International Journal of Remote Sensing
Vol. 00, No. 00, 00 Month 201X, 1–21

RESEARCH ARTICLE

Exploring ELM-Based Spatial-Spectral Classification of Hyperspectral Images

Dora B. Heras[a]*, Francisco Arguello[a]

and Pablo Quesada-Barriuso[a]

[a] *CITIUS (Centro Singular de Investigacin en Tecnoloxas da Informacin), Ra de Jenaro de la Fuente Domnguez, 15782 - Santiago de Compostela, Spain*;

(*Received 00 Month 201X; final version received 00 Month 201X*)

Among the different computational intelligence techniques avalaible for hyperspectral data classification, Support Vector Machines (SVMs) have been playing a dominant role. Recently, a new learning algorithm for single-layer feedforward neural networks called Extreme Learning Machine (ELM) was proposed. This technique is competitive with SVM in terms of accuracy, learning speed and computational scalability. In this paper we propose and evaluate the use of ELM for the land cover classification from hyperspectral images. In addition, we consider two ELM-based techniques integrating spectral and spatial information of the image. The first one is a scheme that uses a majority vote approach in order to combine the results of a pixel-wise spectral classification by ELM and a segmentation map obtained by a watershed algorithm. The second one introduces spatial information from a small spatial neighborhood after the classification by ELM. We show the usefulness of spatial-spectral ELM-based classification techniques in hyperspectral imaging. The results are compared to those obtained by similar SVM-based techniques and show improved classification results and much lower execution time. These simple and computationally cheap methods can be combined with others traditionally applied to hyperspectral images [1].

Keywords: hyperspectral data, remote sensing, spectral-spatial classification, extreme learning machine, watershed, classification accuracy.

1. Introduction

Hyperspectral images are currently widely available as a result of advances in remote sensor technology such as the reduction in the size and cost of the sensors and the increase in spectral resolution. For example, AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) records 220 spectral bands in a spectral range from 0.4 to 2.5 μm, corresponding to the visible and infrared spectrum. ROSIS-03 (Reflective Optics System Imaging Spectrometer) captures 115 spectral bands in the spectral range from 0.43 to 0.86 μm, corresponding mainly to the visible spectrum. Different applications of remote sensing images can be found in the literature: vegetation applications such as forest degradation analysis (Cao et al. 2009), identification of urban areas such as road extraction and urban change detection (Nemmour and

*Corresponding author. Email: dora.blanco@usc.es

[1] This is an Accepted Manuscript of an article published by Taylor & Francis in International Journal of Remote Sensing on 2 Jan 2014, available online: http://wwww.tandfonline.com/10.1080/01431161.2013.869633

Chibani 2006), and mineral mapping (Kruse, Boardman and Huntington 2003), among others. Hyperspectral images consist of hundreds of spectral bands at different wavelength channels, hence efficient techniques are required to extract the huge amount of information they contain. We focus on remote sensing images, i.e. images obtained by airborne or satellite hyperspectrometers corresponding to areas of the surface of the Earth. The problems that are typically addressed for such hyperspectral data include anomaly detection, target recognition and background characterization, including land cover classification, which is the aim of this work.

Although many methods have been applied to different problems associated to classification of multispectral and hyperspectral data (Fauvel, Chanussot, and Benediktsson 2006; Mountrakis, Im, and Ogole 2011; Fauvel et al. 2013), SVM is generally recognized as the method that offers the best results in terms of accuracy of the classification, even when the number of training samples is small or noise caused by measurement errors due to limited precision of sensors, and by atmospheric and topographic distortions, is present. SVM permits the analysis of high-dimensional input spaces where traditional methods are not effective (Mountrakis, Im, and Ogole 2011). SVMs are also, as shown by (Melgani and Bruzzone 2004), more effective than other conventional non-parametric classifiers, such as RBF (radial basis function) neural networks and the KNN (K-nearest neighbor) classifier, in terms of accuracy, computational time, and stability to parameter setting.

For hyperspectral data and, in particular, for remote sensing, SVM-based techniques have recently been extensively applied to different fields demonstrating self-adaptability, rapid learning and limited requirements in the number of training samples (Mountrakis, Im, and Ogole 2011), making them reliable for the intelligent processing of data acquired through remote sensing. SVM provides excellent results when applied to pixel-wise classification (Fauvel, Chanussot, and Benediktsson 2006; Melgani and Bruzzone 2004).

The standard two-class SVM training phase builds a model which is used to predict whether new samples belong to one category or another in a posterior phase of classification (Vapnik 1995). When the data belong to more than two classes it is necessary to define a method in order to solve the multiclass problem. These methods may be of the type one-against-all (OAA), one-against-one (OAO) and all-at-once (Shiego 2005). The SVM classification in the multiclass case always requires classification by pairs of classes to be performed in a OAO fashion, even when the OAA option provided by some libraries is selected (Chang and Lin 2011). If N is the number of classes, $\frac{N^2-N}{2}$ is the number of OAO classifications that must be performed.

Raw SVM pixel-wise classification considers only spectral information but not information on the spatial structures present in the image. The accuracy of the results obtained is limited in this case (Mountrakis, Im, and Ogole 2011). The spatial relationships among pixels can be integrated with the spectral information following mainly two different approaches. The first one consists in introducing spatial information into the kernel function used by the non linear SVM calculation (Camps et al. 2006; Fauvel, Chanussot, and Benediktsson 2006, 2012; Plaza et al. 2009). The second one consists in processing the spatial information outside the SVM and joining the spatial results to the spectral classification results obtained by the raw SVM classification (Bai and Sapiro 2007; Linden et al. 2007; Fauvel, Chanussot, and Benediktsson 2008; Tarabalka, Benediktsson and Chanussot 2009; Tarabalka, Chanussot, and Benediktsson 2010a,b; Tarabalka et al. 2010; Fauvel et al. 2013).

Regarding the last approach, that has been chosen for the present work, the spatial information introduced can be obtained from the closest neighborhood of a pixel by any object-based classification method. This approach uses, for example, Markov random fields (Tarabalka et al. 2010) or morphological profiles (Benediktsson, Palmason and Sveinsson 2005; Fauvel, Chanussot, and Benediktsson 2008). Another option is to consider spatial information in a broader neighborhood and it is based on a segmentation of the image by graph-based segmentation algorithms such as partitional clustering techniques (Tarabalka, Benediktsson and Chanussot 2009), minimum spanning forests (Tarabalka, Chanussot, and Benediktsson 2010b), shortest paths (Bai and Sapiro 2007) or watershed (Tarabalka, Benediktsson and Chanussot 2009) among others.

The spatial and spectral information must be combined in order to compute the final classification map. Three main approaches have been identified in the literature for this task (Fauvel et al. 2013). The first one assigns each region identified by a segmentation map to one of the classes using the vector mean of each class as a feature. This approach has been shown to have low classification accuracy because the use of the vector mean results in a loss of spectral information (Linden et al. 2007). The second one is based on combining the spatial and spectral information in a feature vector that is defined for each pixel and later used for the classification process. The third option is based on combining the spatial segmentation results with the pixel-wise spectral classification results by the known as a plurality vote step, that is usually implemented by majority vote (MV) (Tarabalka, Benediktsson and Chanussot 2009). Different possible schemes based on this approach, and which combine different methods for segmentation with SVM as a spectral classifier, have been studied extensively in (Tarabalka, Chanussot, and Benediktsson 2010a) for remote-sensing images, obtaining good results.

Recently Huang et al. (Huang, Zhu, and Siew 2004) have proposed a technique known as Extreme Learning Machine (ELM) to train Single-hidden Layer Feedforward Neural Networks (SLFNs). Compared to more traditional computational intelligence techniques, including SVM, ELM has proved to be an alternative in terms of generalization performance, learning speed and computational scalability. The principle which ELM is based on is that of an SFLN. The method uses random computational nodes which are independent of the training samples. This makes the algorithm independent of a tuning process, except for the election of the network architecture (number of nodes in the hidden layer) so the number of user-defined parameters is lower than for the SVM with the subsequent saving in computational time. The learning process is faster than traditional gradient-based approaches, such as the back-propagation algorithm (Haykin 2001) and it has good generalization performance (Huang, Zhu, and Siew 2004). The hidden nodes can be radial basis function (RBF) nodes (Huang, Chen, and Siew 2008) or wavelets (Cao, Lin and Huang 2011), among others. The random selection of hidden node parameters and the lack of tuning lead to low computational times for the ELM training phase (Rong et al. 2008). Regarding the universal approximation capability of ELM (Huang, and Chen 2007, 2008) provide a more general theorem which avoids the differential condition on activation functions.

The use of ELM for hyperspectral images has only recently been explored in a small number of works (Pal 2009; Chang 2011; Pal, Maxwell, and Warner 2013). In (Pal 2009) a pixel-wise ELM with the sigmoid activation function applied to land cover classification offers slightly higher accuracy results than a backpropagation trained multilayer neural network for the two datasets considered. (Chang 2011) has applied the PL-ELM classifier (Partial Lanczos ELM) to analyze the change of land use detected in hyperspectral images. The results indicate that PL-ELM outperforms other classification methods, including SVM. More recently (Pal, Maxwell, and Warner 2013) compares a raw pixel-wixe SVM algorithm to a kernel-based ELM algorithm using a RBF function. The ELM results are slightly better in terms of accuracy. Moreover, the method has notably lower computational cost and does not require the implementation of a multiclass strategy.

In this work we study and evaluate the application of two spatial-spectral ELM-based techniques to the land cover classification from hyperspectral images compared to similar published SVM-based techniques. The main contributions of this paper are:

- The use of ELM for the classification of hyperspectral images is validated by its application to real test images extensively used in the literature. Very low execution times for the ELM implementation in OpenMP are obtained on a 4-core multicore CPU. The higher accuracy values obtained by the classification and regularity among classes of the accuracy values as well as the lower execution times in comparison with the SVM classifier, indicate that ELM is a promising classification approach.
- The application of ELM to remote sensing images is further explored. Two spatial-spectral classification techniques based on ELM are proposed. The proposed techniques apply majority vote in order to incorporate the spatial/textural information.

  The first technique, which we call watershed-based ELM, combines, through a majority vote approach, the results of ELM with the spatial information of a segmentation map obtained by a watershed algorithm. In this case the neighborhood is of the variable size of a watershed region. The second technique, which we call spatially regularized ELM, introduces the spatial information of the local neighborhood by taking into account information from the ELM classification on each pixel and on the neighbors in a fixed neighborhood.

The rest of this paper is organized as follows: Section 2 presents the ELM application to pixel-wise classification of hyperspectral images; in Section 3 two proposals are presented for integrating the spatial information with the spectral ELM classification; classification results and execution times for the proposed techniques are included in Section 4; and, finally, in Section 5 the conclusions are presented.

2. ELM

This section briefly describes the raw pixel-wise ELM algorithm. ELM was proposed as an efficient learning algorithm for single-hidden layer feedforward neural networks (SLFNs) (Huang, Zhu, and Siew 2006). The structure of an SLFN is shown in Fig.1. The output function of an SLFN with L hidden nodes an $\boldsymbol{x}$ the input vector can be written as

$$f_L(\boldsymbol{x}) = \sum_{i=1}^{L} \boldsymbol{\beta}_i G(\boldsymbol{a}_i, b_i, \boldsymbol{x}), \quad \boldsymbol{x} \in \mathrm{R}^d,\ \boldsymbol{\beta}_i \in \mathrm{R}^m, \tag{1}$$

where $G(\boldsymbol{a}_i, b_i, \boldsymbol{x})$ denotes the output function of the ith hidden node, being $\boldsymbol{a}_i$, b_i and $\boldsymbol{\beta}_i$ the hidden node parameters and the weight vector connecting the ith hidden node to the output nodes, respectively. For the case of additive nodes with activation function g, it can be expressed as

$$G(\boldsymbol{a}_i, b_i, \boldsymbol{x}) = g(\boldsymbol{a}_i \cdot \boldsymbol{x} + b_i), \quad \boldsymbol{a}_i \in \mathrm{R}^d, \; b_i \in \mathrm{R}, \tag{2}$$

An SLFN with L hidden nodes can approximate N arbitrary distinct samples $(\boldsymbol{x}_i, \boldsymbol{t}_i) \in \mathbf{R}^d \times \mathbf{R}^m$, i.e.,

$$\sum_{j=1}^{N} ||f_L(\boldsymbol{x}_j) - \boldsymbol{t}_j|| = 0, \tag{3}$$

if there exist $\boldsymbol{a}_i$, b_i and $\boldsymbol{\beta}_i$ such that

$$\sum_{i=1}^{L} \boldsymbol{\beta}_i G(\boldsymbol{a}_i, b_i, \boldsymbol{x}_j) = \boldsymbol{t}_j, \quad j = 1, \ldots, N. \tag{4}$$

The above N equations can be written compactly as:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{T} \tag{5}$$

where

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{h}(\boldsymbol{x}_1) \\ \vdots \\ \boldsymbol{h}(\boldsymbol{x}_N) \end{bmatrix} = \begin{bmatrix} G(\boldsymbol{a}_1, b_1, \boldsymbol{x}_1) & \ldots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_1) \\ \vdots & \ldots & \vdots \\ G(\boldsymbol{a}_1, b_1, \boldsymbol{x}_N) & \ldots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_N) \end{bmatrix}_{N \times L}, \tag{6}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_L^T \end{bmatrix}_{L \times N}, \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} \boldsymbol{t}_1{}^T \\ \vdots \\ \boldsymbol{t}_N{}^T \end{bmatrix}_{N \times m}. \tag{7}$$

$\mathbf{H}$ is called the hidden layer output matrix of the neural network. Huang et al. (Huang, Chen, and Siew 2006; Huang, Wang and Lan 2011) have proved that an SLFN with randomly generated additive or RBF nodes in the hidden layer can universally approximate any continuous target function over any compact subset $\boldsymbol{X} \in \mathbf{R}^d$. For the case of additive nodes, the activation function g can be any infinitely differentiable function, and this includes the sigmoidal functions, as well as the radial basis, sine, cosine and exponential functions among others. Hidden node parameters $(\boldsymbol{a}_i, b_i)$ remain fixed after randomly generated and training an SLFN is equivalent to finding a least-squares solution $\hat{\boldsymbol{\beta}}$ of the linear system $\mathbf{H}\boldsymbol{\beta} = \mathbf{T}$:

$$||\mathbf{H}\hat{\boldsymbol{\beta}} - \mathbf{T}|| = \min_{\boldsymbol{\beta}} ||\mathbf{H}\boldsymbol{\beta} - \mathbf{T}||. \tag{8}$$

If the number L of hidden nodes is equal to the number N of training samples, $L = N$, according to Theorem 2.1 in (Huang, Chen, and Siew 2006), matrix $\mathbf{H}$ is square and invertible when hidden node parameters $(\boldsymbol{a}_i, b_i)$ are randomly chosen, and thus SLFNs can approximate these training samples with zero error without the need of the iterative tuning required for other learning mechanisms. However, in most cases the number of hidden nodes is much lower than the number of distinct training samples, therefore $\mathbf{H}$ is a nonsquare matrix and there may not exist $\boldsymbol{a}_i, b_i, \boldsymbol{\beta}_i$ $(i = 1, \dots, L)$ such that $\mathbf{H}\boldsymbol{\beta} = \mathbf{T}$. The smallest norm least-squares solution of the above linear system is:

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T}, \tag{9}$$

where $\mathbf{H}^{\dagger}$ is the *Moore-Penrose generalized inverse* of matrix $\mathbf{H}$ [37, 38]. Thus, ELM can be summarized as follows (Huang, Chen, and Siew 2006; Huang, Wang and Lan 2011):

Algorithm ELM: Given a training set $\aleph = \{(\boldsymbol{x}_i, \boldsymbol{t}_i) | \boldsymbol{x}_i \in \mathbf{R}^d, \mathbf{t}_i \in \mathbf{R}^m, i = 1, \dots, N\}$, hidden node output function $G(\boldsymbol{a}_i, b_i, \mathbf{x})$, and hidden node number L,

(1) Randomly generate hidden node parameters $(\boldsymbol{a_i}, b_i)$, $i = 1, \dots, L$ where $\boldsymbol{a}_i$ and b_i are the input weight and bias values.

(2) Calculate the hidden layer output matrix $\mathbf{H}$.

(3) Calculate the output weight vector, $\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T}$.

As it was stated in (Huang et al. 2012), ELM requires less human intervention than SVM. It also achieves similar or better generalization performance for binary class classification cases, and much better for multiclass classification. In addition, ELM has better scalability and it runs at much faster learning speed than SVM.

3. ELM-based spatial-spectral methods

In this section, two possible schemes for the classification of hyperspectral images, combining spatial and spectral information by different methods, are applied to ELM in order to show its effectiveness as a spectral classifier. Both methods apply a process known as plurality vote or majority vote (Tarabalka, Benediktsson and Chanussot 2009; Cao et al. 2011) in the sense that, in some step, they calculate the class to one pixel based on the class that is majority in a neighborhood of a size depending on the technique. In Section 3.1 a spectral-spatial scheme based on combining the results of spectral classification with the result of a separate non-supervised segmentation is presented. In order to increase the spatial information included in the process, a spatial regularization of the results obtained by ELM is proposed in Section 3.2.

3.1 *Watershed-based spatial-spectral ELM*

The first spatial-spectral classification technique selected is based on capturing the spatial/textural information on the hyperspectral image using an unsupervised watershed algorithm. The segmentation map and the classification outcome obtained by a pixel-wise classifier are combined using a majority vote (MV) process. This approach, first proposed in (Tarabalka, Chanussot, and Benediktsson 2010a) for the case of the SVM classifier, has been successfully applied to a variety of remote sensing images by different authors (Tarabalka, Benediktsson and Chanussot 2009; Tarabalka, Chanussot, and Benediktsson 2010a,b; Fauvel et al. 2013). We propose to extend it for the ELM classifier.

The scheme consists of the stages described in Fig. 2. On one hand, a pixel-wise classifier is applied over the hyperspectral image producing a classification map, as shown in the top of the figure for the case of three classes. The tag of each pixel indicates the class for the pixel, represented in the figure by a color. On the other hand, the spatial processing is performed. The region-based segmentation by a watershed transform is a sound choice as it obtains good results even when the contrast of the image is poor (Serra 1983).

Computing the gradient over a multiband image can proceed in different ways (Tarabalka, Benediktsson and Chanussot 2009). The option selected here is to compute a vectorial gradient than reduces the original multiband image to a one-band image. In this scheme a vectorial gradient operator based on the Euclidean distances of pixel vectors, called Robust Color Morphological Gradient (RCMG), is applied (Evans and Liu 2006). RCMG reduces the dimensionality of the original hyperspectral image so that finally a one-band image whose borders are enhanced and thinned is obtained. Afterwards, a watershed algorithm based on a cellular automaton (Quesada-Barriuso, Argello, and Heras 2012) generates a segmentation map. The watershed algorithm follows the Hill-Climbing algorithm based on the topographical distance by Meyer (Meyer 1994). In this map the different regions are labelled by different letters, as can be shown in the bottom of Fig. 2.

Finally the spectral and the spatial results are combined using a majority vote (MV) process (Tarabalka, Benediktsson and Chanussot 2009). The MV is performed after the ELM within each watershed region in such a way that each pixel in the region is assigned to the most predominant class within the region. The output of this scheme is a more accurate hyperspectral classification of the image compared to the standalone spectral classification. Fig. 3 shows the process of MV considering three classes in the classification map and three watershed regions in the segmentation map, so the neighborhood considered by each pixel is variable and of the size of the watershed region it belongs to.

3.2 *Spatially regularized ELM*

The classification technique described in this section introduces spatial information from the closest neighborhood of a pixel by a regularization process applied over the classification map.

ELM adopts a One-Against-All (OAA) method to decompose multi-classification applications into multiple binary classifiers and transforms the classification application to a multi-output function regression problem (Cao et al. 2011). Considering m possible different class labels, a m-dimensional vector $(t_{i1}, t_{i2}, \ldots t_{im})$ with $t_{ic} \in \{1, -1\}$ $(c = 1, 2, \ldots, m)$ will be computed for each sample $\mathbf{x}_i$ in the training phase. With this One-Against-All approach, if the sample $\mathbf{x}_i$ belongs with the highest probability to a class c, in the training phase the class label t_{ic} of the sample $\mathbf{x}_i$ will be set to 1 and the other values of the vector will set to -1. According to this, in the testing phase, for a testing sample $\mathbf{x}_i$ the index j of the largest value t_{ij} in the output vector will indicate the class of the sample.

With the spatially regularized ELM approach the OAA solution is slightly changed by considering spatial information in the closest neighborhood. First the ELM is applied. For each test sample not only the most probable class, but also the second and the third most probable classes are stored. Afterwards, when all the samples have their class, the set of samples is traversed again and for each sample the class will be set to the most prevalent class in the set of the closest neighborhood if this value is among the three most probable classes for the sample. In other case, the class does not change. The process is repeated over all the samples until stability is achieved.

Thereby disconnected points, which are frequently misclassified (Cao et al. 2011), initially change their class label depending on the class assigned to the neighbors, hence the accuracy of the classification increases. An additional effect is that the edges of the classes are smoothed out solving the problem of classification accuracy usually present in these points. The problem of avoiding disconnected points in the classification map was also studied in (Tarabalka, Benediktsson and Chanussot 2009), in which a regularization step is performed after a spatial-spectral scheme based on SVM by filtering the classification map using Chamfer neighborhood masks.

Figure 4 displays the effect of the regularization explained above. First the pixel-wise ELM classification is performed and, as a result, each pixel is tagged as belonging to one of the three possible classes. Some disconnected points, i.e., which belong to one class but are surrounded by pixels of other classes, are present. Then, the classification map obtained by ELM is regularized. The class for each pixel is decided by analyzing information of other pixels in a fixed size 8-cell neighborhood. As a result, indicated as *regularization* in the figure, the size and shape of the classes change. In a final step, the regularized classification map is combined with the watershed segmentation map using a majority vote with an adaptive neighborhood of the size of the watershed region in question. All the pixels are assigned to the most frequent class within their region.

4. Evaluation

In this section we present and discuss the results obtained after applying the spatial-spectral ELM techniques proposed in the previous sections to land cover from hyperspectral images. A comparison with the results obtained after applying similar SVM-based techniques is studied. The results are analyzed in terms of classification accuracy, quantity and allocation disagreements and computational cost of the methods. First, in Section 4.1 we present the experimental conditions. The results of applying raw pixel-wise ELM and SVM classifiers are compared in Section 4.3. Finally in Sections 4.4 and 4.5 the results applying the proposed spatial-spectral ELM techniques are analyzed.

4.1 *Hyperspectral image dataset and experimental conditions*

The land cover classification is made on publicly available hyperspectral datasets. All experiments with ELM are performed in OpenMP language using the OpenBLAS library (OpenBLAS 2013). The computer platform is a four-core Intel Core i7 860 CPU with 8 GB of RAM using a 64-bit Linux GNU compiler with the -O3 flag.

For the ELM experiments the random values for the input weights and biases were generated by the *rand48* generator from the standard C library. For the inverse calculation the Moore-Penrose generalized method was used (Courrieu 2005). The activation function selected is the sigmoidal activation function $g(x) = \frac{1}{1+e^{-x}}$. As it was explained in Section 2, for the ELM the user only has to decide two parameters: the number of neurons in the hidden layer of the SLFN and the number of training samples per class. An analysis varying these parameters for the two datasets considered is shown in the next section.

In the case of SVM the Gaussian radial basis function (RBF) is used as the activation function. The user must decide the number of training samples per class. Using RBF, two parameters (C and γ) need also to be optimized, corresponding to a penalty term and the width of the radius of the Gaussian function, respectively. In this work, the number of samples per class and the parameters C and γ were chosen as in (Tarabalka, Chanussot, and Benediktsson 2010a,b), because a comparison to results published in that work is performed. The parameters for training the SVM extracted from these publications are $C = 128$ and $\gamma = 2^6$ for the *Indian Pines* image, and $C = 128$, $\gamma = 0.125$ for the *University of Pavia* image. The number of training samples per class will be shown in the next sections. The SVM execution times presented in Section 4.3 were obtained with the same parameters and using the standard LIBSVM library (Chang and Lin 2011).

The hyperspectral datasets used in the experiments are the *University of Pavia* image (Rosis 2013), which was obtained by the ROSIS-03 optical sensor, and the *Indian Pines* image (AVIRIS 2013), obtained by the AVIRIS. Reference data, also called ground truth, are avalaible for both images in order to evaluate the classification results. The input data for the experiments are normalized in the range $[0, 1]$.

The first dataset for the experiments is a 103-band image from the *University of Pavia* at Pavia, Italy, with a spatial dimension of 610 × 340 pixels with spatial resolution of 1.3 m per pixel. The ROSIS-03 sensor captures 115 spectral bands (the twelve most noisy bands of this image were removed) in the spectral range from 0.43 to 0.86 μm, corresponding mainly to the visible spectrum. The reference data contain 9 classes of interest corresponding to different urban elements. A false color image and the reference data are presented in Fig.5. The ground truth covers only 20.62% of this image.

The second image, see Fig.6, is a 220-band dataset of a 2 × 2 mile portion of agricultural area in Northwest Indiana. The image has spatial dimensions of 145 × 145 pixels with a spatial resolution of 20 m per pixel. The AVIRIS sensor has 220 spectral bands and a spectral range from 0.4 to 2.5 μm, corresponding to the visible and infrared spectrum. The scene contains two-thirds agriculture and one-third forest or other natural perennial vegetation. Some of the crops present (corn, soybeans, etc) are in early stages of growth with less than 5% coverage. There are two major dual lane highways, smaller roads, a rail line, some low density housing and other built structures. The reference data contain 16 classes of interest, which cover 49.30% of the full 145 × 145 image.

In all the experiments performed using the different classification methods, a random training sample of pixels for each class was selected from the ground truth. When the SVM is applied, the number of training pixels per class is the same as in (Tarabalka, Chanussot, and Benediktsson 2010a) and the trained classifiers are applied to the remaining known ground pixels. The SVM results for the test images are taken from the literature.

For the case of the ELM in order to select the model a number of trials changing the configuration were conducted, as it will be explained in the next section. Finally, 500 and 950 neurons in the hidden layer for the *University of Pavia* and the *Indian*

Pines images, respectively, and 200 training samples per class, or half the number of samples in the class when this number is lower than 200, were selected for the remaining of the paper. And since, as some authors have reported (Zhu et al. 2005; Wang, Cao, and Yan 2011), the efficiency of the classification is dependent on the random assignment of input weights and biases, we have repeated 100 experiments for each ELM configuration. The result for each pixel is the most repeated class in the 100 experiments.

The classification results are always validated by comparing to the ground truth and in terms of overall accuracy (OA), average accuracy (AA), Kappa coefficient (κ). The OA is the percentage of correctly classified pixels in the whole image. The AA is the mean of the class accuracy for all classes considering that the class-specific accuracy is the percentage of correctly classified pixels for a given class. κ is the percentage of agreement among the classification results and the information from the ground truth corrected by the number of agreements that would be expected accidentally (Richards and Jia 1999). Besides, as (Pontius, and Millones 2010) recommend, two new components of disagreement between classification maps in terms of quantity and spatial allocation of the classes are also computed for the proposed techniques. The quantity disagreement (QD) is defined as the amount of difference between the ground truth and a comparison map that is due to the less than perfect match in the proportions of the classes. The allocation disagreement (AD) is defined as the amount of difference between the ground truth and a classification map that is due to the less than optimal match in the spatial allocation of the categories, given the proportions of the classes in the ground truth and the classification map.

4.2 *ELM model structure*

In the literature, a variety of methods for the proper selection of the model structure of ELM were proposed (Heeswijk et al 2011). Nevertheless, Huang et al. (Huang et al. 2012) proved that with sigmoid additive hidden nodes, as in the present case, ELM can achieve a good generalization performance provided the number of hidden nodes is large enough. Here a study on the influence of the structure of the ELM, number of training samples per class and number of neurons in the hidden layer, is explained.

Figures 7 and 8 show the results of varying the parameters for both test images. Each experimental point in the graphs is the results of 100 executions, as explained in Section 4.1. Considering the relevance of achieving near real-time processing, especially for certain applications, the execution times must also be tested. The ELM implementation has been performed in OpenMP with calls to the OpenBLAS library for the matricial operations. The results are also displayed in the figures (bottom graphs).

The graphs on the left of Figs. 7 and 8 display the classification accuracy and execution times when varying the number of neurons in the hidden layer and considering 200 samples per class or 50% of the available samples of the class are selected. It is interesting to note that for both images when the number of neurons is increased, the accuracy also increases until one value where it becomes constant. If the number of neurons is further increased, beyond one point the accuracy begins to decrease again. The limits for these behaviors depend on the number of channels per pixel and on the number of training and classification samples, so the tuning process must be repeated for each image in order to achieve the best configuration possible. Nevertheless, the low sensitivity to the change in the number of neurons observed in the accuracy indicates that, in line with (Huang et al. 2012), a sufficient number of hidden neurons would provide good enough results.

It can be observed that, for the case of the *University of Pavia* image (Fig. 7), the best number of neurons is around 500. For the case of the *Indian Pines* image (Fig. 8) the best values are obtained around 950 neurons. Comparing the outcomes for both images, as other authors have reported by using different classification methods, worse results are obtained for the *Indian Pines* image as the low spatial resolution of the image leads to the presence of highly mixed pixels which complicates the classification problem (Tarabalka, Chanussot, and Benediktsson 2010a). Furthermore, there are some very small classes for this image.

Regarding the execution times, when the number of neurons in the hidden layer is varied, the values for the whole process are also below 4.5 and 3 seconds for the *University of Pavia* and the *Indian Pines* images respectively. The execution times continuously increase with the number of neurons, the classification time always being higher than the training time.

The results when the number of training samples is modified are displayed on the graphs on the right of Figs. 7 and 8, considering 500 and 950 neurons in the hidden layer for the *University of Pavia* and *Indian Pines* images, respectively. The figures indicate similar trends in accuracy and execution times when the number of samples is increased. It is worthy of note that for these experiments the times are below 2.5 and 1.7 seconds for the *University of Pavia* and *Indian Pines* images respectively.

4.3 *ELM versus SVM*

In this section, the raw pixel-wise ELM method (Huang, Wang and Lan 2011) described in Section 2 is applied to the datasets. The classification results are analyzed and compared to those obtained by the pixel-wise SVM classifier. The analysis is based on the standard accuracy indices OA, AA and κ, and on execution times.

The execution times for the training and classification stages comparing the pixel-wise SVM and ELM classifiers for the test images and for the configurations that will be used in the remaining of the paper are summarized in Table 1. L denotes the number of neurons in the hidden layer. Lower values are obtained for ELM, around 35 and 6 times lower than for SVM. It is well known that SVM is a computationally demanding algorithm for high-dimensional data or when the number of samples is large (Fauvel, Chanussot, and Benediktsson 2008; Tarabalka, Benediktsson and Chanussot 2009). The main difference between SVM and ELM is on the classification time because the SVM classification always requires classification by pairs of classes to be performed in a OAO fashion, as we have explained in Section 1. The classification time increases with the number of classes as where N is the number of classes, $\frac{N^2-N}{2}$ is the number of one-to-one classifications that must be performed. In the case of the ELM, however only one matrix operation is performed for classification, increasing the size of the matrix with the number of classes. In addition, in this work a very efficient implementation of the ELM for multicore architectures has been performed using OpenMP language and performing calls to the OpenBLAS library, thus, the multicore CPU avalaible is fully exploited.

Tables 2 and 3 compare the pixel-wise ELM classifier with SVM highlighting the best results in bold. The SVM results are taken directly from (Tarabalka, Benediktsson and Chanussot 2009) and (Tarabalka, Chanussot, and Benediktsson 2010b) where a 5-fold cross validation was used in order to tune the SVM parameters.

Given that these papers do not include the confusion matrices, QD and AD values can not be calculated for the SVM. The same will be observed in the remaining tables. Better accuracy results are obtained in general for the *Indian Pines* image as for each class the points are more clustered (see the ground truths in Figs. 5 and 6). Inspecting the classification accuracies for the different classes the ELM accuracy results are higher and more uniform among classes than in the case of SVM. As a result, ELM obtains in all cases better OA and AA results than SVM. For the case of the *University of Pavia* image the increase in OA of ELM over SVM is of 5.66 points, and for the case of the *Indian Pines* image, for which the SVM already gives very good results, 2.48.

4.4 *Watershed-based ELM results*

In these experiments, the spatial-spectral classification scheme described in Section 3.1 is tested. First, an unsupervised segmentation by watershed is performed over the test images. The segmentation map is combined with the classification map obtained by SVM or ELM by majority vote inside the watershed regions.

Tables 4 and 5 give class specific accuracies and global OA, AA, and κ values for the spatial-spectral classification schemes based on applying the watershed-based classification to the test images. Additional results in terms of QD and AD are also offered for the ELM-based schemes. Results are shown for the case of considering SVM (SVM+wat column in the table) and ELM (ELM+wat column in the table) as the spectral classifiers. The first conclusion, comparing to the previous Tables 2 and 3, is that the spatial-spectral scheme outperforms the pixel-wise results in all cases. Overall, the *University of Pavia* image gives the best results with an OA improvement of 8.03 points of the spatial-spectral scheme based on ELM over the raw ELM. A similar behavior is observed for the *Indian Pines* image obtaining 9.78 points of maximum improvement. The best overall results are those obtained by the spatial-spectral scheme based on ELM with with OA of 94.70% for *Pavia* and 90.43% for *Indiana*.

4.5 *Spatially regularized ELM results*

In these experiments the spatial regularization process explained in Sect. 3.2 is applied to the pixel-wise ELM classifier, as well as, to the spatial-spectral technique based on the watershed application. The regularization introduces additional spatial/textural information by considering separately for each pixel the classes that the ELM classifier assigns to the neighbors and repeating the regularization process till stabilization of the class tags. All the experiments involving the spatially regularized technique were applied using a fixed 8-cell neighborhood.

Tables 6 and 7 summarize the results obtained by applying the spatial regularization process to the test images. The third column in the tables shows the results of the regularization over the raw ELM. The pixel-wise results are clearly improved by the regularization. The fourth and fifth columns show the effect of the regularization process applied after the watershed-based scheme. Although the spatial-spectral scheme ELM+wat already introduces spatial information from the watershed segmentation, there is still a margin for the improvement, especially, as explained in Section 3.2, on the borders of the classes and on the disconnected points inside the classes that are the objectives of the regularization. The OA values obtained by ELM+wat+reg are 95.12 and 90.93 for the *Indian Pines* and the *University of Pavia* images, respectively. This scheme also offers the best results in terms of QD and AD.

An study of the regularization technique applied after a spatial-spectral SVM scheme is published in the literature (Tarabalka, Benediktsson and Chanussot 2009). In particular, in this last work the regularization is called post-regularization, the segmentation algorithms are ISODATA and EM clustering considering 4 neighbors and the results are available for our test images. Although the method is slightly different from the applied here with ELM, as the regularization is applied after the majority vote, in Tables 8 and 9 the best results for each image, extracted from (Tarabalka, Benediktsson and Chanussot 2009), are incorporated. They are also outperformed by the spatially regularized based ELM combined with watershed (ELM+wat+reg). These tables summarize the OA, AA and κ values for the avalaible techniques and the QD and AD values for the ELM-based techniques. Although good results have been obtained for the spatial-spectral classifier based on SVM, the results show that the same technique with ELM surpasses the SVM-based one, and that the best values in all the parameters are obtained for the spatially regularized watershed-based technique with ELM.

Figures 9 and 10 help in understanding the results. They show, from left to right, the color composite classification maps for the pixel-wise ELM classifier, the watershed-based ELM spatial-spectral technique and the regularized watershed-based ELM spatial-spectral technique. The incorporation of spatial information into the classifier using the majority vote approach (watershed-based ELM) leads to more homogeneous objects than the pixel-wise ELM in the same fashion as it can be observed for SVM in (Tarabalka, Benediktsson and Chanussot 2009) and (Tarabalka, Chanussot, and Benediktsson 2010b). The improvement for the case of spatial regularization is not so obvious from the figures. The reason is that the effect is small and only happens for some classes. In the case of the *Pavia* image the improvement is mainly on the *bricks* class, as can be seen in Table 7, for which there are a large number of edges that are smoothed by the regularization step (See the reference data for the different classes in Fig. 5). For the case of the *Indiana* image, it can be seen in Table 7 that the improvement is mainly on the *Corn-nontill*, *Corn-mintill* and *Soybean-mintill* classes. In particular, in the last of the three classes (see the reference data in Fig. 6) it can be seen that the spatial regularization removes the disconnected points in the flat zone that represents the class, thus avoiding missclassification of that points.

5. Conclusions

In this paper the spatial-spectral classification of land cover from hyperspectral images is addressed. In particular, the benefits of an ELM classification over the traditionally used SVM classification is studied. In addition, two spatial-spectral techniques based on a pixel-wise ELM classifier are explored and compared to the analogous techniques based on a pixel-wise SVM.

The first spatial-spectral technique called ELM+wat leads to more homogeneous regions and higher classification accuracy when compared to pixel-wise classification. The second one, called ELM+wat+reg avoids the missclasification that is usual on isolated pixels and smooths the edges of the classes, solving the classification accuracy problem usually present in this kind of pixels.

The experiments have being carried out on two frequently used land cover images: the *University of Pavia* image and the *Indian Pines* image. The OA values for ELM are 5.66 points over SVM for the *University of Pavia* image and 1.22 points for the *Indian Pines* image. In terms of execution time the ELM is a lower demanding algorithm and thus the times are divided by 35.7 and 6.1 with respect to the SVM. The best results are obtained when the spatial/textural information is introduced in the classification ELM scheme not only by regularization but also by watershed followed by majority vote inside the watershed regions (ELM+wat+reg). For this case, the best results are obtained, OA values of 95.12% and 90.93% for *Pavia* and *Indian Pines*.

The techniques based on ELM yielded higher classification accuracy, more regularity among classes and lower execution time than the analogous SVM-based techniques showing that ELM is competitive with SVM. Hence, more spatial-spectral classification techniques based on ELM need to be explored for land cover classification from hyperspectral images.

Acknowledgements

This work was supported in part by the Ministry of Science and Innovation, Government of Spain, cofounded by the FEDER funds of European Union, under contract TIN 2010-17541, and by Xunta de Galicia, Program for Consolidation of Competitive Research Groups ref. 2010/28. Pablo acknowledges financial support from the Ministry of Science and Innovation, Government of Spain, under a MICINN-FPI grant.

References

AVIRIS Indian Pine Test Site 3 dataset, Available on-line: ftp://ftp.ecn.purdue.edu/biehl/MultiSpec/92AV3C.tif.zip (hyperspectral image) and ftp://ftp.ecn.purdue.edu/biehl/MultiSpec/ThyFiles.zip (ground truth), (accessed April 3, 2013).

Bai, X. and Sapiro, G.. 2007. “A geodesic framework for fast interactive image and video segmentation and matting”. In *Proc. ICCV*. 1–8.

Benediktsson, J. A., Palmason, J. A., and Sveinsson, J. R. 2005. “Classification of hyperspectral data from urban areas based on extended morphological profiles.” *IEEE Trans. Geosci. Remote Sensing* 42 (3): 480–491.

Camps-Valls, G., Gmez-Chova, L., Muoz-Mar, J., Vila-Francs, J., and Calpe-Maravilla, J. 2006. “Composite kernels for hyperspectral image classification.” *Geoscience and Remote Sensing Letters* 3 (1): 93–97.

Cao, X., Chen, J., Matsushita, B., Imura, H., and Wang, L. 2009. An automatic method for burn scar mapping using support vector machines. *International Journal of Remote Sensing* 30 (3): 577–594.

Cao, J. W., Lin, Z. P., and Huang, G. B. 2011. “Composite function wavelet neural networks with differential evolution and extreme learning machine.” *Neural Processing Letters* 33 (3): 251–265.

Cao, J., Lin, Z., Huang, G. B., and Liu, N. 2011. “Voting based extreme learning machine.” *Information Sciences* 185: 66–77.

Chang, Ni-Bin. 2011. “Satellite-based multitemporal-change detection in urban environments.” *SPIE Newsroom*. DOI: 10.1117/2.1201101.003502.

Chang, C. C., and Lin, C. J. 2011. “LIBSVM: a library for support vector machines.” *ACM Transactions on Intelligent Systems and Technology* 2 (3): 27.

Courrieu, P. 2005. “Fast Computation of Moore-Penrose Inverse Matrices.” *Neural Information Processing* 8 (2): 25–29.

Evans, A., and Liu., X. 2006. “A morphological gradient approach to color edge detection. *IEEE Trans. on Image Processing* 15 (6): 1454–1463.

Fauvel, M., Chanussot, J., and Benediktsson, J. A. 2006. “Evaluation of kernels for multiclass classification of hyperspectral remote sensing data." In *IEEE International Conference on Acoustics, Speech and Signal Processing 2006 (ICASSP 2006)* 2: II.

Fauvel, M., Chanussot, J., and Benediktsson,J. A. 2012. "A spatial-spectral kernel-based approach for the classification of remote-sensing images." *Pattern Recogn* 45: 381–392.

Fauvel, M., Chanussot, J., Benediktsson, J. A. and Sveinsson, J. R. 2008. "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles", *IEEE Trans. Geosci. Remote Sens.* 46 (10): 3804–3814.

Fauvel, M., Tarabalka, Y., Benediktsson, J. A., Chanussot, J., and Tilton, J. C. 2013. "Advances in Spectral-Spatial Classification of Hyperspectral Images." *Proceedings of the IEEE* 101 (3): 652–675.

Haykin, S. 2001. *Neural Networks, A Comprehensive Foundation.* Second ed., Pearson education Press.

Van Heeswijk, M., Miche, Y., Oja, E., and Lendasse, A. 2011. "GPU-accelerated and parallelized ELM ensembles for large-scale regression." *Neurocomputing* 74 (16): 2430–2437.

Huang, G. B., and Chen, L. 2007. "Convex Incremental Extreme Learning Machine", *Neurocomputing*, 70: 3056–3062.

G. B. Huang, and L. Chen. 2008. "Enhanced Random Search Based Incremental Extreme Learning Machine", *Neurocomputing* 71: 3460–3468.

Huang, G. -B., Chen, L., and Siew , C. -K. 2006. "Universal approximation using incremental constructive feedforward networks with random hidden nodes." *IEEE Trans Neural Netw* 17 (4): 879–892.

Huang, G. B., Li, M. B., Chen, L., and Siew, C.K. 2008. "Incremental extreme learning machine with fully complex hidden nodes." *Neurocomputing* 4-6 (71): 576–583.

Huang, G -B, Zhu, Q -Y, and Siew, C -K. 2004. "Extreme learning machine: a new learning scheme of feedforward neural networks." In *Proceedings of international joint conference on neural networks (IJCNN2004)* Budapest, Hungary, 25-29 July 2004. 2: 985–990.

Huang, G. B., Zhu, Q. Y. and Siew, C. K. 2006. "Extreme learning machine: theory and applications." *Neurocomputing* 70: 489–501.

Huang, G. -B., Wang, D., Lan, Y,. 2011. "Extreme Learning Machines: A Survey." *Int. J. Mach. Learn. Cybern.* 2 (2): 107–122.

Huang, G. B., Zhou, H., Ding, X., and Zhang, R. 2012. "Extreme learning machine for regression and multiclass classification." *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 42 (2): 513–529.

Kruse, F. A., Boardman, J. W. and Huntington, J. F. 2003. "Comparison of Airborne Hyperspectral Data and EO-1 Hyperion for Mineral Mapping." *IEEE Transactions on Geoscience and Remote Sensing* 41 (6): 1388–1400.

Linden, S. V. D., Janz, A., Waske,B., Ediden, M. and Hostert, P. 2007. "Classifying segmented hyperspectral data from a heterogeneous urban environment using support vector machines." *Journal of Applied Remote Sensing* 1 (1): 1–17. doi:10.1117/1.2813466

Melgani, F., and Bruzzone, L. 2004. "Classification of hyperspectral remote sensing images with support vector machines." *IEEE Transactions on Geoscience and Remote Sensing* 42 (8): 1778–1790.

Meyer, F. 1994. "Topographic distance and watershed lines." *Mathematical Morphology and its Applications to Signal Processing* 38(1): 113–125.

Mountrakis, G., Im, J. and Ogole, C. 2011. "Support vector machines in remote sensing: A review." *ISPRS Journal of Photogrammetry and Remote Sensing* 66 (3): 247-259.

Nemmour, H., and Chibani, Y. 2006. "Multiple support vector machines for land cover change detection: An application for mapping urban extensions." *ISPRS Journal of Photogrammetry and Remote Sensing* 61 (2): 125–133.

OpenBLAS home page: [http://xianyi.github.com/OpenBLAS].

Pal, M. 2009. "Extreme-learning-machine-based land cover classification." *International Journal of Remote Sensing* 30 (14): 3835–3841.

Pal, M., Maxwell, A. E., and Warner, T. A. 2013. "Kernel-based extreme learning machine for remote-sensing image classification." *Remote Sensing Letters* 4 (9): 853–862.

Plaza, A., Benediktsson, J. A., Boardman, J. W., Brazile, J., Bruzzone, L., et al. 2009.

"Recent advances in techniques for hyperspectral image processing." *Remote Sensing of Environment* 113: S110-S122.

Pontius Jr, R. G., and Millones, M. 2010. "Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment." *International Journal of Remote Sensing* 32 (15): 4407–4429.

Quesada-Barriuso, P., Argello, F., and Heras, D. B. 2012. "Efficient segmentation of hyperspectral images on commodity GPUs." *Advances in knowledge-based and intelligent information and engineering systems* 2130–2139. ISBN 978-1-61499-104-5 (print), ISBN 978-1-61499-105-2 (online)

Richards, J. A., and Jia, X. 1999. *Remote sensing digital image analysis: an introduction.* Springer, New York.

Rong, H. J., Ong, Y. S., Tan, A. H., and Zhu, Z. 2008. "A fast pruned-extreme learning machine for classification problem." *Neurocomputing* 72 (1): 359-366.

ROSIS University of Pavia dataset, Available on-line: http://www.ehu.es/ccwintco/uploads/e/ee/PaviaU.mat (hyperspectral image) and http://www.ehu.es/ccwintco/uploads/5/50/PaviaU_gt.mat (ground truth), (accessed April 3, 2013).

Serra, J. 1983. *Image analysis and mathematical morphology.* Orlando, FL, USA: Academic Press, Inc.

Shigeo, A. 2005. *Support vector machines for pattern classification.* Springer-Verlag.

Tarabalka, Y., Benediktsson, J., and Chanussot, J. 2009. Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques. *Geosci. and Remote Sensing, IEEE Trans. on* 47 (8): 2973–2987.

Tarabalka, Y., Chanussot, J., and Benediktsson, J. A. 2010. "Segmentation and classification of hyperspectral images using watershed transformation." *Pattern Recognition* 43 (7): 2367–2379.

Tarabalka, Y., Chanussot, J., and Benediktsson, J. A. 2010. "Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers." *IEEE Trans. Syst., Man, Cybern. B, Cybern.* 40 (5): 1267–1279.

Tarabalka, Y., Fauvel, M., Chanussot, J., and Benediktsson, J. A. 2010. "SVM and MRF-based method for accurate classification of hyperspectral images." *IEEE Geosci. Remote Sens. Lett.* 7 (4): 736–740.

Vapnik, V. 1995. *The nature of statistical learning theory.* Springer-Verlag, New York.

Wang, Y. G., Cao, F. L., Yan, Y. B. 2011. "A study on effectiveness of extreme learning machine". *Neurocomputing* 74 (16): 2483–2490.

Zhu, Q. Y, Quin, A. K., Suganthan, P. N., and Huang, G.B. 2005. "Evolutionary extreme learning machine." *Pattern Recognition* 38: 1759–1763.

Table 1. Execution times in seconds for the raw pixel-wise ELM and SVM classifiers. For ELM 200 training samples per class.

	No. of nodes (L)	training time	class. time	total
University of Pavia image				
ELM	500	0.14	2.01	**2.15**
SVM[a]	-	0.58	76.27	76.85
Indian Pines image				
ELM	950	0.76	0.55	**1.31**
SVM[a]	-	0.28	7.67	7.95

[a]For SVM the parameter values and number of training samples are taken from (Tarabalka, Chanussot, and Benediktsson 2010a).

Table 2. *University of Pavia* image. For ELM average results of 100 executions, $L = 500$ nodes in the hidden layer, and 200 training samples per class or 50% of the avalaible.

Class	No. of samples in reference data	ELM	SVM[a]
1-asphalt	6631	76.36	84.93
2-meadows	18649	87.71	70.79
3-gravel	2099	80.71	67.16
4-trees	3064	94.56	97.77
5-metal sheets	1345	99.24	99.49
6-bare soil	5029	90.21	92.83
7-bitumen	1330	94.06	90.42
8-bricks	3682	82.71	92.78
9-shadows	947	99.78	98.11
overall accuracy (OA)		**86.67**	81.01
average accuracy (AA)		**89.48**	88.25
kappa (κ)		**82.50**	75.86
quantity disagreement (QD)		7.69	-
allocation disagreement (AD)		5.64	-

[a]For SVM results taken from (Tarabalka, Chanussot, and Benediktsson 2010b).

Table 3. *Indian Pines* image. For ELM average results of 100 executions, $L = 950$ nodes in the hidden layer, and 200 training samples per class or 50% of the avalaible.

Class	No. of samples in reference data	ELM	SVM[a]
1-alfalfa	54	79.19	74.36
2-corn-no till	1434	81.32	78.18
3-corn-min till	834	76.12	69.64
4-corn	234	94.09	91.85
5-grass-pasture	497	95.51	92.17
6-grass-trees	747	96.96	91.68
7-grass-pasture-mowed	26	78.92	100
8-hay-windrowed	489	99.22	97.72
9-oats	20	77.90	100
10-soybean-no till	968	83.29	82.03
11-soybean-min till	2468	65.09	58.95
12-soybean-clean	614	91.32	87.94
13-wheat	212	99.67	98.77
14-woods	1294	90.26	93.01
15-bldgs-grass-grees-drives	380	89.32	61.52
16-stone-steel-towers	95	69.73	97.78
overall accuracy (OA)		**80.65**	78.17
average accuracy (AA)		85.49	**85.97**
kappa (κ)		**77.61**	75.33
quantity disagreement (QD)		9.28	-
allocation disagreement (AD)		10.07	-

[a]For SVM results taken from (Tarabalka, Chanussot, and Benediktsson 2010b).

Table 4. Results of the watershed-based ELM and watershed-based SVM classifiers for the *University of Pavia* image. For ELM average results of 100 executions, $L = 500$ nodes, and 200 training samples per class.

Class	No. of samples in reference data	ELM	ELM+wat	SVM+wat[a]
1-asphalt	6631	76.36	88.02	93.64
2-meadows	18649	87.71	95.42	75.09
3-gravel	2099	80.71	90.13	66.12
4-trees	3064	94.56	97.00	98.56
5-metal sheets	1345	99.24	99.91	99.91
6-bare soil	5029	90.21	98.71	97.35
7-bitumen	1330	94.06	99.86	96.23
8-bricks	3682	82.71	93.80	97.92
9-shadows	947	99.78	99.85	96.98
overall acc. (OA)		86.67	**94.70**	85.42
average acc. (AA)		89.48	**95.86**	91.31
kappa (κ)		82.50	**92.95**	81.30
quantity disagreement (QD)		7.69	**3.68**	-
allocation disagreement (AD)		5.64	**1.62**	-

[a]For SVM+wat data taken from (Tarabalka, Chanussot, and Benediktsson 2010b).

Table 5. Results of the watershed-based ELM and SVM classifiers for the *Indian Pines* image. For ELM average results of 100 executions, $L = 950$ nodes in the hidden layer, and 200 training samples per class.

Class	No. of samples in reference data	ELM	ELM+wat	SVM+wat[a]
1-alfalfa	54	81.32	96.56	94.87
2-corn-no till	1434	76.12	95.35	94.22
3-corn-min till	834	94.09	86.77	78.06
4-corn	234	94.09	97.66	88.59
5-grass-pasture	497	95.51	97.26	95.08
6-grass-trees	747	96.96	98.84	97.99
7-grass-pasture-mowed	26	78.92	91.62	100
8-hay-windrowed	489	99.22	99.57	99.54
9-oats	20	77.90	43.90	100
10-soybean-no till	968	83.29	93.96	96.30
11-soybean-min till	2468	65.09	80.87	68.82
12-soybean-clean	614	91.32	97.02	90.78
13-wheat	212	99.67	100	99.38
14-woods	1294	90.26	92.87	97.11
15-buildings-grass-grees-drives	380	89.32	94.62	69.39
16-stone-steel-towers	95	69.73	84.88	95.56
overall accuracy (OA)		80.65	**90.43**	86.63
average accuracy (AA)		85.49	90.74	**91.61**
kappa (κ)		77.61	**88.84**	84.83
quantity disagreement (QD)		9.28	**6.14**	-
allocation disagreement (AD)		10.07	**3.43**	-

[a]For SVM+wat results are taken from Table I of (Tarabalka, Chanussot, and Benediktsson 2010b).

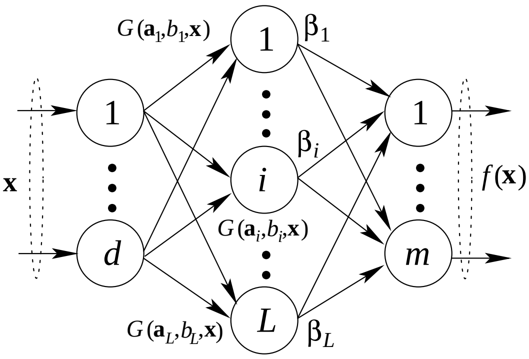

$G(\mathbf{a}_1,b_1,\mathbf{x})$
1
β_1
1
1
β_i
$\mathbf{x}$
i
$f(\mathbf{x})$
$G(\mathbf{a}_i,b_i,\mathbf{x})$
d
m
$G(\mathbf{a}_L,b_L,\mathbf{x})$
L
β_L

Figure 1. Single-hidden layer feedforward network (SLFN).

Table 6. Application of the regularization technique over the ELM and the watershed-based ELM classifiers. *University of Pavia* image. For ELM the same conditions as in Table 4. For SVM, data are not avalaible.

Class		ELM+reg	ELM+wat	ELM+wat+reg
1-asphalt	6631	83.49	88.02	88.88
2-meadows	18649	93.16	95.42	95.83
3-gravel	2099	85.30	90.13	89.59
4-trees	3064	95.93	97.00	97.09
5-metal sheets	1345	99.67	99.91	99.92
6-bare soil	5029	96.92	98.71	99.25
7-bitumen	1330	97.45	99.86	99.93
8-bricks	3682	89.55	93.80	94.48
9-shadows	947	99.78	99.85	99.85
over. acc. (OA)		92.09	94.70	**95.12**
aver. acc. (AA)		93.52	95.86	**96.09**
kappa (κ)		89.53	92.95	**93.51**
quantity disagreement (QD)		4.93	3.68	**3.45**
allocation disagreement (AD)		2.97	1.62	**1.43**

Table 7. Application of the regularization technique over the ELM and watershed-based ELM classifiers. Accuracies in percentage for the *Indian Pines* image. For ELM the same conditions as in Table 5. For SVM, data are not avalaible.

Class	No. of samples in reference data	ELM+reg	ELM+wat	ELM+wat+reg
1-alfalfa	54	80.63	96.56	96.59
2-corn-no till	1434	88.21	95.35	96.02
3-corn-min till	834	81.37	86.77	87.52
4-corn	234	94.09	97.76	97.80
5-grass-pasture	497	95.85	97.26	97.32
6-grass-trees	747	98.25	98.84	98.93
7-grass-pasture-mowed	26	81.13	91.62	90.35
8-hay-windrowed	489	99.27	99.57	99.57
9-oats	20	78.45	43.90	43.12
10-soybean-no till	968	89.98	93.96	94.47
11-soybean-min till	2468	73.25	80.87	82.02
12-soybean-clean	614	94.55	97.02	97.23
13-wheat	212	99.67	100.00	100.00
14-woods	1294	91.93	92.87	92.53
15-bldgs-grass-grees-drives	380	91.07	94.62	94.18
16-stone-steel-towers	95	74.93	84.88	85.79
over. acc. (OA)		85.71	90.43	**90.93**
aver. acc. (AA)		88.28	90.74	**90.85**
kappa (κ)		83.41	88.84	**89.42**
quantity disagreement (QD)		7.67	6.14	**5.98**
allocation disagreement (AD)		6.62	3.43	**3.08**

Table 8. Global classification accuracies in percentage for the *University of Pavia* image. "wat" indicates spatial processing by watershed. "reg" indicates that the results of the spectral classifier are spatially regularized.

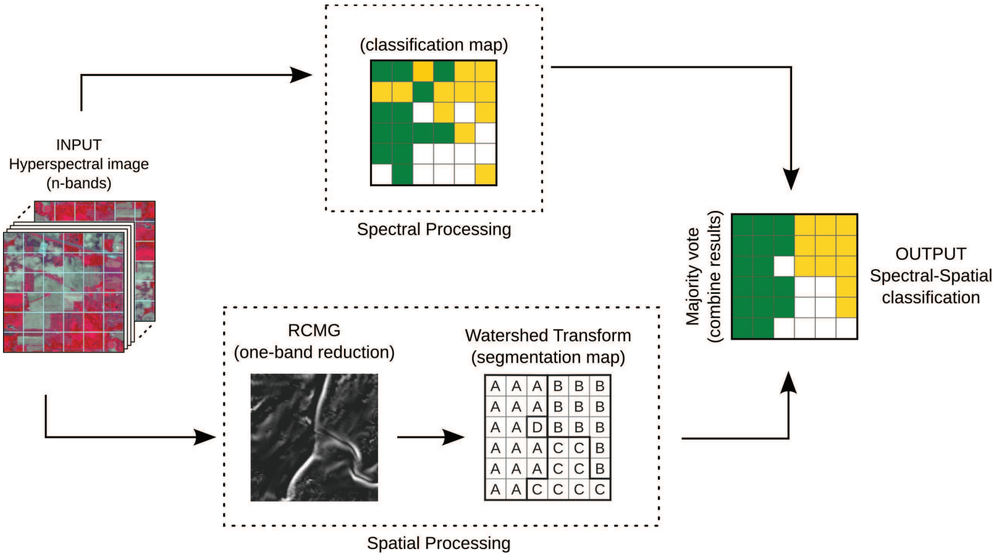
	OA	AA	κ	QD	AD
SVM	81.01	82.85	75.86	-	-
SVM+wat	86.63	91.61	84.83	-	-
SVM+EM+PR	94.68	95.21	92.92	-	-
ELM	86.67	89.48	82.50	7.69	5.64
ELM+wat	94.70	95.86	92.95	3.68	1.62
ELM+reg	92.09	93.52	89.53	4.93	2.97
ELM+wat+reg	**95.12**	**96.09**	**93.51**	**3.45**	**1.43**

[a] Data for SVM are taken from (Tarabalka, Chanussot, and Benediktsson 2010b) (SVM and SVM+wat) and from (Tarabalka, Benediktsson and Chanussot 2009) (SVM+EM+PR).

Table 9. Global classification accuracies for the *Indian Pines* image. ”wat” indicates spatial processing by watershed. ”reg” indicates that the results of the spectral classifier are spatially regularized.

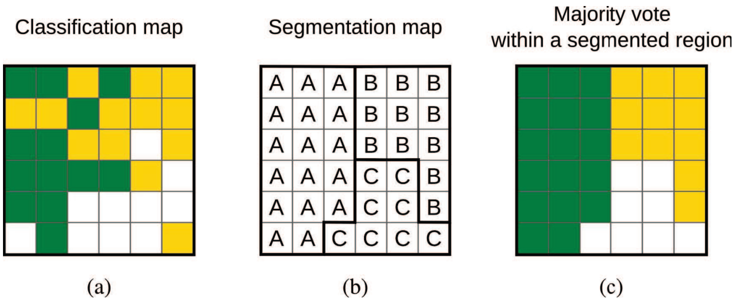
	OA	AA	κ	QD	AD
SVM[a]	78.17	85.97	75.33	-	-
SVM+wat	86.63	91.61	84.83	-	-
SVM+ISODATA+PR	90.64	80.60	89.31	-	-
ELM	80.65	85.49	77.61	9.28	10.07
ELM+wat	90.43	90.74	88.84	6.14	3.43
ELM+reg	85.71	88.28	83.41	7.67	6.62
ELM+wat+reg	**90.93**	**90.85**	**89.42**	**5.98**	**3.08**

[a] Data for SVM are taken from (Tarabalka, Chanussot, and Benediktsson 2010b) (SVM and SVM+wat) and from (Tarabalka, Benediktsson and Chanussot 2009) (SVM+ISODATA+PR).



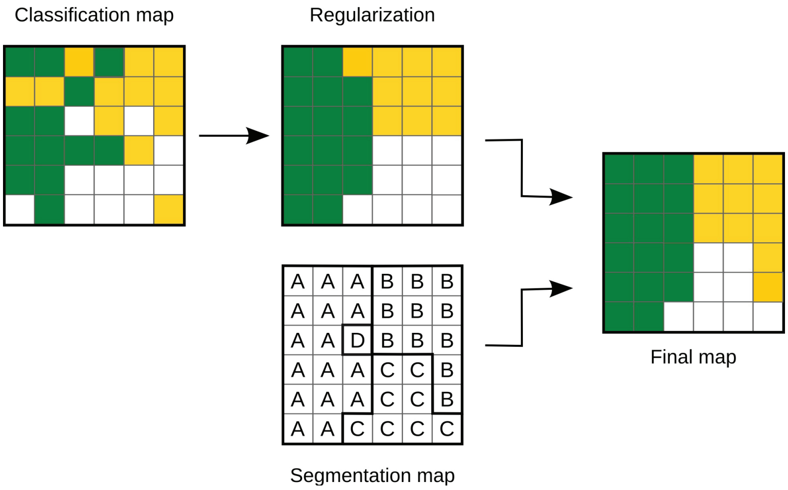
A	A	A	B	B	B
A	A	A	B	B	B
A	A	D	B	B	B
A	A	A	C	C	B
A	A	A	C	C	B
A	A	C	C	C	C

Figure 2. Spectral-spatial classification scheme, which consists of a spectral stage (*top*), a spatial stage (*bottom*), and a final result combination stage.



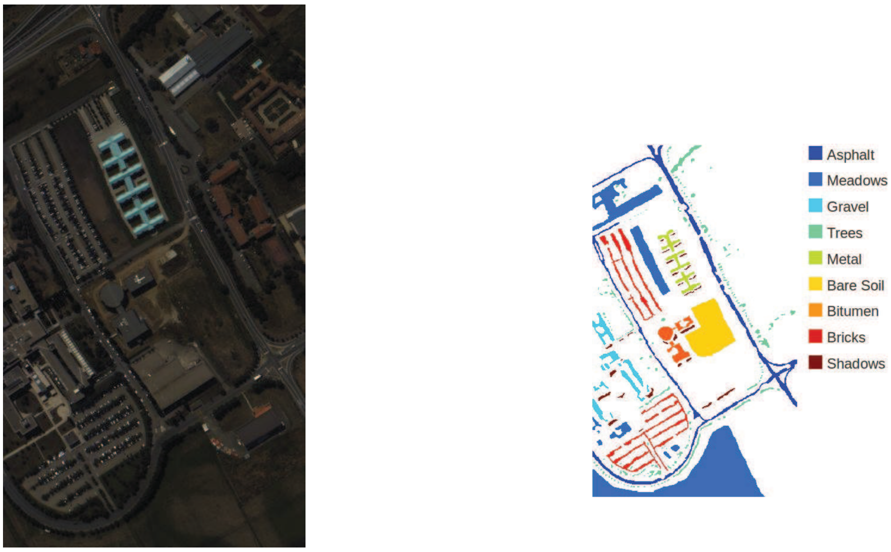
A	A	A	B	B	B
A	A	A	B	B	B
A	A	A	B	B	B
A	A	A	C	C	B
A	A	A	C	C	B
A	A	C	C	C	C

Figure 3. Example of majority vote application for spectral–spatial classification. (a) spectral classification map, (b) three segmented regions represented with different labels, and (c) results combining (a) with (b) using a majority vote process.



A	A	A	B	B	B
A	A	A	B	B	B
A	A	D	B	B	B
A	A	A	C	C	B
A	A	A	C	C	B
A	A	C	C	C	C

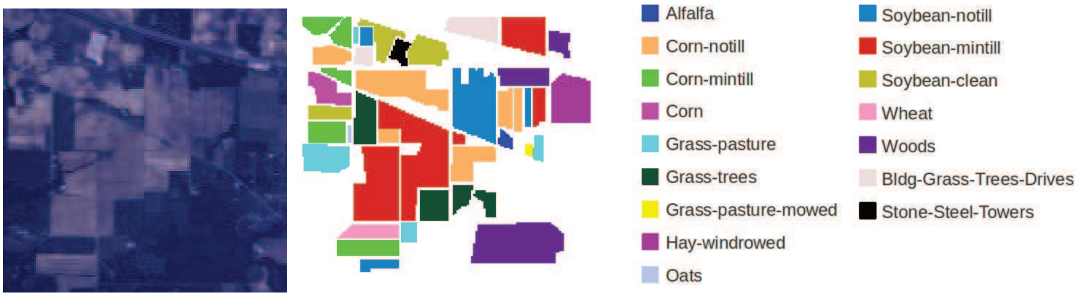
Figure 4. Application of the spatial regularization technique to the result of the spatial-spectral classification scheme based on pixel-wise classification, segmentation and majority vote.



(a) Color composite of the image.

(b) Reference data.

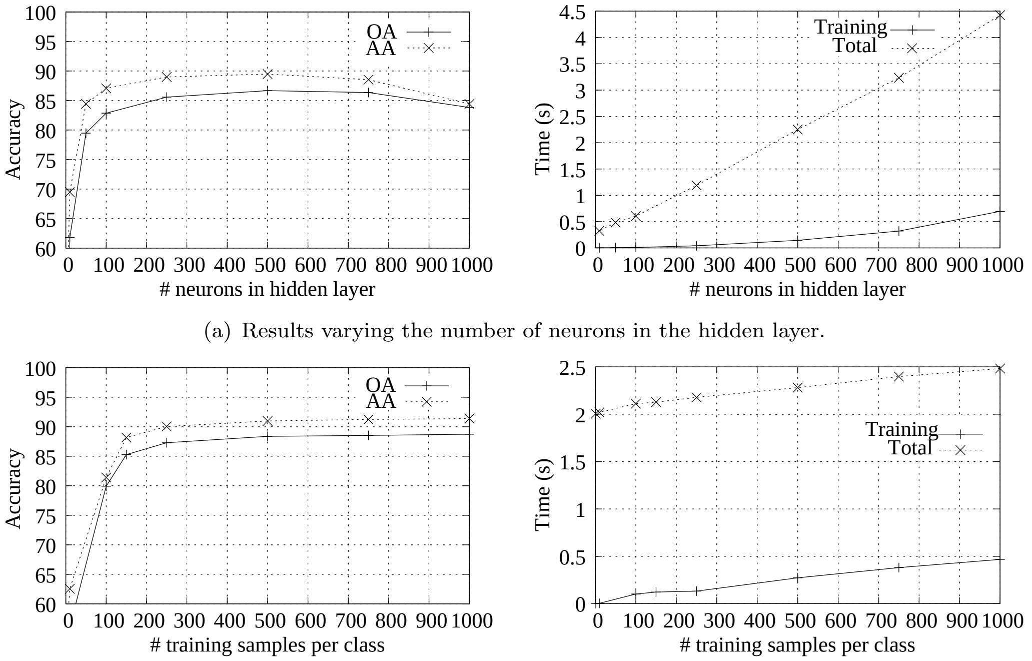
Figure 5. *University of Pavia* image.



(a) Color composite of the image.

(b) Reference data.

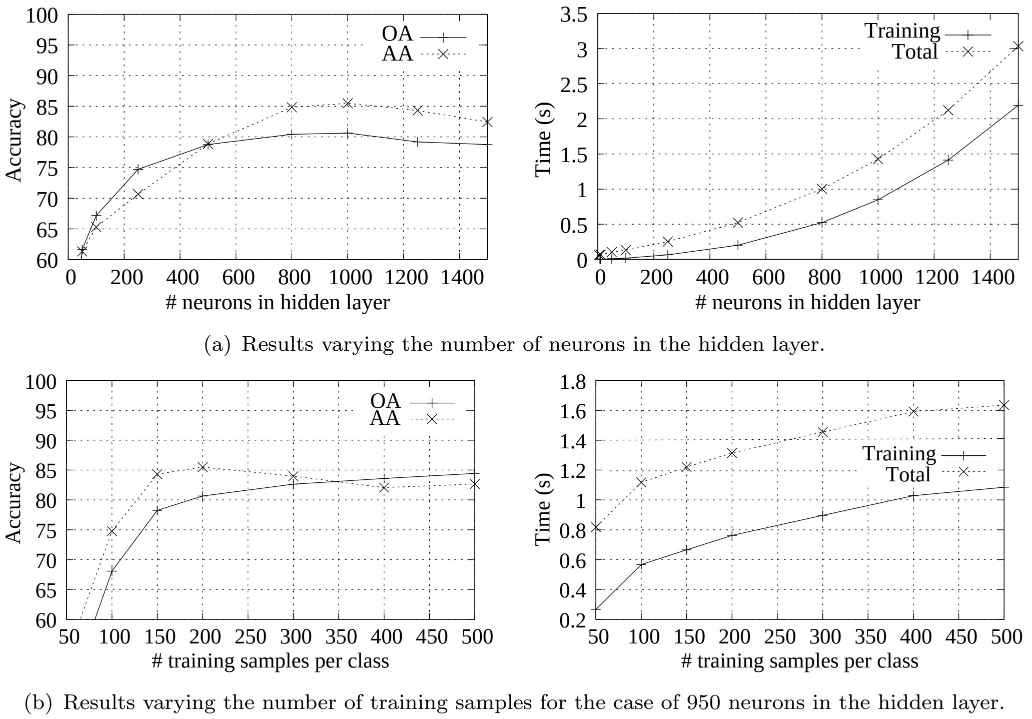
Figure 6. *Indian Pines* image.



(a) Results varying the number of neurons in the hidden layer.

(b) Results varying the number of training samples for the case of 500 neurons in the hidden layer.

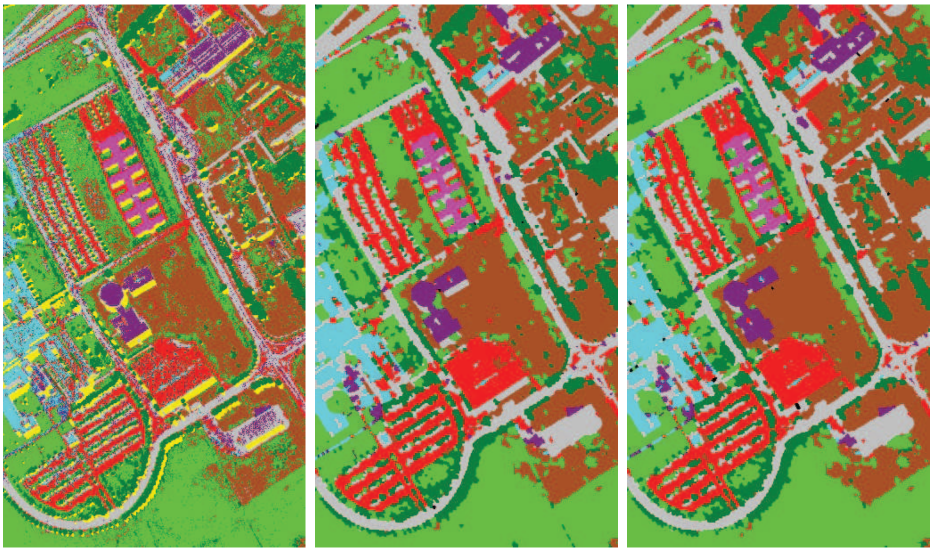
Figure 7. ELM accuracy and execution times for the *University of Pavia* image considering 200 samples per class.



(a) Results varying the number of neurons in the hidden layer.

(b) Results varying the number of training samples for the case of 950 neurons in the hidden layer.

Figure 8. ELM accuracy and execution times for the *Indian Pines* image considering 200 samples per class.

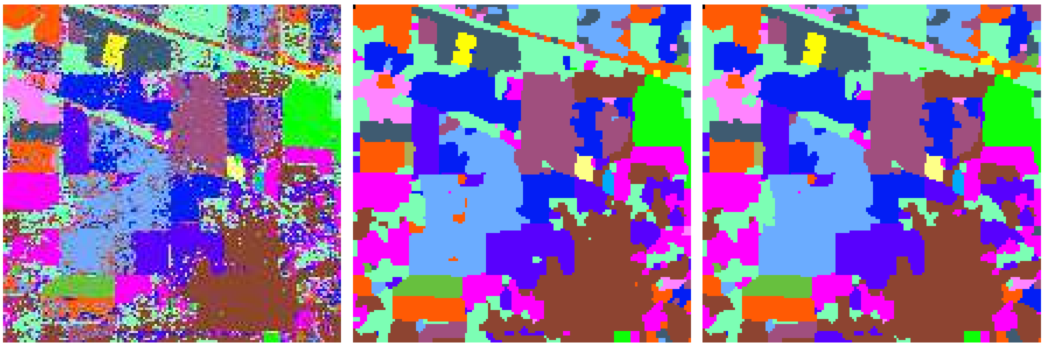


(a) ELM.

(b) Watershed-based ELM.

(c) Regularized watershed-based ELM.

Figure 9. Classification maps for the *University of Pavia* image.



(a) ELM.

(b) watershed-based ELM.

(c) Regularized watershed-based ELM.

Figure 10. Classification maps for the *Indian Pines* image.