

Article

Refinement of Hyperspectral Image Classification with Segment-Tree Filtering

Lu Li ^{1,2}, Chengyi Wang ^{1,*}, Jingbo Chen ¹ and Jianglin Ma ¹

¹ Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Datun Road North 20A, Beijing 100101, China; lili@radi.ac.cn (L.L.); chenjb@radi.ac.cn (J.C.); majianglin2003@gmail.com (J.M.)

² The University of Chinese Academy of Sciences, Beijing 100049, China

* Correspondence: wangcycastle@163.com; Tel.: +86-170-9008-6991

Academic Editors: Qi Wang, Nicolas H. Younan, Carlos López-Martínez, Lenio Soares Galvao and Prasad S. Thenkabail

Received: 11 November 2016; Accepted: 9 January 2017; Published: 16 January 2017

Abstract: This paper proposes a novel method of segment-tree filtering to improve the classification accuracy of hyperspectral image (HSI). Segment-tree filtering is a versatile method that incorporates spatial information and has been widely applied in image preprocessing. However, to use this powerful framework in hyperspectral image classification, we must reduce the original feature dimensionality to avoid the Hughes problem; otherwise, the computational costs are high and the classification accuracy by original bands in the HSI is unsatisfactory. Therefore, feature extraction is adopted to produce new salient features. In this paper, the Semi-supervised Local Fisher (SELF) method of discriminant analysis is used to reduce HSI dimensionality. Then, a tree-structure filter that adaptively incorporates contextual information is constructed. Additionally, an initial classification map is generated using multi-class support vector machines (SVMs), and segment-tree filtering is conducted using this map. Finally, a simple Winner-Take-All (WTA) rule is applied to determine the class of each pixel in an HSI based on the maximum probability. The experimental results demonstrate that the proposed method can improve HSI classification accuracy significantly. Furthermore, a comparison between the proposed method and the current state-of-the-art methods, such as Extended Morphological Profiles (EMPs), Guided Filtering (GF), and Markov Random Fields (MRFs), suggests that our method is both competitive and robust.

Keywords: hyperspectral image classification; SELF; SVMs; Segment-Tree Filtering

1. Introduction

Hyperspectral image (HSI) classification is important for urban land use monitoring, crop growth monitoring, environmental assessment, etc. Various machine learning algorithms that process high-dimension data can be employed in pixel-wise classification, such as Support Vector Machines (SVMs) [1], Logistic Regression [2,3], Artificial Neural Networks (ANNs) [4], etc. However, these conventional approaches do not consider spatial HSI information between neighboring pixels, which can lead to noisy classification output. Including the spatial relationships between pixels can enhance the classification accuracy. For example, there is a high probability that a pixel shares the same class as its neighboring pixels if the similarity measure between them is high. Otherwise, if the similarity measure is low, this probability decreases. Therefore, HSI classification could be improved further by combining spatial and spectral features.

Many spatial-spectral methods have been proposed to incorporate spatial or contextual information. For example, spatial information was represented using Markov Random Fields (MRFs) in [5,6], and classification has been performed using α -Expansion [7] and Belief Propagation [8], which are commonly used max-flow/min-cut algorithms in MRF optimization. Another method is presented

in [9], namely Extended Morphological Profiles (EMPs). After the first two principal components of the HSI are computed using the Principal Component Analysis (PCA) method, spatial features are extracted by morphological operations. Together with spectral information, they are concatenated for HSI classification. As morphological operations such as opening and closing involve neighboring pixel calculations, contextual information is naturally utilized in this manner. Another example of employing contextual information is via texture analysis. In [10], a Gray Level Co-occurrence Matrix (GLCM) is used to extract this type of contextual information, which is then employed to concatenate spectral features used for classification. In [11], segmentation is employed to represent spatial information based on a minimum spanning tree method, and majority voting is used to assign a class label to each region. Similar to [11], methods based on segmentation [12–15] have attracted increased attention because they produce satisfactory results. However, there are some drawbacks to algorithms based on hard segmentation. For example, they assume that all pixels in the same region are homogenous. After segmentation is completed, the relationship between pixels in different regions is fully disconnected; thus, if the segmentation is incorrect, the accuracy decreases dramatically. Although an over-segmentation approach is applied in [11,15] to improve the similarity in a region, the computational complexity increases considerably. Therefore, these algorithms are not efficient because of the complex voting processes in thousands of regions. In [16], super-pixel segmentation is applied to feature extraction and then classification is conducted in a novel framework via multiple kernels, which avoid voting but the super-pixel method still needs over-segmentation. To make classification more efficient, Edge-Aware Filtering and Edge-Preserving Filtering (EAF and EPF) methods [17,18] can be applied. These methods have been adopted successfully in many computer vision applications, such as stereo matching [19], optical flow [20], image fusion [21], etc. Unlike image segmentation, the most prominent merit of the EAF method is that in homogeneous image areas, EAFs can generate smooth output, while in inhomogeneous image areas, they can adaptively preserve boundaries, even in challenging situations. In this paper, we implement a tree-structure EAF for HSIs that is based on the segment-tree algorithm [22,23] and combines the advantages of segmentation and EAF. Unlike other EAF methods, the window size does not need to be set in this scheme. It is difficult to establish a proper window size for bilateral and guided filters [17], largely because objects of interest display the most prominent features at different scales. Another merit of this scheme is that the segment-tree filter is more efficient than other EAFs because of its tree structure [24]. By traversing the tree in two sequential passes, from the leaves to the root and from the root to the leaves, every pixel in HSI can be filtered and labeled.

However, the Segment-Tree Filter cannot be used for original HSI directly because the computational cost of this method is extremely high and the Hughes phenomenon always makes the classification accuracy unsatisfactory. The hyperspectral bands that are contaminated by noise may destroy the true connection between neighboring pixels. To avoid this problem, there are several literature that introduce how to choose the bands of original HSI [25,26] or produce new salient features. In this paper, the Semi-supervised Local Fisher (SELF) discriminant analysis method [27] is employed to reduce dimensionality. The SELF method of feature reduction is used because it retains prior knowledge from training sets and statistical distribution of clusters, unlike methods such as PCA, Linear Discriminant Analysis (LDA)/Fisher Discriminant Analysis (FDA) [28], and Local FDA (LFDA) [29]. In practice, the segmentation will be less sensitive to the number of training samples based on the SELF method. Additionally, we can construct the Segment-Tree Filter using a limited number of bands, which can reduce the computational cost of segmentation. Because the extracted SELF bands can reduce the effect of noise, a limited number of bands can well represent the inherent spatial structure of the image, which can lead to better output.

The remainder of this paper is organized as follows. In Section 1, we discuss some related methods and processes, including initial classification, SELF, Graph-based Segment-Tree Filter construction, and filtering. In Section 2, the proposed HSI classification scheme is described in detail. The experimental

results are presented in Section 3. Finally, we draw our conclusions and present our outlooks for future research in Section 4.

2. HSI Classification Refinement Using Segment-Tree Filtering

A schematic diagram of the proposed method is shown in Figure 1.

1. Step 1: Construct the Segment-Tree Filter, which involves feature extraction using the SELF method followed by building a tree-structure filter for an HSI based on dimensionality reduction.
2. Step 2: Use a Multi-class SVM method to obtain the initial classification map.
3. Step 3: Perform Segment-Tree Filtering based on the Multi-class SVM, pixel-based initial classification map. By combining this initial classification map and the Segment-Tree Filter, we can incorporate spatial information and spectral features, adaptively. Finally, the HSI classification map can be derived from the result of Segment-Tree Filtering.

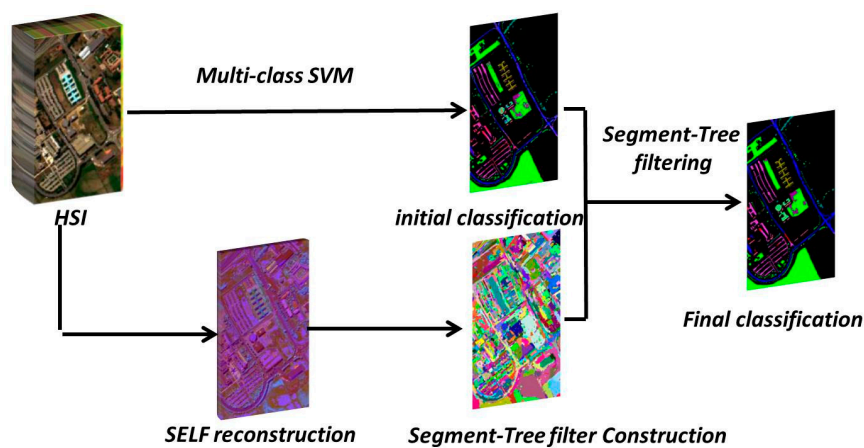


Figure 1. Workflow of Segment-Tree Filtering for HSI classification.

2.1. Initial Classification

In this paper, a Multi-class SVM classifier is adopted in the initial classification step. SVMs are widely used classifiers in remote sensing image classification. These supervised learning models are used for classification and regression in binary classification problems. In this paper, we utilize a Multi-class SVM from LIBSVM library with a radial basis function (RBF) kernel. In this method, a “one against one” strategy [30] is employed to extend the binary SVM to multi-class cases. The punishment parameter C and the spread of the kernel γ are optimally determined by cross-validation.

In most cases, the output of the initial classification is a probability map, which can be represented as a tensor [18] as follows:

$$\{\mathbf{M}_p^k \mid p(i, j); i = 1, 2, \dots, H; j = 1, 2, \dots, W; k = 1, 2, \dots, S\} \quad (1)$$

where (i, j) is the position of sample p in the image; H and W are the height and width of the image, respectively; k is the label of the sample; S is the total number of classes in the classification; and \mathbf{M}_p^k is the probability that the sample p belongs to the k th class. Based on the Multi-class SVM classifier, \mathbf{M}_p^k is either 0 or 1 depending on whether the sample belongs to the k th class.

$$\mathbf{M}_p^k = \begin{cases} 1 & \text{if } class(p) = k \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

2.2. Semi-Supervised Local Fisher Discriminant Analysis

In this section, we review SELF briefly. The SELF method seeks an embedding transformation such that the local, between-class scatter is maximized and the local, within-class scatter is minimized in both the training and test sets. We assume that HSI \mathbf{X} has m hyperspectral bands and n samples, and the training set \mathbf{X}' has n' samples. Then, the test set has $n - n'$ samples. There are three pre-defined input parameters: the trade-off parameter β , the dimensionality of the reconstruction space r , and the KNN parameter K . The five steps in this process are as follows:

1. Local scaling coefficient σ_i is pre-computed for each sample in the training set, which is equal to the Euclidean distance between the sample x_i and its K th nearest neighbor x_i^K among all samples in both the training and test sets.

$$\sigma_i = \|x_i - x_i^K\| \quad (3)$$

2. Local between-class weight matrix \mathbf{W}^{lb} and local within-class weight matrix \mathbf{W}^{lw} are computed as Equations (4) and (5), respectively. In this step, if two samples have the same label in the training set, σ_i is used to scale the local geometric structure with heat kernel weighting.

$$\mathbf{W}_{i,j}^{lb} = \begin{cases} (\frac{1}{n'} - \frac{1}{n'_{y_i}}) \exp(\frac{-\|x_i - x_j\|^2}{\sigma_i \sigma_j}) & \text{if } y_i = y_j \\ \frac{1}{n'} & \text{otherwise} \end{cases} \quad (4)$$

$$\mathbf{W}_{i,j}^{lw} = \begin{cases} \frac{1}{n'_{y_i}} \exp(\frac{-\|x_i - x_j\|^2}{\sigma_i \sigma_j}) & \text{if } y_i = y_j \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

3. The local between-class scatter matrix \mathbf{S}^{lb} and local, within-class scatter matrix \mathbf{S}^{lw} are calculated by Equations (6) and (7):

$$\mathbf{S}^{lb} = \mathbf{X}' \{diag(\mathbf{W}^{lb} \mathbf{1}_{n'})\} \mathbf{X}'^T; \quad (6)$$

$$\mathbf{S}^{lw} = \mathbf{X}' \{diag(\mathbf{W}^{lw} \mathbf{1}_{n'})\} \mathbf{X}'^T; \quad (7)$$

Note that $\mathbf{1}_{n'}$ is a unit column vector of size $n' \times 1$. Steps 1–3 are the same as those used in the LFDA procedure. In our procedure, only samples in the training set have been used at this point, and the statistical distribution of clusters has not been assessed or applied.

4. The covariance matrix \mathbf{S}^t is computed based on all samples in both the training and test sets as below:

$$\mathbf{S}^t = \frac{\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T}{n}; \quad (8)$$

where \bar{x} is the mean of all samples. Then, the regularized, local, between-class scatter matrix \mathbf{S}^{rlb} and the regularized, local, within-class scatter matrix \mathbf{S}^{rlw} are derived by Equations (9) and (10), respectively.

$$\mathbf{S}^{rlb} = (1 - \beta) \mathbf{S}^{lb} + \beta \mathbf{S}^t; \quad (9)$$

$$\mathbf{S}^{rlw} = (1 - \beta) \mathbf{S}^{lw} + \beta \mathbf{I}_m; \quad (10)$$

β is the trade-off parameter based on prior knowledge from the training set and the statistical distribution of clusters. Therefore, SELF maintains the advantages of LFDA and PCA. Note that \mathbf{I}_m is an $m \times m$ identity matrix.

5. Transformation matrix \mathbf{T} can be computed based on generalized eigenvalue decomposition. \mathbf{T} consists of weighted eigenvectors corresponding to the r largest eigenvalues. After \mathbf{T} is determined, all the pixels in the HSI can be reprojected to a new low-dimensional space.

The parameter β plays an important role in the algorithm. When it is relatively small (e.g., $\beta = 0.01$), SELF is nearly identical to LFDA, and SELF becomes increasingly similar to PCA when β approaches 1. β balances prior knowledge regarding the labels in the training set and the statistical distribution of clusters in the test set. In our experiment, we set β to 0.6, as we found that this value fully utilizes the advantages of the algorithm.

As a Semi-Supervised Learning (SSL) method, SELF can adapt based on the number of training samples. When the number of samples in a training set is small and prior knowledge about labels is limited and/or noisy, SELF can use the statistical distribution of clusters in the test set to offset these issues. We demonstrate this advantage in Figures 2 and 3. Figure 2 illustrates the reconstructed image of the Indian Pines dataset using PCA, LDA, LFDA, and SELF when the training samples accounted for only 1% of all samples. Figure 3 shows the same reconstructed image when the training percentage increases to 20%. In both figures, the color images (R, G, and B) are composed of the first three bands extracted using the corresponding feature-transformation methods. In Figure 2a, because the reconstruction based on LDA only relies on a small number of training set samples, the image exhibits considerable noise and error. When the number of training samples increases, LDA reconstruction is improved (e.g., there are fewer fractions and more homogenous areas in Figure 3a than in Figure 2a). Additionally, LFDA is more robust than LDA, even if the number of training samples is limited. However, as the number of samples increases, numerous false edges and fractions can be observed in the reconstructed image based on LFDA, as shown in Figures 2b and 3b. The boundary of the purple trapezoid in the bottom-left portion of Figure 2b was correctly extracted; however, it was incorrectly extracted in Figure 3b. As shown in Figures 2d and 3d (see the red rectangular region), SELF is robust regardless of the number of samples in the training set. Conversely, PCA does not depend on the training set; therefore, Figures 2c and 3c display the same reconstruction result. However, the comparison between PCA and SELF in the rectangular region in Figure 3d shows that PCA creates more segments and fractions because it aligns the boundaries of objects rather than classes, which causes more errors in subsequent processing steps compared to using SELF. In our experiment, the best reconstructed images based on SELF had 10 bands. Thus, the spectral dimension of the original HSI was dramatically reduced, but the most discriminatory information within the spectral bands was retained.

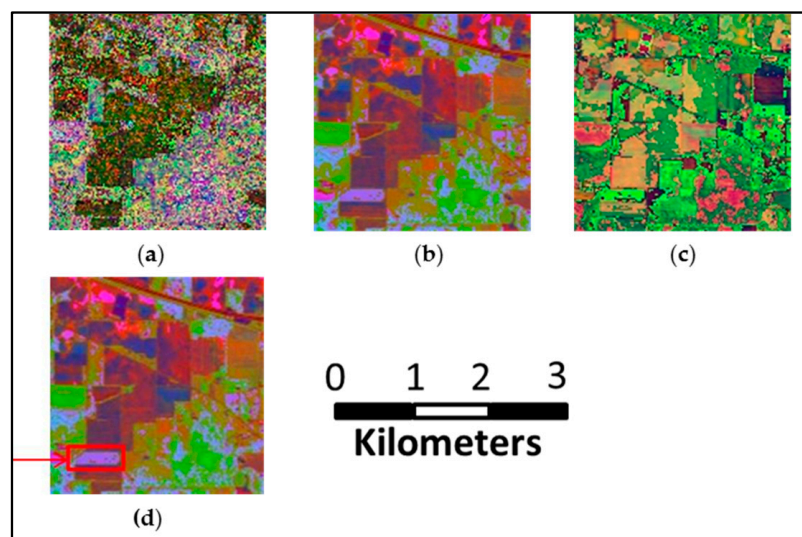


Figure 2. Indian Pines reconstructed using different methods of dimensional reduction: (a) LDA; (b) LFDA; (c) PCA; (d) SELF. The number of samples in the training set accounts for only 1% of all samples.

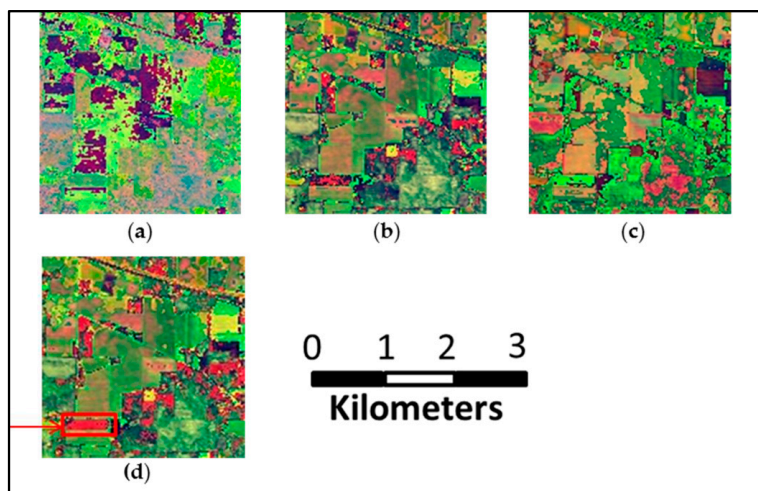


Figure 3. Indian Pines reconstructed using different methods of dimensional reduction: (a) LDA; (b) LFDA; (c) PCA; (d) SELF. The number of samples in the training set accounts for 20% of all samples.

2.3. Segment-Tree Filter Construction

The image transformed using SELF is used as input for the graph-based Segment-Tree Filter. The implementation of Segment-Tree Filtering is based on the methodology presented in [22,23], which use the Kruskal algorithm to construct a Minimum Spanning Tree (MST). The general workflow is summarized as follows. First, a graph $G = \{V, E\}$ is constructed for an image ($m \times n$ pixels), where V represents the vertices and each pixel is a vertex. E represents an edge that links four neighbors, and there are $m(n - 1) + n(m - 1)$ edges in total. A weight w_e is assigned for each edge E to represent the dissimilarity between the linked vertices. Several dissimilarity measures, such as the L1-norm, L2-norm, L_∞ -norm, and Spectral Angle Mapper (SAM), have been proposed in the literature [11]. In our experiment, SAM is used as the dissimilarity measure.

1. All the edges are sorted in ascending order according to their weights. This step can be performed efficiently using a quicksort algorithm [31], even if the number of edges is very large.
2. For each vertex, we initialize a tree $T_i(V_i, E_i)$.
3. A subtree is then built for each segment. Then, subtrees are merged based on the order of sorted edges. Segment-Tree Filtering is a variant of the conventional MST approach that considers an extra criterion to merge trees [22,32], as shown in Equation (11):

$$w_e \leq \min\left(\max(w_{T_p}) + \frac{k}{|T_p|}, \max(w_{T_q}) + \frac{k}{|T_q|}\right) \quad (11)$$

where w_e is the weight of the edge between subtrees T_p and T_q , $|T_p|$ is the number of vertices in the subtree T_p , and k is a constant. In our experiments, k is set to five times the standard deviation of all weights in the graph. If criterion (3) is satisfied, subtrees T_p and T_q are merged. Criterion (3) establishes a trade-off between the edge weights and the numbers of pixels in the subtrees. Initially, merging subtrees is easy because the number of pixels in each subtree is small. As the number of pixels increases, the criterion becomes increasingly rigorous; therefore, it is adaptive.

4. All the remaining edges that are not part of any subtree are sorted again. If the number of vertices in a subtree is smaller than a threshold T_0 , then the subtrees should be merged. In our experiment, $T_0 = 6$. This processing step is based on the improvement presented in [23] to omit small fractions caused by noise. The obtained subtrees are illustrated in Figure 4, in which each color represents an obtained subtree. As shown, constructed subtrees can be used to segment HSIs adaptively.

5. Finally, subtrees are merged until all vertices are included in the trees. For each tree, all the connected vertices exhibit the highest similarity and are within the shortest possible distance. As shown in Figure 5b, the edges of the final tree minimally cross the boundaries between two regions.

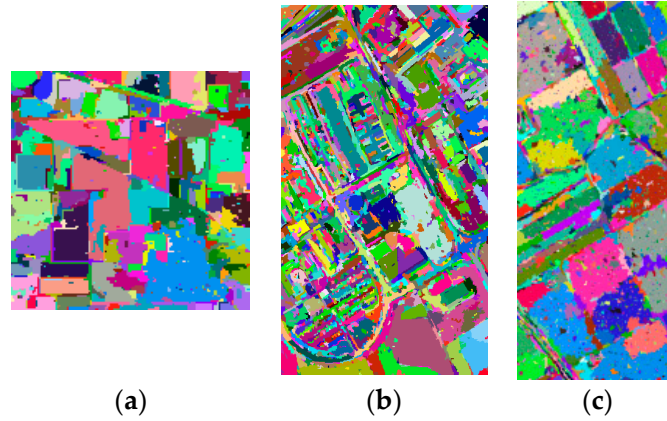


Figure 4. Subtrees constructed for different standard benchmarks: (a) Indian Pines; (b) University of Pavia; (c) Salinas. Each color segment represents a subtree.

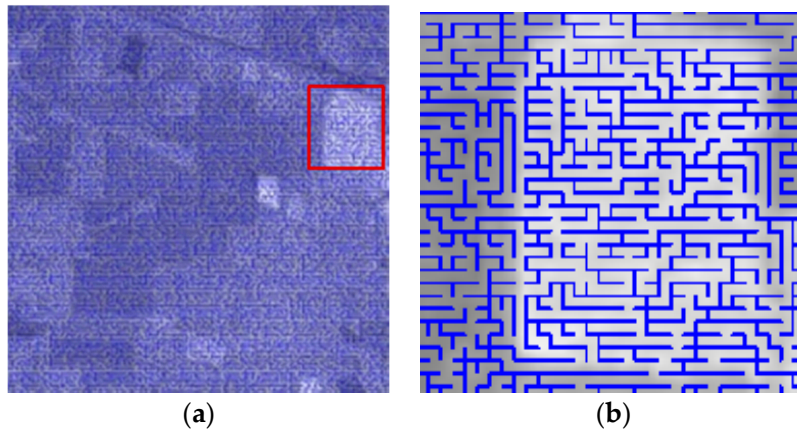


Figure 5. Tree structure of the Segment-Tree Filter for the Indian Pines dataset: (a) Image of the segment tree; (b) Close-up of the red rectangular region in (a).

2.4. Segment-Tree Filtering

The final step is to filter the initial probability maps using the tree-structure filter. The objective of the filtering process is to compute the aggregated probabilities. In the proposed approach, all vertices contribute to the aggregated probabilities, unlike using local neighbor methods. The non-local, aggregated probabilities $\bar{\mathbf{M}}_p^d$ can be defined as follows:

$$\bar{\mathbf{M}}_p^d = \sum_{q \in I} S(p, q) \mathbf{M}_q^d \quad (12)$$

where \mathbf{M}_p^d is defined in Equation (1). $S(p, q)$ is a weighting function that denotes the weight contribution of pixel q to p :

$$S(p, q) = \exp\left(- \sum_{i \in \text{path}(p, q)} w_{ei} / \gamma\right) \quad (13)$$

where w_{ei} is the weight of an edge in the tree structure connecting p and q and γ is a constant parameter.

Due to the tree structure, all the aggregated probabilities of class d in the image can be computed efficiently through traversing the tree in two sequential passes. In the first pass, forward filtering occurs from the leaf pixels to the root:

$$\mathbf{M}_p^{d\uparrow} = \mathbf{M}_p^d + \sum_{q \in c(p)} S(p, q) \mathbf{M}_q^{d\uparrow} \quad (14)$$

where $c(p)$ represents all the children of vertex $c(p)$. In the second pass, backward filtering occurs from the root to the leaf pixels:

$$\bar{\mathbf{M}}_p^d = S(pa(p), p) \bar{\mathbf{M}}_{pa(p)}^d + (1 - S^2(pa(p), p)) \mathbf{M}_q^{d\uparrow} \quad (15)$$

where $pa(p)$ represents the parent of vertex p .

As Figure 6 shows, vertex V_4 aggregates the probabilities of V_5 , V_6 , V_7 , and itself during the forward filtering step using Equation (16). During the backward filtering step, the probabilities of V_1 , V_2 , and V_3 contribute to V_4 based on Equation (17). After only two filtering steps, the aggregated probabilities of all vertices are computed, which reflects an extremely low computational complexity. Finally, the classification map is obtained using a simple Winner-Take-All (WTA) rule.

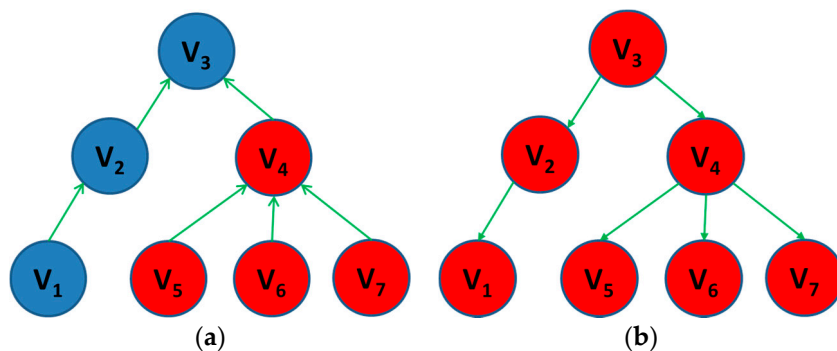


Figure 6. Segment-Tree Filtering in two sequential passes: (a) Forward filtering from the leaves to root; (b) Backward filtering from the root to leaves.

3. Experiments and Results

The proposed method has been implemented in C++ with the OpenCV library and Lapack library. The implemented code is available by contracting author. Evaluations were performed using three hyperspectral benchmark datasets as below:

1. The first HSI is a 2×2 mile portion of agricultural area over the Indian Pines region in Northwest Indiana, which was acquired by NASA's Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor. This scene with a size of 145×145 pixels, comprises 202 spectral bands in the wavelength range from 0.4 to $2.5 \mu\text{m}$, with spatial resolution of 20 m. The ground truth of scene (see Figure 7a) contains 16 classes of interest and total 10,366 samples. Due to the imbalanced number of available labeled pixels and a large number of mixed pixels per class, this dataset creates a challenge in HSI classification.
2. The second HSI is a 103-band image acquired by Reflective Optics Spectrographic Image System (ROSIS-03) sensor over the urban area of the University of Pavia, Italy. The spatial resolution is 1.3 m and the scene contains 610×340 pixels and nine classes. The number of samples is 42,776 in total. The ground truth of the scene is shown in Figure 8a.
3. The third HSI is also derived by AVIRIS sensor over Salinas Valley, California. This scene with a size of 512×217 pixels, and 204 spectral bands is used for classification. There are 16 classes in the ground truth image, which is shown in Figure 9a.

The overall accuracy (OA), average accuracy (AA), Kappa coefficient, and producer accuracy (PA) are used to assess the classification accuracy.

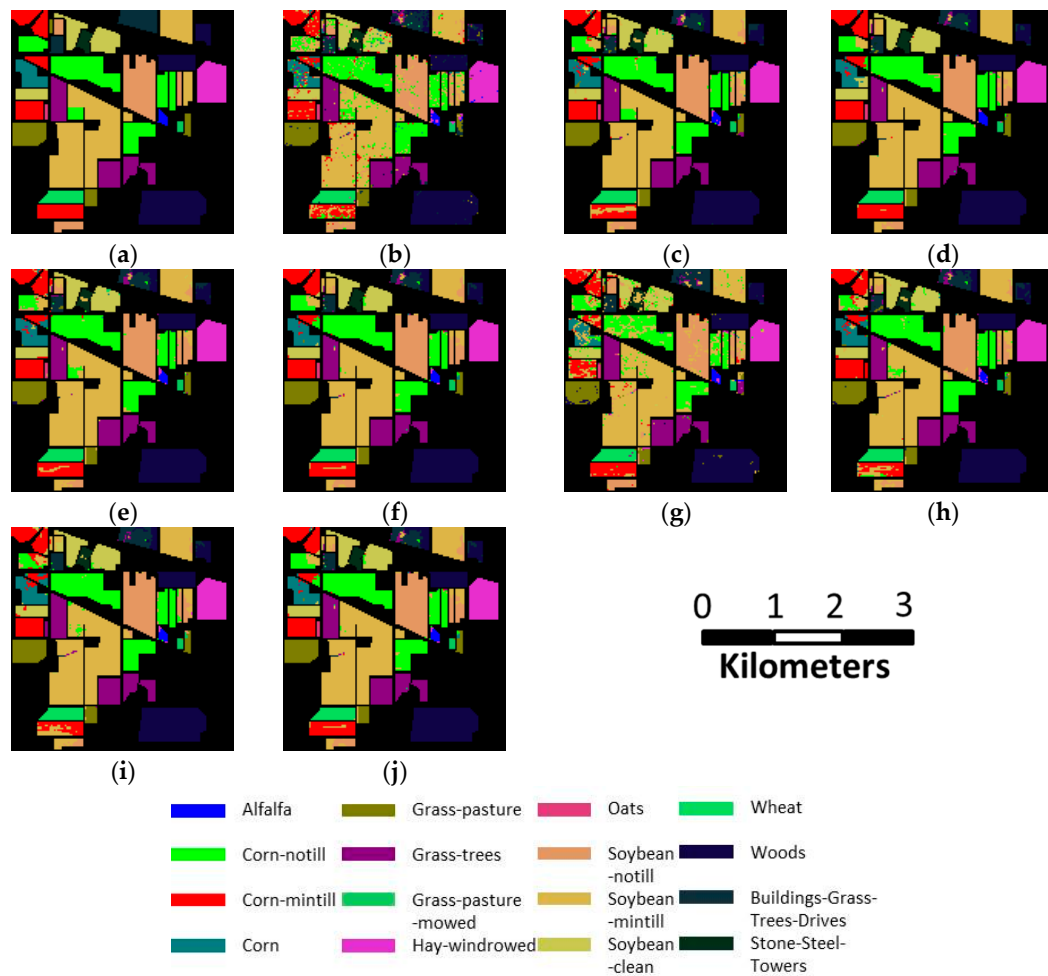


Figure 7. Classification results for Indian Pines: (a) Actual values; (b) Multi-class SVM; (c) ST; (d) PCA + ST; (e) LDA + ST; (f) LFDA + ST; (g) EMPs; (h) SVM + MRF; (i) PCA + GF ([18]); (j) The proposed method; (d–f) combine different dimensionality reduction methods (before “+”) with Segment-Tree Filtering (after “+”); (g–i) are other methods of spatial-spectral classification for HSIs.

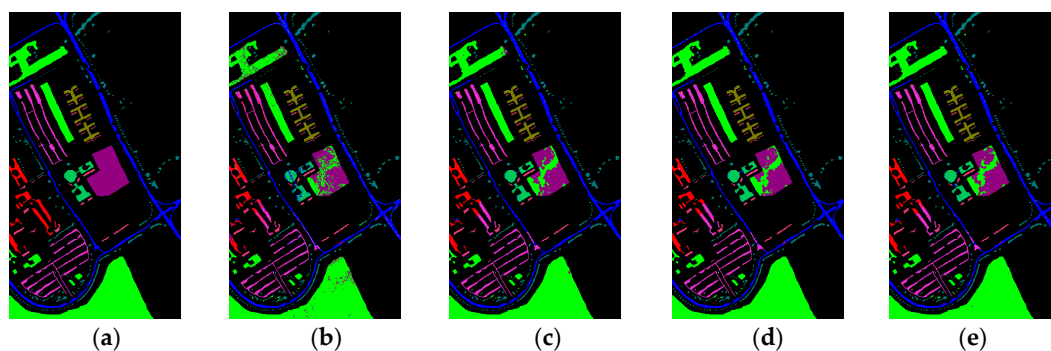


Figure 8. Cont.

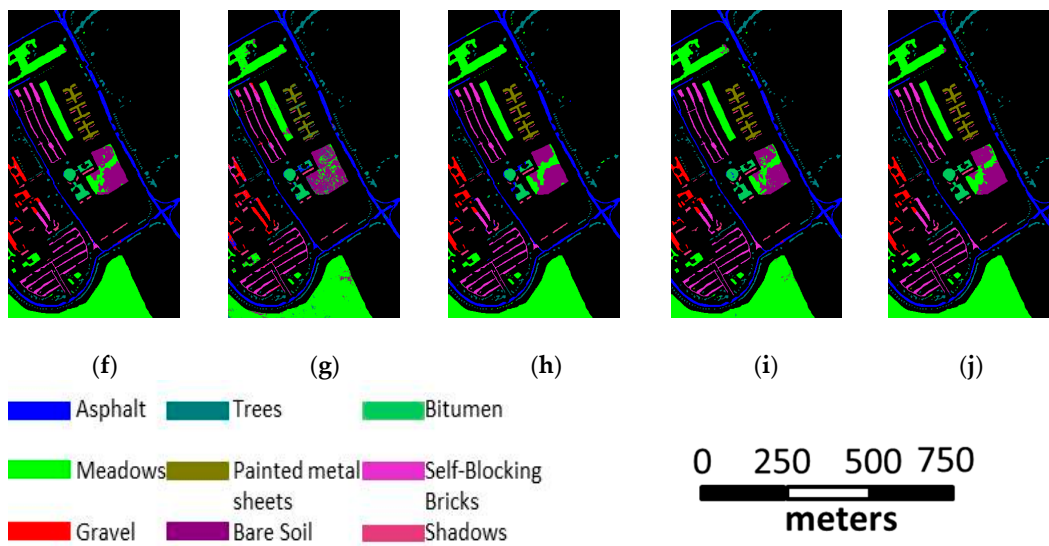


Figure 8. Classification results for Pavia University: (a) Actual values; (b) Multi-class SVM; (c) ST; (d) PCA + ST; (e) LDA + ST; (f) LFDA +ST; (g) EMPs; (h) SVM + MRF; (i) PCA + GF ([18]); (j) The proposed method. (d–f) combine different dimensionality reduction methods (before “+”) with Segment-Tree Filtering (after “+”); (g–i) are other methods of spatial-spectral classification for HSIs.

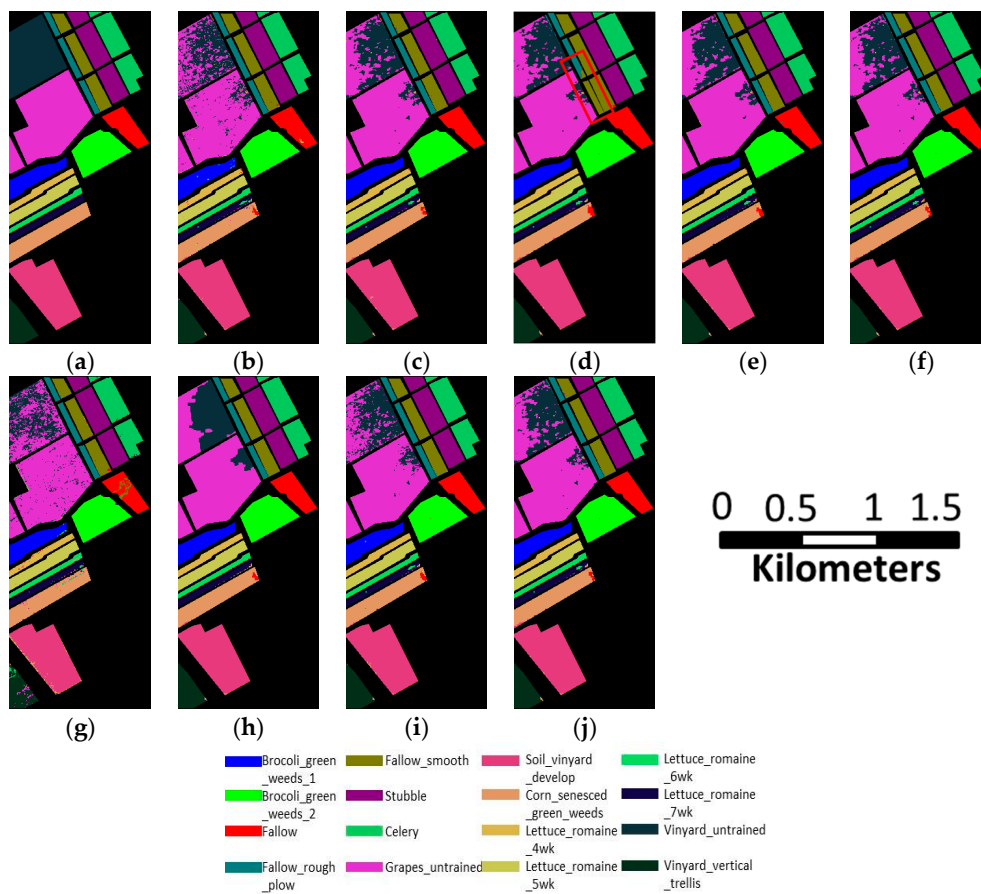


Figure 9. Classification results for Salinas; (a) Actual values; (b) Multi-class SVM; (c) ST; (d) PCA + ST; (e) LDA + ST; (f) LFDA + ST; (g) EMPs; (h) SVM + MRF; (i) PCA + GF ([18]); (j) The proposed method; (d–f) combine different dimensionality reduction methods (before “+”) with Segment-Tree Filtering (after “+”); (g–i) are other methods of spatial-spectral classification for HSIs.

3.1. Influence of Different Parameters

Some parameters in our proposed method may affect the classification accuracy, such as β , K , and r . Therefore, the Indian Pines dataset is used to test the importance of these parameters to the classification. In this case, the training samples account for 15% of all samples, regardless of their class. When one of the parameters is measured, the other parameters are fixed. Five-fold cross validation is used to tune all parameters. The influences of β , K , and r on the classification accuracy are shown in Figures 10–12, respectively. The influences of β and r are less than 1%, while the influence of K is greater than 1%. Figures 10–12 illustrate that the classification accuracy is the most sensitive to the influence of parameter K . Different dissimilarity measures are adopted during graph-based Segment-Tree Filter construction in our experiments, including the Minkowski distance (from 1 to 6 and infinity) and SAM, as shown in Figure 13. All the parameters affect the classification accuracy by approximately 1% to 2%, and the SAM dissimilarity measure was the largest in our Segment-Tree Filtering approach.

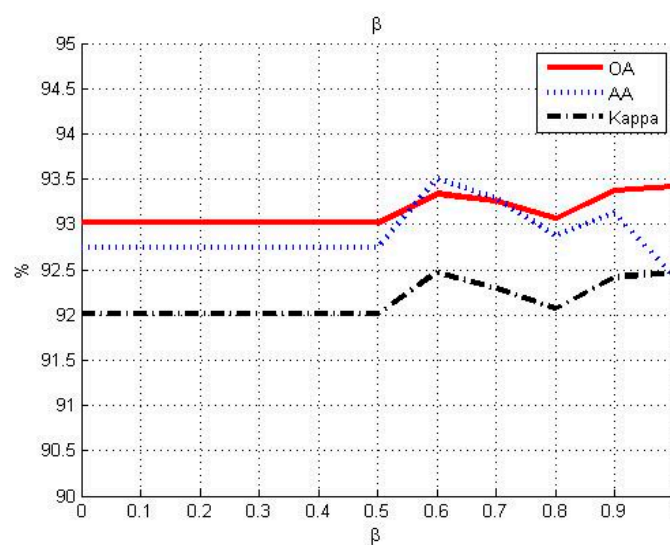


Figure 10. Influence of parameter β on the classification accuracy.

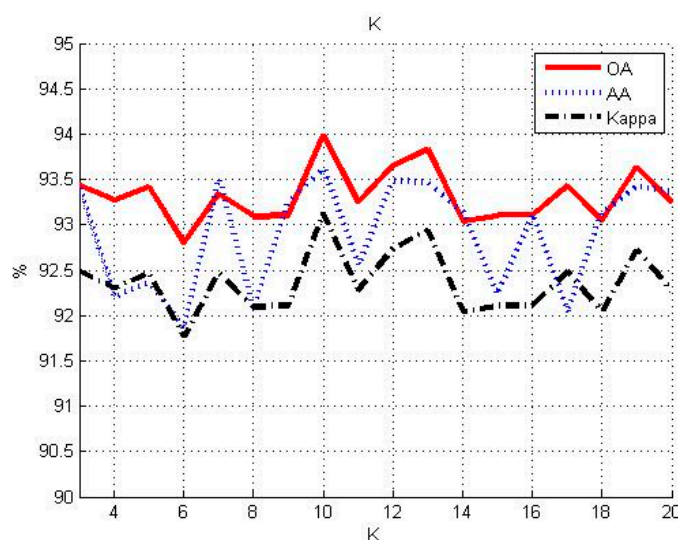


Figure 11. Influence of parameter K on the classification accuracy.

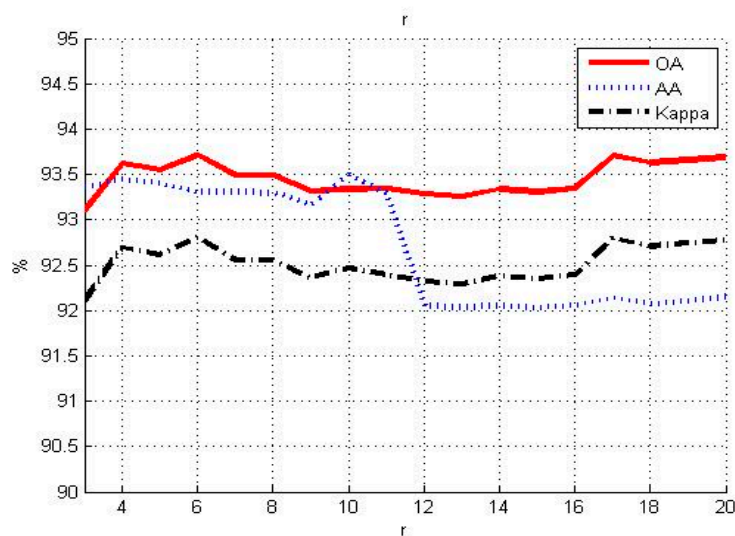


Figure 12. Influence of parameter r on the classification accuracy.

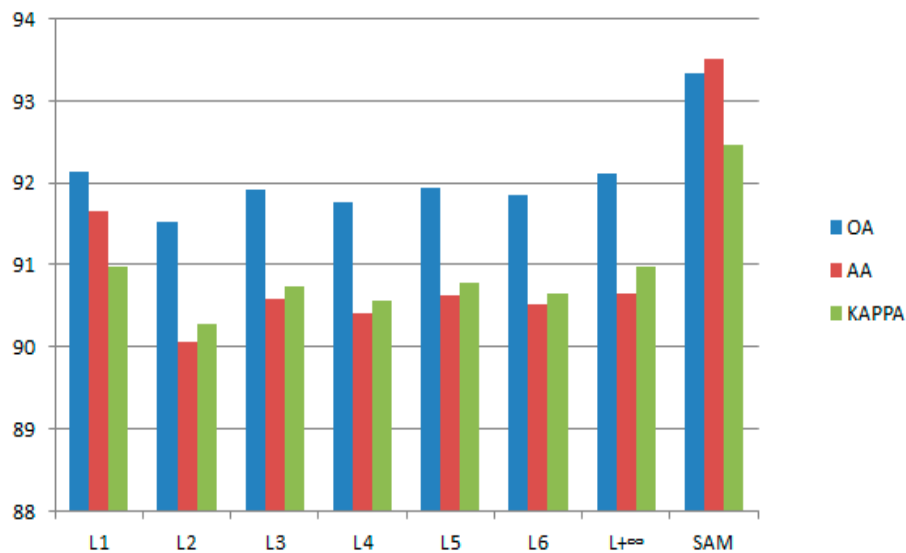


Figure 13. Influences of different dissimilarity measures on the classification accuracy.

3.2. Classification Accuracy Analysis

In this experiment, the training samples account for 15% of all the available samples, regardless of their class. The parameter settings are summarized as follows. In the initial classification step, the parameters were based on observed values, as discussed in Section 2.1. In the SELF transformation step, $\beta = 0.6$, $K = 7$, and $r = 10$. In the graph-based Segment-Tree construction step, SAM is used as the dissimilarity measure, and in the Segment-Tree Filtering step, γ is set as three times the standard deviation of w_e .

All experiments would be repeated five times according to different sampling training-set. The average of the five classification accuracies is recorded. The results of our proposed method based on analyses of the Indian Pines, University of Pavia, and Salinas datasets are shown in Tables 1–3, respectively. The visual results of the initial classification of each HSI are shown in Figure 7b, Figure 8b, and Figure 9b, respectively. Although SVMs are powerful classifiers, the results of the initial classification based solely on spectral features contain substantial noise. However, our proposed method greatly improves the classification accuracy after Segment-Tree Filtering, as shown

in Figure 7j, Figure 8j, and Figure 9j. The OA increases by 8.56% for the Indian Pines dataset, 4.64% for the Pavia University dataset, and 1.22% for the Salinas dataset. The largest PA increase is 32.60% for the Indian Pines dataset, followed by 22.71% for the Pavia University dataset and 3.41% for the Salinas dataset.

Table 1. Number of training and test samples from the Indian Pines dataset and the classification accuracies (in percentages) of different methods (the bolded item in each line means the best accuracy).

Class	Training/Test	SVM	ST	PCA + ST	LDA + ST	LFDA + ST	EMP	SVM + MRF	PCA + GF	Proposal
Alfalfa	7/46	58.70	71.74	91.30	76.09	82.61	82.61	83.30	80.43	91.30
Corn-N	214/1428	81.65	92.86	89.22	90.68	89.64	71.57	88.51	89.29	92.37
Corn-M	125/830	75.90	83.25	96.51	91.81	92.65	76.99	80.84	81.20	92.16
Corn	36/237	64.56	85.23	76.79	91.56	93.67	62.87	87.34	89.87	88.61
Grass-Pa	73/483	89.23	90.89	92.75	91.10	92.13	78.05	91.72	93.17	93.37
GrassT	109/730	96.99	99.73	99.45	99.59	99.45	98.36	99.18	100	99.73
GrassP	5/28	89.29	96.43	92.85	100	96.42	89.29	96.43	96.43	96.43
Hay-W	72/478	98.74	99.79	100	100	100	98.95	100	100	100
Oats	5/20	75.00	100	100	90.00	100	90.00	0.00	0.00	100
Soy-N	146/972	82.10	89.20	86.31	88.17	88.16	84.77	87.14	84.26	88.17
Soy-M	368/2455	85.62	94.70	95.62	95.11	94.50	93.40	93.40	95.52	94.58
Soy-C	89/593	74.20	94.77	99.66	96.46	97.64	71.16	93.76	95.45	97.98
Wheat	31/205	96.59	99.51	99.51	99.51	99.51	97.56	99.51	100	99.51
Woods	190/1265	95.42	97.23	98.50	97.94	98.26	98.26	96.60	98.66	98.10
Building	58/386	61.14	59.07	60.62	58.55	60.33	85.23	62.79	76.42	64.88
Stone-ST	14/93	87.10	87.10	98.92	87.10	98.92	60.22	88.17	94.62	98.92
OA		84.78	92.11	93.32	92.83	93.01	70.48	91.22	92.20	93.34
AA		82.01	90.09	92.44	90.85	92.74	67.26	89.07	85.96	93.50
Kappa		82.60	90.97	92.47	91.80	92.00	67.28	90.11	91.08	92.47

Table 2. Number of training and test samples from the Pavia University dataset and the classification accuracies (in percentages) of different methods (the bolded item in each line means the best accuracy).

Class	Training/Test	SVM	ST	PCA + ST	LDA + ST	LFDA + ST	EMP	SVM + MRF	PCA + GF	Proposal
Asphalt	995/6631	93.03	96.95	95.97	96.03	95.99	93.53	97.10	96.44	96.67
Meadows	2797/18,649	95.06	99.33	99.88	99.81	99.88	95.69	100	99.18	99.79
Gravel	315/2099	66.32	72.46	71.46	72.74	71.46	76.13	69.03	73.80	71.08
Trees	460/3064	93.05	94.65	94.39	94.09	94.45	98.63	90.31	97.03	93.79
P-M-S	202/1345	99.70	99.55	99.78	99.63	99.78	84.31	100	100	99.70
Bare Soil	754/5029	66.65	70.33	71.01	70.01	71.12	76.89	70.87	68.50	71.30
Bitumen	200/1330	77.21	96.99	97.07	95.78	97.74	84.81	83.38	97.22	99.92
Self-B B	552/3682	91.55	98.37	97.99	98.09	97.83	89.19	98.29	99.76	98.80
Shadows	142/947	100	98.94	96.09	91.12	96.09	77.82	98.10	100	98.32
OA		89.25	93.74	93.76	93.51	93.78	90.75	93.21	93.78	93.89
AA		86.95	91.95	91.52	90.81	91.59	86.34	89.68	92.43	92.15
Kappa		85.60	91.58	91.59	91.26	91.62	87.72	90.82	91.62	91.77

Table 3. Number of training and test samples from the Salinas dataset and the classification accuracies (in percentages) of different methods (the bolded item in each line means the best accuracy).

Class	Training/Test	SVM	ST	PCA + ST	LDA + ST	LFDA + ST	EMP	SVM + MRF	PCA + GF	Proposal
g_w_1	301/2009	98.80	100	100	100	100	98.95	100	99.80	100
g_w_2	491/3276	99.97	100	99.97	100	99.97	99.68	100	99.97	99.97
Fallow	296/1976	98.68	100	100	99.75	100	90.13	100	99.89	100
Fallow_r_p	209/1394	99.35	98.21	60.04	99.50	98.92	99.14	99.43	100	98.92
Fallow_s	401/2678	98.28	98.81	99.22	98.62	98.62	96.83	99.10	98.92	98.62
Stubble	593/3959	99.97	99.92	99.90	99.97	99.87	99.75	100	99.97	99.87
Celery	536/3579	99.35	99.61	99.66	99.52	99.64	98.46	99.89	99.78	99.64
Grapes_u	1690/11,271	92.11	95.03	95.20	95.16	95.51	95.40	94.58	95.11	95.52
Soil_v_d	930/6203	99.43	99.50	99.87	99.48	99.69	96.45	99.79	99.47	99.68
C_s_g_w	491/3278	93.62	96.06	94.63	95.18	95.79	94.97	96.86	95.73	95.64
L_r_4	160/1068	96.44	99.25	96.72	97.94	99.63	94.66	98.69	99.25	99.63
L_r_5	289/1927	99.53	100	95.69	100	100	99.79	100	100	100
Le_r_6	137/916	98.03	97.82	98.25	97.49	98.14	95.09	98.25	98.80	98.14
L_r_7	160/1070	92.05	93.74	93.74	93.93	93.48	96.07	96.26	93.64	93.49
V_u	1090/7268	56.89	57.33	56.70	56.84	57.75	50.30	57.04	57.66	57.85
V_v	271/1807	98.83	99.06	99.11	99.00	99.06	89.60	99.39	99.45	99.06
OA		91.56	92.59	91.35	92.49	92.77	90.32	92.76	92.72	92.78
AA		95.08	95.89	93.05	95.77	96.00	93.45	96.17	96.09	96.00
Kappa		90.57	91.73	90.33	91.60	91.91	89.17	91.92	91.85	91.92

3.3. Influences of Different Techniques for Dimensionality Reduction

In the above section, we illustrated that incorporating spatial information can improve the classification accuracy. In the following section, we will evaluate how different techniques for dimensionality reduction can affect the classification accuracy. First, we assess the classification accuracy with/without dimensionality reduction. If no dimensionality reduction method is used, the Segment-Tree Filter is constructed using the original HSI. Figure 7c, Figure 8c, and Figure 9c show the results of classification using this scheme. Because redundant bands negatively affect segmentation, the classification accuracy using the original bands in the three HSI datasets is less than that produced using the proposed method. The fourth column in Tables 1–3 illustrates that reducing the dimensionality is necessary to increase the classification accuracy, and the average OA increased by approximately 0.53%. In addition, dimensionality can considerably improve the computational speed.

Next, we examine how different methods of dimensionality reduction can affect the classification accuracy. Figures 7–9 show the classification results for various methods, including the PCA, LDA, and LFDA methods; however, the classification accuracy produced by SELF is better than the accuracies of those methods.

Figure 7d–f, Figure 8d–f, and Figure 9d–f show the classification results for PCA, LDA, and LFDA, respectively. As expected, the OA and Kappa coefficient of SELF is the highest among these methods and the AA is the highest for the Indian Pines dataset and the second highest for the Pavia University and Salinas datasets. Although PCA sometimes performed better than SELF in some PAs, PCA with Segment-Tree Filtering is not a robust algorithm and can easily over-smooth spatial information, as illustrated in Figure 9d. The results illustrated in the red rectangle exhibit considerable classification error-based PCA, and the PA of Fallow_r_p decreases from 99.35% to 60.04% in the fourth row of Table 3.

3.4. Comparison to Other Methods of Spectral-Spatial Classification

As discussed in Section 1, spectral-spatial classification is a powerful method of combining contextual information. Therefore, we compared the proposed method to other common methods of spectral-spatial classification. We implemented the following spectral-spatial classification algorithms in our analysis.

1. The first algorithm is based on EMPs [9]. In [9], a neural network classifier was applied; however, an SVM is used instead of a back-propagation neural network to create a fair comparison. The EMPs are shown in Figure 7g, Figure 8g, and Figure 9g.
2. The second approach uses MRFs [6] with Multi-class SVM, which is, to the best of our knowledge, the state of the art method for spatial-spectral image classification based on remote sensing. Multi-class SVM is used as the initial classifier, and the spatial optimization is performed using the max-flow/min-cut algorithms. In our experiment, α -expansion is adopted, and the regularization coefficient is fixed to 0.5. The results of the SVM with MRFs are shown in Figure 7h, Figure 8h, and Figure 9h.
3. The third approach is based on a Guided Filter with PCA [18]. The size and blur degree in Guided Filtering are tuned adaptively by cross-validation. The results of this classification are shown in Figure 7i, Figure 8i, and Figure 9i.

The proposed method produced results that were more accurate than those of the EMP-based method for the Indian Pines dataset; however, the results were similar for the other datasets. This result suggests that the proposed method is more suitable for different datasets compared to the EMP-based approach.

The classification accuracies of the proposed method and the SVM method with MRFs were nearly equal. This result indicates that the proposed method achieved an accuracy comparable to that of a state of the art method for spatial-spectral HSI classification. Furthermore, when the number of training

samples within a class is very small, e.g., the PAs of the “Oats” class in Table 1, the classification accuracy of the proposed approach is perfect, while the SVM with MRFs method fails. This occurs because MRFs over smooth spatial features; thus, the regularization parameter requires complex tuning steps. Therefore, our proposed approach is more robust than the SVM with MRFs method.

The classification accuracy of the proposed method is slightly higher than that of the Guided Filter based on the OAs of the three datasets.

However, we computed the computational times associated with the three HSIs based on GF with PCA [18] and our method. We assume that N pixels, M bands, D classes, and local window size R are used for the reconstruction of the HSIs. The complexity of Segment-Tree Filtering is $O(ND)$, and that of Guided Filtering is $O(NDM)$. For the Indian Pines dataset, GF needed 2.6138 s to process 10 bands in the compressed dataset, while the proposed method required only 0.0536 s (all programs were executed using an Intel(R) Xeon(R) CPU E5-2620 with 24 GB of RAM.). Guided Filtering is slower because it computes the inverse covariance matrix for each sample. In extreme cases, Guided Filtering using an original HSI as the guide image can be time consuming and ineffective.

3.5. Effect of the Training Set on Classification

In this section, we assess how the number of training samples affects the classification accuracy of the proposed method. Thus, we varied the training sample size from 1% of all samples to 20% of all samples. We found that when the training sample size increases, the classification accuracy also increases. Therefore, we only illustrate how the number of training samples affects the classification accuracy of the Indian Pines dataset, as shown in Figure 14. The classification accuracy improves considerably until the number of pixels in the training set reaches 5% of the total pixel number. Then, the accuracy continues to improve but at a lower rate.

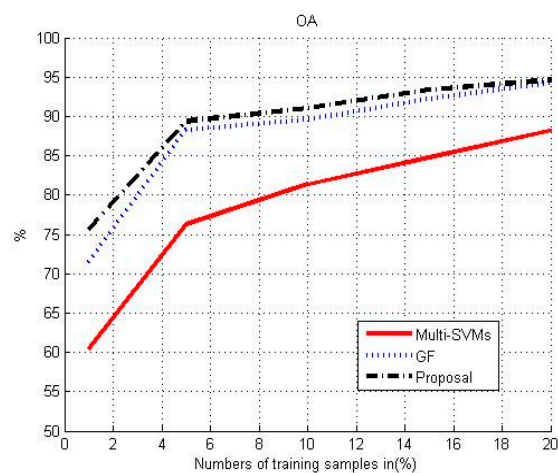


Figure 14. Classification accuracy based on the number of training samples for the Indian Pines dataset.

We also evaluate the effects of the training sample size on the classification accuracies of the Guided Filter [18] and Multi-class SVM methods. As shown in Figure 14, the proposed method and Guided Filter approach improve the classification accuracy regardless of the size of the training set, and the proposed method yields better results. Furthermore, the advantage is larger when the size of the training set is small.

4. Conclusions

A novel and efficient approach based on a Segment-Tree Filter has been proposed for hyperspectral image classification. Our proposed approach is based on spatial-spectral filtering, which is a special EAF. This filter construction utilizes both spectral features using a SELF transformation and spatial

information using a Segment-Tree algorithm. After an initial classification map is generated by Multi-class SVM, we can filter the map using the Segment-Tree Filter. One advantage of our proposed approach is that the classification accuracy has been improved dramatically. Experimental results show that the proposed method produced a high classification accuracy for hyperspectral image benchmark sets, including 93.34% for the Indian Pines dataset, 93.89% for the Pavia University dataset, and 92.78% for the Salinas dataset. Compared to other spatial-spectral methods, another advantage of the proposed method is that it provides a more robust classification approach for different datasets and training sets of different sizes.

In the future, two major aspects of our approach could be improved. First, dimensionality reduction using could be performed in SELF to reconstruct HSIs with nonlinear projections. Second, other classifiers, including fuzzy classifiers, could be applied to improve the classification accuracy.

Acknowledgments: The authors would like to thank D. Landgrebe and P. Gamba for providing the Indian Pines and Pavia University respectively hyperspectral data set available to the community, and Website http://www.ehu.es/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes for downloading all the three hyperspectral data sets. This research was supported in part by the Natural Science Foundation of China with Project IDs 41301499 and 41401474, and the National Science and Technology Major Project with Project ID 30-Y20A03-9003-15/16.

Author Contributions: Lu Li and Chengyi Wang conceived and designed the experiments; Lu Li performed the experiments; Lu Li and Jingbo Chen analyzed the data; Lu Li and Jianglin Ma wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [[CrossRef](#)]
2. Böhning, D. Multinomial logistic regression algorithm. *Ann. Inst. Stat. Math.* **1992**, *44*, 197–200. [[CrossRef](#)]
3. Li, J.; Bioucas-Dias, J.; Plaza, A. Semi-supervised hyperspectral image segmentation using multinomial logistic regression with active learning. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4085–4098.
4. Ratle, F.; Camps-Valls, G.; Weston, J. Semi-supervised neural networks for efficient hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 2271–2282. [[CrossRef](#)]
5. Li, W.; Prasad, S.; Fowler, E. Hyperspectral image classification using Gaussian mixture models and Markov random fields. *IEEE Lett. Geosci. Remote Sens.* **2014**, *11*, 153–157. [[CrossRef](#)]
6. Li, J.; Bioucas, M.; Antonio, P. Spectral-spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 809–823. [[CrossRef](#)]
7. Boykov, O.; Veksler, O.; Zabih, R. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 1222–1239. [[CrossRef](#)]
8. Li, J.; Bioucas, M.; Antonio, P. Spectral-spatial classification of hyperspectral data using loopy belief propagation and active learning. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 844–856. [[CrossRef](#)]
9. Benediktsson, A.; Palmason, A.; Sveinsson, R. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 480–491. [[CrossRef](#)]
10. Zortea, M.; de Martino, M.; Serpico, S. A SVM Ensemble approach for spectral-contextual classification of optical high spatial resolution imagery. In *Proceeding of the IEEE International Conference on Geoscience and Remote Sensing Symposium*, Boston, MA, USA, 23–28 July 2007.
11. Yuliya, T.; Jocelyn, C.; Jón Atli, B. Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers. *IEEE Trans. Syst. Man Cybern. B Cybern.* **2010**, *40*, 1267–1279. [[CrossRef](#)]
12. Fu, W.; Li, S.; Fang, L. Spectral-spatial Hyperspectral image classification via superpixel merging and sparse representation. In *Proceeding of the IEEE Proceedings on Geoscience and Remote Sensing Symposium*, Milan, Italy, 26–31 July 2015.
13. Duan, W.; Li, S.; Fang, L. Spectral-spatial hyperspectral image classification using superpixel and extreme learning machines. In *Pattern Recognition, Proceedings the 6th Chinese Conference, Changsha, China, 17–19 November 2014*; Springer: Berlin/Heidelberg, Germany, 2014; Volume 483, pp. 159–167. [[CrossRef](#)]

14. Weihua, S.; David, M. Trilateral filter on multispectral imagery for classification and segmentation. *Proc. SPIE* **2011**. [[CrossRef](#)]
15. Tarabalka, Y.; Benediktsson, A.; Chanussot, J. Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 2973–2987. [[CrossRef](#)]
16. Fang, L.; Li, S.; Duan, W. Classification of hyperspectral images by exploiting spectral-spatial information of superpixel via multiple kernels. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 6663–6673. [[CrossRef](#)]
17. He, K.; Sun, J.; Tang, X. Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1397–1409. [[CrossRef](#)] [[PubMed](#)]
18. Kang, X.; Li, S.; Jón Alti, B. Spectral-spatial hyperspectral image classification with edge-preserving filtering. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 2666–2677. [[CrossRef](#)]
19. Hosni, A.; Rhemann, C.; Bleyer, M.; Rother, C.; Gelautz, M. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 504–511. [[CrossRef](#)] [[PubMed](#)]
20. Hosni, A.; Rhemann, C.; Bleyer, M.; Gelautz, M. Temporally consistent disparity and optical flow via efficient spatio-temporal filtering. In Proceedings of the 5th Pacific Rim Symposium, Gwangju, Korea, 20–23 November 2011; Springer: Berlin, Germany, 2011; pp. 165–177.
21. Shutao, L.; Xudong, K.; Jianwen, H. Image fusion with guided filtering. *IEEE Trans. Image Process.* **2013**, *22*, 2864–2875. [[CrossRef](#)] [[PubMed](#)]
22. Felzenszwalb, F.; Huttenlocher, P. Efficient graph-based image segmentation. *Int. J. Comput. Vis.* **2004**, *59*, 167–181. [[CrossRef](#)]
23. Mei, X.; Sun, X.; Dong, W.; Wang, H.; Zhang, X. Segment-tree based cost aggregation for stereo matching. In Proceeding of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013.
24. Yang, X. A Non-local cost aggregation method for stereo matching. In Proceeding of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012.
25. Wang, Q.; Lin, J.; Yuan, Y. Salient band selection for hyperspectral image classification via manifold ranking. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 1279–1289. [[CrossRef](#)] [[PubMed](#)]
26. Yuan, Y.; Lin, J.; Wang, Q. Dual clustering based hyperspectral band selection by contextual analysis. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 1431–1445. [[CrossRef](#)]
27. Sugiyama, M.; Idé, T.; Nakajima, S.; Sese, J. Semi-supervised local Fisher discriminant analysis for dimensionality reduction. *Mach. Learn.* **2010**, *78*, 35–61. [[CrossRef](#)]
28. Fisher, R.A. The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **1936**, *7*, 179–188. [[CrossRef](#)]
29. Sugiyama, M. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *J. Mach. Learn. Res.* **2007**, *8*, 1027–1061.
30. Huang, T.K.; Weng, R.C.; Lin, C.J. Generalized bradley-terry models and multi-class probability estimates. *J. Mach. Learn. Res.* **2006**, *7*, 85–115.
31. Cormen, T.; Leiserson, C.; Rivest, R.; Stein, C. *Introduction to Algorithms*, 3rd ed.; MIT Press: London, UK, 2005; pp. 170–174.
32. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905. [[CrossRef](#)]



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).