Journal of
Applied Remote Sensing

RemoteSensing.SPIEDigitalLibrary.org

Hyperspectral data classification improved by minimum spanning forests

Ricardo Dutra da Silva
Helio Pedrini

SPIE.

Ricardo Dutra da Silva, Helio Pedrini, "Hyperspectral data classification improved by minimum spanning forests," *J. Appl. Remote Sens.* **10**(2), 025007 (2016), doi: 10.1117/1.JRS.10.025007.

Hyperspectral data classification improved by minimum spanning forests

Ricardo Dutra da Silva[a] **and Helio Pedrini**[b,*]
[a]Federal University of Technology, Informatics Department, Av. Sete de Setembro 3165, Curitiba, Paraná 80230-901, Brazil
[b]University of Campinas, Institute of Computing, Av. Albert Einstein 1251, Campinas, São Paulo 13083-852, Brazil

Abstract. Remote sensing technology has applications in various knowledge domains, such as agriculture, meteorology, land use, environmental monitoring, military surveillance, and mineral exploration. The increasing advances in image acquisition techniques have allowed the generation of large volumes of data at high spectral resolution with several spectral bands representing images collected simultaneously. We propose and evaluate a supervised classification method composed of three stages. Initially, hyperspectral values and entropy information are employed by support vector machines to produce an initial classification. Then, the *K*-nearest neighbor technique searches for pixels with high probability of being correctly classified. Finally, minimum spanning forests are applied to these pixels to reclassify the image taking spatial restrictions into consideration. Experiments on several hyperspectral images are conducted to show the effectiveness of the proposed method. © *2016 Society of Photo-Optical Instrumentation Engineers (SPIE)* [DOI: 10.1117/1.JRS.10.025007]

Keywords: hyperspectral images; supervised classification; minimum spanning forests; image analysis.

Paper 15782 received Nov. 8, 2015; accepted for publication Mar. 24, 2016; published online Apr. 26, 2016.

1 Introduction

The technology of remote sensing[1–4] has applications in several fields, including agriculture, forestry, hydrology, meteorology, mapping, environmental monitoring, land use, mineral exploration, and military surveillance.

Airborne and satellite sensors are typically used to measure the electromagnetic radiation reflected from the Earth's surface. Different types of surface materials, such as soil, water, vegetation, and minerals, can be distinguished from spectral reflectance signatures.

Advances in image acquisition techniques have allowed the extraction of large volumes of data with high spectral resolution. Such data, called hyperspectral, refer to the collection of information consisting of various spectral bands, which are defined in a wavelength range.

Hyperspectral image analysis[5–11] is a challenging task and has received increasing interest in the community. Some inherent difficulties in the analysis of such data include its high dimensionality and noise contamination.

Although the spectral bands provide valuable information with high discriminative power that allows the identification of distinct classes of objects, some initiatives for combination of spatial and spectral information have been also presented in the literature.[12–15]

This work investigates the use of spectral and spatial information to classify hyperspectral data. A three-stage supervised classifier is employed to determine the class for each pixel. An initial classification is performed by support vector machines (SVMs), whose result is used as input to *K*-nearest neighbor (K-NN) that searches for correctly classified pixels. Such pixels are used as seeds for a minimum spanning forests (MSF) algorithm to generate a final spatial-based classification. The proposed method is applied to a set of hyperspectral images, and its

*Address all correspondence to: Helio Pedrini, E-mail: helio@ic.unicamp.br

performance is compared to other classification approaches, including two which are also based on MSF.[13,15]

The remainder of the paper is organized as follows. Section 2 briefly describes concepts related to the topic under investigation. The proposed method for hyperspectral data is presented in Sec. 3. Experimental results are described and discussed in Sec. 4. Section 5 presents the conclusions and directions for future work.

2 Background

The remote sensing area[1–3,16] has employed aircraft and satellites to acquire information about the Earth's surface. In addition to traditional aerial photographs operating in the visible portion of the electromagnetic spectrum, continuous technological development has allowed the acquisition of information in other portions, such as microwave and thermal infrared.

Hyperspectral sensors, also known as imaging spectrometers, are instruments that capture images at several contiguous bands of spectral radiance with wavelengths typically varying between 0.4 and 2.5 μm. Therefore, each pixel in the image contains a continuous spectrum that can be used to characterize objects in the scene.

Several hyperspectral sensors have been used to acquire data for a number of different applications, such as thematic mapping, geology, hydrology, agricultural assessment, mineral exploration, and study of global environmental and climate changes. Some important sensors include the hyperspectral digital imagery collection experiment,[17] airborne visible/infrared imaging spectrometer,[18] reflective optics system imaging spectrometer,[19] compact airborne spectrographic imager,[20] airborne imaging spectrometer,[21] airborne imaging spectrometer for different applications,[22] geophysical and environmental research imaging spectrometer,[23] hyperspectral mapper,[24] aerosol mass spectrometer,[25] airborne prism experiment,[26] compact high-resolution imaging spectrometer,[27] digital airborne imaging spectrometer,[28] and thermal infrared multispectral scanner.[29]

The analysis of hyperspectral data has received growing interest from the community, since they have spectral attributes with high discriminative power. Several classification methods have been proposed in the literature for labeling the image pixels according to specific classes, including Bayesian networks,[30] SVM,[31] neural networks,[32] nearest neighbors,[33] clustering algorithms,[34] and decision trees,[35] among others.

The process of classification consists in categorizing pixels or regions of the images into a set of different classes, such that pixels with similar attributes belong to the same class. Hyperspectral image classification methods can be generally divided into unsupervised and supervised techniques. Unsupervised techniques identify groups of pixels without prior knowledge of the data, whereas supervised techniques employ known labeled samples to assign pixels to one of the classes.

Many pixelwise classification approaches[36,37] consider only spectral information of the pixels. Other methods take spatial information present in the image into account.[12,38–40] Some strategies for including spatial information in the classification are watershed,[41] minimum spanning forests,[13] graph cuts,[42] shortest paths,[43] random walks,[44] and Markov random fields.[45]

The image analysis process typically uses a more succinct representation of the data or its components (objects) called the feature vector, which stores the most representative attributes of the regions of the image. The main features extracted from images to assist the analysis process are the color, texture, and shape of objects. The number of attributes or characteristics determines the dimension of the feature vector and usually depends on the application area and the properties that need to be discriminated.

Due to the high dimensionality[46] of the feature vector that represents the spectral information contained in hyperspectral images, feature selection techniques[47] are usually applied as preprocessing to the data classification. A widely used technique for data compression and dimensionality reduction is the principal component analysis (PCA).[48]

3 Methodology

A general scheme for the proposed classification method is shown in Fig. 1. Given a set of hyperspectral bands and a set of classified samples (training data), the classification process

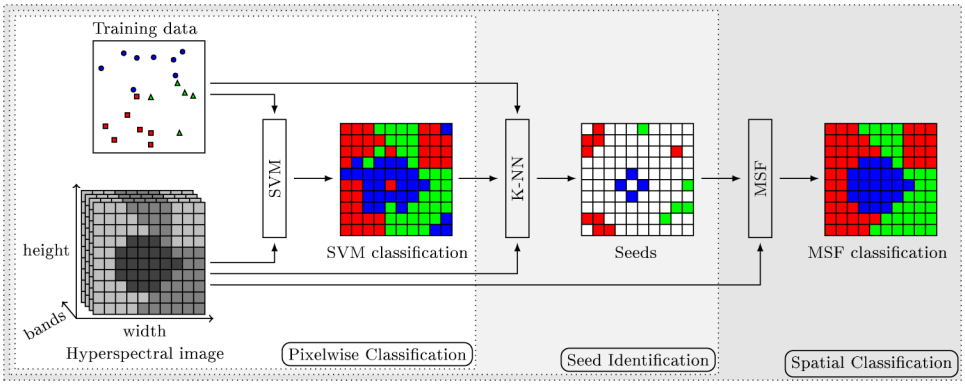


Fig. 1 Overview of the proposed classification method. The first step produces an initial classification through SVM. The second step compares, for each pixel, the classification assigned by SVM and the classes of K-NN. If the classes agree, the pixel is selected as a seed. The final step uses the seeds in an MSF algorithm to produce a spatial classification.

of the hyperspectral image is initially obtained by using SVM. In a second step, a K-NN search is performed in order to choose the pixels that are most likely to be correctly classified by SVM. Such pixels are chosen as seeds for the MSF algorithm to produce a spatial-based classification. The details on each step are described as follows.

The hyperspectral image is a matrix of size $m \times n \times l$, so that each pixel is a vector $\mathbf{p}(x,y) = [p(x,y,1), p(x,y,2), \ldots, p(x,y,l)]$. The set S_u of samples to be classified is the set of $m \times n$ pixels. For simplicity, a general pixel will be referred to as $\mathbf{p} = (p_1, p_2, \ldots, p_l)$, dropping the two-dimensional indices. The classified samples form a set S_c of pairs $s_i = (\mathbf{x}_i, y_i)$ and $i = \{1,2,\ldots,j\}$, such that the i'th pair has feature vector $\mathbf{x}_i \in \mathbb{R}^p$ and belongs to class y_i.

Two measures are investigated as descriptors for the elements of both S_u and S_c. The first corresponds to the hyperspectral values. In such a case, the descriptor of a pixel $\mathbf{p}$ is the pixel itself, the vector $\mathbf{p} = (p_1, p_2, \ldots, p_l)$. The second measure is the local entropy, which measures the intensity variation around a pixel. This information can be useful in distinguishing homogeneous regions (smaller values) from heterogeneous regions (larger values). The feature vector for a pixel $\mathbf{p}$ is a vector $\mathbf{e} = [e(p_1), e(p_2), \ldots, e(p_l)]$, such that $e(p_i)$ is the entropy of p_i computed over a 9×9 neighborhood centered at p_i.

The hyperspectral and entropy descriptors for a sample $s_i \in S_c$ are given beforehand, so that the pair $(\mathbf{p}_i, y_i)$ denotes s_i with hyperspectral descriptor and $(\mathbf{e}_i, y_i)$ denotes s_i with entropy descriptor.

3.1 *Support Vector Machines Classification*

An SVM classifier receives as input the sets S_u and S_c. The PCA technique is applied to the entropy vectors $\mathbf{e}$ in order to obtain reduced vectors $\mathbf{e}'$. This dimensionality reduction process is important to remove redundant or highly correlated features, which improves the performance of the classifier.

The SVM classifier is trained by using the pairs $(\mathbf{e}'_i, y_i)$ of S_c. Given the trained model, SVM classifies the samples $\mathbf{p}$ of S_u such that $L_1(\mathbf{p})$ is the assigned class. The assignment of all pixels produces an image L_1 of size $m \times n$.

3.2 *Seed Identification*

The SVM classification is checked against a K-NN search in order to identify pixels that have high probability of being correctly classified. The pixels identified in this step are called seeds. These pixels are considered in the next step as starting points for an MSF method which introduces spatial restrictions in the classification. All pixels that are not seeds will be reclassified by expanding the classification from the seeds. Algorithm 1 summarizes the process of seed identification.

Algorithm 1 Seed Identification.

Input:

S_u set of pixels

S_c set of classified samples

L_1 image of classes assigned by SVM

Output:

L_2 image of seeds

C image of costs

1 **for** $\mathbf{p} \in S_u$ **do**

2 Find the pairs $\{(\mathbf{e}'_{c_1}, y_{c_1}), (\mathbf{e}'_{c_2}, y_{c_2}), \ldots, (\mathbf{e}'_{c_k}, y_{c_k})\}$ from S_c which are the k nearest neighbors to the descriptor $\mathbf{e}'$ of $\mathbf{p}$

3 **if** $L_1(\mathbf{p}) = y_{c_1} = y_{c_2} = \ldots = y_{c_k}$ **then**

4 Make $L_2(\mathbf{p})$ equal to $L_1(\mathbf{p})$

5 Make $C(\mathbf{p})$ equal to 0

6 else

7 Make $L_2(\mathbf{p})$ equal to -1

8 Make $C(\mathbf{p})$ equal to ∞

9 end

10 **end**

For each pixel $\mathbf{p} \in S_u$, a K-NN search is performed in the feature space to retrieve its k nearest neighbors among the set S_c. The entropy descriptors are also used in this step. Therefore, given the descriptor $\mathbf{e}'$ of $\mathbf{p}$, the pairs of closest samples $\{(\mathbf{e}'_{c_1}, y_{c_1}), (\mathbf{e}'_{c_2}, y_{c_2}), \ldots, (\mathbf{e}'_{c_k}, y_{c_k})\}$ from S_c are retrieved.

If the class of $\mathbf{p}$ assigned by SVM is the same class of the pairs found with the K-NN search, $L_1(\mathbf{p}) = y_{c_1} = y_{c_2} = \ldots = y_{c_k}$, then $\mathbf{p}$ is included as a seed pixel in the set L_2, such that its class is the same as the one from SVM, $L_2(\mathbf{p}) = L_1(\mathbf{p})$. In addition to the class labeling, a cost of zero is also assigned to a seed, that is, $C(\mathbf{p}) = 0$. L_2 and C are images of size $m \times n$. These two images are further used in the MSF classification, such that $L_2(\mathbf{p})$ will store the classification of $\mathbf{p}$ given by an MSF, whereas C will store the cost of such assignment.

All the pixels $\mathbf{p}$ for which the class assigned by SVM differs from any of the K-NN will be considered in the next step of the classification. These pixels are assigned an invalid class $L_2(\mathbf{p}) = -1$ and a high cost, $C(\mathbf{p}) = \infty$.

3.3 *Minimum Spanning Forests Classification*

The final step of the classification uses an MSF approach so that the classification takes spatial restrictions into account. This step performs its computations over the hyperspectral feature space; that is, the similarity of pixels is computed using the pixel vectors of values $\mathbf{p} = (p_1, p_2, \ldots, p_l)$.

The classification is stored in the image L_2. The cost image C and a minimum priority queue Q are used as auxiliary data structures. The seeds found in the previous step are initially inserted into the priority queue Q. The priority queue Q is maintained according to the costs of C. The MSF computation is performed as shown in Algorithm 2.

A minimum cost pixel $\mathbf{p}$ is retrieved from Q, and its four-neighborhood is analyzed. Suppose $\mathbf{q}$ is a neighbor pixel of $\mathbf{p}$ and $d(\mathbf{p}, \mathbf{q})$ is the Euclidean distance of their feature vectors

Algorithm 2 Minimum spanning forests.

Input:

S_u set of pixels

L_2 image of classes

C image of costs

Q auxiliary minimum priority queue

Output:

L_2 classified image

```
1 Insert into Q all p ∈ S_u such that C(p) = 0
2 while Q is not empty do
3     Remove p from Q
4     for q ∈ S_u in the 4-neighborhood of p do
5         if C(q) > d(p,q) then
6             Make L_2(q) equal to L_2(p)
7             Make C(q) equal to d(p,q)
8             if q ∈ Q then
9                 Update priority queue Q
10            else
11                Insert q into Q
12            end
13    end
14 end
```

$\mathbf{p} = (p_1, p_2, \ldots, p_l)$ and $\mathbf{q} = (q_1, q_2, \ldots, q_l)$. If the cost $C(\mathbf{q})$ is greater than $d(\mathbf{p}, \mathbf{q})$, then $\mathbf{q}$ is assigned to the class of $\mathbf{p}$ ($L_2(\mathbf{q}) = L_2(\mathbf{p})$) and its cost is updated to the distance between the descriptors, $C(\mathbf{q}) = d(\mathbf{p}, \mathbf{q})$. In case $\mathbf{q}$ is already in Q, the priority queue is updated; otherwise $\mathbf{q}$ is inserted into Q.

The process just described is the basic iteration of the MSF algorithm. At each iteration, a pixel of minimum cost is retrieved from Q. Since the seeds are elements with cost zero, the classification will expand from them. The process is repeated until the priority queue is empty, such that all pixels have been classified.

4 Experimental Results

In this section, the proposed method is compared to four others: (i) SVM, (ii) SVM with PCA, (iii) the approach based on stochastic minimum spanning forest by Bernard et al.,[13] referred here to as SMSF, and (iv) the approach based on probabilistic minimum spanning forest by Tarabalka et al.,[15] referred here to as PMSF.

The SMSF and PMSF approaches are closely related to our proposed method, which are also based on the SVM classifier to obtain an initial classification and based on MSF for improving the results. On the other hand, the SMSF uses a stochastic scheme for the second step. A set of n maps is created and, for each one of these maps, a percentage of the pixels classified by SVM is randomly chosen to form the seeds for the MSF method. Then, MSF is applied to each map and the final classification is given by a voting scheme considering the results for all these maps.

Table 1 Summary of the main dataset characteristics.

Dataset	Rows	Columns	Bands	Classes
Indian Pines	145	145	200	16
Pavia University	610	340	103	9
Pavia Centre	1096	715	102	9
Salinas-A	83	86	204	6
Salinas	512	217	204	16
Kennedy Space Center	512	614	176	13
Botswana	1476	256	145	14

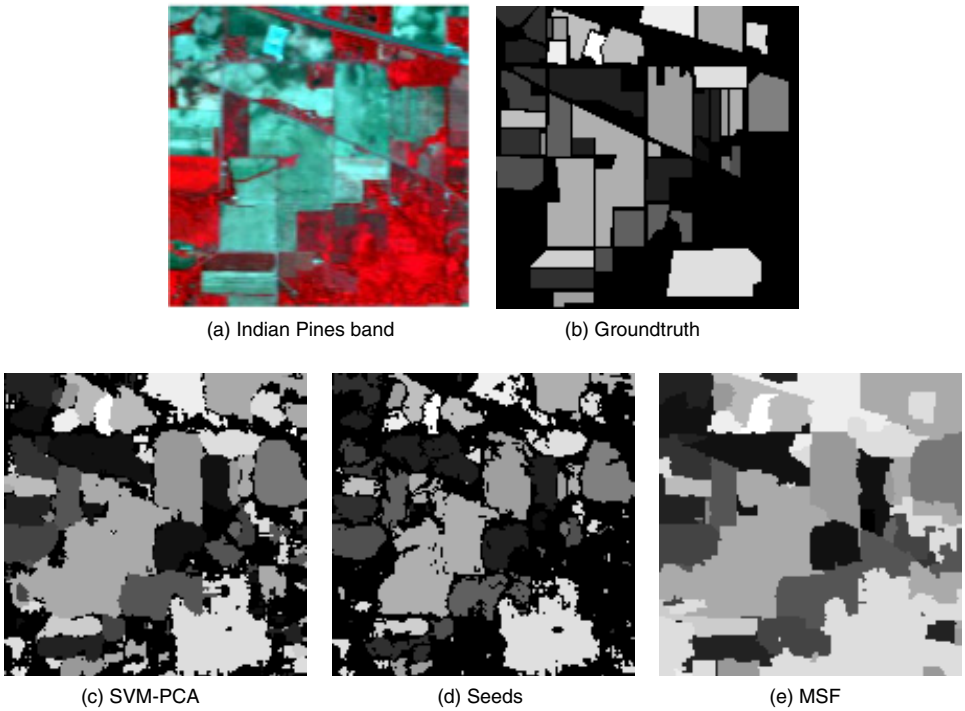
The PMSF also differs in the second step, where the most reliable pixels classified by SVM are chosen as seeds for the MSF method.

Experiments are conducted on seven benchmark datasets.[18,19,49] The main characteristics of each dataset (image dimensions, number of bands, and number of classes) are summarized in Table 1.

The algorithms were implemented in MATLAB®.[50] The LIBSVM,[51] with default parameters, was used to perform the classification based on SVM. The methods were evaluated through the overall and class accuracy rates.

Figure 2 shows an overview of the proposed classification method. Figures 2(a) and 2(b) depict one band of the Indian Pines dataset and its ground truth. The SVM classification is given in Fig. 2(c). The resulting seeds are shown in Fig. 2(d), whereas the final classification is shown in Fig. 2(e).

In the following sections, we initially tuned the parameters for SVM, our method, SMSF, and PMSF. Hyperspectral and entropy descriptors were compared. Entropy improved the results obtained with SVM when compared to the results through hyperspectral values. The same was not true for MSF classification, where the hyperspectral values tend to perform better than entropy. We presented some comparisons for SVM. For the MSF step, we used hyperspectral



(a) Indian Pines band (b) Groundtruth

(c) SVM-PCA (d) Seeds (e) MSF

Fig. 2 Example illustrating an overview of the proposed classification method.

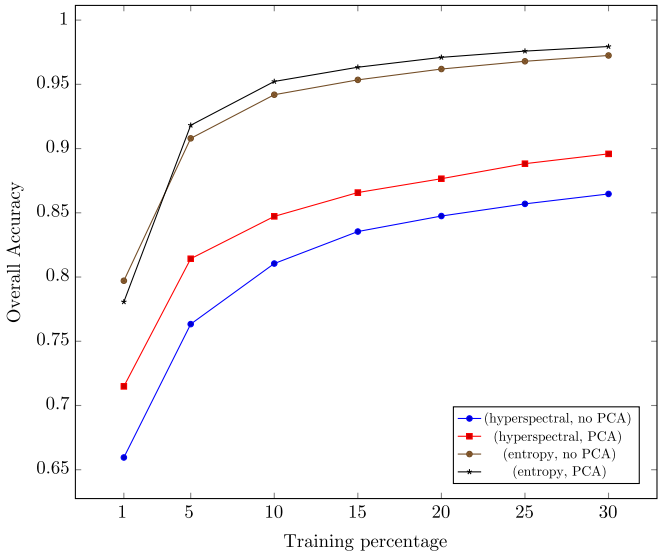


Fig. 3 Comparison of parameters for SVM classification. The lines are labeled as a pair defining which descriptor was used and whether PCA has been performed.

descriptors. The parameter k for the k-nearest neighbors used in our method was evaluated, as well as the parameters for SMSF and PMSF. After the tuning steps, we compared the methods.

4.1 *Support Vector Machines and Descriptors*

Figure 3 shows the behavior of the SVM classification according to the used descriptor, hyperspectral values, or entropy, as well as the use of PCA. Entropy provides a significant improvement on the classification results when compared to hyperspectral values, whereas the use of PCA helped to achieve better classification results relative to each descriptor.

4.2 *K-Nearest Neighbor Setup*

Figure 4 shows the impact on the classification according to the choice of the k value (of K-NN) for selecting the seeds in our method. The best results were obtained with the hyperspectral

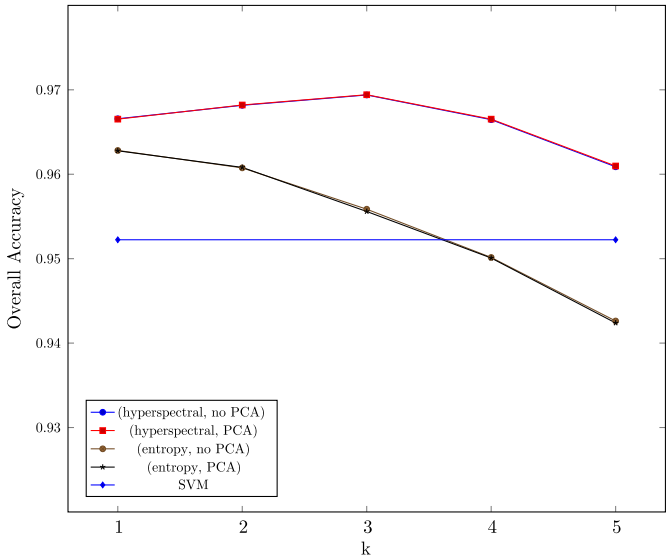


Fig. 4 Impact of k-nearest neighbors for the selection of seed pixels. The lines are labeled as a pair defining which descriptor was used and whether PCA has been performed. The SVM result is depicted as a baseline for comparison.

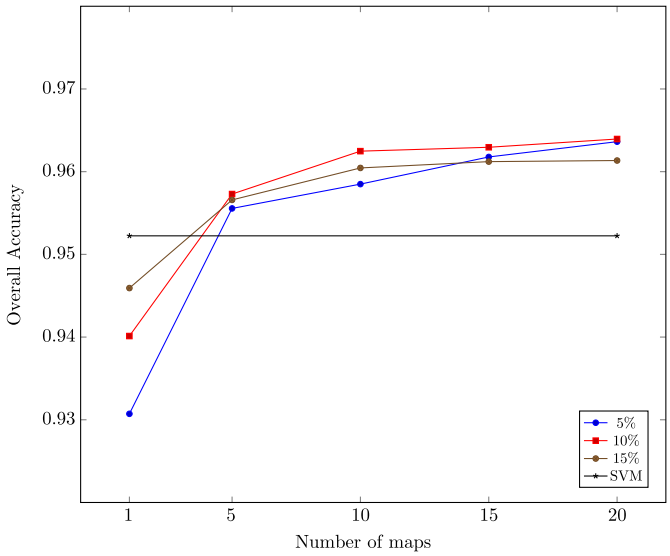


Fig. 5 Results obtained with the SMSF using different numbers of maps and percentages of pixels randomly selected as seeds. Each line reports the results for a certain percentage of seeds as the number of maps varies. The SVM result is depicted as a baseline for comparison.

descriptor. PCA did not provide any improvement. All the experiments were performed with 10% of samples for training the SVM. The SVM result is depicted as a baseline for comparison. The best result is achieved with $k = 3$.

4.3 *SMSF Setup*

The impact of the choice in terms of the number of maps and marks for the SMSF method is summarized in Fig. 5. All experiments used 10% of the pixels for the SVM classification. Even though the original paper[13] does not use entropy and PCA for SVM, we have used these settings for a fair comparison of the MSF step; otherwise the results for SMSF would be degraded. Similarly to what occurred with our method, PCA and entropy do not improve the results of the MSF step. We suppress the results for such parameters and show the best results using

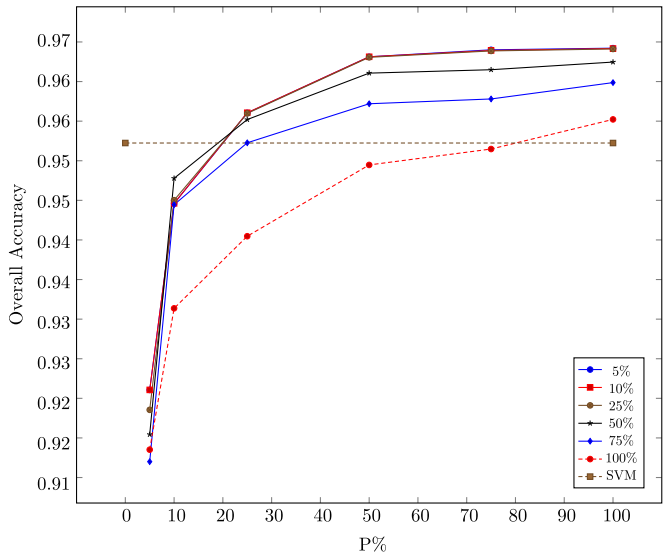


Fig. 6 Results obtained with the PMSF using different percentages of pixels selected as seeds and different reliability thresholds. Each line reports the results for a certain reliability threshold. The SVM result is depicted as a baseline for comparison.

Table 2 Comparison for the Indian Pines dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
Alfalfa	0.521	0.312	0.708	0.854	0.854
Corn-notill	0.798	0.827	0.869	0.891	0.850
Corn-min	0.781	0.817	0.884	0.893	0.905
Corn	0.657	0.738	0.752	0.786	0.862
Grass/pasture	0.792	0.828	0.846	0.906	0.937
Grass/trees	0.805	0.798	0.878	0.906	0.964
Pasture-mowed	0.739	0.000	0.000	0.000	0.000
Hay-windrowed	0.952	0.968	1.000	0.998	0.989
Oats	0.000	0.000	0.000	0.000	0.000
Soybeans-notill	0.889	0.907	0.946	0.929	0.938
Soybeans-min	0.943	0.954	0.977	0.965	0.978
Soybeans-clean	0.830	0.857	0.904	0.906	0.861
Wheat	0.753	0.858	1.000	0.926	0.984
Woods	0.932	0.930	0.967	0.942	0.972
Building-trees	0.711	0.737	0.851	0.713	0.661
Stone-steel	0.871	0.906	1.000	1.000	0.953
Overall	0.856	0.873	0.919	0.915	0.921

a feature descriptor based on hyperspectral values and no PCA step. This is also the setting used in the original paper. The best results were obtained with 20 maps (as in the original paper) and 10% of seeds randomly selected. In the original paper, the value is 3% for some datasets and 11% for other datasets. However, some of the datasets used here were not considered in the paper. Therefore, we choose 10% random samples for seed selection since this value achieves the best results for the datasets evaluated in our experiments.

Table 3 Comparison for the Pavia University dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
Asphalt	0.942	0.962	0.966	0.959	0.973
Meadows	0.971	0.978	0.984	0.973	0.989
Gravel	0.723	0.787	0.843	0.776	0.850
Trees	0.885	0.890	0.915	0.894	0.884
Painted metal sheets	0.973	0.988	0.999	0.999	0.995
Bare Soil	0.832	0.870	0.893	0.903	0.921
Bitumen	0.870	0.866	0.939	0.880	0.907
Self-blocking bricks	0.915	0.931	0.959	0.958	0.938
Shadows	0.842	0.864	0.836	0.845	0.887
Overall	0.921	0.938	0.953	0.941	0.955

Table 4 Comparison for the Pavia Centre dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
Water	0.999	0.999	0.999	0.999	0.999
Trees	0.970	0.971	0.972	0.954	0.974
Asphalt	0.886	0.898	0.898	0.930	0.923
Self-blocking bricks	0.832	0.873	0.887	0.835	0.914
Bitumen	0.884	0.910	0.938	0.906	0.912
Tiles	0.961	0.964	0.963	0.952	0.965
Shadows	0.940	0.951	0.970	0.954	0.957
Meadows	0.990	0.992	0.991	0.995	0.995
Bare soil	0.611	0.626	0.677	0.589	0.694
Overall	0.971	0.975	0.979	0.974	0.979

Table 5 Comparison for the Salinas-A dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
Brocoli	1.000	0.986	0.997	0.929	1.000
Corn	1.000	1.000	1.000	1.000	0.997
Lettuce1	1.000	1.000	1.000	1.000	1.000
Lettuce2	1.000	1.000	1.000	1.000	1.000
Lettuce3	1.000	1.000	0.959	0.969	1.000
Lettuce4	1.000	1.000	0.996	0.983	0.999
Overall	1.000	0.999	0.994	0.988	0.999

Table 6 Comparison for the Salinas dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
Brocoli1	0.999	0.999	1.000	0.999	0.998
Brocoli2	0.983	0.987	0.992	0.971	0.993
Fallow	0.998	0.999	0.997	0.995	0.999
Fallow rough	0.942	0.940	0.986	0.971	0.967
Fallow smooth	0.998	0.999	0.988	0.989	0.997
Stubble	0.999	0.997	0.995	0.991	1.000
Celery	0.999	0.998	1.000	0.998	0.999
Grapes	0.959	0.969	0.981	0.993	0.983
Soil	0.999	0.999	0.998	0.997	0.999
Corn	0.983	0.994	0.980	0.986	0.993
Lettuce1	0.985	0.976	0.968	0.958	0.978
Lettuce2	1.000	0.999	0.999	0.992	1.000
Lettuce3	0.987	0.971	0.922	0.916	0.990
Lettuce4	0.993	0.996	0.975	0.977	0.988
Vinyard1	0.925	0.944	0.952	0.991	0.962
Vinyard2	0.990	0.988	0.994	0.966	0.995
Overall	0.976	0.981	0.983	0.988	0.988

Table 7 Comparison for the Kennedy Space Center dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
1	0.994	0.996	1.000	0.990	1.000
2	0.954	0.963	0.927	0.972	0.950
3	0.978	0.991	0.991	0.987	1.000
4	0.991	0.996	0.991	1.000	0.929
5	0.931	0.944	0.993	0.986	1.000
6	0.995	0.995	0.956	0.995	0.995
7	0.979	0.989	1.000	1.000	1.000
8	0.956	0.956	1.000	0.953	0.977
9	0.953	0.955	0.964	0.962	0.983
10	0.997	0.994	0.975	0.978	0.986
11	0.989	0.989	0.987	0.966	0.984
12	0.965	1.000	0.991	0.998	0.973
13	1.000	1.000	1.000	1.000	1.000
Overall	0.980	0.985	0.986	0.984	0.985

4.4 *PMSF Setup*

These experiments were also conducted with 10% of the pixels for the SVM training. As in SMSF, the use of entropy and PCA for SVM was kept for a fair comparison of the MSF step. We only show the results for hyperspectral descriptors for the MSF step, once PCA and entropy did not result in any improvement.

Table 8 Comparison for the Botswana dataset.

Class	SVM	SVM-PCA	SMSF	PMSF	SVM-MSF
1	0.885	0.897	0.934	0.877	0.963
2	0.544	0.567	0.522	0.522	0.533
3	1.000	1.000	0.996	1.000	0.996
4	0.808	0.860	0.927	0.850	0.689
5	0.860	0.839	0.884	0.860	0.860
6	0.554	0.603	0.616	0.653	0.678
7	0.948	0.961	0.948	0.940	0.991
8	1.000	1.000	1.000	0.978	0.995
9	0.879	0.865	0.968	0.936	0.979
10	0.951	0.951	0.915	0.942	0.978
11	0.869	0.872	0.887	0.909	0.909
12	0.988	0.963	1.000	1.000	1.000
13	0.917	0.888	0.876	0.846	1.000
14	0.000	0.024	0.012	0.024	0.071
Overall	0.846	0.851	0.870	0.858	0.883

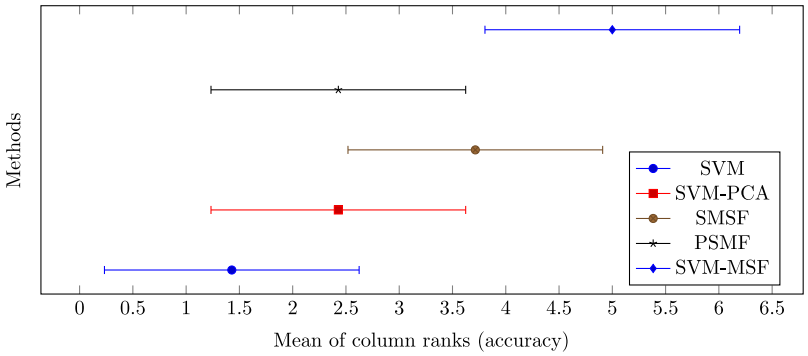


Fig. 7 Post-hoc analysis of Friedman test for the different evaluated methods.

After the SVM step, PMSF computes the connected components on the resulting classification and the probability (reliability) of each classification being correct (for more details, refer to Ref. 15). These components and the following three parameters are used to select the seeds of the method:

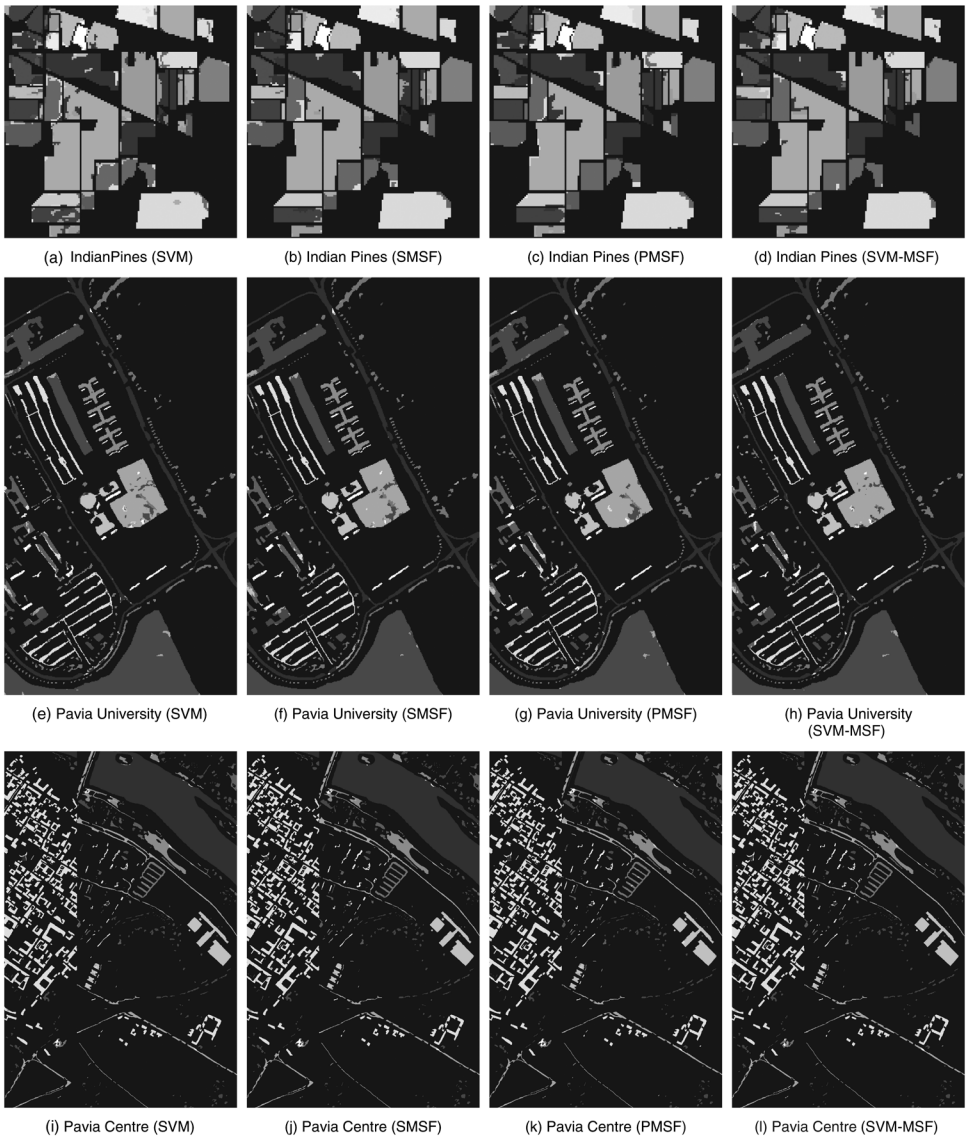


Fig. 8 Visual comparison of the classification results given by SVM, SMSF, and SVM-MSF.

- the way of selecting the seeds depends on whether a connected component is larger or smaller than a threshold M;
- if the connected component is larger than M, then a percentage P of its most reliable classifications is kept as seeds;
- otherwise, the pixels with reliability greater than a threshold S are kept as seeds.

The parameter S is the smaller probability value of the $T\%$ most reliable SVM classified pixels for the whole image. We will discuss the choice of T, as in the original paper. The parameter M was chosen to be 20, since it produced good results in our experiments, reflecting the choice in the original paper. The parameters P and T were set to $\sim$5% as in the original paper; however, we did not obtain good results with such a choice. The differences may be due to the probability computations, even though we used LIBSVM,[51] as in the original paper. The choice of these parameters for the PMSF method is summarized in Fig. 6. The overall accuracy stabilizes with $P = 50\%$ and $T = 5\%$. The results get worse for T greater than 25%. The best results are obtained with T in the range (5%, 25%) (overlaid curves in Fig. 6).

4.5 *Comparison*

Tables 2–8 show the gradual improvement on the classification achieved by the proposed method when using SVM with PCA and then by adding the MSF step. The method also achieves

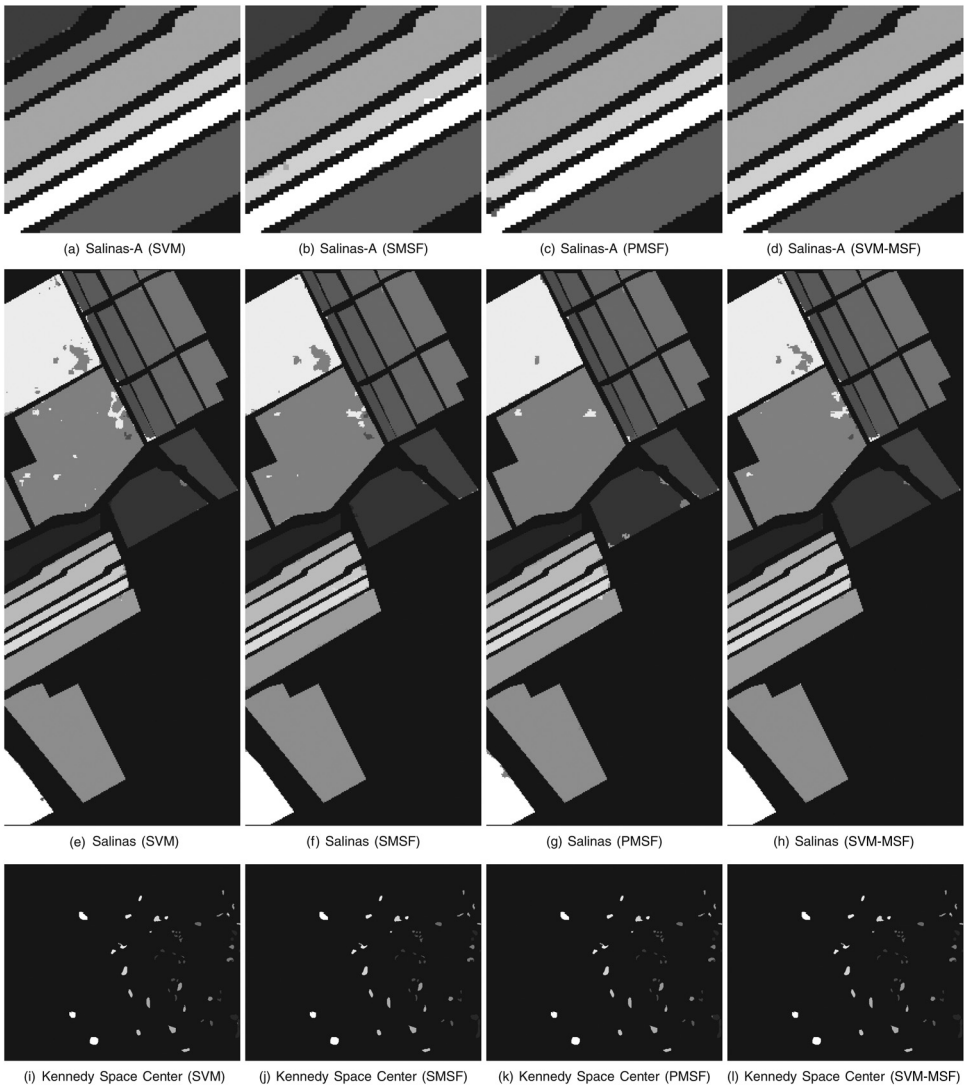
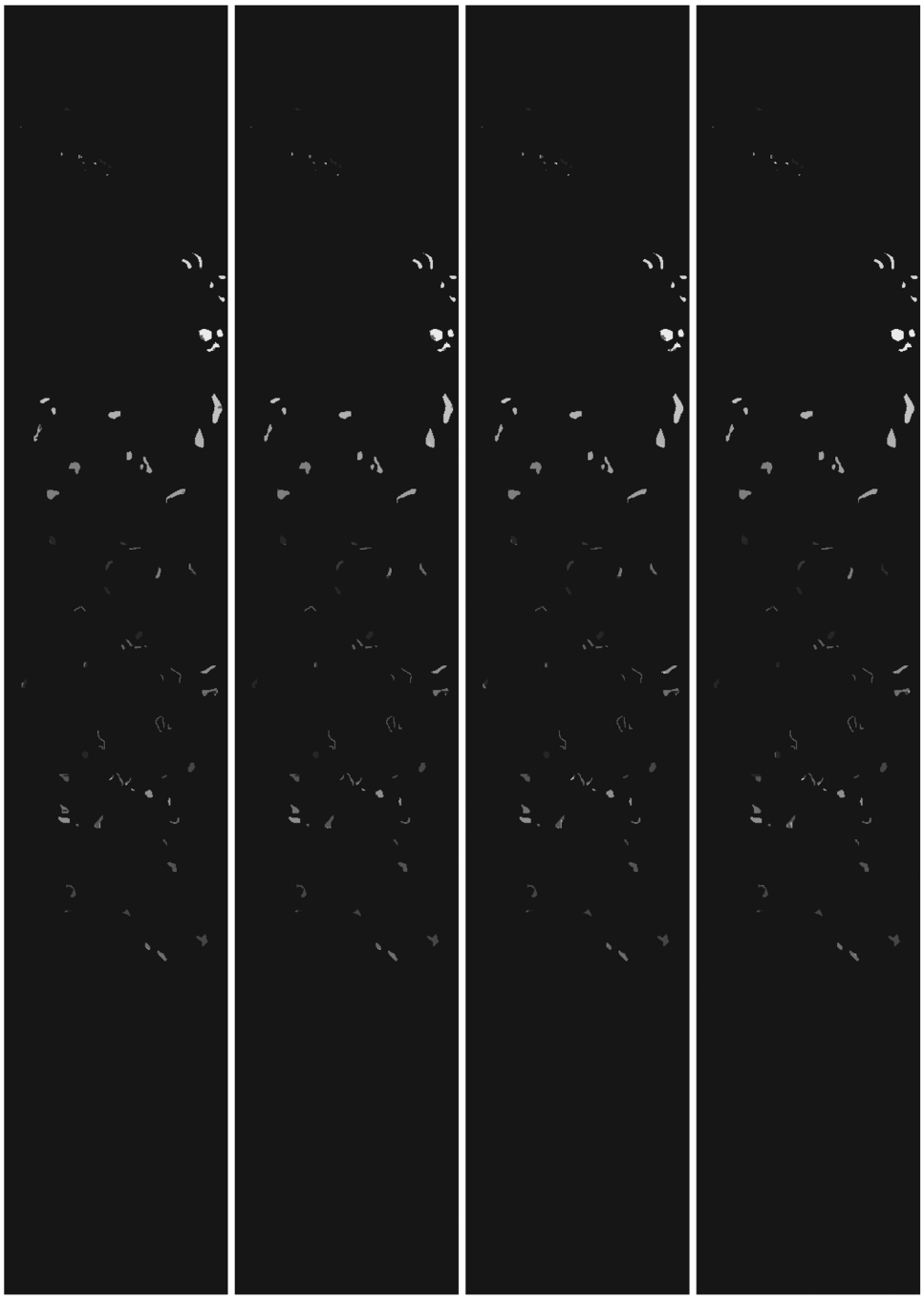


Fig. 9 Visual comparison of the classification results given by SVM, SMSF, and SVM-MSF.

competing values of class accuracy and superior overall accuracy when compared to SMSF and PMSF. The experiments for these images were performed by using 10% of the pixels as training samples. The remaining pixels were used as test samples.

We performed the Friedman test and obtained a p-value of 0.0002, which suggests there is a significant difference between at least two methods. A post-hoc analysis based on the Tukey–Kramer test is shown in Fig. 7. The analysis shows that our method performs better than three methods (SVM, SVM-PCA, and PMSF) and has statistically equal results when compared against SMSF.

An interesting aspect of the proposed classification method is that, besides correcting erroneous classifications, the seed identification step also confirms many of the correct



(a) Botswana (SVM) (b) Botswana (SMSF) (c) Botswana (PMSF) (d) Botswana (SVM-MSF)

Fig. 10 Visual comparison of the classification results given by SVM, SMSF, and SVM-MSF.

classifications produced by the SVM classifier. The method preserves as many as possible of these correct classifications and works on the weak classified pixels, such as the ones close to spatial boundaries between classes.

Figures 8–10 show a visual comparison of the classification results produced by SVM, SMSF, and SVM-MSF. The images depict only the classes; that is, the respective background is removed and shown in black.

5 Conclusions

In this work, we described a method for hyperspectral data classification through the combination of spectral and spatial information. An SVM was employed to produce an initial classification, which was then refined by applying K-NN and MSF approaches, incorporating spatial information into the spectral classification process.

Experiments were conducted on several datasets, and the proposed approach was demonstrated to be very effective, providing results with high accuracy. Intermediate results also show that the entropy measure and PCA technique can be used to improve the classification obtained by the SVM classifier.

As directions for future work, we intend to investigate the combination of other complementary features, as well as the impact of using different parameters in the classification process.

Acknowledgments

The authors are thankful to FAPESP (Grant No. 2011/22749-8) and CNPq (Grant No. 307113/2012-4) for their financial support.

References

1. M. T. Eismann, *Hyperspectral Remote Sensing*, SPIE Press, Bellingham, Washington (2012).
2. T. Lillsand, R. W. Kiefer, and J. Chipman, *Remote Sensing and Image Interpretation*, John Wiley & Sons, Inc., USA (2014).
3. W. G. Rees, *Physical Principles of Remote Sensing*, Cambridge University Press, New York, NY (2013).
4. R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, 3rd ed., Academic Press, Burlington, Massachusetts (2006).
5. G. Bilgin, S. Erturk, and T. Yildirim, "Unsupervised classification of hyperspectral-image data using fuzzy approaches that spatially exploit membership relations," *IEEE Geosci. Remote Sens. Lett.* **5**(4), 673–677 (2008).
6. C.-I. Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*, 1st ed., John Wiley & Sons, Inc., Hoboken, New Jersey (2013).
7. R. Ji et al., "Spectral-spatial constraint hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.* **52**(3), 1811–1824 (2014).
8. W. R. Johnson et al., "Hyperspectral imaging in the thermal infrared: existing and future instruments," in *Hyperspectral Imaging and Sounding of the Environment*, HM4B–3, Optical Society of America, Washington, DC (2015).
9. T. Liu et al., "Remotely sensed image retrieval based on region-level semantic mining," *Eurasip J. Image Video Process.* **2012**(1), 1–11 (2012).
10. Q. Lu, X. Huang, and L. Zhang, "A novel clustering-based feature representation for the classification of hyperspectral imagery," *Remote Sens.* **6**(6), 5732–5753 (2014).
11. S. Suess et al., "Import vector machines for quantitative analysis of hyperspectral data," *IEEE Geosci. Remote Sens. Lett.* **11**(2), 449–453 (2014).
12. M. Fauvel et al., "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE* **101**(3), 652–675 (2013).
13. K. Bernard et al., "Spectral-spatial classification of hyperspectral data based on a stochastic minimum spanning forest approach," *IEEE Trans. Image Process.* **21**, 2008–2021 (2012).
14. Y. Tarabalka, J. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.* **47**, 2973–2987 (2009).

15. Y. Tarabalka, J. Chanussot, and J. Benediktsson, "Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers," *IEEE Trans. Syst. Man Cybern.* **40**, 1267–1279 (2010).
16. E. C. Barrett and L. Curtis, *Introduction to Environmental Remote Sensing*, Stanley Thornes Ltd., Cheltenham, Gloucestershire, UK (2013).
17. P. A. Mitchell, "Hyperspectral digital imagery collection experiment (HYDICE)," *Proc. SPIE* **2587**, 70–95 (1995).
18. AVIRIS, "Airbone visible/infrared imaging spectrometer," 2015, http://aviris.jpl.nasa.gov/ (16 October 2015).
19. B. Kunkel et al., "ROSIS imaging spectrometer and its potential for ocean parameter measurements (airbone and spaceborner)," *Int. J. Remote Sens.* **12**, 753–761 (1991).
20. CASI, "Compact airborne spectrographic imager," 2016, http://geo.arc.nasa.gov/sge/jskiles/top-down/OTTER/OTTER_docs/CASI.html (16 October 2015).
21. G. Vane, A. F. Goetz, and J. B. Wellman, "Airborne imaging spectrometer: a new tool for remote sensing," *IEEE Trans. Geosci. Remote Sens.* **GE-22**, 546–549 (1984).
22. B. M. Braam et al., "Design and first test results of the finnish airborne imaging spectrometer for different applications (AISA)," *Proc. SPIE* **1937**, 142–151 (1993).
23. W. E. Collins and S.-H. Chang, "Geophysical environmental research corporation 63-channel airborne imaging spectrometer and 12-band thermal scanner," *Proc. SPIE* **1298**, 62–71 (1990).
24. T. Cocks et al., "The HyMapTM airborne hyperspectral sensor: the system, calibration and performance," in *First EARSel Workshop on Imaging Spectroscopy*, European Association of Remote Sensing Laboratories, Zurich, Switzerland (1998).
25. K. Hartonen, T. Laitinen, and M.-L. Riekkola, "Current instrumentation for aerosol mass spectrometry," *TrAC Trends in Anal. Chem.* **30**(9), 1486–1496 (2011).
26. APEX, "Airbone Prism Experiment," 2016, http://www.apex-esa.org/ (16 October 2015).
27. M. A. Cutter, D. R. Lobb, and R. A. Cockshott, "Compact high resolution imaging spectrometer (CHRIS)," *Acta Astronaut.* **46**(2–6), 263–268 (2000).
28. F. Lehmann et al., "The digital airborne imaging spectrometer DAIS 7915, a new sensor for the surveillance of ecosystems," in *Thematic Conf. on Geologic Remote Sensing*, Vol. 1, pp. 1–427, Environmental Research Institute of Michigan (1996).
29. F. Palluconi and G. R. Meeks, "Thermal infrared multispectral scanner (TIMS): an investigator's guide to TIMS data," Tech. Rep. JPL-PuB-85-32, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California (1985).
30. N. Bali and A. Mohammad-Djafari, "Bayesian approach with hidden markov modeling and mean field approximation for hyperspectral data analysis," *IEEE Trans. Image Process.* **17**, 217–225 (2008).
31. G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: a review," *ISPRS J. Photogramm. Remote Sens.* **66**(3), 247–259 (2011).
32. E. Merényi et al., "Classification of hyperspectral imagery with neural networks: comparison to conventional tools," *EURASIP J. Adv. Signal Process.* **2014**(1), 1–19 (2014).
33. E. Blanzieri and F. Melgani, "Nearest neighbor classification of remote sensing images with the maximal margin principle," *IEEE Trans. Geosci. Remote Sens.* **46**(6), 1804–1811 (2008).
34. B. Zhang et al., "A neighbourhood-constrained k-means approach to classify very high spatial resolution hyperspectral imagery," *Remote Sens. Lett.* **4**(2), 161–170 (2013).
35. P. K. Goel et al., "Classification of hyperspectral data by decision trees and artificial neural networks to identify weed stress and nitrogen status of corn," *Comput. Electron. Agr.* **39**, 67–93 (2003).
36. H. Chang, Q. Yang, and B. Parvin, "A Bayesian approach for image segmentation with shape priors," in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1–8, IEEE (2008).
37. L. Zhang et al., "On combining multiple features for hyperspectral remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.* **50**, 879–893 (2012).
38. X. Hadoux et al., "A spectral–spatial approach for hyperspectral image classification using spatial regularization on supervised score image," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **8**(6), 2361–2369 (2015).

39. Y. Liu et al., "Hyperspectral classification via learnt features," in *IEEE Int. Conf. on Image Processing*, pp. 2591–2595 (2015).
40. T. Takayama, A. Iwasaki, and O. Kashimura, "Optimal segmentation of classification and prediction maps for monitoring forest condition with spectral and spatial information from hyperspectral data," in *IEEE Int. Geoscience and Remote Sensing Symp.*, pp. 3498–3501 (2014).
41. Y. Yang and G. Cao, "Combining watersheds and conditional random fields for image classification," in *10th Int. Conf. on Fuzzy Systems and Knowledge Discovery*, pp. 805–810 (2013).
42. H. Zhou, J. Zheng, and L. Wei, "Texture aware image segmentation using graph cuts and active contours," *Pattern Recognit.* **46**, 1719–1733 (2013).
43. J. J. de Mesquita Sá, Jr, A. R. Backes, and P. C. Cortez, "Texture analysis and classification using shortest paths in graphs," *Pattern Recognit. Lett.* **34**(11), 1314–1319 (2013).
44. L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 1768–1783 (2006).
45. Y. Xu, Z. Wu, and Z. Wei, "Markov random field with homogeneous areas priors for hyperspectral image classification," in *IEEE Int. Geoscience and Remote Sensing Symp.*, pp. 3426–3429 (2014).
46. J. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sens.* **32**, 779–785 (1994).
47. S. Deng et al., "A feature-selection algorithm based on support vector machine-multiclass for hyperspectral visible spectral analysis," *J. Food Eng.* **119**(1), 159–166 (2013).
48. I. Jolliffe, *Principal Component Analysis*, 2nd ed., Springer-Verlag, New York, NY (2002).
49. L. L. Biehl, "Hyperspectral images," 2015, https://engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html (16 October 2015).
50. MATLAB, *MATLAB version R2015a*, The MathWorks Inc., Natick, MA (2015).
51. C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Trans. Intell. Syst. Technol.* **2**, 1–27 (2011).

Ricardo Dutra da Silva received his PhD degree in computer science from University of Campinas and his MSc and BSc degrees in computer science from Federal University of Parana. He is currently a professor in the Informatics Department at Federal University of Technology, Curitiba, Brazil. His research interests include image processing and analysis, computer vision, computational geometry, and computer graphics.

Helio Pedrini received his PhD degree in electrical and computer engineering from Rensselaer Polytechnic Institute, Troy, New York, USA. He received his MSc degree in electrical engineering and his BSc in computer science, both from University of Campinas, Brazil. He is currently a professor in the Institute of Computing at the University of Campinas, Brazil. His research interests include image processing, computer vision, and pattern recognition and computer graphics.