro, Liang Xiao, Muhammad Moazzam Jawaid, Shahzad Hyder Soomro, and Naveed Ahmed Jaffari. A bilateral filter based post-processing approach for supervised spectral-spatial hyperspectral image classification. *Mehran University Research Journal of Engineering & Technology*, 37(3):615–630, 2018.

[25] Xudong Kang, Shutao Li, and Jon Atli Benediktsson. Spectral–spatial hyperspectral image classification with edge-preserving filtering. *IEEE transactions on geoscience and remote sensing*, 52(5):2666–2677, 2013.

[26] Jaime Zabalza, Jinchang Ren, Zheng Wang, Stephen Marshall, and Jun Wang. Singular spectrum analysis for effective feature extraction in hyperspectral imaging. *IEEE Geosci. Remote. Sens. Lett.*, 11(11):1886–1890, 2014.

[27] Jaime Zabalza, Jinchang Ren, Zheng Wang, Huimin Zhao, Jun Wang, and Stephen Marshall. Fast implementation of singular spectrum analysis for effective feature extraction in hyperspectral imaging. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 8(6):2845–2853, 2015.

[28] Hang Fu, Genyun Sun, Jinchang Ren, Aizhu Zhang, and Xiuping Jia. Fusion of pca and segmented-pca domain multiscale 2-d-ssa for effective spectral-spatial feature extraction and data classification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2020.

[29] Xiangyong Cao, Zongben Xu, and Deyu Meng. Spectral-spatial hyperspectral image classification via robust low-rank feature extraction and markov random field. *Remote Sensing*, 11(13):1565, 2019.

[30] Xianghai Cao, Xiaozhen Wang, Da Wang, Jing Zhao, and Licheng Jiao. Spectral–spatial hyperspectral image classification using cascaded markov random fields. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(12):4861–4872, 2019.

[31] Guixin Zhao, Xuesong Wang, and Yuhu Cheng. Hyperspectral image classification based on local binary pattern and broad learning system. *International Journal of Remote Sensing*, 41(24):9393–9417, 2020.

[32] You Ma, Zhi Liu, and C. L. Philip Chen Chen Chen. Hybrid spatial-spectral feature in broad learning system for hyperspectral image classification. *Applied Intelligence*, 52(3):2801–2812, 2022.

[33] Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected topics in applied earth observations and remote sensing*, 7(6):2094–2107, 2014.

[34] Ronghua Shang, Keyao Zhu, Huidong Chang, Weitong Zhang, Jie Feng, and Songhua Xu. Hyperspectral image classification based on mixed similarity graph convolutional network and pixel refinement. *Applied Soft Computing*, 170:112657, 2025.

[35] Prince Yaw Owusu Amoako, Guo Cao, Boshan Shi, Di Yang, and Benedict Boakye Acka. Orthogonal capsule network with meta-reinforcement learning for small sample hyperspectral image classification. *Remote Sensing*, 17(2):215, 2025.

[36] Xiangyong Cao, Jing Yao, Zongben Xu, and Deyu Meng. Hyperspectral image classification with convolutional neural network and active learning. *IEEE Transactions on Geoscience and Remote*

Sensing, 58(7):4604–4616, 2020.

[37] Swalpa Kumar Roy, Gopal Krishna, Shiv Ram Dubey, and Bidyut B Chaudhuri. Hybridsn: Exploring 3-d–2-d cnn feature hierarchy for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 17(2):277–281, 2019.

[38] Yimin Xu, Zhaokui Li, Wei Li, Qian Du, Cuiwei Liu, Zhuoqun Fang, and Lin Zhai. Dual-channel residual network for hyperspectral image classification with noisy labels. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2021.

[39] Chao Ma, Minjie Wan, Jian Wu, Xiaofang Kong, Ajun Shao, Fan Wang, Qian Chen, and Guohua Gu. Light self-gaussian-attention vision transformer for hyperspectral image classification. *IEEE Trans. Instrum. Meas.*, 72:1–12, 2023.

[40] Weiliang Huang, Wenxuan He, Shuhong Liao, Zhen Xu, and Jingwen Yan. Efficient spectralformer for hyperspectral image classification. *Digit. Signal Process.*, 143:104237, 2023.

[41] Weidong Zhang, Gongchao Chen, Peixian Zhuang, Wenyi Zhao, and Ling Zhou. Catnet: Cascaded attention transformer network for marine species image classification. *Expert Syst. Appl.*, 256:124932, 2024.

[42] Weidong Zhang, Zexu Li, Guohou Li, Peixian Zhuang, Guojia Hou, Qiang Zhang, and Chongyi Li. Gacnet: Generate adversarial-driven cross-aware network for hyperspectral wheat variety identification. *IEEE Trans. Geosci. Remote. Sens.*, 62:1–14, 2024.

[43] Weidong Zhang, Zexu Li, Hai-Han Sun, Qiang Zhang, Peixian Zhuang, and Chongyi Li. Sstnet: Spatial, spectral, and texture aware attention network using hyperspectral image for corn variety identification. *IEEE Geosci. Remote. Sens. Lett.*, 19:1–5, 2022.

[44] Danfeng Hong, Zhu Han, Jing Yao, Lianru Gao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Spectralformer: Rethinking hyperspectral image classification with transformers. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2021.

[45] Baisen Liu, Yuanjia Liu, Wulin Zhang, Yiran Tian, and Weili Kong. Spectral swin transformer network for hyperspectral image classification. *Remote Sensing*, 15(15):3721, 2023.

[46] Jie Sun, Jing Yang, Wang Chen, Sujie Ding, Shaobo Li, and Jianjun Hu. Lctcs: Low-cost and two-channel sparse network for hyperspectral image classification. *Neural Processing Letters*, 56(3):181, 2024.

[47] Jing Yao, Danfeng Hong, Chenyu Li, and Jocelyn Chanussot. Spectralmamba: Efficient mamba for hyperspectral image classification. *CoRR*, abs/2404.08489, 2024.

[48] Yan He, Bing Tu, Bo Liu, Jun Li, and Antonio Plaza. 3dss-mamba: 3d-spectral-spatial mamba for hyperspectral image classification. *IEEE Trans. Geosci. Remote. Sens.*, 62:1–16, 2024.

[49] Jun Wang, Wen-chuan Wang, Xiao-xue Hu, Lin Qiu, and Hong-fei Zang. Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems. *Artificial Intelligence Review*, 57(4):98, 2024.

[50] Philippe Thévenaz, Daniel Sage, and Michael Unser. Bi-exponential edge-preserving smoother. *IEEE transactions on image processing*, 21(9):3924–3936, 2012.

[51] C. L. Philip Chen and Zhulin Liu. Broad learning system: An effective and efficient incremental learning system without the need for deep architecture. *IEEE Trans. Neural Networks Learn. Syst.*, 29(1):10–24, 2018.

[52] Chunhui Zhao, Wu Liu, Yan Xu, and Jinhuan Wen. Hyperspectral image classification via spectral–spatial shared kernel ridge regression. *IEEE Geoscience and Remote Sensing Letters*, 16(12):1874–1878, 2019.

[53] Hang Fu, Genyun Sun, Jinchang Ren, Aizhu Zhang, and Xiuping Jia. Fusion of pca and segmented-pca domain multiscale 2-d-ssa for effective spectral-spatial feature extraction and data classification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2020.

[54] Mingsong Li, Yikun Liu, Guangkuo Xue, Yuwen Huang, and Gongping Yang. Exploring the relationship between center and neighborhoods: Central vector oriented self-similarity network for hyperspectral image classification. *IEEE Transactions on Circuits and Systems for Video*

Technology, 33(4):1979–1993, 2022.

[55] Mingsong Li, Wei Li, Yikun Liu, Yuwen Huang, and Gongping Yang. Adaptive mask sampling and manifold to euclidean subspace learning with distance covariance representation for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–18, 2023.

[56] Lingbo Huang, Yushi Chen, and Xin He. Spectral-spatial masked transformer with supervised and contrastive learning for hyperspectral image classification. *IEEE Trans. Geosci. Remote. Sens.*, 61:1–18, 2023.

[57] Lingbo Huang, Yushi Chen, and Xin He. Spectral-spatial mamba for hyperspectral image classification. *Remote. Sens.*, 16(13):2449, 2024.

[58] Peng Chen, Wenxuan He, Feng Qian, Guangyao Shi, and Jingwen Yan. A synergistic cnn-transformer network with pooling attention fusion for hyperspectral image classification. *Digit. Signal Process.*, 160:105070, 2025.