Guided Filter of Random Patches Network and Relaxed-Collaborative-Representation-Based Hyperspectral Image Classification

Tugcan Dundar and Taner Ince, *Member, IEEE*

***Abstract*—Feature extraction and accurate classification are crucial tasks in the land-cover classification of the hyperspectral image (HSI). We propose a guided filter (GF) of a random patches network (RPNet) and a relaxed collaborative representation (RCR)-based HSI classification (HSIC) method called GRR. The shallow and deep features are extracted using RPNet that requires no pretraining stage. In addition to the obtained feature set, the original HSI and extracted features are then filtered by GF to preserve the edge details. After that, all the distinct feature sets are separately concatenated with the original HSI to keep the original structure of the data. The high-dimensional feature sets are then processed by a linear discriminant analysis (LDA) to increase class separability and to select the most representative features. Since few train samples are available in the HSIC task, the efficiency of LDA is improved using superpixel segmentation to generate pseudosamples. In the final stage, the reduced-dimension feature sets are classified by the use of superpixel-guided RCR, which utilizes the resemblance and discrimination of the feature sets efficiently. The extensive experiments on the real HSIs are carried out to validate the efficacy of the proposed method.**

***Index Terms*—Guided filter (GF), hyperspectral image classification (HSIC), random patches network (RPNet), relaxed collaborative representation (RCR).**

I. Introduction

HYPERSPECTRAL remote sensing is a measurement technique that collects data from the land surface over a broad spectrum [1]. The availability of rich spectral information allows the identification of materials. Therefore, it is used in many areas such as mineral detection [2], [3], [4], target detection [5], [6], [7], classification of land surfaces [8], [9], [10], [11], etc.

Hyperspectral image classification (HSIC) methods can be categorized into spectral and spectral–spatial classifiers. Spectral classifiers utilize spectral information of a single pixel. Support vector machines (SVM) [12], sparse representation classifier (SRC) [13], and collaborative representation classifier (CRC) [14] are popular spectral-based classifiers. While only one single-pixel information is used in spectral classifiers, neighborhood information of a pixel is also utilized in spectral–spatial classifiers [15]. SRC assumes the test sample is a linear combination of all train samples. However, CRC formulates the test sample as a collaboration of all train data. Wei et al. [14] indicated that the collaborative representation of training samples is as important as sparsity. Therefore, several CRC-based HSIC methods have been proposed [16], [17], [18], [19], [20]. Li et al. [16] use a distance-weighted Tikhonov regularization to measure the similarity between the test sample and atoms in the dictionary and, then, couple it to a CRC-based nearest-regularized subspace (NRS) classifier. The joint collaborative representation (JCR) version of NRS is proposed in [17] by averaging the spectra of the neighborhood of the test sample and the spectrum of training examples in each class. A weighted JCR (WJCR) is proposed in [18] by assigning appropriate weights to the neighbor of the test pixel. Chen et al. [19] proposed a weighted regularized collaborative representation optimized classifier, which combines the advantages of CRC, NRS, and WJCR. Shen et al. [20] calculate class-specific group weights of train samples to reveal the impact of different classes and proposed a grouped CRC algorithm.

Not only spectral and spatial features but also different features are extracted then followed by a CRC-based classification. 3-D discrete wavelet transform [21] is applied in [22] to the original hyperspectral image (HSI) to extract features, and then, these are classified using CRC. Jia et al. [23] employ 3-D Gabor features [24] and classify the features using CRC. Yang and Qian [25] utilized multiple features such as the spectral value feature (SVF), extended multiattribute profile (EMAP) [26], and Gabor features to represent HSI better than the raw data. Then, both the residuals of SRC and CRC were combined to classify a given test sample. In [27], extracted EMAP features were filtered by the predefined 3-D Gabor filters, and the resulting features were classified with CRC. The obtained pixelwise classification map was regularized using multiscale superpixels constituted from the EMAP features. Liu et al. [28] proposed the Tikhonov regularized CRC-based multitask correlation adaptive representation classifier using local binary pattern (LBP) [29], Gabor, differential morphological profiles (DMP) [30], and SVF. The designed optimization problem was solved for each modality and the label of a given sample was assigned to the class with minimum residual calculated over all modalities. All of

Manuscript received 14 December 2023; revised 21 January 2024 and 23 February 2024; accepted 3 March 2024. Date of publication 5 March 2024; date of current version 18 March 2024. *(Corresponding author: Tugcan Dundar.)*

The authors are with the Department of Electrical and Electronics Engineering, Gaziantep University, 27310 Gaziantep, Türkiye (e-mail: tugcandundar@gantep.edu.tr; tanerince@gantep.edu.tr).

Digital Object Identifier 10.1109/JSTARS.2024.3373600

these methods transform the original feature space into another space and find an optimal representation between test and train samples. Meanwhile, the distinct contribution of each feature to the classification stage is ignored due to the classical CRC-based representation.

In order to overcome the aforementioned issues, the CRC-based method called relaxed collaborative representation (RCR) [31] has been introduced. The objective function of RCR is to utilize the resemblance and discrimination of extracted features and; in addition, it finds an optimal solution for the representation coefficients at the same time. RCR has been used in anomaly detection [32], [33] and classification [34], [35], [36] problems due to the efficient usage of the extracted features. Li et al. [34] extracted different features such as SVF, spectral gradient feature [37], Gabor, and DMP features from the HSI. In order to use the similarities and distinctiveness of the features, the RCR-based classification called JCR with multitask learning (JCRC-MTL) was performed. In addition, the contextual information of a test pixel was considered using a fixed-size square patch to increase the classification results of the pixel-based RCR. In [35], multifeature-superpixel-based RCR with band weighting (MSRCR-BW) using EMAP, LBP, and Gabor features was proposed. Each spectral band of HSI was considered as an image and each image was segmented into the same superpixel regions. Instead of a random selection of train samples, the dictionary was composed of the samples selected from each superpixel. Then, the extracted feature set of each superpixel was classified with RCR. In addition to the feature weighting, HSI bands were also weighted by RCR in the MSRCR-BW method. Gao et al. [36] proposed region-based self-balancing dictionary learning for RCR (RSBDL-RCR), which learns a comprehensive dictionary at first and classifies the samples with RCR using a superpixel segmentation strategy. Instead of calculating the residuals of RCR, two new indicators called self-contribution representing the discrimination capability and average contribution indicating the significance of each class were used to assign a class label. The proposed RCR-based JCRC-MTL, MSRCR-BW, and RSBDL-RCR HSIC methods give satisfactory classification results. However, the methods use feature extraction algorithms, such as LBP, Gabor, DMP, and EMAP. Each has different parameters that need to be tuned to obtain the features better. After parameter adjustments are fulfilled, features need to be extracted by calculating them separately, which increases the computation time of the proposed methods considerably. In addition, JCRC-MTL creates a fixed-size patch around each test sample to obtain neighborhood information, which may increase the computation time and may cause misclassification near the class boundaries. In MSRCR-BW, the dictionary is generated by selecting samples from each superpixel. If the train samples are randomly selected from the data, some superpixels may not contain samples. Additionally, since each spectral band of HSI is treated as a single image and band-by-band representation coefficients are calculated with RCR, the high dimension of the dataset negatively affects the computation time. Due to the dictionary learning stage of RSBDL-RCR, it imposes quite a computational burden.

In recent years, a fast and efficient deep-learning-based HSIC method called random patches network (RPNet) [38] requiring no pretraining stage has been proposed. RPNet selects random convolution kernels from HSI and convolves the HSI with the chosen kernels. After repeating the process up to the predefined number of layers, both shallow and deep features of HSI are extracted in simple, fast, and efficient way. Then, all the features are stacked together and classified by SVM. Over the years, different RPNet-based HSIC methods have been proposed. In random patches convolution and local covariance (RPCC) [39], the dimension of HSI was reduced by a maximum noise fraction (MNF) [40]. Then, RPNet was used to extract spatial features and local covariance matrices were utilized to extract spectral features. The obtained feature sets were concatenated and classified with SVM. Cheng et al. [41] presented spectral–spatial RPNet (SSRPNet), which extracts LBP features on the dimension-reduced HSI and stacks the extracted features with HSI bands. After that, the dimension was again reduced with principal component analysis (PCA) [42] and fed into RPNet. Then, all the obtained spectral–spatial features were classified by a graph-based learning method. Shenming et al. [43] proposed the Gabor filter with random patch convolution (GRPC), which applies PCA, linear discriminant analysis (LDA) [44], and Gabor filter. The dimension-reduced and filtered HSI was processed by RPNet; then, both the output of the Gabor filter and RPNet were stacked to be classified with SVM. RPNet with the recursive filtering (RPNET-RF) [45] method was proposed to improve the result of RPNet using RF, which is an edge-preserving filter. The dimension of RPNet feature was reduced and filtered by PCA and RF, respectively. The filtered output was joined with the original HSI and the combined features were assigned labels with SVM. Multiscale superpixelwise RPNet (MSRPNet) was proposed in [46], which employs 2-D singular spectrum analysis (2D-SSA) [47] for noise elimination and spectral—spatial and superpixelwise PCA (S^3-PCA) [48] for acquiring local and global features. Then, RPNet was used to obtain shallow and deep features of the output of S^3-PCA. In the last step, RPNet and raw HSI features were gathered and classified with SVM. RPNet and its variants provide promising performance in terms of accuracy and computation time. Since the raw HSI spectral bands and obtained features are concatenated in the RPNet-based methods, the resultant feature matrix has high dimensions, which adversely affect the classification [49]. In addition, the similarity and diversity of the extracted features are not efficiently utilized due to the SVM.

Motivated by the success of RPNet and RCR, a novel HSIC method called GRR is proposed. It unifies the favorable aspects of RPNet, which extracts shallow and deep features in the HSI, and RCR, which is capable of the classification of the feature sets efficiently. In addition, the proposed method takes into account the spatial structure of the HSI by the use of GF and superpixel segmentation. Each available RCR-based HSIC methods utilize different conventional feature extraction algorithms that require many mathematical formulations to compute and the selection of the right feature extraction algorithm is also a challenging task. Moreover, RPNet-based HSIC methods use solely SVM to classify HSI, which ignores the relation between the features

and spatial structure of the HSI. In order to cope with the aforementioned issues, the proposed GRR combines the superior aspects of RPNet and RCR and then enhances the classification performance with additional stages such as guided filtering of the features, superpixel-guided pseudosample generation, and superpixel-based classification. The stages of the proposed GRR can be summarized as follows. First, PCA is applied to reduce the dimension of HSI, which is next processed by RPNet to extract shallow and deep features. Then, RPNet features and raw HSI features are filtered by GF to retain the sharp edges of the features, which improves the accuracy [50], [51], [52], [53]. At the end of this stage, unique feature sets are obtained as namely; RPNet, GF of RPNet, and GF of raw HSI. In order to incorporate the original structure of the data, the raw HSI is separately concatenated with these feature sets. Then, the high dimensionality of the obtained feature sets is reduced by LDA, which provides the selection of more representative features and the facility of more separable classes. Since LDA is a supervised method and its performance is directly related to the plenty of train samples, superpixel-guided pseudosample generation is utilized to increase the available samples before the LDA process. Finally, all the dimension-reduced feature sets are classified with RCR using the segmented HSI. The superpixel-guided classification enables the use of spatial structures and fast classification compared to the pixelwise classification.

The contributions of the proposed method can be listed as follows.

1) A simple and fast HSI feature extraction method is developed to explore the shallow and deep features efficiently, which is computationally less expensive compared to traditional methods.
2) Sharp feature edge details are preserved in the process by means of GF on both raw HSI and RPNet features.
3) Separate concatenation of the extracted features with the raw HSI allows us to maintain the original structure of HSI inside each feature set.
4) Superpixel-guided pseudosample generation approach for LDA provides to identify the samples correctly even if the number of available train samples is low.
5) RCR-based classification on the segmented HSI assures the efficient use of the extracted features, spatial neighborhood information, and computation time.

The rest of this article is organized as follows. Section II describes and formulates the stages of the proposed GRR. The experimental setup and results are presented in Section III. Finally, Section IV concludes this article.

II. GF of RPNet and RCR

The proposed GRR starts with the extraction of features using RPNet. Then, these features and original HSI features are separately filtered using GF to retain the edge details in the features. Since HSIC suffers from the limited training samples, superpixel-guided pseudosamples are generated to overcome this problem. These pseudosamples are processed using LDA to reduce the high dimension of each feature set and increase the class discrimination capability. Finally, the superpixel-guided RCR classification is performed to predict the class labels. A visual representation of the proposed GRR is presented in Fig. 1. The following sections describe and formulate each stage of the GRR in detail.

A. RPNet Feature Extraction

RPNet differs from the conventional deep-learning-based methods [54], [55], [56] since it does not require a pretraining stage and many train samples. It randomly selects convolution kernels from the HSI and convolves the whole HSI with a cascaded structure containing dimension reduction, whitening, patch extraction, convolution, and activation operations, respectively. In order to decrease computation time and increase the effectiveness of the convolution operation, RPNet first reduces the dimension using PCA, then a whitening operation is performed to ensure that the variances of each spectral band are similar and the correlation between the spectral channels are minimized [57]. Then, random t pixels are chosen from the whitened data and t convolution kernels are created using each selected pixel as center. Afterward, the whitened data are convolved with the obtained kernels to extract t feature maps. Finally, the extracted features are fed into the activation operator called rectified linear units (ReLU). RPNet reimplements all the processes throughout the predefined number of layers.

Let $\mathcal{H} \in \mathbb{R}^{r\times c\times b}$ be the 3-D representation of HSI where r and c are row and column size, respectively, and b is the spectral size. Let $\mathcal{H}_P \in \mathbb{R}^{r\times c\times h}$ represent the first h principal components of $\mathcal{H}$ computed by PCA and the result of whitening operation on $\mathcal{H}_P$ is denoted as $\mathcal{H}_W \in \mathbb{R}^{r\times c\times h}$. The convolution operation using convolution kernels $\mathcal{C}_{1,2,\ldots,t} \in \mathbb{R}^{w\times w\times h}$ with a patch size of w is formulated as

$$\mathbf{N}_j = \sum_{i=1}^{h} \mathcal{H}_W^i * \mathcal{C}_j^i \quad ; \quad j = 1, 2, \ldots, t \tag{1}$$

where $*$ is the 2-D convolution operator, $\mathbf{N}_j \in \mathbb{R}^{r\times c}$ represents the jth feature matrix, $\mathcal{H}_W^i \in \mathbb{R}^{r\times c}$ denotes the ith dimension of $\mathcal{H}_W$, and $\mathcal{C}_j^i$ indicates the jth patch of dimension i. We concatenate all $\mathbf{N}_j$ matrices into $\mathcal{N} \in \mathbb{R}^{r\times c\times t}$ to include all features in each layer. In order to advance sparsity in $\mathcal{N}$, ReLU is applied, and then, the features $\mathcal{U}^1 \in \mathbb{R}^{r\times c\times t}$ of the first layer are extracted as

$$\mathcal{U}^1 = \max(0, \mathcal{N} - \mathcal{M}) \tag{2}$$

where $\mathcal{M} \in \mathbb{R}^{r\times c\times t}$ is a 3-D matrix calculated as the mean of $\mathcal{N}$ and duplicated t times in the third dimension as

$$\mathcal{M}(:,:,j) = \frac{1}{t}\sum_{j=1}^{t} \mathcal{N}(:,:,j), \quad j = 1, 2, \ldots, t. \tag{3}$$

The feature $\mathcal{U}^1$ represents the output of the first layer of RPNet and, thus, holds the low-level features. To acquire the high-level features, the output of the previous layer is given as input to the next layer. Let $\mathcal{U}^{(\ell-1)}$ denote the extracted feature set of layer $(\ell - 1)$ and be the input for layer ℓ. In this case, only the input $\mathcal{H}$ of the first layer is replaced by $\mathcal{U}^{(\ell-1)}$ for the subsequent layers and all the calculations starting from PCA are repeated

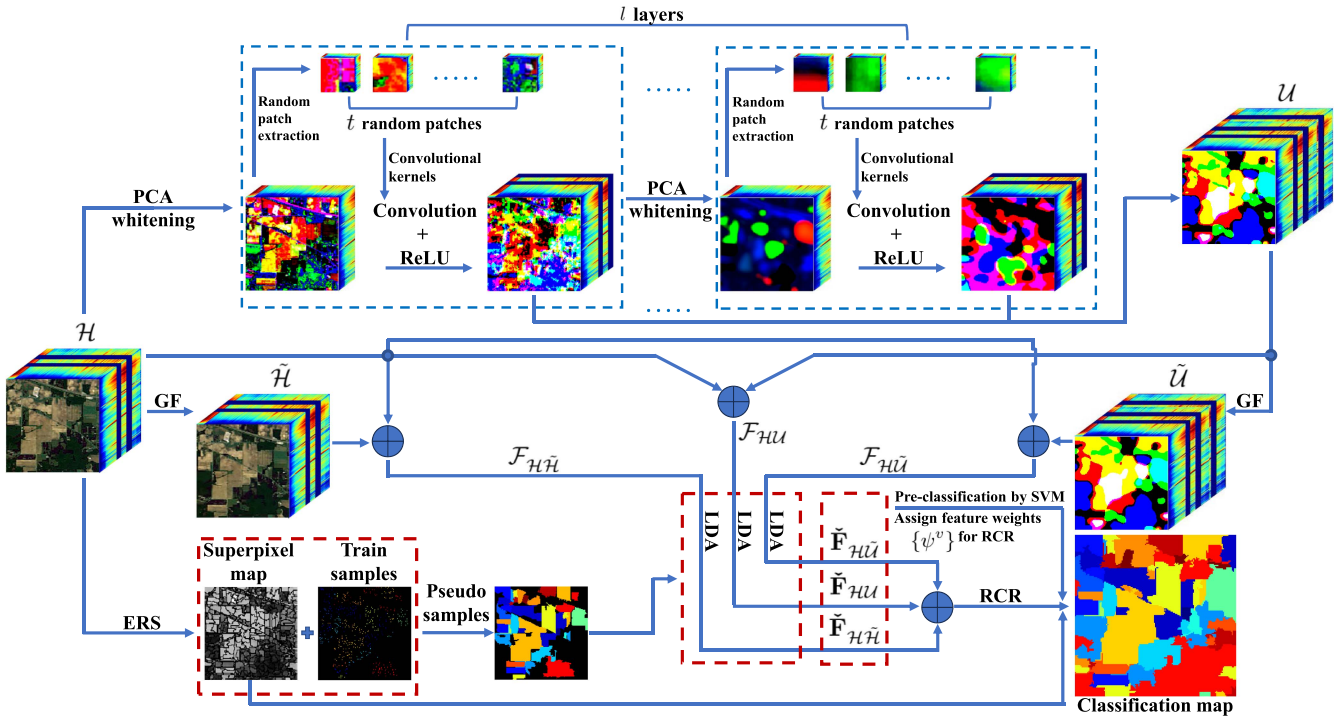

Fig. 1. Flowchart of the proposed GRR.

until the predefined number of layers l is reached. Hence, both the shallow and deep features $\mathcal{U}^{1,2,\ldots,l}$ are extracted from the layers and stacked into the $\mathcal{U} = \{\mathcal{U}^1, \mathcal{U}^2, \ldots, \mathcal{U}^l\} \in \mathbb{R}^{r\times c\times tl}$.

B. Guided Filter

GF is a key step for extracting and emphasizing edge details in an image. The main principle of GF is to obtain more structural filter output compared to the input by using a guidance image. The guidance image can be either a gray-level or a colorful image. However, the first three principal components of PCA are usually chosen as a guidance image [51], [52] in HSIC because local linearity in the color image is more effective than the gray image [50]. In addition, it is required to apply GF to each spectral band separately when the input has more than a single channel as in the case of color image [50].

Since there is a local linear relation between the output and guidance image in a patch ω_z whose center is positioned at the pixel z [50], the local linear transformation by use of color guidance image $\mathcal{G} \in \mathbb{R}^{r\times c\times 3}$ in a local patch is evaluated as follows:

$$q_u = \boldsymbol{\alpha}_z^\top \mathbf{g}_u + \beta_z \quad \forall u \in \omega_z \tag{4}$$

where the scalar $q_u \in \mathbb{R}^{1\times 1}$ is the filter output at the pixel location u in ω_z, $\boldsymbol{\alpha}_z \in \mathbb{R}^{3\times 1}$ and $\beta_z \in \mathbb{R}^{1\times 1}$ are the filter coefficients, $\mathbf{g}_u \in \mathbb{R}^{3\times 1}$ is the color vector of $\mathcal{G}$ at pixel u, and the operand $(\cdot)^\top$ is transpose operation. In order to obtain the coefficients $\boldsymbol{\alpha}_z$ and β_z, the following linear-ridge-regression-based energy function is formed:

$$E(\boldsymbol{\alpha}_z, \beta_z) = \sum_{u\in\omega_z} \left((\boldsymbol{\alpha}_z^\top \mathbf{g}_u + \beta_z - p_u)^2 + \epsilon \boldsymbol{\alpha}_z^2 \right) \tag{5}$$

where the parameter ϵ controls the amount of smoothing and $p_u \in \mathbb{R}^{1\times 1}$ is the pixel value of the input image $\mathbf{P} \in \mathbb{R}^{r\times c}$ at the pixel u. After the optimal coefficients $\boldsymbol{\alpha}_z$ and β_z in a local patch are found, the filter output q_u is easily determined [50]. If the filtering process is repeated for all pixels, the filtered output $\tilde{\mathbf{P}} \in \mathbb{R}^{r\times c}$ of the input $\mathbf{P}$ is obtained.

GRR applies GF on raw HSI and RPNet features simultaneously. Suppose that the guidance images of raw HSI and RPNet features are given as $\mathcal{G}^{\mathcal{H}} \in \mathbb{R}^{r\times c\times 3}$ and $\mathcal{G}^{\mathcal{U}} \in \mathbb{R}^{r\times c\times 3}$, respectively. Since GF is applicable for single-band images, each spectrum of both $\mathcal{H}$ and $\mathcal{U}$ are separately filtered with the help of $\mathcal{G}^{\mathcal{H}}$ and $\mathcal{G}^{\mathcal{U}}$ by following (4) and (5); then, the filtered outputs $\tilde{\mathcal{H}} \in \mathbb{R}^{r\times c\times b}$ and $\tilde{\mathcal{U}} \in \mathbb{R}^{r\times c\times tl}$ are obtained as mentioned earlier.

After the GF process, the features of the proposed GRR are constructed as

$$\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}} = \{\mathcal{H}, \tilde{\mathcal{H}}\} \in \mathbb{R}^{r\times c\times 2b}$$
$$\mathcal{F}_{\mathcal{H}\mathcal{U}} = \{\mathcal{H}, \mathcal{U}\} \in \mathbb{R}^{r\times c\times (b+tl)}$$
$$\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}} = \{\mathcal{H}, \tilde{\mathcal{U}}\} \in \mathbb{R}^{r\times c\times (b+tl)}$$

where the features $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathcal{F}_{\mathcal{H}\mathcal{U}}$, and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$ are obtained by concatenation of the raw HSI $\mathcal{H}$ with the guided filtered HSI $\tilde{\mathcal{H}}$, extracted RPNet features $\mathcal{U}$, and guided filtered RPNet features $\tilde{\mathcal{U}}$, respectively.

C. Pseudosample Generation

Since all obtained feature sets are high-dimensional, LDA is later utilized to reduce the dimension, increase the class-discrimination, and decrease the computation time. The performance of LDA is completely related to the available dictionary

because it is a supervised technique; however, HSIC suffers from the lack of train samples [1]. Therefore, we propose a simple yet efficient superpixel-based temporary pseudosample generation method to alleviate this problem. First, entropy rate superpixel (ERS) segmentation [58] is utilized to segment the HSI into many nonoverlapping spatial regions using the first principal component of $\mathcal{H}$ as the base image. Then, the number of superpixels $K = m/T^2$ is determined where T controls the approximate number of pixels in each segmented region and $m = r \times c$ denotes the number of pixels in the image.

Let n train samples given as $\mathbf{d}_{1,\ldots,n} \in \mathbb{R}^{b\times 1}$ are used to construct training dictionary matrix $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n] \in \mathbb{R}^{b\times n}$ with P classes and the pth class has n_p samples where $n = \sum_{p=1}^{P} n_p$. Suppose $L(\cdot)$ is an operator indicating the label of a given sample or region and $S = (s_1, s_2, \ldots, s_K)$ be the set of superpixels $s_{k=1,2,\ldots,K}$ where each includes many pixels inside. Then the following process is repeated for each s_k as

$$L(s_k) = \begin{cases} \text{mode}(L\{\mathbf{d}_{s_k}\}), & \text{if any } \mathbf{d}_{1,\ldots,n} \in s_k \\ \emptyset, & \text{otherwise} \end{cases} \tag{6}$$

where $L(s_k)$ refers to the label of superpixel s_k and $\{\mathbf{d}_{s_k}\}$ is a set of train samples inside s_k. The null set $\emptyset$ implies that the superpixel s_k does not contain any train samples inside; thus, no label is assigned. The function mode$(L\{\mathbf{d}_{s_k}\})$ counts the train labels inside s_k, if present, then finds the most repeated label inside s_k.

Consequently (6) states that if a superpixel contains train samples, then the label of that superpixel is assigned as the most frequent label of train samples inside; otherwise, the superpixel has no label. With the help of pseudosample generation, the number of training samples is temporarily increased from $\mathbf{D} \in \mathbb{R}^{b\times n}$ to $\hat{\mathbf{D}} \in \mathbb{R}^{b\times \eta}$ ($\eta \gg n$) to implement LDA effectively.

D. Linear Discriminant Analysis

LDA is a supervised technique that projects the high-dimensional data into low-dimensional domain while maximizing class discrimination simultaneously. As it is stated in [59], the main drawback of LDA arises when the available train sample size is small compared to the dimension of the data. This issue makes the solution singular; therefore, LDA cannot be implemented. Since the pseudosample is temporarily generated and, thus, the dictionary is enlarged, we successfully overcome this problem.

Assume that the enlarged dictionary $\hat{\mathbf{D}}$ contains η train samples as $\hat{\mathbf{D}} = [\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \ldots, \hat{\mathbf{d}}_\eta] \in \mathbb{R}^{b\times\eta}$ and each class p has η_p samples, i.e., $\eta = \sum_{p=1}^{P} \eta_p$. The mean of each class $\mathbf{m}_p \in \mathbb{R}^{b\times 1}$ and the mean of all train samples $\mathbf{m} \in \mathbb{R}^{b\times 1}$ are calculated as

$$\mathbf{m}_p = \frac{1}{\eta_p} \sum_{\iota=1}^{\eta_p} \hat{\mathbf{d}}_{p_\iota} \tag{7}$$

$$\mathbf{m} = \frac{1}{P} \sum_{p=1}^{P} \mathbf{m}_p \tag{8}$$

where $\hat{\mathbf{d}}_{p_\iota}$ refers to the ιth sample in the pth class. Then, the intraclass scatter matrix $\mathbf{T}_W \in \mathbb{R}^{b\times b}$ and interclass scatter matrix $\mathbf{T}_B \in \mathbb{R}^{b\times b}$ are calculated as

$$\mathbf{T}_W = \sum_{p=1}^{P} \sum_{\iota=1}^{\eta_p} (\hat{\mathbf{d}}_{p_\iota} - \mathbf{m}_p)(\hat{\mathbf{d}}_{p_\iota} - \mathbf{m}_p)^\top \tag{9}$$

$$\mathbf{T}_B = \sum_{p=1}^{P} \eta_p (\mathbf{m}_p - \mathbf{m})(\mathbf{m}_p - \mathbf{m})^\top. \tag{10}$$

In order to increase interclass variance and decrease intraclass variance, the following is maximized in accordance with the transformation matrix $\mathbf{V} \in \mathbb{R}^{b\times b}$ as

$$\max_{\mathbf{V}} \left\{ \frac{\mathbf{V}^\top \mathbf{T}_B \mathbf{V}}{\mathbf{V}^\top \mathbf{T}_W \mathbf{V}} \right\} \tag{11}$$

where $\mathbf{V}$ is simply calculated by using a generalized eigenvalue problem as follows:

$$\mathbf{T}_B \mathbf{V} = \mathbf{\Phi} \mathbf{T}_W \mathbf{V} \tag{12}$$

where the columns of $\mathbf{V}$ are the generalized eigenvectors associated with the generalized eigenvalues, which are the diagonal elements of $\mathbf{\Phi}$. After computing $\mathbf{V}$ and $\mathbf{\Phi}$, the eigenvalues are arranged in descending order with their corresponding eigenvectors. Then, only the first $P-1$ columns of the sorted eigenvectors are selected because the matrix $\mathbf{T}_B$ has a rank of $P-1$ and also the first $P-1$ sorted eigenvalues are nonzero. As a result, the first $P-1$ columns of the sorted $\mathbf{V}$, i.e., $\mathbf{V}_d \in \mathbb{R}^{b\times(P-1)}$, is used as the transformation matrix for dimensionality reduction.

Suppose that the obtained features $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathcal{F}_{\mathcal{H}\mathcal{U}}$, and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$ are converted to the 2-D feature matrices as $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{2b\times m}$, $\mathbf{F}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(b+tl)\times m}$, and $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(b+tl)\times m}$ and their corresponding temporarily enlarged dictionaries are also given as $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{2b\times\eta}$, $\hat{\mathbf{D}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(b+tl)\times\eta}$, and $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(b+tl)\times\eta}$, i.e., $(\eta < m)$, respectively. After the equations from (7) to (12) are calculated for $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\hat{\mathbf{D}}_{\mathcal{H}\mathcal{U}}$, and $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}}$, the transformation matrices are obtained as $\mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{H}}}} \in \mathbb{R}^{2b\times(P-1)}$, $\mathbf{V}_{d_{\mathcal{H}\mathcal{U}}} \in \mathbb{R}^{(b+tl)\times(P-1)}$, and $\mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{U}}}} \in \mathbb{R}^{(b+tl)\times(P-1)}$ independently. Then, in order to reduce the high dimensionality of the features, the transformations are evaluated as follows:

$$\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}} = \mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{H}}}}^\top \mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}} \tag{13}$$

$$\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}} = \mathbf{V}_{d_{\mathcal{H}\mathcal{U}}}^\top \mathbf{F}_{\mathcal{H}\mathcal{U}} \tag{14}$$

$$\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}} = \mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{U}}}}^\top \mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}} \tag{15}$$

where $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{(P-1)\times m}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(P-1)\times m}$, and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(P-1)\times m}$ are the low-dimensional spaces of $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathbf{F}_{\mathcal{H}\mathcal{U}}$, and $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}}$, respectively. Before classification, all of the low-dimensional feature sets $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}$, and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$ are normalized to have columnwise unit norm.

E. Relaxed Collaborative Representation

RCR relies on the observation that the representation coefficients of the extracted feature sets should share similarities but also should reflect dissimilarities to emphasize different characteristics of the features. RCR is mainly proposed for the pixelwise classification. However, it is time consuming to handle

the HSIC task due to the presence of many unlabeled pixels. Therefore, the segmented HSI in the pseudosample generation stage is used for RCR. In this way, the computation time is reduced and the neighborhood information is utilized simultaneously. Suppose that ϑ different features are extracted for each pixel and each feature is represented as $v = 1, 2, \ldots, \vartheta$. In the case of GRR, $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}$, and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$ are the feature sets and $\check{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{(P-1)\times n}$, $\check{\mathbf{D}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(P-1)\times n}$, and $\check{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(P-1)\times n}$ are the corresponding dictionaries of the feature sets, respectively.

Let $\mathbf{X}_k^v$ be the vth feature representation coefficient matrix affiliated with the corresponding dictionary $\mathbf{A}_k^v$ and the samples $\mathbf{Y}_k^v$ in the kth superpixel. Then, the objective function of RCR is given as follows:

$$\arg\min_{\{\mathbf{X}_k^v\},\{\psi^v\}} \sum_{v=1}^{\vartheta} \left(\|\mathbf{Y}_k^v - \mathbf{A}_k^v \mathbf{X}_k^v\|_F^2 + \lambda \|\mathbf{X}_k^v\|_F^2 + \tau\psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2 \right) \quad \text{s.t. prior } \{\psi^v\}_{v=1,2,\ldots,\vartheta} \tag{16}$$

where λ and τ are the positive regularization parameters. The terms ψ^v and $\bar{\mathbf{X}}_k^v$ denote the weight of each feature and the mean of all coefficient matrices $\{\mathbf{X}_k^v\}_{v=1,2,\ldots,\vartheta}$, i.e., $\sum_{v=1}^{\vartheta} \mathbf{X}_k^v/\vartheta$, respectively. The regularization $\psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2$ ensures that all the coding matrices $\{\mathbf{X}_k^v\}$ are similar because each feature coding matrix $\mathbf{X}_k^v$ is tried to minimize as close to the mean coefficient matrix $\bar{\mathbf{X}}_k^v$. On the other hand, the weight ψ^v is utilized to emphasize the contribution of each future. According to the work in [31], the proper selection of *prior* $\{\psi^v\}$ depends on three cases such as strong, moderate, and weak priors. In the case of moderate and weak priors, the problem (16) is valid, which requires an additional calculation to optimize $\{\psi^v\}$. However, since the proper weight of each future is known beforehand for the case of strong prior, the problem (16) is reduced to optimize only the coding matrices $\{\mathbf{X}^v\}$.

To decrease the computation time of the proposed GRR and assign appropriate weight to each feature, suppose the obtained features after LDA process are preclassified by SVM. Then, the overall accuracy (OA) results of the features, which is the ratio of correct predictions divided by the amount of unlabeled samples to be classified, are used to determine the weights ψ^v as follows:

$$\{\psi^v\}_{v=1,2,\ldots,\vartheta} = \frac{\mathrm{OA}_v}{\sum_{v=1}^{\vartheta} \mathrm{OA}_v} \tag{17}$$

where OA_v represents the OA result of the vth feature. In addition, (17) assures that the sum of the weights is equal to unity. The optimization problem in (16) can be thought as minimizing only $\{\mathbf{X}_k^v\}$ because the weights are $\{\psi^v\}$ prelearned using the preclassification of the features. Therefore, (16) can be rewritten as

$$\arg\min_{\{\mathbf{X}_k^v\}} \sum_{v=1}^{\vartheta} \left(\|\mathbf{Y}_k^v - \mathbf{A}_k^v \mathbf{X}_k^v\|_F^2 + \lambda \|\mathbf{X}_k^v\|_F^2 + \tau\psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2 \right) \tag{18}$$

which has a closed-form solution [31], [34] for $v = 1, 2, \ldots, \vartheta$ as follows:

$$\mathbf{X}_k^v = \mathbf{X}_k^{0,v} + \tau \frac{\psi^v}{\sum_{\gamma=1}^{\vartheta} \psi^\gamma} \mathbf{B}_k^v \mathbf{Q}_k \sum_{\gamma=1}^{\vartheta} \psi^\gamma \mathbf{X}_k^{0,v} \tag{19}$$

$$\bar{\mathbf{X}}_k^v = \sum_{v=1}^{\vartheta} \psi^v \mathbf{X}_k^v \Big/ \sum_{v=1}^{\vartheta} \psi^v \tag{20}$$

where $\mathbf{B}_k^v = ((\mathbf{A}_k^v)^\top \mathbf{A}_k^v + \mathbf{I}(\lambda + \tau\psi^v))^{-1}$, $\mathbf{X}_k^{0,v} = \mathbf{B}_k^v (\mathbf{A}_k^v)^\top \mathbf{Y}_k^v$, $\mathbf{Q}_k = (\mathbf{I} - \sum_{\gamma=1}^{\vartheta} \bar{\psi}^\gamma \mathbf{B}_k^\gamma)^{-1}$, $\bar{\psi}^\gamma = \tau(\psi^\gamma)^2 / \sum_{v=1}^{\vartheta} \psi^v$, and $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix. After the optimal solutions of $\{\mathbf{X}_k^v\}$ are found, the reconstruction residual error e for class p is calculated as

$$e_p = \sum_{v=1}^{\vartheta} \psi^v \|\mathbf{Y}_k^v - \mathbf{A}_{k_p}^v \mathbf{X}_{k_p}^v\|_F^2 \tag{21}$$

where $\mathbf{A}_{k_p}^v$ and $\mathbf{X}_{k_p}^v$ represent the dictionary and coding matrix of the vth feature in the kth superpixel belonging to pth class, respectively. Then, the labels of the samples inside a superpixel set is determined as

$$L(\mathbf{Y}_k^v) = \arg\min_{p=1,2,\ldots,P} e_p \tag{22}$$

where $L(\mathbf{Y}_k^v)$ indicates the label of the samples $\mathbf{Y}_k^v$. As a result of calculating all the equations from (16) to (22) for all superpixels, the classification map of a given HSI is obtained. Algorithm 1 presents the pseudocode of the proposed GRR.

III. Experimental Analysis and Results

This section describes the datasets, analyzes the effects of the parameters, and presents the results of the fair comparisons with the recent methods. All of the experiments are separately conducted on the Indian Pines (IP)[60], Pavia University (PU)[61], and large-scale WHU-Hi-HongHu (WHHH)[62] HSIC datasets. The parameters of the GRR are the number of random patches t, number of principal components h, patch size w, layer depth l, superpixel patch size T, RCR regularizers λ, and τ. While adjusting a parameter to an optimum value in a specified range, the remaining parameters are kept constant. In order to validate the effectiveness of the GRR, we select some recent methods called JCRC-MTL [34], RPNet [38], RPCC [39], SSRPNet [41], GRPC [43], RPNet-RF [45], and MS-RPNet [46] for the comparison. In addition, an SVM-based version of GRR (GR-SVM), which classifies the features with SVM instead of RCR is also taken into account to prove the effectiveness of the GRR. The experiments were realized on a PC with an Intel Core i7-6700HQ 3.50-GHz processor and 16 GB of RAM.

A. IP Dataset

The IP dataset was obtained from the IP test zone in Northwestern Indiana [60]. It has a spatial size of 145×145 pixels where spatial resolution is 20 m/pixel and covers 220 spectral bands ranging between $0.4-2.5$ μm in 10 nm steps. However, The bands numbered [104–108, 150–163, 220] are discarded due to the water absorption. Therefore, the IP with a size

Algorithm 1: Pseudocode of the Proposed GRR.

Input:
1) Hyperspectral image $\mathcal{H}$
2) RPNet parameters: t, h, w, and l
3) Superpixel segmentation parameter: T
4) RCR parameters: λ and τ

S1: Extract the RPNet features
for each layer ℓ up to l
1) Apply PCA to $\mathcal{H}$ and obtain $\mathcal{H}_P$ with first h PCs.
2) Whiten the data $\mathcal{H}_P$ and obtain $\mathcal{H}_W$.
3) Extract t convolution kernels $\mathcal{C}_{1,2,\ldots,t}$ from $\mathcal{H}_W$.
4) Convolve the $\mathcal{H}_W$ with each of $\mathcal{C}_{1,2,\ldots,t}$ by (1) and then concatenate the features into the $\mathcal{N}$.
5) Obtain the feature $\mathcal{U}^\ell$ of the corresponding layer after applying the ReLU activation function to $\mathcal{N}$ by (2) and (3).
6) If $\ell < l$, repeat the process by changing $\mathcal{H}$ with $\mathcal{U}^\ell$.
end

S2: Stack the features into $\mathcal{U} = \{\mathcal{U}^1, \mathcal{U}^2, \ldots, \mathcal{U}^l\}$
S3: Apply GF on $\mathcal{U}$ and $\mathcal{H}$ by (4) and (5), then obtain $\tilde{\mathcal{U}}$ and $\tilde{\mathcal{H}}$.
S4: Stack the extracted features $\mathcal{U}$, $\tilde{\mathcal{U}}$, and $\tilde{\mathcal{H}}$ with $\mathcal{H}$ separately and construct the feature sets $\mathcal{F}_{\mathcal{H}\mathcal{U}}$, $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$, and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$.
S5: Segment $\mathcal{H}$ into many superpixels with ERS and generate pseudosamples by (6).
S6: Apply LDA on the pseudosamples from (7) to (12) and evaluate the dimension-reduced feature sets $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}$, $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$, and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}}$ from (13) to (15).
S7: Preclassify the features by SVM to assign appropriate feature weights to each future.
S8: Superpixel-guided RCR classification
for each superpixel region k up to K
1) Solve the RCR problem from (16) to (20) and find coefficient matrices $\{\mathbf{X}_k^v\}$.
2) Evaluate the residuals by (21).
3) Assign a label to superpixel by (22).
end

Output: Predicted class labels of the $\mathcal{H}$.

TABLE I
SIXTEEN GROUND-TRUTH CLASSES OF THE IP DATASET

Class	Name	Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93
	Total	10 249

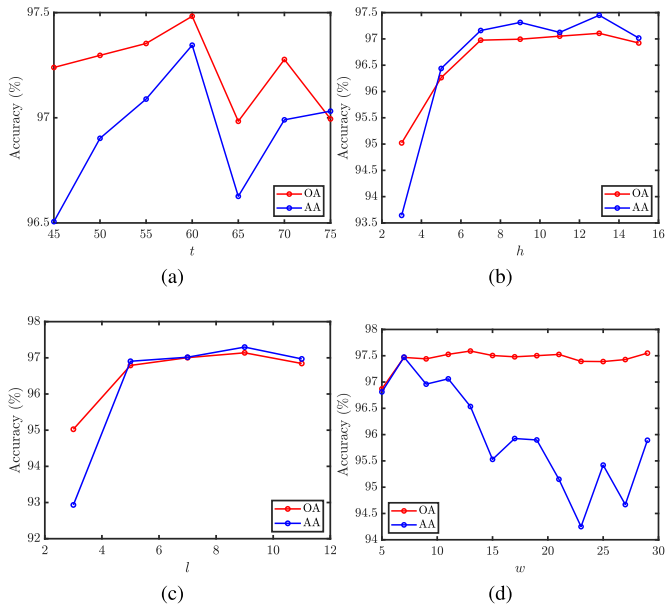

Fig. 2. IP dataset: OA and AA results of the proposed GRR under different (a) random patches t, (b) principal components h, (c) layer depths l, and (d) patch sizes w.

of $145 \times 145 \times 200$ containing 16 land-cover classes is used throughout the experiments. Table I summarizes the class-wise information. The optimal parameters are adjusted using 5% and 95% of randomly selected samples from each class as train and test samples, respectively. All of the experiments are repeated ten times and the results are averaged due to the randomly chosen train and test samples.

Fig. 2 presents the influences of the parameters t, h, l, and w on the performance of GRR in terms of OA and AA. As it is seen from Fig. 2(a), the results of OA and AA reach the maximum at $t = 60$. The effect of number of principal components suddenly arises from 3 to 7; then, the individual results of OA and AA seem almost stable. Both OA and AA have the highest values when $h = 13$ according to Fig. 2(b). The depth l of the network significantly increases the classification performance changing from 3 to 5. After this value, OA and AA results slightly increase toward $l = 9$ and reach the maximum at this point as can be seen from Fig. 2(c). The optimal value of the patch size is chosen as $w = 7$ because AA gradually falls after this point and OA seems stable according to Fig. 2(d). Hence, the optimal values are adjusted as $t = 60$, $h = 13$, $l = 9$, and $w = 7$ for the RPNet-based feature extraction stage.

To generate the pseudosamples for LDA and classify the features with RCR, GRR utilizes superpixel segmentation. Fig. 3 shows the impact of the different superpixel patch sizes T in terms of the OA and AA results. While OA reaches its maximum at $T = 8$, AA reaches the maximum at $T = 4$. When $T = 4$, the OA is almost 1% lower than at $T = 8$. Meanwhile, the AA

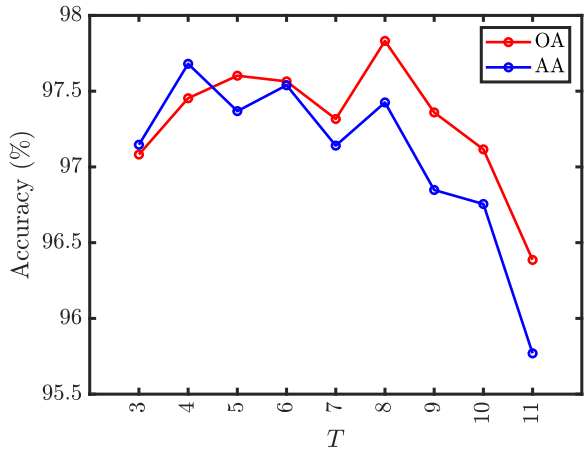

Fig. 3. IP dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

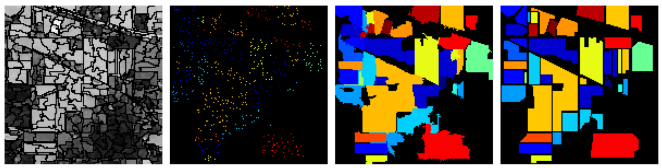

Fig. 4. IP dataset: The maps of (a) superpixel segmentation, (b) train samples, (c) pseudosamples for LDA, and (d) ground-truth.

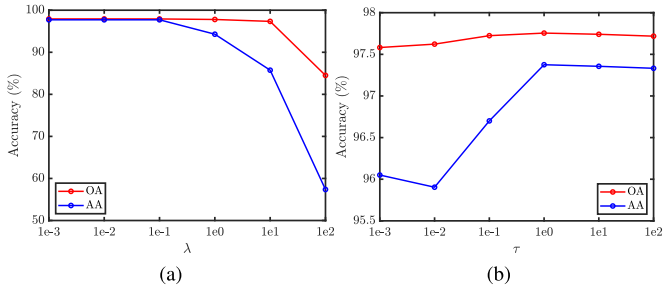

Fig. 5. IP dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ.

decreases by approximately 0.5% when $T = 8$ compared to $T = 4$. Therefore, T is selected as 8 for the convenience. Fig. 4 shows the segmentation map, selected random train samples, generated pseudosamples, and ground-truth image.

The selection of regularization parameters for the RCR-based classification is important. Therefore, these parameters are varied through $\lambda = [1e{-}3,\ 1e2]$ and $\tau = [1e{-}3,\ 1e2]$ to see the effect on the OA and AA results, which is shown in Fig. 5. Except for $\lambda = 1e2$, the OA values are quite similar; however, the AA result starts to drop after $\lambda = 1e{-}1$. Hence, the optimal value of λ is set as $1e{-}1$. Similarly, the OA values are almost the same between $\tau = 1e{-}1$ and $\tau = 1e2$; however, the AA value gets the highest value at $\tau = 1e0$. Therefore, the optimal value of τ is tuned as $1e0$.

Table II reports the comparison results by selecting 5% and 95% of samples from each class used as train and test samples, respectively. The quantitative results are evaluated in terms of classwise accuracies, OA, AA, and kappa (κ) with standard deviations along with computation times. As seen, the proposed GRR achieves the best overall performance followed by GR-SVM. This proves that the RCR-based classification provides better results than the SVM-based classification. The OA, AA, and κ of GRR are approximately 1%, 2%, and 1.5% higher than the GR-SVM. It can be observed that GRR classifies the samples better than other methods under comparison. Efficient feature extraction and RCR-based robust classification greatly affect the performance. JCRC-MTL, which extracts the features with traditional methods and then applies RCR, has lower accuracies than GRR. As seen, GRR outperforms JCRC-MTL by nearly 2.8% and 8% in terms of OA and AA, respectively. The advantage of the GRR can be seen from class 9, which has only one training sample. While GRR completely classifies all the test samples accurately, the other methods produce lower results except for GR-SVM. The best OA attained by RPNet-based methods is SSRPNet due to the graph-learning-based classification of the samples in their methods. Moreover, feature extraction or feature processing stages in RPNet-based models improve the results as well as GRR. For the computation time of the algorithms, GRPC is the fast method followed by RPNet-RF. Among the all methods, the running time of the JCRC-MTL is quite high due to the samplewise patch construction and classification stages. GRR has a slightly higher computation time than GR-SVM due to the fact that RCR requires additional computations.

The classification maps of the methods are shown in Fig. 6. Since all the methods except RPNet utilize additional spatial feature processing stages, the corresponding classification maps have smooth edges. The effect of the use of edge preserving filter GF and superpixels can be seen in the classification map of the proposed GRR because the borders of the classes are preserved better than the others. In addition, all methods except GRR failed to correctly classify class 9, which contains few samples and has a small area within the land cover. Finally, the classification map of GRR is closest to the ground-truth map visually.

Fig. 7 shows the OA results by changing the train sample percentage. After 7% train percent, the OA results of the GRR, GR-SVM, and SSRPNet are close to each other. For the percentage limits below 7%, GRR has the highest OA than the others. When only 1% of the IP dataset is selected as the training sample, GRR provides almost 5% higher OA than the methods that provide the closest OAs such as SSRPNet and JCRC-MTL. Therefore, it can be concluded that GRR has the ability to efficiently classify the samples under a few labeled samples compared to the methods under comparison.

B. PU Dataset

The PU dataset was collected by ROSIS sensor over the PU located in Northern Italy. The PU dataset consists of 610×340 pixels and 103 spectral bands spanned between 0.43 and 0.86 μm. It has a spatial resolution of 1.3 m/pixel and

TABLE II
CLASSIFICATION ACCURACY RESULTS WITH 5% LABELED SAMPLES OF THE IP DATASET

Class	Train	Test	RPNet	RPNet-RF	RPCC	GRPC	MS-RPNet	JCRC-MTL	SSRPNet	GR-SVM	GRR
1	3	43	75.23	90.58	92.44	82.55	92.55	95.81	**99.53**	95.34	96.04
2	72	1356	91.11	93.19	94.03	91.90	93.57	91.65	**97.34**	97.33	96.66
3	42	788	87.81	92.60	91.85	92.39	93.73	88.56	95.73	98.85	**99.74**
4	12	225	60.26	90.31	93.82	**97.88**	95.55	94.00	97.86	93.24	96.08
5	25	458	89.75	90.25	93.16	92.60	**94.93**	92.29	94.19	91.96	93.82
6	37	693	97.51	93.51	93.05	93.53	94.52	98.62	95.15	96.65	**99.30**
7	2	26	85.38	95.00	93.07	90.96	93.84	98.07	80.76	**98.46**	96.53
8	24	454	96.60	99.57	98.81	98.97	98.98	99.95	99.91	**100**	**100**
9	1	19	35.26	49.21	55.00	66.05	51.84	0	24.21	89.47	**100**
10	49	923	88.55	90.11	89.97	88.62	91.37	90.96	92.97	94.95	**95.26**
11	123	2332	95.77	94.41	93.91	94.21	94.90	97.37	**98.54**	98.00	98.39
12	30	563	81.19	87.54	86.59	90.82	88.59	92.62	93.99	95.13	**98.54**
13	11	194	98.24	94.43	91.08	93.58	93.42	99.17	87.73	**99.48**	99.27
14	64	1201	98.34	97.33	97.42	98.01	98.46	**99.99**	99.85	97.36	99.92
15	20	366	89.30	94.19	94.23	95.09	95.72	99.04	**99.45**	87.97	93.79
16	5	88	64.26	76.64	81.98	87.72	88.92	95.34	59.54	**97.04**	**97.04**
OA	–	–	91.61 (± 1.06)	93.26 (± 0.63)	93.29 (± 0.96)	93.51 (± 0.95)	94.39 (± 0.86)	95.08 (± 0.58)	96.39 (± 0.54)	96.66 (± 0.33)	**97.85** (± 0.60)
AA	–	–	83.41 (± 1.82)	89.30 (± 1.99)	90.02 (± 1.74)	90.93 (± 2.45)	91.31 (± 2.07)	89.59 (± 0.71)	88.55 (± 3.69)	95.70 (± 1.53)	**97.52** (± 0.62)
κ	–	–	90.38 (± 1.22)	92.32 (± 0.72)	92.35 (± 1.10)	92.61 (± 1.08)	93.61 (± 0.99)	94.38 (± 0.66)	95.88 (± 0.63)	96.18 (± 0.38)	**97.55** (± 0.69)
Runtime (s)	–	–	2.76	2.39	10.51	**2.19**	17.21	39.53	22.47	7.72	8.92

The highest result is bolded and the following result is underlined in each line.

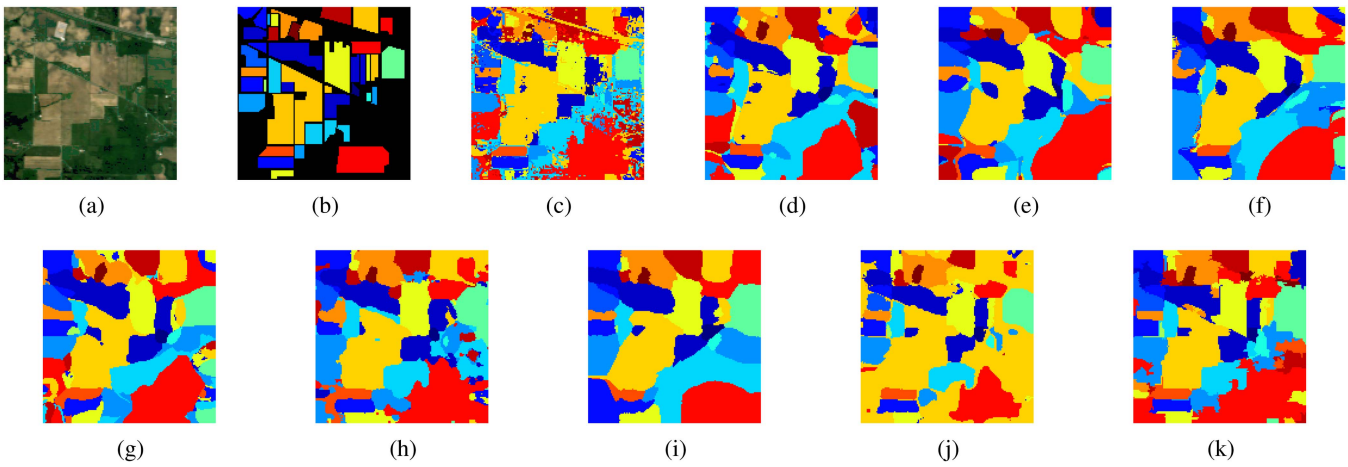

Fig. 6. IP dataset. (a) False-color image. (b) Ground-truth. Classification maps with OAs obtained by (c) RPNet (91.97%), (d) RPNet-RF (93.39%), (e) RPCC (93.94%), (f) GRPC (94.51%), (g) MS-RPNet (95.19%), (h) JCRC-MTL (95.31%), (i) SSRPNet (96.71%), (j) GR-SVM (96.92%), and (k) GRR (98.24%).

covers 9 ground-truth classes. Table III presents the class-based information of the scene. Among the available samples, 1% and 99% of each class are randomly partitioned as train and test samples, respectively. Simulations are repeated 10 times, and then, the results are averaged.

The optimal parameters of RPNet-based feature extraction are evaluated using the same strategy used in the case of IP where the parameters t, h, l, and w are varied. Fig. 8 shows the variations of OAs and AAs with respect to the different values of the parameters. According to Fig. 8(a), the number of random patches $t = 50$ provides the highest OA and AA. After this value, both OA and AA decrease dramatically. Starting from the smallest value in the range of h, the number of principal components continuously increases the result until $h = 13$. In accordance with Fig. 8(b), it seems that there is a slow decline

TABLE III
NINE GROUND-TRUTH CLASSES OF THE PU DATASET

Class	Name	Samples
1	Asphalt	6631
2	Meadows	18 649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare Soil	5029
7	Bitumen	1330
8	Self-Blocking Bricks	3682
9	Shadows	947
	Total	42 776

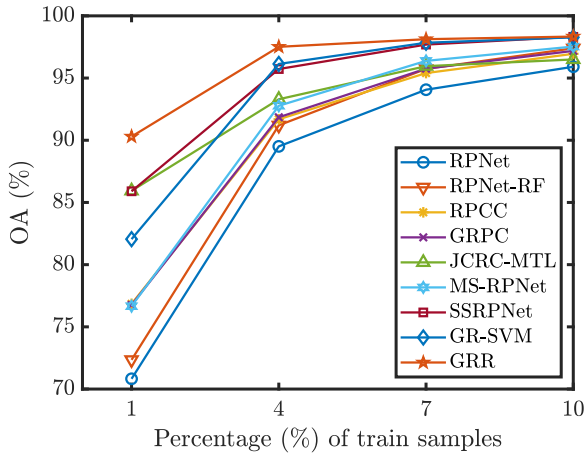

Fig. 7. IP dataset: OA results of the different methods under varying training percentages.

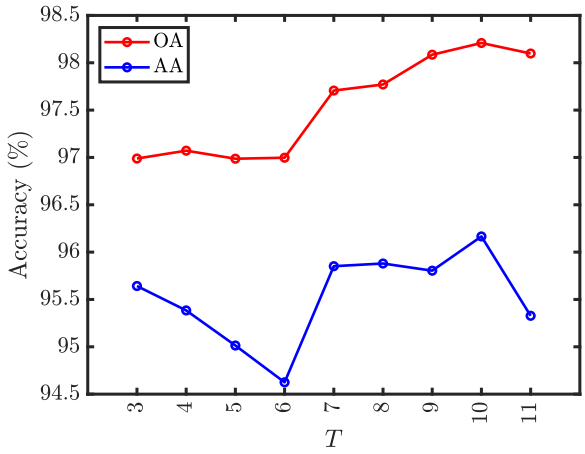

Fig. 9. PU dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

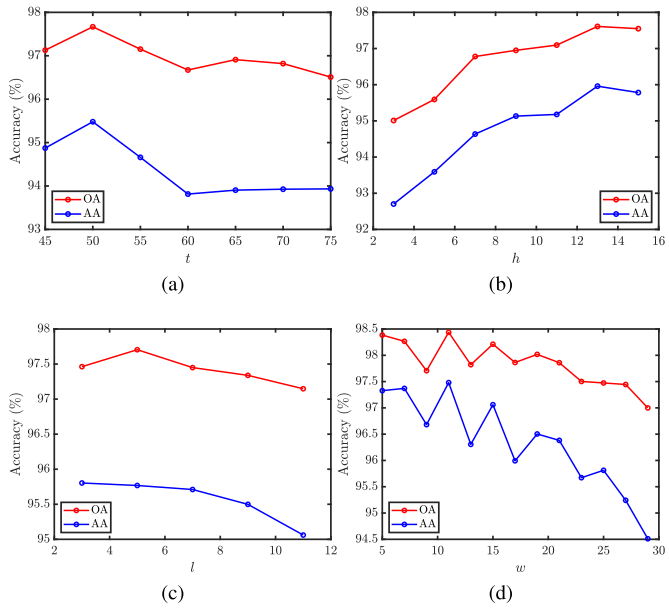

Fig. 8. PU dataset: OA and AA results of the proposed GRR under different (a) random patches t, (b) principal components h, (c) layer depths l, and (d) patch sizes w.

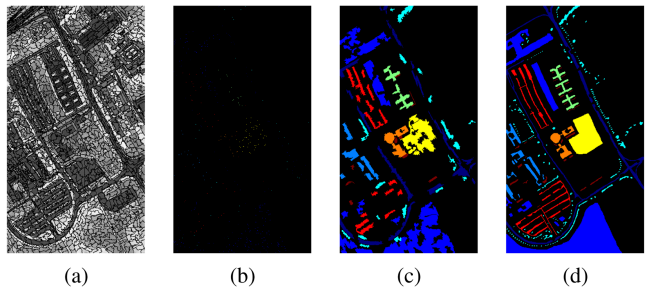

Fig. 10. PU dataset: The maps of (a) superpixel segmentation, (b) train samples, (c) pseudosamples for LDA, and (d) ground-truth.

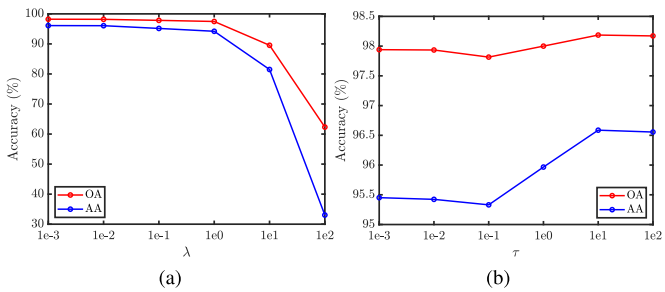

Fig. 11. PU dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ.

for OA and AA after this value. The optimal value of the layer number l is determined as 5 because the amount of increase in OA is remarkable and the result of AA is satisfactory as seen in Fig. 8(c). Since the geometrical structures of the classes in the PU dataset are very different, i.e., asphalt, meadows, bare soil, and shadows, the patch sizes w of the convolution kernels significantly affect the results, which can be seen from Fig. 8(d) clearly. However, both OA and AA reach the highest value when $w = 11$ at which the features of the geometrical structures are better maintained. Eventually, the optimal parameters are tuned as $t = 50$, $h = 13$, $l = 5$, and $w = 11$ for the RPNet-based feature extraction in the PU dataset.

Fig. 9 represents the changes in OA and AA values with the corresponding superpixel patch sizes when T varies from 3 to 11. The selection of T is significant because it is used in generating pseudosamples of LDA as well as the classification of the samples. As it can be observed, the OA value increases from $T = 3$ to $T = 10$; then, it starts to decrease. Meanwhile, the AA value first decreases from $T = 3$ to $T = 6$, then it increases up to $T = 10$ at which the AA is maximum. Hence, the optimal value is selected as $T = 10$ for best results. Fig. 10 depicts the segmentation map, random train samples, generated pseudosamples, and ground-truth, respectively.

The optimal regularization parameters λ and τ are selected using the same idea in the IP case. Fig. 11(a) shows the OAs and AAs, which are quite close for $\lambda = 1e{-}3$ and $\lambda = 1e0$,

TABLE IV
CLASSIFICATION ACCURACY RESULTS WITH 1% LABELED SAMPLES OF THE PU DATASET

Class	Train	Test	RPNet	RPNet-RF	RPCC	GRPC	MS-RPNet	JCRC-MTL	SSRPNet	GR-SVM	GRR
1	67	6564	91.02	91.46	91.76	92.96	89.14	92.69	**99.44**	98.01	98.51
2	187	18462	99.32	99.17	99.53	99.01	99.01	99.54	99.24	99.43	**99.96**
3	21	2078	75.70	76.08	87.26	87.68	91.60	96.51	95.44	96.02	**96.93**
4	31	3033	87.74	84.60	70.84	76.05	**91.72**	72.97	73.08	88.92	91.58
5	14	1331	50.33	97.55	91.29	92.51	99.19	**100**	98.44	99.87	99.21
6	51	4978	88.75	95.81	99.62	99.44	88.30	99.84	**100**	98.78	98.01
7	14	1316	80.16	76.73	95.11	93.29	95.69	99.28	99.52	98.73	**99.68**
8	37	3645	94.26	85.71	88.00	86.36	92.10	96.59	**98.98**	95.37	96.26
9	10	937	70.90	77.46	50.53	54.48	82.26	77.40	46.91	77.13	**85.84**
OA	–	–	91.60 (± 0.76)	93.02 (± 0.78)	93.21 (± 0.34)	93.47 (± 0.93)	94.27 (± 1.34)	95.72 (± 0.52)	96.10 (± 0.51)	97.37 (± 0.44)	**98.10** (± 0.51)
AA	–	–	82.02 (± 2.40)	87.17 (± 1.24)	85.99 (± 1.66)	86.87 (± 1.74)	92.11 (± 3.20)	92.76 (± 1.02)	90.12 (± 1.41)	94.70 (± 1.67)	**96.22** (± 1.01)
κ	–	–	88.58 (± 1.09)	90.69 (± 1.04)	90.95 (± 0.46)	91.31 (± 1.24)	92.38 (± 1.78)	94.29 (± 0.71)	94.83 (± 0.69)	96.51 (± 0.59)	**97.47** (± 0.68)
Runtime (s)	–	–	**5.98**	8.32	81.49	9.20	127.43	267.64	223.87	34.16	38.68

The highest result is bolded and the following result is underlined in each line.

both are maximum at $\lambda = 1e{-}3$. After $\lambda = 1e{-}3$, both OAs and AAs decrease seriously. Fig. 11(b) shows the OAs and AAs by varying τ in a specified interval. There are no remarkable changes in the OAs; however, the AA is at the highest value when $\tau = 1e1$. Consequently, the optimal parameters are set to $\lambda = 1e{-}3$ and $\tau = 1e1$, respectively.

Table IV presents the comparative classification results of the methods under 1% labeled samples in each class of the PU using the optimal parameters. The proposed GRR achieves the best or the second-best classwise accuracies for many classes. Compared to the GR-SVM, the OA, AA, and κ of GRR is about 1% better due to the RCR-based classification. The results of JCRC-MTL are approximately 3% lower than the results of GRR. It means that the extracted features of GRR are more meaningful than the features of the JCRC-MTL. This is because the original and guided filtered RPNet features. Among all RPNet-based methods, GRR has the highest OA, AA, and κ. For example, the results of the SSRPNet are the highest one among RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, and SSRPNet. However, GRR outperforms the SSRPNet by nearly 2%, 6%, and 3% in terms of OA, AA, and κ, respectively. Since the spatial size of the PU is much larger than the IP, the computation time of the methods is directly affected as seen in Table IV. For example, the runtime of the RPCC increases considerably compared to the IP dataset because the local covariance matrix calculation requires much time for the spatial size of the PU. 2D-SSA and S3-PCA steps of the MS-RPNet methods take much time due to the large spatial dimension of the PU. The abundance of the test samples negatively affect the computation time of the JCRC-MTL because the patch construction around each test sample is time consuming. For the SSRPNet, the weight calculation between the nodes in graph-based learning consumes much time for the large spatial size of the PU. In the proposed GRR, most of the computation time is spent on the GF process because the edge details are extracted in a local window. Compared to the GR-SVM, GRR takes approximately 4.5 s longer due to the superpixel-based classification with RCR. Among the all methods, RPNet has the lowest running time followed by the RPNet-RF.

In order to see the performance of methods visually, classification maps of all methods are provided in Fig. 12. It can be observed that the classification map of GRR is quite close to the reference map and false-color image. In addition, the edges of the classes and the small local spatial regions are preserved and classified better compared to the other methods since GF preserves edge details and superpixel-guided RCR classifies the samples better.

An additional comparative experiment is realized under different numbers of train samples, such as 10, 40, 70, and 100 per class. According to Fig. 13, the proposed GRR is able to classify the dataset better than the other methods under a few labeled samples. As the number of train samples increases, the OA results of other methods get close to the OA of GRR.

C. WHHH Dataset

The large-scale WHHH dataset was obtained over HongHu City, China, in 2017. It has a size of 940 × 475 × 270 whose wavelength changes from 0.4 to 1 μm. The dataset has quite high spatial and spectral resolutions, such as 4.3 cm and 6 nm, respectively. There are 22 land-cover classes, which include different crop types. Table V summarizes the class-based information of the dataset. Due to the plenty of available samples, only 0.5% of the samples are used as train and the rest are utilized as test samples in each class. All of the experiments are reiterated 10 times; then, the results are averaged.

Fig. 14 shows the variation of OA and AA results with respect to the t, h, l, and w, respectively. According to Fig. 14(a), the number of random convolution kernels t is selected as 120 because the higher OA and AA results are obtained. The number of principal components h increases the results up to $h = 15$ as seen in Fig. 14(b); then, both OA and AA values decrease. The layer depth l is an important parameter that has an effect on the extraction of deep features. Fig. 14(c) depicts the impact of l on the OA and AA results. It is seen that $l = 9$ provides the highest results. The optimal value of the patch size w is determined as 25 due to the highest values of OA and AA according to Fig. 14(d).

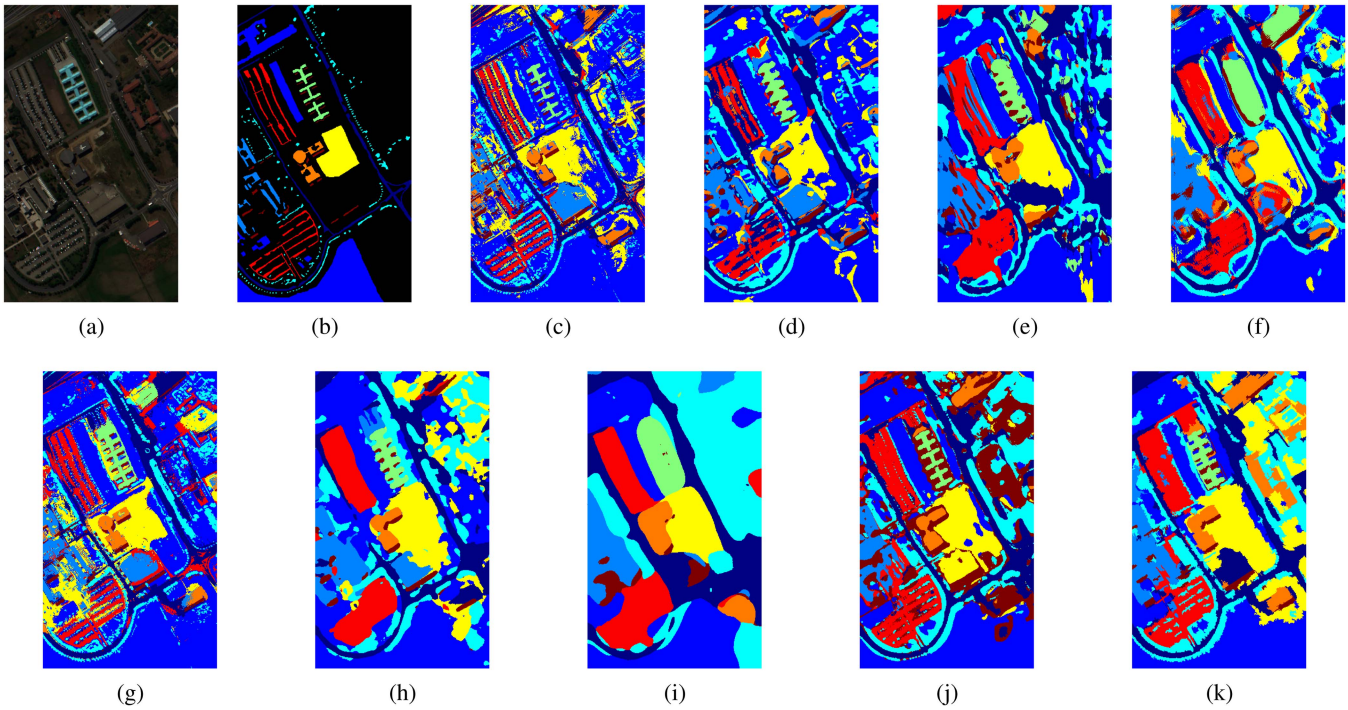

Fig. 12. PU dataset. (a) False-color image. (b) Ground-truth. Classification maps with OAs obtained by (c) RPNet (92.03%), (d) RPNet-RF (93.34%), (e) RPCC (93.53%), (f) GRPC (94.27%), (g) MS-RPNet (95.32%), (h) JCRC-MTL (96.18%), (i) SSRPNet (96.59%), (j) GR-SVM (97.51%), and (k) GRR (98.58%).

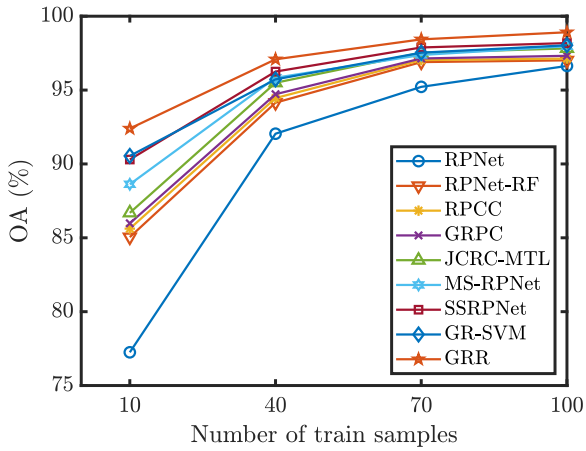

Fig. 13. PU dataset: OA results of the different methods under varying training samples per class.

TABLE V
TWENTY-TWO GROUND-TRUTH CLASSES OF THE WHHH DATASET

Class	Name	Samples
1	Red roof	14 041
2	Road	3512
3	Bare soil	21 821
4	Cotton	163 285
5	Cotton firewood	6218
6	Rape	44 557
7	Chinese cabbage	24 103
8	Pakchoi	4054
9	Cabbage	10 819
10	Tuber mustard	12 394
11	Brassica parachinensis	11 015
12	Brassica chinensis	8954
13	Small Brassica chinensis	22 507
14	Lactuca sativa	7356
15	Celtuce	1002
16	Film covered lettuce	7262
17	Romaine lettuce	3010
18	Carrot	3217
19	White radish	8712
20	Garlic sprout	3486
21	Broad bean	1328
22	Tree	4040
	Total	386 693

Consequently, the optimal parameters are adjusted as $t = 120$, $h = 15$, $l = 9$, and $w = 25$, respectively.

The effect of superpixel patch size T is analyzed in Fig. 15. As it is seen, an increase in T decreases the OAs and AAs. This may be due to the small spatial regions such as class 2 and small regions belonging to a class located inside a different class such as class 14 found in class 13 and class 7–10 found in class 14. In such cases, there would be misclassification because the large superpixel patch size could contain different classes. According to Fig. 15, $T = 5$ is the best choice to achieve higher OA and AA values. Fig. 16 illustrates the segmentation map with respect to $T = 5$, randomly chosen train samples, formed pseudosamples, and reference ground-truth map.

The impact of the RCR regularization parameters λ and τ is presented in Fig. 17. It is seen from Fig. 17(a) that the OA results are similar between $\lambda = 1e{-}3$ and $\lambda = 1e1$, then

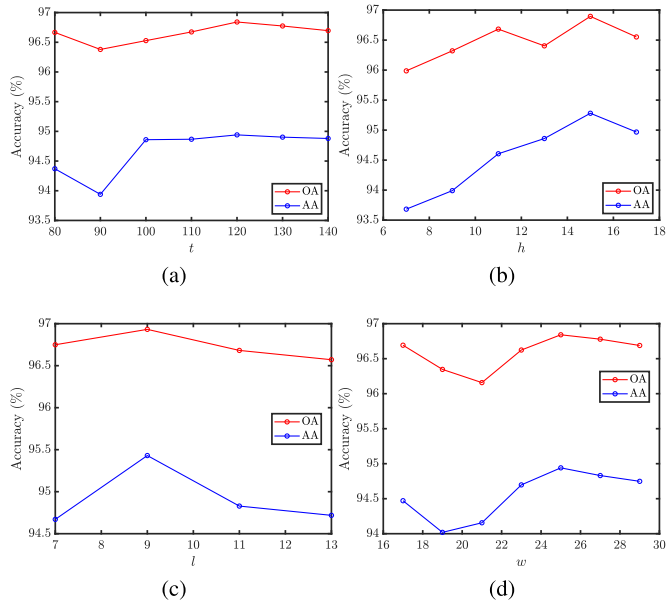

Fig. 14. WHHH dataset: OA and AA results of the proposed GRR under different (a) random patches t, (b) principal components h, (c) layer depths l, and (d) patch sizes w.

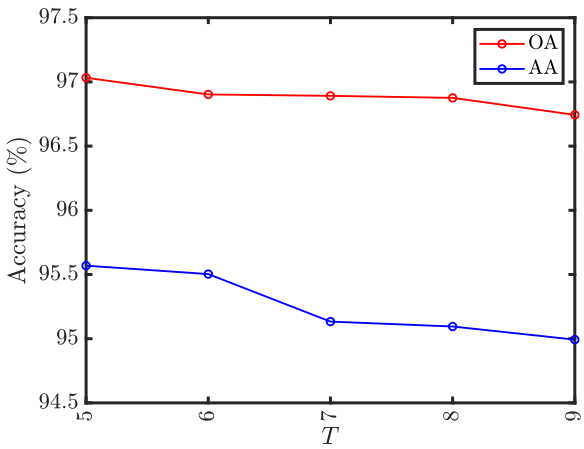

Fig. 15. WHHH dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

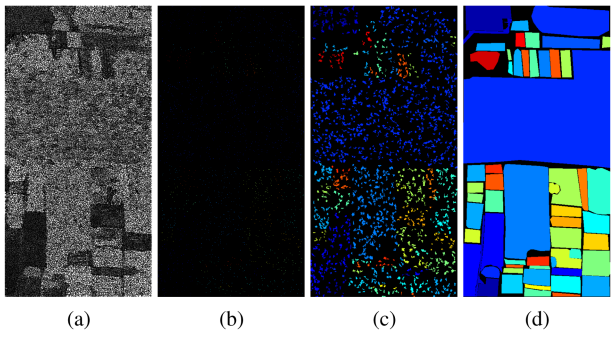

Fig. 16. WHHH dataset: The maps of (a) superpixel segmentation, (b) train samples, (c) pseudosamples for LDA, and (d) ground-truth.

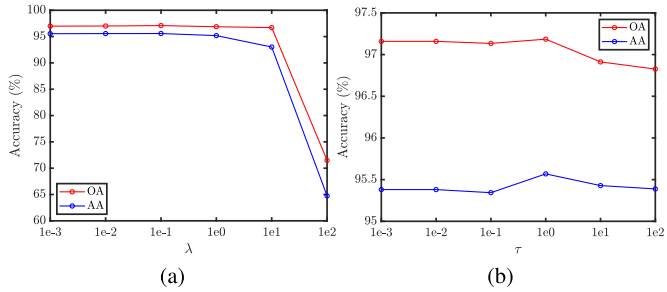

Fig. 17. WHHH dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ.

suddenly decrease after $\lambda = 1e1$. By the way, the AA results are similar between $\lambda = 1e{-}3$ and $\lambda = 1e{-}1$, and start to fall after $\lambda = 1e{-}1$. Hence, the best parameter is decided as $\lambda = 1e{-}1$. For the parameter τ, it is seen that both OA and AA attain the maximum at $\tau = 1e0$ according to Fig. 17(b). Therefore, the optimal parameters are tuned as $\lambda = 1e{-}1$ and $\tau = 1e0$, respectively.

A fair comparison results are indicated in Table VI by randomly distributing 0.5% train samples per class. It is observed that the proposed GRR has the highest OA, AA, and κ values followed by the results of the GR-SVM. It can be concluded that superpixel-based RCR provides superior results compared to SVM since RCR utilizes the extracted features efficiently and superpixel segmentation provides spatially correlated regions. Compared to the JCRC-MTL, GRR outperforms the results in terms of OA, AA, and κ by approximately 3%, 8.5%, and 4%, respectively. It shows that extracted features with RPNet and superpixel-based RCR provide better performance than traditional feature extraction methods and patch-based RCR. The other RPNet-based methods such as RPNet-RF, RPCC, GRPC, MS-RPNet, and SSRPNet obtain higher accuracy results than RPNet due to the additional spatial filtering operations. The efficacy of the proposed GRR can be observed from classes 2 and 15 whose spatial regions are small and available train samples are low. Except for GRR with 94.40% class accuracy, the other methods have class accuracies lower than 90% for class 2. In class 15, GRR and GR-SVM have results greater than 90% while other methods remain below this percentage. Due to the large-scale of WHHH dataset, the computation time of all methods increases compared to the IP and PU datasets. Due to the great number of test samples and patch-based classification, JCRC-MTL is the slowest method followed by SSRPNet. The evaluation of node weights in graph-based SSRPNet requires a high-computational cost. In the method MS-RPNet, the 2D-SSA and S3-PCA algorithms take long time; therefore, the computational time is high. Among all methods, RPNet-RF is the fastest method followed by RPNet. The computation times of the proposed GR-SVM and GRR are lower than the JCRC-MTL, SSRPNet, and MS-RPNet; however, higher than the others since the guided filtering operation spends time due to the large scale of the dataset. Compared to the GR-SVM, the computation time

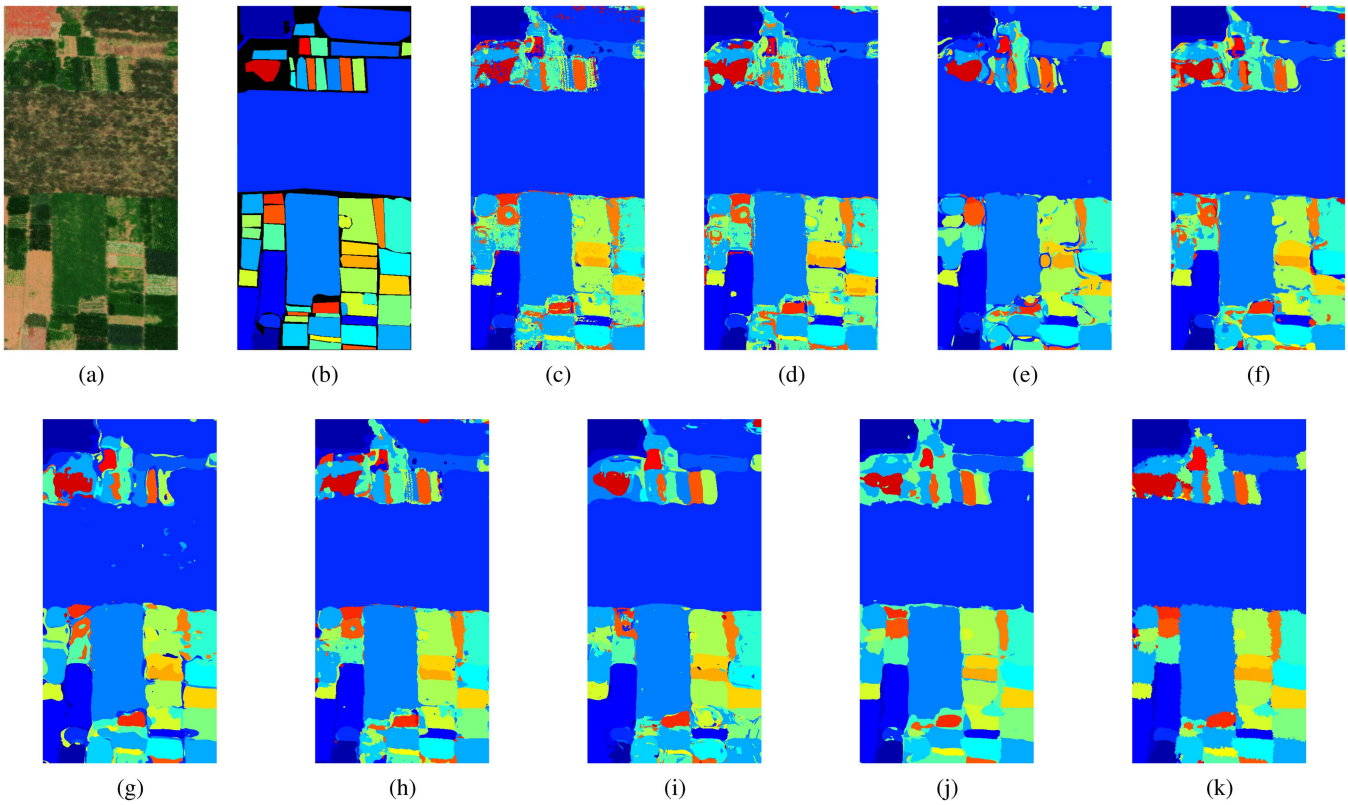

Fig. 18. WHHH dataset. (a) False-color image. (b) Ground-truth. Classification maps with OAs obtained by (c) RPNet (91.78%), (d) RPNet-RF (92.34%), (e) RPCC (92.58%), (f) GRPC (93.18%), (g) MS-RPNet (93.97%), (h) JCRC-MTL (94.51%), (i) SSRPNet (94.92%), (j) GR-SVM (95.83%), and (k) GRR (97.12%).

TABLE VI
CLASSIFICATION ACCURACY RESULTS WITH 0.5% LABELED SAMPLES OF THE WHHH DATASET

Class	Train	Test	RPNet	RPNet-RF	RPCC	GRPC	MS-RPNet	JCRC-MTL	SSRPNet	GR-SVM	GRR
1	71	13 970	97.95	97.07	91.90	**98.73**	94.94	92.06	92.76	98.60	97.69
2	18	3494	88.52	87.29	68.05	68.91	58.98	62.97	61.17	88.32	**94.40**
3	110	21 711	95.35	94.71	94.91	97.54	94.76	97.54	**98.53**	95.60	97.73
4	817	162 468	99.20	99.58	99.81	99.48	99.05	99.50	99.47	99.24	**99.92**
5	32	6186	87.24	86.76	88.57	94.07	91.01	95.59	95.16	99.11	**99.21**
6	223	44 334	95.12	95.38	94.82	96.94	94.52	97.36	97.48	96.96	**98.97**
7	121	23 982	88.16	87.52	85.19	88.01	89.93	94.22	**94.59**	93.96	93.42
8	21	4033	53.16	63.64	70.74	77.81	89.11	86.55	87.19	78.38	**93.18**
9	55	10 764	81.56	91.24	86.91	86.87	93.30	85.29	85.70	91.43	**93.93**
10	62	12 332	81.30	79.04	90.20	84.17	89.22	89.61	90.21	**94.56**	92.40
11	56	10 959	68.35	75.33	77.31	72.38	71.89	79.20	80.41	**91.31**	89.40
12	45	8909	68.03	67.10	81.40	78.35	81.27	77.21	82.77	95.33	**95.47**
13	113	22 394	82.21	79.69	83.75	81.97	87.34	90.67	91.87	87.33	**91.94**
14	37	7319	78.01	86.29	87.47	83.61	**93.29**	74.68	77.07	91.71	89.73
15	6	996	88.65	70.88	33.03	67.03	53.31	73.73	78.46	90.06	**96.78**
16	37	7225	90.57	87.28	72.96	87.53	94.20	94.08	94.73	93.23	**97.07**
17	16	2994	57.08	76.11	81.73	82.01	89.61	88.70	88.79	88.41	**99.67**
18	17	3200	80.84	80.53	67.81	74.28	78.68	86.04	87.19	78.02	**92.06**
19	44	8668	75.44	72.30	85.92	71.92	63.64	73.60	74.05	**90.62**	86.13
20	18	3468	81.02	86.33	52.62	70.41	87.37	80.82	81.03	84.28	**97.40**
21	7	1321	62.52	64.79	79.93	73.18	99.84	93.37	93.52	92.51	**99.93**
22	21	4019	80.04	91.71	88.38	85.40	94.30	89.61	92.58	89.97	**94.40**
OA	–	–	91.25 (± 0.92)	91.96 (± 0.72)	92.07 (± 0.68)	92.54 (± 0.98)	93.12 (± 1.12)	93.98 (± 0.63)	94.46 (± 0.61)	95.73 (± 0.38)	**97.08** (± 0.57)
AA	–	–	80.92 (± 2.11)	82.75 (± 1.86)	80.15 (± 1.69)	82.75 (± 1.96)	85.89 (± 2.76)	86.47 (± 0.92)	87.49 (± 2.53)	91.31 (± 1.32)	**95.04** (± 0.83)
κ	–	–	88.95 (± 1.10)	89.84 (± 0.88)	89.83 (± 0.78)	90.57 (± 1.19)	91.27 (± 1.53)	92.39 (± 0.69)	92.99 (± 0.65)	94.62 (± 0.43)	**96.31** (± 0.61)
Runtime (s)	–	–	103.04	**73.22**	1355.96	116.51	2074.68	4564.29	3261.71	361.46	520.55

The highest result is bolded and the following result is underlined in each line.

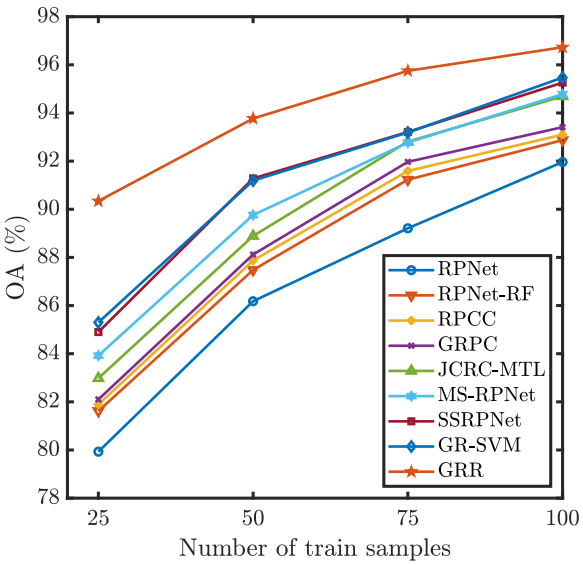

Fig. 19. WHHH dataset: OA results of the different methods under varying training samples per class.

of the GRR is higher due to the preclassification with SVM for the weight assignment of the RCR.

Fig. 18 presents the classification maps of the methods. It is seen that all methods except RPNet have smooth classification maps due to the spatial filtering. By considering both false-color and ground-truth images, the proposed GRR provides more correct spatial borders and less misclassification of the samples. In addition, the oversmoothing of class borders is lesser than the others. It implies that the GF-based filtering and superpixel-based classification preserve the local spatial structures.

In order to see the performance of the methods under equal train samples per class such as 25, 50, 75, and 100, Fig. 19 is introduced in terms of OA. When using 25 train samples per class, GRR provides OA above 90% while other methods remain between 80% and 85%. Although the OAs of other methods approach the OA of GRR as the number of train samples increases, the proposed GRR always provides the best result. Therefore, GRR achieves the best classification performance not only in the separation of percentage-based train samples but also in the partition of equal numbers of train samples per class.

IV. Conclusion

This article proposed an efficient spectral–spatial HSIC method utilizing RPNet and RCR. First, shallow and deep features are extracted using RPNet. GF is utilized to extract edge detailed features of RPNet and original HSI spectral features, respectively. Then, RPNet features and edge-detailed features of RPNet and original HSI are separately stacked with the original HSI spectral features. In this way, the feature sets containing different attributes extracted from the same HSI are obtained. Since the dimension of a stacked feature set is high, LDA is applied to reduce the dimension and increase class separability. In order to maximize the class separability capacity of LDA under a few train samples, pseudosamples are formed by means of superpixel segmentation. Finally, the dimension-reduced feature sets are fed into the superpixel-guided RCR classifier to use the similarity and distinction of the extracted features better. The conducted experiments on the real HSI datasets verified the effectiveness of the proposed GRR method.

The method still needs some improvements to achieve better classification performance. Instead of using fixed-size convolution kernels in RPNet, shape-adaptive kernels can be used. In this way, the more representative features may be obtained. In addition, single-scale superpixel segmentation in the classification stage can be replaced by multiscale segmentation. In this manner, spectral–spatial features of both small and large areas within the HSI can be better characterized.

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] T. Ince and N. Dobigeon, "Fast hyperspectral unmixing using a multiscale sparse regularization," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6015305.

[3] T. Ince, "Double spatial graph Laplacian regularization for sparse unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8009605.

[4] T. Ince, "Superpixel-based graph Laplacian regularization for sparse hyperspectral unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5501305.

[5] D. Manolakis and G. Shaw, "Detection algorithms for hyperspectral imaging applications," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 29–43, Jan. 2002.

[6] H. Ren and C.-I. Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1232–1249, Oct. 2003.

[7] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Sparse representation for target detection in hyperspectral imagery," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 629–640, Jun. 2011.

[8] T. Dundar and T. Ince, "Weighted and label-consistent joint sparse hyperspectral image classification," in *Proc. 13th Int. Conf. Elect. Electron. Eng.*, 2021, pp. 215–219.

[9] T. Dündar and T. İnce, "Locality constraint joint-sparse and weighted low-rank based hyperspectral image classification," in *Proc. 10th Int. Conf. Recent Adv. Space Technol.*, 2023, pp. 1–6.

[10] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral–spatial hyperspectral image classification via multiscale adaptive sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7738–7749, Dec. 2014.

[11] W. Fu, S. Li, L. Fang, X. Kang, and J. A. Benediktsson, "Hyperspectral image classification via shape-adaptive joint sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 2, pp. 556–567, Feb. 2016.

[12] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[13] 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.

[14] L. Wei and D. Qian, "A survey on representation-based classification and detection in hyperspectral remote sensing imagery," *Pattern Recognit. Lett.*, vol. 83, pp. 115–123, Nov. 2016.

[15] K. Bernard, Y. Tarabalka, J. Angulo, J. Chanussot, and J. A. Benediktsson, "Spectral–spatial classification of hyperspectral data based on a stochastic minimum spanning forest approach," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2008–2021, Apr. 2012.

[16] W. Li, E. W. Tramel, S. Prasad, and J. E. Fowler, "Nearest regularized subspace for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 477–489, Jan. 2014.

[17] W. Li and Q. Du, "Joint within-class collaborative representation for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2200–2208, Jun. 2014.

[18] M. Xiong, Q. Ran, W. Li, J. Zou, and Q. Du, "Hyperspectral image classification using weighted joint collaborative representation," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 6, pp. 1209–1213, Jun. 2015.
[19] X. Chen, S. Li, and J. Peng, "Hyperspectral imagery classification with multiple regularized collaborative representations," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7, pp. 1121–1125, Jul. 2017.
[20] X. Shen, W. Bao, H. Liang, X. Zhang, and X. Ma, "Grouped collaborative representation for hyperspectral image classification using a two-phase strategy," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 5505305.
[21] M. Weeks and M. Bayoumi, "3D discrete wavelet transform architectures," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1998, pp. 57–60.
[22] S. Jia, L. Deng, and L. Shen, "An effective collaborative representation algorithm for hyperspectral image classification," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2014, pp. 1–6.
[23] S. Jia, L. Shen, and Q. Li, "Gabor feature-based collaborative representation for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 1118–1129, Feb. 2015.
[24] L. Shen and S. Jia, "Three-dimensional Gabor wavelets for pixel-based hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 5039–5046, Dec. 2011.
[25] J. Yang and J. Qian, "A weighted multiple-feature fusion classifier for hyperspectral images with limited training samples," *Eur. J. Remote Sens.*, vol. 51, no. 1, pp. 1006–1021, 2018.
[26] B. W. M. D. Mura, J. A. Benediktsson, and L. Bruzzone, "Extended profiles with morphological attribute filters for the analysis of hyperspectral data," *Int. J. Remote Sens.*, vol. 31, no. 22, pp. 5975–5991, 2010.
[27] S. Jia, X. Deng, J. Zhu, M. Xu, J. Zhou, and X. Jia, "Collaborative representation-based multiscale superpixel fusion for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7770–7784, Oct. 2019.
[28] G. Liu, L. Gao, and L. Qi, "Hyperspectral image classification via multi-feature-based correlation adaptive representation," *Remote Sens.*, vol. 13, no. 7, 2021, Art. no. 1253.
[29] 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.
[30] J. Benediktsson, M. Pesaresi, and K. Amason, "Classification and feature extraction for remote sensing images from urban areas based on morphological transformations," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 9, pp. 1940–1949, Sep. 2003.
[31] M. Yang, L. Zhang, D. Zhang, and S. Wang, "Relaxed collaborative representation for pattern classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2224–2231.
[32] Z. Wu et al., "Hyperspectral anomaly detection with relaxed collaborative representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5533417.
[33] H. Su, H. Zhang, Z. Wu, and Q. Du, "Relaxed collaborative representation with low-rank and sparse matrix decomposition for hyperspectral anomaly detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6826–6842, 2022.
[34] J. Li, H. Zhang, L. Zhang, X. Huang, and L. Zhang, "Joint collaborative representation with multitask learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5923–5936, Sep. 2014.
[35] H. Su, Y. Gao, and Q. Du, "Superpixel-based relaxed collaborative representation with band weighting for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5525416.
[36] Y. Gao, H. Su, H. Lu, and Q. Du, "Self-balancing dictionary learning for relaxed collaborative representation of hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5539918.
[37] E. Angelopoulou, S. Lee, and R. Bajcsy, "Spectral gradient: A material descriptor invariant to geometry and incident illumination," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, pp. 861–867.
[38] Y. Xu, B. Du, F. Zhang, and L. Zhang, "Hyperspectral image classification via a random patches network," *ISPRS J. Photogrammetry Remote Sens.*, vol. 142, pp. 344–357, 2018.
[39] Y. Sun, Z. Fu, and L. Fan, "A novel hyperspectral image classification pattern using random patches convolution and local covariance," *Remote Sens.*, vol. 11, no. 16, 2019, Art. no. 1954. [Online]. Available: https://www.mdpi.com/2072-4292/11/16/1954
[40] A. Green, M. Berman, P. Switzer, and M. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Trans. Geosci. Remote Sens.*, vol. GRS-26, no. 1, pp. 65–74, Jan. 1988.
[41] C. Cheng, H. Li, J. Peng, W. Cui, and L. Zhang, "Hyperspectral image classification via spectral-spatial random patches network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 4753–4764, 2021.
[42] C. Rodarmel and J. Shan, "Principal component analysis for hyperspectral image classification," *Surveying Land Inf. Sci.*, vol. 62, no. 2, pp. 115–122, Jun. 2002.
[43] Q. Shenming, L. Xiang, and G. Zhihua, "A new hyperspectral image classification method based on spatial-spectral features," *Sci. Rep.*, vol. 12, 2022, Art. no. 1541.
[44] Q. Du and N. H. Younan, "Dimensionality reduction and linear discriminant analysis for hyperspectral image classification," in *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.*, 2008, pp. 392–399.
[45] D. Uchaev and D. Uchaev, "Small sample hyperspectral image classification based on the random patches network and recursive filtering," *Sensors*, vol. 23, no. 5, 2023, Art. no. 2499.
[46] H. Chen, T. Wang, T. Chen, and W. Deng, "Hyperspectral image classification based on fusing S3-PCA, 2D-SSA and random patch network," *Remote Sens.*, vol. 15, no. 13, 2023, Art. no. 3402.
[47] J. Zabalza et al., "Novel two-dimensional singular spectrum analysis for effective feature extraction and data classification in hyperspectral imaging," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4418–4433, Aug. 2015.
[48] X. Zhang, X. Jiang, J. Jiang, Y. Zhang, X. Liu, and Z. Cai, "Spectral–spatial and superpixelwise PCA for unsupervised feature extraction of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. pp. 1–10, .
[49] Y. Dong, B. Du, L. Zhang, and L. Zhang, "Dimensionality reduction and classification of hyperspectral images using ensemble discriminative local metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2509–2524, May 2017.
[50] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.
[51] 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.
[52] T. Dundar and T. Ince, "Sparse representation-based hyperspectral image classification using multiscale superpixels and guided filter," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 246–250, Feb. 2019.
[53] Y. Shen, Y. Yu, Y. Wei, and H. Guo, "Hyperspectral image classification based on combining texture and multiscale guided filter feature of principal component," *Infrared Phys. Technol.*, vol. 127, 2022, Art. no. 104415.
[54] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.
[55] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.
[56] X. X. Zhu et al., "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.
[57] M. Caywood, B. Willmore, and D. Tolhurst, "Independent components of color natural scenes resemble V1 neurons in their spatial and color tuning," *J. Neurophysiol.*, vol. 91, pp. 2859–2873, 2004.
[58] M. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 2097–2104.
[59] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 3, pp. 862–873, Mar. 2009.
[60] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, "220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian Pine test site 3," *Purdue Univ. Res. Repository*, vol. 10, 2015, Art. no. 991.
[61] Y. Tarabalka, J. Chanussot, and J. Benediktsson, "Segmentation and classification of hyperspectral images using watershed transformation," *Pattern Recognit.*, vol. 43, no. 7, pp. 2367–2379, 2010.
[62] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, "WHU-HI: UAV-borne hyperspectral with high spatial resolution (H2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with CRF," *Remote Sens. Environ.*, vol. 250, 2020, Art. no. 112012.

Tugcan Dundar received the B.Sc. and M.Sc. degrees in electrical and electronics engineering from Gaziantep University, Gaziantep, Türkiye, in 2016 and 2019, respectively. He has been working toward the Ph.D. degree in electrical and electronics engineering from Gaziantep University, Gaziantep, Türkiye, since 2019.

From 2017 to 2019, he was a Research Assistant with Gaziantep University, where he is currently a Lecturer. His research interests include the areas of representation-based classification and machine learning with applications to remote sensing.

Taner Ince (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in electrical and electronics engineering from Gaziantep University, Gaziantep, Türkiye, in 2003, 2006, and 2012, respectively.

He was a Visiting Researcher with the University of California, Los Angeles, CA, USA, during August–April 2014, supported by the International Research Fellowship Program (2219) of the Scientific and Technological Research Council of Türkiye. From 2004 to 2012, he was a Research Assistant with Gaziantep University, where he is currently an Associate Professor. His research interests include compressed sensing, sparse representation, and machine learning with applications to remote sensing image analysis.