

Article

Texture Extraction Techniques for the Classification of Vegetation Species in Hyperspectral Imagery: Bag of Words Approach Based on Superpixels

Sergio R. Blanco ^{1,*} , Dora B. Heras ¹  and Francisco Argüello ² 

¹ Centro Singular de Investigación en Tecnologías Inteligentes, Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain; dora.blanco@usc.es

² Departamento de Electrónica y Computación, Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain; francisco.arguello@usc.es

* Correspondence: sergiorey.blanco@usc.es

Received: 9 July 2020; Accepted: 12 August 2020; Published: 14 August 2020



Abstract: Texture information allows characterizing the regions of interest in a scene. It refers to the spatial organization of the fundamental microstructures in natural images. Texture extraction has been a challenging problem in the field of image processing for decades. In this paper, different techniques based on the classic Bag of Words (BoW) approach for solving the texture extraction problem in the case of hyperspectral images of the Earth surface are proposed. In all cases the texture extraction is performed inside regions of the scene called superpixels and the algorithms profit from the information available in all the bands of the image. The main contribution is the use of superpixel segmentation to obtain irregular patches from the images prior to texture extraction. Texture descriptors are extracted from each superpixel. Three schemes for texture extraction are proposed: *codebook-based*, *descriptor-based*, and *spectral-enhanced descriptor-based*. The first one is based on a codebook generator algorithm, while the other two include additional stages of keypoint detection and description. The evaluation is performed by analyzing the results of a supervised classification using Support Vector Machines (SVM), Random Forest (RF), and Extreme Learning Machines (ELM) after the texture extraction. The results show that the extraction of textures inside superpixels increases the accuracy of the obtained classification map. The proposed techniques are analyzed over different multi and hyperspectral datasets focusing on vegetation species identification. The best classification results for each image in terms of Overall Accuracy (OA) range from 81.07% to 93.77% for images taken at a river area in Galicia (Spain), and from 79.63% to 95.79% for a vast rural region in China with reasonable computation times.

Keywords: hyperspectral; texture extraction; superpixel; BoW; SVM; vegetation

1. Introduction

Monitoring vegetation species in a natural area is an important task in the context of human intervention planning. Specifically, the observation of the dynamic behavior of the vegetation provides useful insights for biodiversity conservation and forestry, among other fields. Hyperspectral imagery for remote sensing has been revealed as a powerful technique in this field, and many examples can be mentioned: from land cover changes [1] to mapping vegetation species [2,3]. Although satellite-based remote sensing is a way of obtaining consistent and comparable data, Unmanned Aerial Vehicles (UAVs) provide a more flexible platform with higher spatial resolution. The price of multi- or hyperspectral sensors used on board UAVs has decreased during the last few years. This fact makes them widely used even by small companies for an increasing number of tasks.

In the case of images for land cover analysis, supervised classification solves the problem of, given a hyperspectral image and its reference data, obtaining and distinguishing the different vegetation species or artificial elements present in the scene. In order to perform this classification, texture features can be extracted from the image [4], thus improving the classification accuracy. These features characterize the visual structures present in the scene. As it is a powerful visual cue, texture supplies information to identify objects or uniform regions of interest in the images. Texture can be differentiated from color in the sense that it refers to the spatial organization of a set of basic elements or primitives over the image called textons. These can be defined as the fundamental microstructures in natural images and the atoms of preattentive human visual perception [5].

Texture classification deals with designing algorithms or processing schemes for declaring a given texture region as belonging to one out of a set of categories (in a context where training samples have been provided). Research on texture features is mainly focused on three well-established approaches: Bag of Words (BoW)-based [6], Convolutional Neural Network (CNN)-based [7], and attribute-based [8]. The goal of BoW texture feature extraction is the statistical representation of texture images as histograms over a texton dictionary. The approach of CNNs aims to leverage large labeled datasets to learn high quality features, which can then be categorized using a simple classifier. In the case of the attribute-based approach, there are three essential issues: the identification of a universal texture attribute vocabulary, the establishment of an annotated benchmark texture dataset, and the estimation of texture attributes from images based on low level texture representations. One of the first attempts was carried out in [9], where a set of seventeen human comprehensible attributes (seven related to color and ten to structure) for color texture characterization were introduced.

Different papers focused on the classification of vegetation species using texture features in color, multi-, or hyperspectral imagery can be found in the literature. The simplest methods to characterize vegetation using textures are based on color histograms, statistical measures (mean, standard deviation, skewness, kurtosis, or entropy, among others), and clustered centers of filter bank responses. Following this approach, a classification scheme for the canopy cover mapping of spekboom in a large semiarid region in South Africa is presented in [10]. The scheme is based on a set of spectral features and vegetation indices, including several statistical measures in sliding windows of several sizes. A different scheme for natural roadside vegetation classification is presented in [11]. This scheme learns two individual sets of BoW dictionaries from color and filter-bank texture features.

Two simple methods for texture extraction, based on the analysis of patterns in the neighborhood of a pixel, are Local Binary Pattern (LBP) and Gray-Level Co-occurrence Matrix (GLCM). LBP is used in [12] for the classification of tree species using hyperspectral data and an aerial stereo camera system. Feature extraction is performed following a patch-based approach. On the other hand, a large number of publications on the classification of vegetation are based on the GLCM texture method. Vegetation mapping in complex urban landscapes using a hybrid method combining Random Forest and GLCM texture analysis at nine different window sizes is presented in [13]. The classification is done using ultra-high resolution imagery acquired at low altitudes. A proposal of crop classification method for hyperspectral images combining spectral indices and GLCM texture information is presented in [14]. An object-based GLCM texture extraction method for the classification of man-planted forests on mountainous areas using satellite data is the contribution presented by [15]. As a preprocessing step, the texture feature of segmented image objects are enhanced using a 2D Gabor filter. Using very high resolution images acquired by UAVs, a study to identify the most relevant image parameters for tree species discrimination is conducted in [16]. Specifically, classification of savannah tree species is carried out by using chromatic coordinates, spectral indices, the canopy height model, and GLCM texture measures in different window sizes. Similarly, the potential of combining spectral measures and GLCM texture information for crop classification in time-series UAV images is investigated in [17].

More elaborate texture methods based on local invariant descriptors such as SURF and SIFT can also be used for characterizing vegetation species. For example, a methodology for vegetation segmentation in cornfield images obtained by UAVs is presented in [18]. Specifically, it focuses on

finding an appropriate set of different color vegetation indices and local descriptors for vegetation characterization. The classification of weeds growing among crops using a BoW model based on SIFT or SURF features is presented in [19]. Finally, a study on the application of SIFT to cropland mapping in the Brazilian Amazon based on vegetation index time series is conducted in [20].

A classification scheme based on textures can be combined with other types of features obtained by UAVs to improve the classification results [10,16]. Among them are spectral features, vegetation indices, and morphological measures. For the detection of the extent of trees and shrubs, the canopy height model (CHM) is the one most commonly used. LiDAR sensors have been widely used in order to collect high resolution information on forest structure. Surface reconstruction by image matching can also be used to estimate CHM. It is achieved by exploiting the redundancy of multiple overlapping aerial images [21,22]. CHM is not used in this paper since the available datasets in many cases do not provide multiple images for the same area.

For the classification of a image using textures it is necessary to delimit regions over which the texture features are computed. Most of the vegetation classification methods proposed in the literature use regular patches [6,8,10,12,13,15,16,19]. In other cases, segmentation or object detection algorithms for dividing the image into regions are used [14,17,18]. A technique commonly used for the extraction of uniform regions in images is the segmentation based on superpixels [23,24]. A superpixel is a set of neighboring pixels (segment) which are similar in terms of low-level properties (such as spatial proximity, color, intensity, or other criteria). They differ from other segmentation methods in that the size and regularity of the superpixels are similar throughout the image. Superpixels provide a convenient and compact representation of images that allow to reduce the computational cost of the processing algorithms [25]. In the schemes presented in this paper, a texture feature vector is computed for each superpixel.

In this paper, different techniques for vegetation classification in multi and hyperspectral images based on texture extraction and BoW are proposed. The techniques are grouped into three categories: codebook-based, descriptor-based, and spectral enhanced descriptor-based schemes. The main contribution of this work is that in all the presented schemes the texture algorithms are computed inside superpixels, in contrast to most of the methods previously published in the literature, in which the vegetation textures are extracted from patches or objects. Moreover, some of the descriptor-based methods have not been applied before to multi- and hyperspectral images. Finally, a detailed comparison of the different techniques is carried out in terms of classification accuracy for several land cover remote sensing datasets.

The rest of the paper is organized into four sections. Section 2 presents a description of the proposed schemes involving superpixel computation and texture extraction. The experimental results for the evaluation in terms of classification performance and computational cost are presented in Section 3. The discussion is carried out in Section 4. Finally, Section 5 summarizes the main conclusions.

2. Methods

Three different schemes for texture extraction in order to obtain superpixel descriptors (i.e., a vector which describes the texture or visual properties of each superpixel in the scene) were proposed. The main novelty is that the texture features were computed inside these irregular patches of the images called superpixels and that the schemes were adapted to profit from the information available in all the bands of the hyperspectral images. Different texture extraction techniques can be derived from the proposed schemes depending on the algorithms selected for their stages as it will be explained throughout the paper.

The different stages of the proposed schemes, shown in Figure 1, are the following.

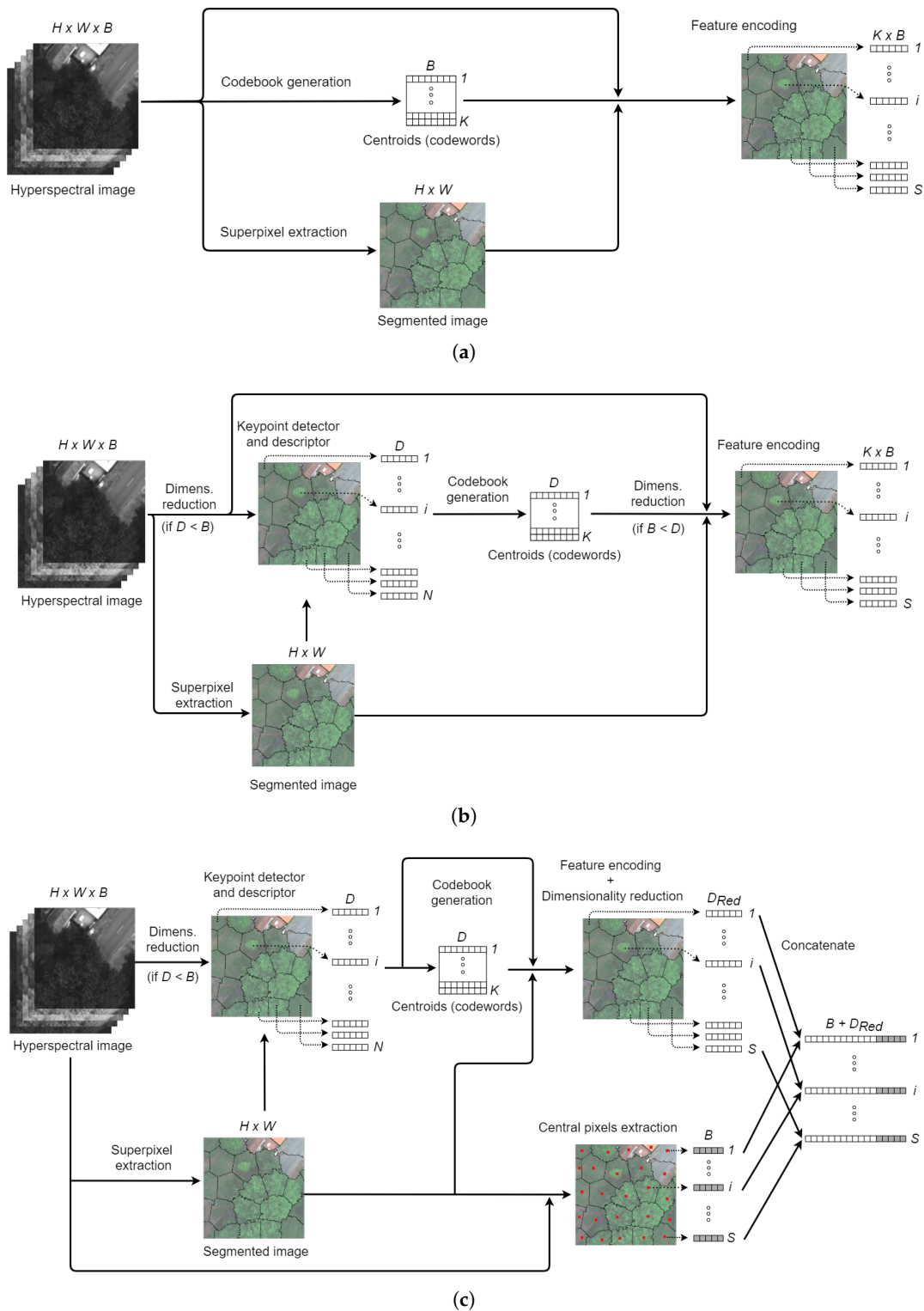


Figure 1. Texture extraction schemes for hyperspectral imagery using the Bag of Words (BoW) superpixel-based approach proposed: (a) *codebook-based*, (b) *descriptor-based*, and (c) *spectral-enhanced descriptor-based*.

Superpixel extraction. This is a particular type of segmentation stage. As previously mentioned, a superpixel is a set of pixels which are similar in terms of spatial proximity, color, intensity, or other properties. There is a relationship between these superpixels and the objects present in the scene.

In this stage, a set of S superpixels was extracted from the image. The computed superpixels are irregular, which is a difference between this process and other similar ones (e.g., creation of a grid of square patches). The differences in size and shape among superpixels are due to the adaption of each superpixel to the objects appearing in the scene.

In our case, the algorithm used for superpixel extraction was Simple Linear Iterative Clustering (SLIC) [26], although other options such as those based on watershed [27] or Efficient Topology Preserving Segmentation (ETPS) [28] would obtain similar results. SLIC clusterises pixels into superpixels, taking into account their relative position and spectral values, so both spatial and spectral information are considered. This algorithm is an adaptation of k-means for superpixel generation that begins defining clusters. Each pixel is associated to the nearest cluster center. Then, the cluster centers are adjusted to be the mean of all pixels belonging to the cluster. The assignment and update steps are iteratively repeated until the convergence criteria (a maximum number of iterations or an error value) is met. Finally, a postprocessing step enforces connectivity by reassigning disjoint pixels to nearby superpixels. It offers good results for segmenting hyperspectral images [24].

After the segmentation was computed, most of the subsequent stages in the proposed schemes were calculated at the superpixel level instead of at the pixel level. In particular, only one label from the reference data was considered for each superpixel, which is the one associated to the central pixel of this superpixel. Moreover, texture extraction was performed inside each superpixel.

Keypoint detection and description. A set of points of interest or keypoints were extracted from the image for each band and each superpixel. This stage was used in two of the schemes as shown in Figure 1b,c. These keypoints may be extracted in the positions given by a keypoint detector [29] or densely at each pixel position over a fixed grid. In addition, they should be distinctive and robust to image transformations.

Given a keypoint and its neighboring pixels, a set of features were computed, obtaining a local texture descriptor. In our case, the algorithms used to create texture descriptors were Scale-Invariant Feature Transform (SIFT) [30], Histogram of Oriented Gradients (HOG) [31], Dense SIFT (DSIFT) [32] and Local Intensity Order Pattern (LIOP) [33]. SIFT and HOG algorithms include both a descriptor and a keypoint detector. In LIOP a fixed grid was created, because this technique does not include a keypoint detector algorithm, only a descriptor one. DSIFT also uses a similar dense approach, but the detector is built-into the technique.

This process was applied to each spectral band, and then descriptors from all the bands were grouped for each superpixel taking into account their location in the XY plane. At the end of the process, a variable number of keypoints (with their corresponding descriptors) was assigned to each superpixel. The dimension of each descriptor was denoted as D and their number as N , as shown in Figure 1b,c.

Codebook generation. The objective of this stage was to create a texton dictionary with K codewords based on all the bands of the input image. This codebook can be learned [34,35] or predefined [36]. In this paper the codebook was learned using k-means [6] or Gaussian Mixture Modeling (GMM) [37] algorithms. The size and nature of the codebook greatly affects the performance of the classification. The key was to generate a compact but yet discriminatory one.

Feature encoding. Given the codebook and the computed local features (i.e., vector descriptors corresponding to each superpixel), this encoding process mapped the latter to one or a variable number of codewords. The result was a feature coding vector per superpixel. Thereby, the aim of this process was mapping each superpixel description (object representation) to one or more codewords. This is a core component of the scheme, influencing texture classification in terms of both accuracy and speed. The feature encoding algorithms employed in this paper were Vector of Locally Aggregated Descriptors (VLAD) [38] and Fisher Vectors (FV) [37]. Once the desired vector representation for each superpixel was obtained, this representation was used as representative of the superpixel for the later stages.

Dimensionality Reduction. This is a stage where a set of vectors obtained in the previous stage (e.g., descriptors or coding vectors) were reduced. This reduction was performed if the number of

bands B of the image was higher than the dimension D of the descriptors or coding vectors, in which case the image was reduced to D bands (see Figure 1 for details). This step was also used in Figure 1c in order to transform the descriptors from dimension D to D_{red} . The techniques used for the reduction could be any of the traditional functions of aggregation such as the sum or the mean, or any other algorithm related to feature extraction such as, for example, Principal Component Analysis (PCA). PCA constitutes a quite popular method for feature extraction [39,40]. PCA estimates projections of the original data so that most of the variance is concentrated in a few components.

Feature classification. This last stage was not part of the texture extraction schemes and performed a classification of the images based on the features produced by the previous texture extraction technique. Texture features were inputs to a superpixel-level classification, i.e., the training and testing sets consisted of superpixels described by their texture features. Once the classification finished the same class was assigned to all the pixels in each superpixel. SVMs were selected as classifiers. They are usually presented as standard non-contextual classifiers for remote sensing classification [41], and can handle scenarios with a low number of training samples [42]. Results obtained by other two standard classifiers in remote sensing, RF and ELMs [43] were also obtained.

Figure 1 illustrates the three different proposed texture extraction schemes showing the different stages according to the previous description. All of them have the image as input and one feature vector per superpixel as output.

2.1. Codebook-Based Scheme

The first scheme (named *codebook-based* from now on and shown in Figure 1a) began by performing two tasks in parallel: segmenting the image and creating a codebook. In terms of codebook generation, a texton dictionary with K codewords was created. The final set of codewords obtained was of size $K \times B$ (being B the number of bands of the input image).

Given the generated codebook and the computed superpixel segmentation, the next stage was the feature encoding. A vector representation of each superpixel, a texture vector, was obtained by mapping each superpixel to one or more codewords. In the case of k-means, this assignment can be done using the centroid with the shortest Euclidean distance to the superpixel.

2.2. Descriptor-Based Scheme

The second scheme in Figure 1 (named *descriptor-based*) represents an increase in complexity with respect to the first one. In parallel to the superpixel generation, a conditional dimensionality reduction step was performed. Only if the number of bands B of the image is larger than the resulting dimension D of the descriptors that were created, the image was reduced to D spectral bands. The next stage was keypoint detection and description. An algorithm such as SIFT or HOG was applied over each one of the bands of the image. The algorithm carried out two sequential stages. First N keypoints (points of interest) were detected and then, using their neighborhood pixels, a local texture descriptor algorithm was applied to obtain a set or pool of texture features of dimension D . Each one of these vectors was assigned to a superpixel according to the location of the keypoint. The number of keypoints per superpixel is variable, being possible to obtain even zero keypoints for a particular one. The implications of this variable number of descriptors per superpixel are not important in this scheme because they are used only when computing the codebook where they are stacked together. Further implications will be pointed out when describing the feature encoding stage in the next scheme.

After keypoint detection and description, the codebook generator was applied to the stacked descriptor vectors from all bands, obtaining K codewords of size D each. Another conditional dimensionality reduction stage was then performed in case the number of bands of the input image is lower than the descriptor dimension. After the dimensionality reduction (to the dimension of the input image in the first step or to the dimension of the descriptors later), the dimensions of both the codewords and the image pixel-vectors were equal (which means that now B is equal to D).

The output obtained (a texture vector describing each superpixel) is equivalent to the one obtained by the *codebook-based* scheme in Figure 1a.

2.3. Spectral-Enhanced Descriptor-Based Scheme

The last scheme (named *spectral-enhanced descriptor-based*), shown in Figure 1c, differs slightly from the previous one. The main novelty is that the feature coding stage operates at superpixel level and that a concatenation of spectral information to the texture descriptors at the end of the texture extraction process is performed.

As it can be observed in the figure, the input hyperspectral image for the encoding stage in the previous schemes was replaced here by the image descriptors obtained for the different superpixels. If a superpixel has no associated texture descriptor, the resulting vector is zero. However, if it has one or more descriptors, all of them are compared to each codeword in order to obtain the resulting vector.

As the feature encoding process took the texture descriptors as input (unlike the previous schemes), some kind of spectral information needed to be added. With this objective, a new stage called central pixel extraction was executed. It searched for the central pixel of each superpixel in the spatial coordinates of the image and extracted the corresponding spectral values (central pixel-vector). The resulting data structure once the central pixels were extracted consisted of S vectors (as many as superpixels in the segmentation) each one of dimension B (the number of bands). Finally, a concatenation was performed: a new vector per superpixel was created by stacking the texture vector from the feature encoding with the pixel-vector from the central pixel extraction stage. The output is equivalent to the one obtained by the previous schemes but differing in the dimension of the superpixel feature vector: $B + D_{Red}$.

2.4. Dataset Description

Three datasets were used to evaluate the schemes proposed: a set of three hyperspectral images (from now on *standard dataset*), a set of multispectral scenes from river basins for which only vegetation classes are taken into account (*Galicia dataset*), and a large set of multispectral images from a vast region in China called *Gaofen dataset*. The *standard dataset* was used for comparison purposes as the scenes in it are usually present in land cover classification papers. Both the *Galicia dataset* and the *Gaofen dataset* were used because they contain large images with a wide range of vegetation species (both forests and crops). For the *Galicia dataset* only vegetation classes will be classified although the images also contain other materials, while for the remaining two datasets all classes including a non-vegetation one are classified.

The previously mentioned *standard dataset* corresponds to two images commonly used in the remote sensing literature: Pavia University (Pavia) and Salinas Valley (Salinas) [44]. Pavia was obtained by the ROSIS-03 (Reflective Optics System Imaging Spectrometer) sensor over the city of Pavia, Italy, with a spatial resolution of 2.6 m/pixel and covering the spectral range from 430 to 860 nm. Its dimensions are 610×340 pixels and 103 bands and its corresponding latitude and longitude are $45^{\circ}11'23.66''N$ and $09^{\circ}08'57.06''E$, respectively. Salinas was obtained by the AVIRIS (Airborne Visible Infrared Imaging Spectrometer) sensor with a spectral range from 400 to 2500 nm. The main properties of this image are a resolution of 3.7 m/pixel, dimensions of 512×217 and 224 spectral bands. Moreover, its corresponding latitude and longitude are $36^{\circ}39'33.8''N$ and $121^{\circ}39'58.7''W$. Figure 2 shows the false color composite images and the reference data corresponding to this dataset, while Table 1 displays the classes available in the reference data and the number of disjoint superpixels used for classification in training (15%) and testing (85%). Fifteen percent of superpixels corresponds to between 14% and 15% of the pixels in the image.

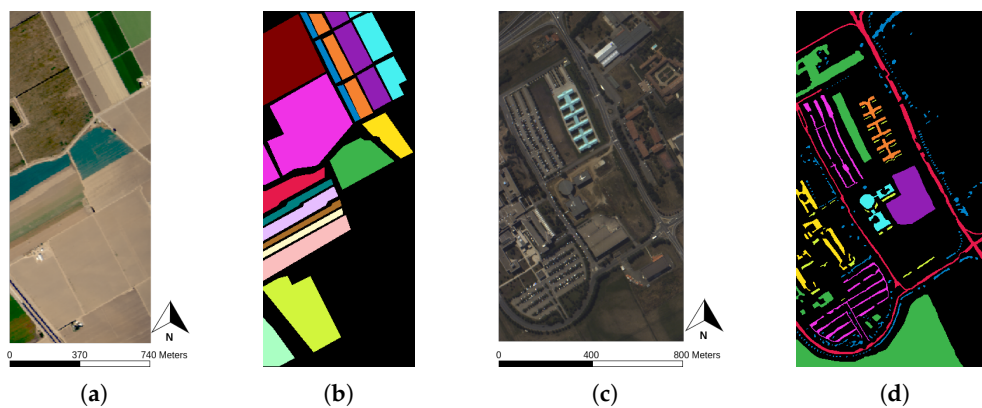


Figure 2. Standard dataset hyperspectral images and their corresponding reference maps for classification: (a,b) image and reference map for Salinas, respectively, and (c,d) image and reference map for Pavia, respectively. Each color in a reference map corresponds to a particular class for the image. No reference information is available for the regions marked in black.

Table 1. Standard dataset. Classes available in the reference data and number of superpixels used for training (15%) and testing (85%).

Salinas				Pavia			
# and color	Classes	Superpixels		Classes	Superpixels		
		Train (15%)	Test (85%)		Train (15%)	Test (85%)	
1.	Broccoli gr. weeds 1	6	38	Asphalt	14	82	
2.	Broccoli gr. weeds 2	7	42	Meadows	59	336	
3.	Fallow	5	32	Gravel	8	50	
4.	Fallow rough plow	6	36	Trees	1	7	
5.	Fallow smooth	10	62	Metal	5	29	
6.	Stubble	9	57	Bare soil	15	85	
7.	Celery	13	77	Bitumen	7	41	
8.	Grapes untrained	30	174	Bricks	10	57	
9.	Soil vineyard dev.	12	68	Shadows	1	9	
10.	Corn gr. weeds	9	56				
11.	Lettuce rom 4 weeks	4	28				
12.	Lettuce rom 5 weeks	6	39				
13.	Lettuce rom 6 weeks	3	20				
14.	Lettuce rom 7 weeks	3	23				
15.	Vineyard untrained	22	126				
16.	Vineyard ver. trellis	4	26				

The *Galicia dataset* is made up of four multispectral images, and the objective of their creation was to monitor the interaction of masses of native vegetation with artificial structures and river beds. Four locations in the Galician provinces of A Coruña and Pontevedra were selected in an area comprised between Eiras Dam and River Mestas, with a distance of approximately 145.6 kilometers end-to-end. The datasets were captured by a MicaSense RedEdge multispectral camera [45] mounted on a custom UAV. Its 5 discrete sensors provide spectral channels at wavelengths of 475 nm (Blue), 560 nm (Green), 668 nm (Red), 717 nm (Edge), and 840 nm (Near infrared). The spatial resolution is 8.2 cm/pixel at a height of 120 m.

The four images in the dataset are the following; River Oitavén (Oitavén from now on) of size 6689×6722 pixels and is located in $42^{\circ}22'27.8''N$ $8^{\circ}25'30.3''W$, River Mestas (Mestas from now on) of size 4915×9040 pixels and is located in $43^{\circ}38'38.5''N$ $7^{\circ}59'04.3''W$, River Ferreiras (Ferreiras from now on) which is of size 9335×9219 pixels and is located in $43^{\circ}32'58.8''N$ $7^{\circ}57'33.2''W$, and Eiras dam (Eiras from now on) whose dimensions are $5176 \times 18,224$ pixels and is located in $42^{\circ}22'26.0''N$ $8^{\circ}25'41.3''W$.

Figure 3 shows the false color composite images and their reference data (constructed in a long-term process involving forestry experts and the authors of the paper) corresponding to each one of the scenes, while Table 2 shows the classes available in the reference data for classification and the number of superpixels used for training (15%) and testing (85%). Moreover, as the objective is the identification of plant species, only vegetation classes are considered.

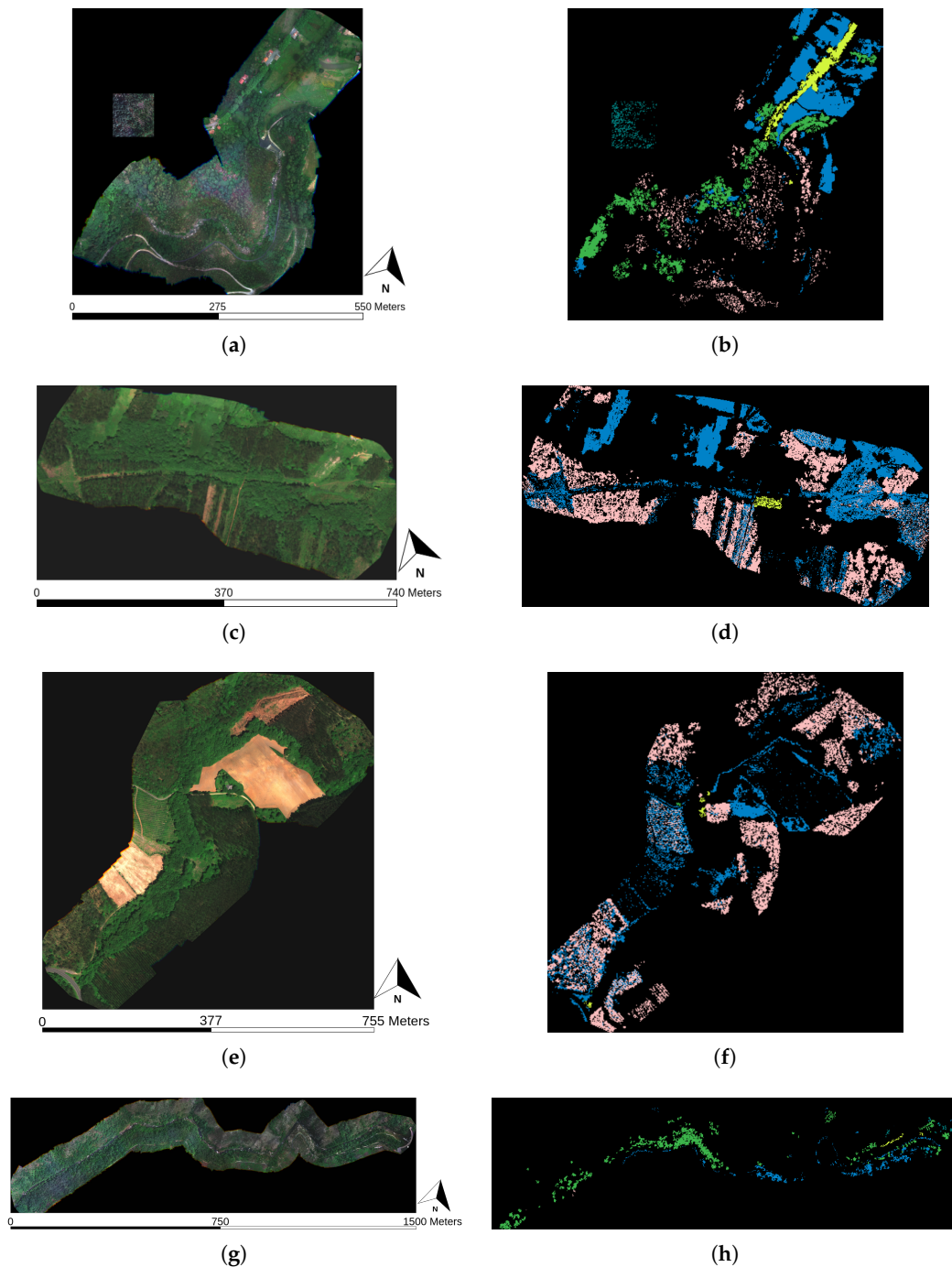


Figure 3. Galicia dataset multispectral images and their corresponding reference maps for classification: (a,b) image and reference map for Oitavén, respectively; (c,d) image and reference map for Mestas, respectively; (e,f) image and reference map for Ferreiras, respectively; and (g,h) image and reference map for Eiras, respectively. Each color in a reference map corresponds to a different class for the image. Only vegetations classes are represented and considered for the experiments. No reference information is available for the regions marked in black.

Table 2. Galicia dataset. Classes available in the reference data and number of superpixels in the disjoint training (15%) and testing (85%) sets. Only vegetation classes are considered in the experiments, the rest of the classes are marked with “-”. NA indicates that the image does not contain samples for a specific vegetation class [46].

# and color	Classes	Oitavén		Mestas		Ferreiras		Eiras	
		Superpixels		Superpixels		Superpixels		Superpixels	
		Train (15%)	Test (85%)	Train (15%)	Test (85%)	Train (15%)	Test (85%)	Train (15%)	Test (85%)
1.	Water	-	-	-	-	-	-	-	-
2.	Oak	268	1523	NA	NA	1	8	395	2240
3.	Tiles	-	-	-	-	-	-	-	-
4.	Meadows	342	1943	661	3748	557	3159	114	646
5.	Asphalt	-	-	-	-	-	-	-	-
6.	Bare Soil	-	-	-	-	-	-	-	-
7.	Rock	-	-	-	-	-	-	-	-
8.	Concrete	-	-	-	-	-	-	-	-
9.	Authohtonous vegetation	90	514	13	74	8	46	8	51
10.	Eucalyptus	208	1182	683	3873	957	5427	1	10
11.	Pines	14	81	NA	NA	1	4	16	91

Finally, the *Gaofen dataset* was used [47]. It is a large-scale land use grouping of images containing 160 annotated Gaofen-2 (GF-2) satellite scenes. GF-2 is the second satellite of the High-definition Earth Observation System promoted by the China National Space Administration. The spectral range goes from 0.45 to 0.89 μm (blue to near-infrared) and the spatial dimension of the images is 6908×7300 pixels. Some of the dataset advantages are its large coverage, wide distribution, and high spatial resolution. It is remarkable that this dataset has high intra-class and low inter-class differences. The images cover an area of more than 50,000 km^2 in China. Specifically, it can be divided into two sets. A large-scale classification set made up of 150 high-resolution images acquired from more than 60 different cities in China and where 5 major categories are annotated. From now on, this dataset will be named *GID5* (Gaofen Image Dataset 5 classes). A fine land-cover classification set composed of 30,000 multi-scale image patches coupled with 10 pixel-level annotated images and made up of 15 sub-categories. From now on *GID15* (Gaofen Image Dataset 15 classes). Figure 4 shows the false color composite images and their reference data of two images from *GID5* and two others from *GID15*. Table 3 shows the classes available in the reference data and the number of superpixels used for training (15%) and testing (85%).

Table 3. Gaofen dataset. Classes available in the reference data and number of superpixels used for training (15%) and testing (85%). NA values indicate the non-existence of samples for a specific vegetation class and image, while “-” implies the non-existence of samples for the specific non-vegetation class [47].

# and Color	Classes	GF1		GF2		# and color	Classes	GF3		GF4	
		Superpixels		Superpixels				Superpixels		Superpixels	
		Train (15%)	Test (85%)	Train (15%)	Test (85%)			Train (15%)	Test (85%)	Train (15%)	Test (85%)
1.	Built-up	123	698	34	194	1.	Industrial land	533	3024	-	-
2.	Farmland	2323	13,169	796	4511	2.	urban residential	952	5401	6	34
3.	Forest	348	1978	187	1066	3.	Rural residential	291	1651	76	433
4.	Meadow	668	3791	114	648	4.	Traffic land	496	2812	38	221
5.	Water	-	-	-	-	5.	Paddy field	NA	NA	NA	NA
						6.	Irrigated land	2258	12,801	2159	12,235
						7.	Dry cropland	118	675	-	-
						8.	Garden plot	13	77	NA	NA
						9.	Arbor woodland	NA	NA	707	4010
						10.	Shrub land	14	80	NA	NA
						12.	Natural grassland	NA	NA	803	4551
						13.	Artificial grassland	NA	NA	NA	NA
						13.	River	-	-	-	-
						14.	Lake	-	-	-	-
						15.	Pond	12	71	4	26

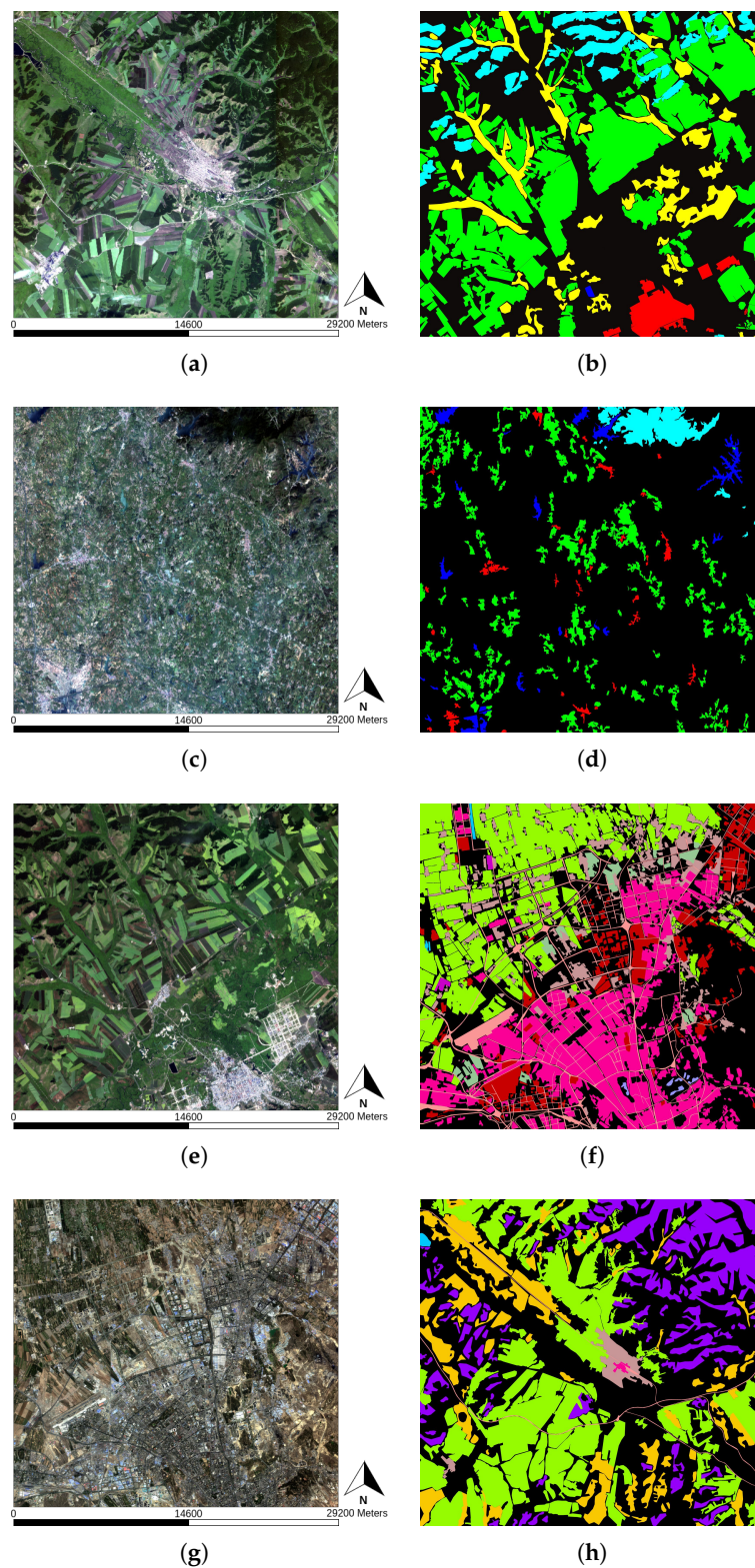


Figure 4. *Gaofen* dataset multispectral images and their corresponding reference maps for classification: (a,b) image and reference map for GF1 (original name GF2_PMS2__L1A0001708261-MSS2), respectively; (c,d) image and reference map for GF2 (original name GF2_PMS1__L1A0001798942-MSS1), respectively; (e,f) image and reference map for GF3 (original name GF2_PMS2__L1A0001821754-MSS2), respectively; and (g,h) image and reference map for GF4 (original name GF2_PMS1__L1A0001680858-MSS1), respectively. The different colors in the reference data indicate different classes for the images. No reference information is available for the regions marked in black.

2.5. Accuracy Assessment and Set-Up Description

The classification accuracy obtained by classifying the features provided by the proposed schemes was reported in terms of the usual measures in remote sensing. The first measure Overall Accuracy (OA) is the most widely used [48]. It provides the percentage of correctly classified pixels, and it is presented for every experiment. Besides, Quantity Disagreement (QD) and Allocation Disagreement (AD), which measure the disagreement between classification map and reference data in terms of proportion and spatial allocation of the classes, respectively, were also provided [49].

The input data for the experiments were standardized (a mean of 0 and a standard deviation of 1). In addition, with the aim of evaluating the computational cost associated to each method, execution time measures were performed. All the results presented in the paper are the mean value of 3 independent runs for each scenario, each one being obtained under identical experimental conditions.

Regarding the configuration parameters, as mentioned above, SLIC was used as the superpixel extraction algorithm. There are two parameters for SLIC: superpixel size and regularity of the superpixels. The superpixel size is the desired average size of each superpixel (in terms of area in pixel units). Ten and 1100 were the values selected for the *standard dataset* (as the images in the dataset are small) and the other two datasets, respectively. For superpixel regularity, the larger the value, the more regular the superpixels obtained. A value of 20 was selected for all datasets. These values were experimentally decided and depend mainly on the resolution of the images and on the size of the structures present on them, being in general bigger superpixels more adequate for higher resolution images.

The classification was performed by using SVM, Random Forest, and ELM. More precisely, for SVM the LIBSVM implementation version 3.24 in C/C++ was chosen [50] selecting a linear kernel. The parameter C was determined for each SVM and values of 0.02 for parameter C (the same value for all datasets) gave the best results. In the case of Random Forest, the OpenCV implementation was used [51]. The only parameter set for this algorithm is the number of trees. After a search for the considered datasets, a value of 200 trees was chosen. Last, the ELM implementation selected was [52]. The number of neurons in the hidden layer was 250 for the small images and 500 for the bigger ones. These are standard values for the datasets considered [43].

The classification is performed at superpixel level, i.e., all the pixels in a superpixel are assigned the same label. As far as the training and testing features are concerned, two disjoint sets were set up for each image. Specifically, after segmenting the images using SLIC, 15% of the superpixels from each class were randomly taken for training and the remaining 85% for testing in the general scenario. For training, only one label per superpixel, the label of the spatially central pixel, is considered. The number of 15% of superpixels is equivalent to choose ~13% of the pixels of each class. This percentage of training samples is reasonable as it is shown in [43]. Results for 10% and 20% of samples for training were also tested for all the images in the *Galicia dataset* as this is the most representative dataset in the study, containing large images and including only vegetation classes.

Regarding other set-up details, the VLFeat library version 0.9.21 was used [32]. Specifically, the implementations of the texture extraction related algorithms SIFT, DSIFT, LIOP, HOG, GMM, VLAD, and FV available in the library were used. All the experiments were carried out using C/C++ compiled with gcc 7.5.0. Additionally, a first generation Intel Core i7-8700K CPU at 3.70 GHz and a 64-bit Ubuntu 18.04 were used for all the experiments including the execution time evaluation.

The classification accuracy results were obtained for each one of the three datasets described in Section 2.4 and considering different texture extraction techniques. The selection of specific algorithms for each stage of the proposed texture extraction schemes resulted in 16 different techniques for the experiments. The mapping between these techniques, the schemes they correspond to and the specific algorithms used are presented in Table 4. The row *Without Texture Features* shows the configuration without performing feature extraction at all and using the central pixel of each superpixel as input for the classification.

Table 4. Texture extraction techniques considered for the schemes proposed in Figure 1 detailing the configuration of the different stages.

Scheme	Technique	Superpixel Extraction	Keypoint Detection and Description	Codebook Generation	Feature Encoding	Dimensionality Reduction	Feature Classification
-	<i>Without Texture Features</i>	SLIC	-	-	-	-	SVM
Codebook-based scheme	<i>k-means + VLAD</i>	SLIC	-	k-means	VLAD	-	SVM
	<i>k-means + BOW</i>	SLIC	-	k-means	BoW	-	SVM
	<i>GMM + FV</i>	SLIC	-	GMM	FV	-	SVM
Descriptor-based scheme	<i>SIFT + k-means + VLAD</i>	SLIC	SIFT	k-means	VLAD	PCA	SVM
	<i>SIFT + GMM + FV</i>	SLIC	SIFT	GMM	FV	PCA	SVM
	<i>DSIFT + k-means + VLAD</i>	SLIC	DSIFT	k-means	VLAD	PCA	SVM
	<i>DSIFT + GMM + FV</i>	SLIC	DSIFT	GMM	FV	PCA	SVM
	<i>LIOP + k-means + VLAD</i>	SLIC	LIOP	k-means	VLAD	PCA	SVM
	<i>LIOP + GMM + FV</i>	SLIC	LIOP	GMM	FV	PCA	SVM
	<i>HOG + k-means + VLAD</i>	SLIC	HOG	k-means	VLAD	PCA	SVM
Spectral-enhanced descriptor-based scheme	<i>SIFT + k-means + VLAD + Spec</i>	SLIC	SIFT	k-means	VLAD	PCA	SVM
	<i>SIFT + GMM + FV + Spec</i>	SLIC	SIFT	GMM	FV	PCA	SVM
	<i>DSIFT + k-means + VLAD + Spec</i>	SLIC	DSIFT	k-means	VLAD	PCA	SVM
	<i>DSIFT + GMM + FV + Spec</i>	SLIC	DSIFT	GMM	FV	PCA	SVM

3. Experimental Results

The aim of this section is to analyze how the use of the different schemes influences the results of the classification. Classification results and computational efficiency in terms of Overall Accuracy (OA), Quantity Disagreement (QD), Allocation Disagreement (AD), and execution time are presented. The results are obtained for the three hyperspectral and multispectral datasets described above and for the texture extraction techniques that were described in Table 4. Three classifiers are considered: SVM, RF, and ELM.

The classification accuracy results for the *standard dataset* are shown in Tables 5 for the SVM classifier, Table 6 when RF is used for classification, and Table 7 for the experiments with the ELM classifier. The listed techniques are grouped according to the texture scheme followed: *codebook-based*, *descriptor-based*, and *spectral-enhanced descriptor-based*. For each technique, 15 % of the superpixels were randomly selected for training and the remaining 85 % for testing. The best results for each image in terms of OA are highlighted with a gray background.

The results in Tables 5–7 show the same trends. As the images in the *standard dataset* are very small, they do not benefit from a superpixel level classification independently of the classifier considered, so for each classifier all techniques offer similar results and, in general, the OA values are low. The techniques from the *descriptor-based* scheme and *spectral-enhanced descriptor-based* scheme present a slightly higher OA, being the SIFT-based ones, specifically *SIFT + GMM + FV* and *SIFT + k-means + VLAD*, the best methods for the Salinas and Pavia images respectively. The QD and AD values are low for all the experiments and lower for the higher OA values, as expected. As the *standard dataset* does not focus on vegetation, experiments with the *Galicia dataset* considering only vegetation classes were performed.

Table 5. Classification results in terms of OA (%), QD (%), and AD (%) obtained by the techniques detailed in Table 4 for the images of the *standard dataset* and using a SVM classifier. Fifteen percent of the superpixels are used for training. The best OA results are in a gray background.

Scheme	Technique	Salinas			Pavia		
		OA	QD	AD	OA	QD	AD
-	Without Texture Features	76.48 ± 0.46	15.19 ± 2.72	3.12 ± 1.36	65.77 ± 0.97	21.48 ± 9.67	6.75 ± 1.94
Codebook-based scheme	<i>k</i> -means + VLAD	75.07 ± 1.55	15.88 ± 2.95	3.74 ± 1.46	65.92 ± 2.97	20.28 ± 8.32	6.55 ± 1.01
	<i>k</i> -means + BOW	77.93 ± 1.08	14.97 ± 2.56	3.04 ± 1.27	60.93 ± 1.32	27.98 ± 10.17	7.95 ± 2.34
	GMM + FV	74.86 ± 1.66	16.64 ± 3.52	4.07 ± 1.89	66.84 ± 3.52	21.28 ± 9.37	6.05 ± 1.52
Descriptor-based scheme	SIFT + <i>k</i> -means + VLAD	82.11 ± 0.14	9.41 ± 0.60	1.48 ± 0.36	71.41 ± 3.60	16.88 ± 7.61	4.57 ± 0.99
	SIFT + GMM + FV	83.38 ± 1.54	9.12 ± 0.57	1.18 ± 0.22	70.12 ± 1.36	17.83 ± 6.39	5.77 ± 1.24
	DSIFT + <i>k</i> -means + VLAD	79.90 ± 0.98	13.41 ± 2.29	2.88 ± 1.11	68.71 ± 0.29	19.88 ± 8.13	5.69 ± 1.84
	DSIFT + GMM + FV	82.70 ± 1.12	9.36 ± 0.67	1.54 ± 0.46	70.36 ± 0.30	19.28 ± 8.67	5.42 ± 1.94
	LIOP + <i>k</i> -means + VLAD	76.90 ± 1.28	15.89 ± 2.78	3.43 ± 1.51	67.89 ± 3.48	21.93 ± 9.25	6.73 ± 2.95
	LIOP + GMM + FV	78.09 ± 3.53	15.96 ± 2.14	3.12 ± 1.20	68.96 ± 3.54	20.03 ± 8.87	5.93 ± 2.15
	HOG + <i>k</i> -means + VLAD	78.66 ± 3.46	15.81 ± 2.26	3.31 ± 1.25	69.35 ± 3.36	20.36 ± 8.54	5.83 ± 1.65
Spectral-enhanced descriptor-based scheme	HOG + GMM + FV	79.19 ± 3.40	16.06 ± 2.58	3.10 ± 1.49	69.76 ± 3.20	20.17 ± 8.83	5.13 ± 1.35
	SIFT + <i>k</i> -means + VLAD + Spec	79.28 ± 3.16	16.32 ± 2.18	3.90 ± 1.82	69.66 ± 2.99	20.03 ± 8.52	4.87 ± 1.75
	SIFT + GMM + FV + Spec	79.61 ± 3.11	16.12 ± 2.34	3.15 ± 1.29	69.42 ± 2.91	20.94 ± 8.51	5.02 ± 1.85
	DSIFT + <i>k</i> -means + VLAD + Spec	79.66 ± 2.99	16.96 ± 2.88	3.76 ± 1.52	69.40 ± 2.75	19.74 ± 8.12	4.83 ± 1.72
	DSIFT + GMM + FV + Spec	79.89 ± 2.94	16.66 ± 2.83	3.14 ± 1.79	69.38 ± 2.62	20.97 ± 7.96	4.33 ± 1.63

Table 6. Classification results in terms of OA (%), QD (%), and AD (%) obtained by the techniques detailed in Table 4 for the images of the *standard dataset* and using a RF classifier. Fifteen percent of the superpixels are used for training. The best OA results are in a gray background.

Scheme	Technique	Salinas			Pavia		
		OA	QD	AD	OA	QD	AD
-	Without Texture Features	77.32 ± 0.56	13.27 ± 2.86	3.01 ± 1.16	66.69 ± 1.24	20.28 ± 9.37	5.91 ± 1.72
Codebook-based scheme	<i>k</i> -means + VLAD	75.77 ± 1.13	14.68 ± 2.35	3.20 ± 0.68	66.82 ± 2.05	19.88 ± 7.76	5.10 ± 1.42
	<i>k</i> -means + BOW	78.53 ± 1.68	14.21 ± 2.06	3.14 ± 1.58	61.93 ± 1.02	24.18 ± 9.72	7.55 ± 2.04
	GMM + FV	75.26 ± 1.60	15.64 ± 2.63	3.87 ± 1.74	67.38 ± 2.22	20.08 ± 8.27	5.73 ± 1.22
Descriptor-based scheme	SIFT + <i>k</i> -means + VLAD	83.20 ± 0.54	8.31 ± 1.82	1.18 ± 0.25	72.83 ± 2.85	15.92 ± 6.81	4.37 ± 0.79
	SIFT + GMM + FV	84.26 ± 1.43	8.82 ± 0.37	1.04 ± 0.36	71.58 ± 1.38	16.94 ± 5.97	5.03 ± 1.73
	DSIFT + <i>k</i> -means + VLAD	80.63 ± 0.85	12.82 ± 2.19	2.38 ± 1.10	69.82 ± 0.25	18.71 ± 7.93	5.09 ± 1.51
	DSIFT + GMM + FV	83.67 ± 1.02	8.56 ± 0.97	1.29 ± 0.36	71.76 ± 0.24	18.84 ± 8.07	4.86 ± 1.32
	LIOP + <i>k</i> -means + VLAD	77.13 ± 1.44	15.19 ± 2.28	3.34 ± 1.25	68.69 ± 3.04	20.33 ± 9.03	6.42 ± 2.30
	LIOP + GMM + FV	79.49 ± 3.23	15.09 ± 2.28	2.87 ± 1.16	69.45 ± 3.19	19.07 ± 8.34	5.27 ± 1.77
	HOG + <i>k</i> -means + VLAD	79.36 ± 3.06	14.91 ± 1.93	2.24 ± 1.23	70.95 ± 3.06	19.46 ± 7.74	4.32 ± 1.15
Spectral-enhanced descriptor-based scheme	HOG + GMM + FV	80.91 ± 2.31	15.63 ± 2.06	2.88 ± 0.97	70.96 ± 2.17	19.67 ± 6.93	4.46 ± 1.55
	SIFT + <i>k</i> -means + VLAD + Spec	80.38 ± 2.86	15.87 ± 2.98	3.56 ± 1.62	70.86 ± 2.74	19.43 ± 8.02	3.74 ± 1.05
	SIFT + GMM + FV + Spec	80.62 ± 3.10	15.63 ± 2.14	2.75 ± 1.30	70.12 ± 1.91	19.84 ± 8.11	4.92 ± 1.45
	DSIFT + <i>k</i> -means + VLAD + Spec	80.63 ± 2.93	16.26 ± 2.28	3.36 ± 1.54	70.41 ± 2.25	18.94 ± 8.02	4.33 ± 1.68
	DSIFT + GMM + FV + Spec	80.35 ± 2.75	15.36 ± 2.33	2.67 ± 1.82	70.31 ± 2.41	20.07 ± 7.16	4.56 ± 1.57

Table 7. Classification results in terms of OA (%), QD (%), and AD (%) obtained by techniques detailed in Table 4 for the images of the *standard dataset* and using a ELM classifier. Fifteen of the superpixels are used for training. The best OA results are in a gray background.

Scheme	Technique	Salinas			Pavia		
		OA	QD	AD	OA	QD	AD
-	Without Texture Features	75.64 ± 0.87	16.30 ± 2.38	3.33 ± 1.30	64.49 ± 0.15	22.69 ± 8.86	6.65 ± 1.47
Codebook-based scheme	<i>k</i> -means + VLAD	74.58 ± 1.14	16.48 ± 2.50	3.96 ± 1.72	66.01 ± 2.11	21.53 ± 8.87	6.72 ± 1.10
	<i>k</i> -means + BOW	76.08 ± 1.50	15.12 ± 2.38	3.63 ± 1.55	59.04 ± 1.89	28.77 ± 11.03	7.67 ± 2.94
	GMM + FV	73.01 ± 1.82	17.52 ± 3.10	5.24 ± 1.28	65.94 ± 3.09	23.82 ± 5.91	6.66 ± 1.69
Descriptor-based scheme	SIFT + <i>k</i> -means + VLAD	80.07 ± 0.93	11.24 ± 0.01	1.23 ± 0.31	72.07 ± 3.88	18.93 ± 7.58	5.31 ± 0.84
	SIFT + GMM + FV	82.58 ± 1.82	11.49 ± 0.13	1.38 ± 0.36	68.92 ± 1.60	18.13 ± 6.32	7.76 ± 1.42
	DSIFT + <i>k</i> -means + VLAD	79.47 ± 0.29	15.48 ± 1.65	3.49 ± 1.17	70.34 ± 0.51	20.01 ± 8.96	6.77 ± 1.02
	DSIFT + GMM + FV	83.93 ± 1.60	8.30 ± 0.57	1.75 ± 0.88	69.81 ± 0.90	19.28 ± 8.10	5.43 ± 1.49
	LIOP + <i>k</i> -means + VLAD	74.25 ± 1.42	16.05 ± 2.36	5.83 ± 1.10	69.29 ± 3.20	23.44 ± 9.41	8.10 ± 2.83
	LIOP + GMM + FV	76.72 ± 3.71	15.02 ± 2.56	4.07 ± 1.22	67.86 ± 3.46	22.22 ± 8.51	7.39 ± 2.01
	HOG + <i>k</i> -means + VLAD	76.83 ± 3.81	16.24 ± 2.35	4.82 ± 1.87	67.44 ± 3.50	22.10 ± 8.62	6.11 ± 1.17
Spectral-enhanced descriptor-based scheme	HOG + GMM + FV	77.82 ± 3.72	18.70 ± 2.73	3.90 ± 1.18	67.76 ± 3.38	23.23 ± 8.53	7.36 ± 1.07
	SIFT + <i>k</i> -means + VLAD + Spec	78.06 ± 3.31	18.92 ± 2.33	5.07 ± 1.93	68.11 ± 2.39	21.16 ± 8.17	4.74 ± 1.02
	SIFT + GMM + FV + Spec	78.60 ± 3.63	18.47 ± 2.72	3.48 ± 1.85	67.24 ± 2.08	21.57 ± 8.26	5.62 ± 1.45
	DSIFT + <i>k</i> -means + VLAD + Spec	77.07 ± 2.52	18.08 ± 2.96	5.59 ± 1.07	67.45 ± 2.17	20.89 ± 8.29	5.75 ± 1.07
	DSIFT + GMM + FV + Spec	77.11 ± 2.35	18.90 ± 2.39	3.60 ± 1.09	68.98 ± 2.26	21.22 ± 7.73	5.14 ± 1.41

The results for the *Galicia dataset* are detailed in Tables 8–10 for the results with a SVM, a RF, and a ELM classifier, respectively. Unlike the *standard dataset*, the *Galicia dataset* is made up of larger images,

so the execution time is very relevant. This is the reason for displaying execution times in the Tables. The best results for each image in terms of OA are highlighted with a gray background.

It can be observed that in this case, two techniques based on k-means as codebook generator, *k-means + BoW* and *LIOP + k-means + VLAD*, offer the best results for all the classifier. The best result is achieved for only one of the images, Mestas, and the RF classifier by a different technique: *SIFT + GMM + FV + Spec*. The AD and QD values are lower for higher OA values, as expected. Regarding execution times, the techniques with the highest ones correspond to those using a SIFT-based keypoint detection and description algorithm. On the contrary, the methods with the lowest computational cost are those based on LIOP or HOG as keypoint detection and description algorithms, while those based on the simpler *codebook-based* scheme present reasonable computational costs. Focusing on the best two techniques, *LIOP + k-means + VLAD* displays lower execution times than *k-means + BoW*.

In order to determine whether the previous results for the *Galicia dataset* are statistically reliable, Table 11 shows results varying the percentage of superpixels considered in the training set. Ten percent, 15%, and 20 % of the superpixels from each of the five vegetation classes were selected for training. For each percentage, the superpixels are randomly picked. The four best techniques extracted from the previous table were chosen to perform this comparison: *k-means + BoW*, *GMM + FV*, *LIOP + k-means + VLAD*, and *SIFT + k-means + VLAD + Spec*. It can be observed that the standard deviation decreases as the size of the training set increases. The highlighted best results show that the *k-means + BoW* technique outperforms the other methods 7 out of 16 times and presents the lowest standard deviation values. *LIOP + k-means + VLAD* obtains the best results in 2 out of 16 times and competitive results for the remaining experiments.

Table 8. Classification results in terms of OA (%), QD (%), AD (%), and execution times (seconds) obtained by the techniques detailed in Table 4 for the *Galicia dataset* images and using a SVM classifier. Fifteen percent of the superpixels are used for training. The best OA results are in a gray background.

	Oitavén				Mestas			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	72.71 ± 0.53	3.23 ± 1.88	2.22 ± 1.12	6 ± 0	86.10 ± 0.36	1.75 ± 0.48	1.06 ± 0.09	5 ± 0
<i>k-means + VLAD</i>	76.68 ± 1.06	2.08 ± 0.11	1.81 ± 0.41	239 ± 6	89.44 ± 0.13	0.73 ± 0.05	0.19 ± 0.09	161 ± 1
<i>k-means + BoW</i>	81.94 ± 0.54	2.71 ± 0.48	1.39 ± 0.38	239 ± 6	92.67 ± 0.41	0.50 ± 0.01	0.54 ± 0.04	160 ± 0
<i>GMM + FV</i>	79.62 ± 0.57	1.87 ± 0.79	1.15 ± 0.11	360 ± 7	90.27 ± 0.25	0.61 ± 0.07	0.99 ± 0.40	399 ± 1
<i>SIFT + k-means + VLAD</i>	79.52 ± 0.43	2.26 ± 0.79	1.15 ± 0.07	280 ± 2	86.92 ± 0.33	2.75 ± 0.66	1.45 ± 0.47	479 ± 2
<i>SIFT + GMM + FV</i>	64.58 ± 0.50	5.26 ± 1.56	3.58 ± 0.34	268 ± 2	90.08 ± 0.24	1.51 ± 0.25	0.98 ± 0.08	264 ± 0
<i>DSIFT + k-means + VLAD</i>	75.22 ± 0.48	3.43 ± 0.71	2.13 ± 1.10	1279 ± 6	86.09 ± 0.36	1.77 ± 0.60	0.86 ± 0.09	1369 ± 5
<i>DSIFT + GMM + FV</i>	77.32 ± 0.33	3.51 ± 0.73	1.61 ± 0.58	1136 ± 3	91.85 ± 0.67	1.01 ± 0.50	0.75 ± 0.08	2257 ± 4
<i>LIOP + k-means + VLAD</i>	82.74 ± 0.47	2.52 ± 0.43	2.48 ± 0.68	68 ± 1	90.24 ± 3.67	1.79 ± 0.03	1.78 ± 0.04	26 ± 0
<i>LIOP + GMM + FV</i>	71.72 ± 0.49	6.02 ± 1.42	4.51 ± 0.73	21 ± 0	90.33 ± 0.02	1.64 ± 0.73	1.12 ± 0.02	14 ± 0
<i>HOG + k-means + VLAD</i>	73.14 ± 0.26	5.65 ± 0.89	3.13 ± 0.88	111 ± 0	90.00 ± 0.11	1.34 ± 0.67	0.99 ± 0.18	49 ± 0
<i>HOG + GMM + FV</i>	60.43 ± 0.58	10.98 ± 1.47	6.16 ± 1.13	111 ± 0	90.33 ± 0.02	1.71 ± 0.02	0.30 ± 0.18	49 ± 0
<i>SIFT + k-means + VLAD + Spec</i>	77.86 ± 0.28	5.28 ± 1.24	3.39 ± 1.44	1068 ± 3	92.48 ± 0.02	0.95 ± 0.14	0.40 ± 0.04	1076 ± 2
<i>SIFT + GMM + FV + Spec</i>	77.97 ± 0.03	5.81 ± 1.66	3.78 ± 1.83	1068 ± 3	92.53 ± 0.02	0.25 ± 0.04	0.18 ± 0.02	1076 ± 2
<i>DSIFT + k-means + VLAD + Spec</i>	77.64 ± 0.33	5.63 ± 1.25	3.49 ± 1.55	1135 ± 4	92.51 ± 0.03	0.18 ± 0.07	0.36 ± 0.01	1501 ± 5
<i>DSIFT + GMM + FV + Spec</i>	78.03 ± 0.27	5.56 ± 1.28	3.14 ± 1.47	1135 ± 4	92.47 ± 0.03	0.50 ± 0.09	0.42 ± 0.09	1988 ± 4
	Ferreiras				Eiras			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	76.84 ± 0.71	7.86 ± 1.49	3.70 ± 1.08	9 ± 0	78.40 ± 0.79	6.05 ± 1.86	4.37 ± 1.28	11 ± 0
<i>k-means + VLAD</i>	83.58 ± 0.35	2.43 ± 1.07	2.30 ± 0.49	230 ± 1	86.58 ± 1.09	2.24 ± 0.75	1.78 ± 0.79	245 ± 2
<i>k-means + BoW</i>	84.93 ± 0.47	3.13 ± 0.43	2.77 ± 0.57	228 ± 0	85.19 ± 0.81	2.30 ± 1.16	1.57 ± 0.28	243 ± 0
<i>GMM + FV</i>	82.77 ± 0.34	3.95 ± 0.99	2.28 ± 1.12	702 ± 0	85.61 ± 0.41	2.83 ± 0.61	1.79 ± 0.69	887 ± 0
<i>SIFT + k-means + VLAD</i>	80.75 ± 0.08	3.41 ± 0.58	2.23 ± 1.07	682 ± 6	84.32 ± 0.24	3.83 ± 0.45	1.64 ± 0.81	503 ± 9
<i>SIFT + GMM + FV</i>	83.99 ± 0.28	3.87 ± 1.30	2.05 ± 0.84	600 ± 6	81.42 ± 0.48	4.97 ± 1.59	2.89 ± 0.76	493 ± 7
<i>DSIFT + k-means + VLAD</i>	83.29 ± 0.13	3.15 ± 0.73	1.49 ± 0.32	1951 ± 19	73.35 ± 4.56	5.76 ± 1.79	3.74 ± 0.65	2081 ± 1.89
<i>DSIFT + GMM + FV</i>	84.38 ± 1.09	3.29 ± 1.17	2.67 ± 0.96	2547 ± 131	81.17 ± 0.17	4.26 ± 1.17	2.28 ± 0.39	3953 ± 30
<i>LIOP + k-means + VLAD</i>	83.08 ± 0.41	3.40 ± 1.34	2.05 ± 0.43	118 ± 0	88.21 ± 0.37	3.99 ± 1.61	2.93 ± 0.48	99 ± 0
<i>LIOP + GMM + FV</i>	84.82 ± 0.19	5.34 ± 1.58	3.64 ± 0.61	52 ± 0	69.24 ± 0.35	6.64 ± 1.19	3.16 ± 1.15	73 ± 0
<i>HOG + k-means + VLAD</i>	82.98 ± 0.01	4.51 ± 1.49	2.43 ± 0.25	216 ± 0	84.41 ± 0.17	4.13 ± 1.04	2.79 ± 0.36	308 ± 0
<i>HOG + GMM + FV</i>	83.50 ± 0.03	3.77 ± 1.16	2.82 ± 0.44	216 ± 0	69.01 ± 0.26	6.22 ± 1.07	3.66 ± 0.81	308 ± 0
<i>SIFT + k-means + VLAD + Spec</i>	84.08 ± 0.10	3.96 ± 1.04	2.90 ± 0.86	1477 ± 7	84.96 ± 0.78	2.91 ± 0.97	1.94 ± 0.82	1707 ± 7
<i>SIFT + GMM + FV + Spec</i>	84.12 ± 0.16	3.72 ± 1.40	2.23 ± 0.02	1477 ± 0	85.14 ± 0.51	2.15 ± 0.98	1.17 ± 0.82	1707 ± 7
<i>DSIFT + k-means + VLAD + Spec</i>	84.06 ± 0.13	3.09 ± 1.05	2.31 ± 0.51	2195 ± 26	85.21 ± 0.59	2.84 ± 0.42	1.47 ± 0.28	2385 ± 14
<i>DSIFT + GMM + FV + Spec</i>	84.05 ± 0.13	3.42 ± 1.43	2.97 ± 0.87	2064 ± 476	85.11 ± 0.57	2.01 ± 0.26	1.34 ± 0.67	4037 ± 148

Table 9. Classification results in terms of OA (%), QD (%), AD (%), and execution times (seconds) obtained by the techniques detailed in Table 4 for the *Galicia dataset* images and using a RF classifier. Fifteen percent of the superpixels are used for training. The best OA results are in a gray background.

	Oitavén				Mestas			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	73.66 ± 0.27	2.38 ± 0.75	1.69 ± 1.12	8 ± 0	87.65 ± 0.32	1.85 ± 0.10	1.56 ± 0.93	7 ± 0
<i>k-means + VLAD</i>	78.75 ± 1.83	2.56 ± 0.50	1.41 ± 0.94	243 ± 5	91.21 ± 0.12	0.92 ± 0.06	0.16 ± 0.02	164 ± 2
<i>k-means + BoW</i>	82.86 ± 0.83	2.13 ± 0.64	1.80 ± 0.79	242 ± 6	92.44 ± 0.68	0.57 ± 0.02	0.65 ± 0.05	165 ± 2
<i>GMM + FV</i>	80.35 ± 0.69	1.13 ± 0.51	1.74 ± 0.05	362 ± 4	91.31 ± 0.15	0.48 ± 0.06	0.25 ± 0.08	402 ± 2
<i>SIFT + k-means + VLAD</i>	81.38 ± 0.48	2.35 ± 0.59	1.33 ± 0.04	285 ± 2	87.85 ± 0.39	2.90 ± 0.60	1.45 ± 0.27	477 ± 4
<i>SIFT + GMM + FV</i>	66.93 ± 0.12	6.98 ± 1.23	3.26 ± 0.77	264 ± 1	90.34 ± 0.39	1.13 ± 0.57	0.46 ± 0.06	272 ± 2
<i>DSIFT + k-means + VLAD</i>	76.60 ± 0.33	3.21 ± 0.34	2.83 ± 1.31	1291 ± 9	88.63 ± 0.11	1.47 ± 0.22	0.55 ± 0.07	1345 ± 6
<i>DSIFT + GMM + FV</i>	78.78 ± 0.89	2.94 ± 0.35	1.33 ± 0.25	1139 ± 2	91.37 ± 0.90	1.94 ± 0.29	0.49 ± 0.01	2263 ± 5
<i>LIOP + k-means + VLAD</i>	83.29 ± 0.80	2.86 ± 0.85	2.36 ± 0.23	69 ± 1	91.24 ± 3.99	1.09 ± 0.03	1.16 ± 0.02	29 ± 0
<i>LIOP + GMM + FV</i>	73.89 ± 0.41	7.61 ± 1.29	4.90 ± 0.37	22 ± 1	91.44 ± 0.04	1.66 ± 0.83	1.77 ± 0.03	15 ± 0
<i>HOG + k-means + VLAD</i>	74.24 ± 0.10	5.45 ± 0.62	3.87 ± 0.06	113 ± 0	91.08 ± 0.18	1.84 ± 0.45	0.24 ± 0.01	52 ± 0
<i>HOG + GMM + FV</i>	62.04 ± 0.38	10.59 ± 1.98	6.29 ± 1.51	115 ± 3	91.33 ± 0.09	1.56 ± 0.08	0.28 ± 0.01	51 ± 0
<i>SIFT + k-means + VLAD + Spec</i>	78.26 ± 0.28	4.21 ± 0.87	2.51 ± 0.46	1076 ± 3	93.67 ± 0.62	0.95 ± 0.03	0.64 ± 0.01	1089 ± 1
<i>SIFT + GMM + FV + Spec</i>	78.92 ± 0.55	4.04 ± 0.31	2.08 ± 0.32	1078 ± 2	93.77 ± 0.79	0.45 ± 0.07	0.74 ± 0.02	1086 ± 1
<i>DSIFT + k-means + VLAD + Spec</i>	78.85 ± 0.69	4.97 ± 0.80	2.88 ± 0.25	1097 ± 3	93.34 ± 0.42	0.99 ± 0.03	0.59 ± 0.03	1189 ± 1
<i>DSIFT + GMM + FV + Spec</i>	78.04 ± 0.59	4.14 ± 0.25	6.29 ± 0.92	1089 ± 3	93.45 ± 0.91	0.93 ± 0.04	0.33 ± 0.02	1146 ± 3
	Ferreiras				Eiras			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	78.64 ± 0.99	7.19 ± 1.62	3.64 ± 0.38	11 ± 0	81.70 ± 0.78	6.60 ± 1.95	4.68 ± 1.43	14 ± 1
<i>k-means + VLAD</i>	84.19 ± 0.40	2.36 ± 0.44	1.07 ± 0.05	240 ± 2	87.38 ± 1.19	2.61 ± 0.54	1.58 ± 0.17	263 ± 2
<i>k-means + BoW</i>	85.99 ± 0.44	3.36 ± 0.60	2.01 ± 0.26	235 ± 0	86.46 ± 0.59	2.54 ± 1.74	1.82 ± 0.89	267 ± 4
<i>GMM + FV</i>	83.47 ± 0.96	3.48 ± 0.14	2.10 ± 1.40	732 ± 1	85.41 ± 0.85	2.39 ± 0.09	1.66 ± 0.53	896 ± 1
<i>SIFT + k-means + VLAD</i>	81.82 ± 0.31	3.22 ± 0.61	2.16 ± 0.58	701 ± 3	85.57 ± 0.24	3.71 ± 0.57	1.93 ± 0.64	533 ± 12
<i>SIFT + GMM + FV</i>	84.09 ± 0.82	3.18 ± 1.07	2.27 ± 0.12	608 ± 4	81.84 ± 0.49	4.96 ± 1.41	2.01 ± 0.27	515 ± 5
<i>DSIFT + k-means + VLAD</i>	83.24 ± 0.91	3.41 ± 0.86	1.04 ± 0.06	1962 ± 14	73.54 ± 4.13	5.11 ± 1.25	3.74 ± 0.80	2099 ± 2
<i>DSIFT + GMM + FV</i>	85.76 ± 1.92	3.35 ± 0.80	2.49 ± 0.46	2588 ± 56	82.56 ± 0.76	4.40 ± 1.59	2.81 ± 0.22	3943 ± 4
<i>LIOP + k-means + VLAD</i>	84.56 ± 0.65	3.68 ± 0.27	2.94 ± 0.10	123 ± 3	89.20 ± 0.96	3.39 ± 1.54	2.25 ± 0.56	117 ± 2
<i>LIOP + GMM + FV</i>	85.96 ± 0.84	5.19 ± 1.16	3.27 ± 0.54	58 ± 0	71.03 ± 0.60	6.65 ± 1.32	3.16 ± 0.25	87 ± 0
<i>HOG + k-means + VLAD</i>	82.74 ± 0.01	4.91 ± 1.32	2.43 ± 0.11	228 ± 1	85.92 ± 0.81	4.29 ± 0.49	2.72 ± 0.52	315 ± 1
<i>HOG + GMM + FV</i>	84.63 ± 0.09	3.86 ± 1.28	2.14 ± 0.16	239 ± 0	69.01 ± 0.54	6.59 ± 1.07	3.97 ± 0.07	308 ± 0
<i>SIFT + k-means + VLAD + Spec</i>	84.93 ± 0.09	3.22 ± 1.04	2.02 ± 0.45	1497 ± 5	84.99 ± 0.54	2.41 ± 0.75	1.06 ± 0.87	1712 ± 4
<i>SIFT + GMM + FV + Spec</i>	84.64 ± 0.37	3.10 ± 1.30	2.53 ± 0.49	1481 ± 0	85.36 ± 0.53	2.40 ± 0.92	1.47 ± 0.59	1757 ± 5
<i>DSIFT + k-means + VLAD + Spec</i>	84.38 ± 0.65	3.16 ± 1.81	2.69 ± 0.29	2208 ± 20	85.87 ± 0.59	2.94 ± 0.01	1.89 ± 0.74	2399 ± 3
<i>DSIFT + GMM + FV + Spec</i>	84.62 ± 0.94	3.17 ± 1.90	2.15 ± 0.61	2084 ± 217	85.18 ± 0.53	2.46 ± 0.08	1.45 ± 0.08	4056 ± 176

It can be concluded from Table 11 that *k-means + BoW* is a technique that offers the best consistent results because it outperforms the other ones in terms of OA and obtains reasonable results regarding execution times. *LIOP + k-means + VLAD* is also an interesting technique, although it was experimentally checked that its results are highly dependent on the optimization of the tuning of its input parameters, which is a resource expensive process.

Finally, Table 12 shows the results obtained for the *Gaofen dataset* as it is a dataset with a large number of scenes. It is divided into *GID5* (150 scenes) and *GID15* (10 scenes) [47], as it was detailed in the dataset description. The data trends are similar to those of the *Galicia dataset*, being the *codebook-based* scheme the best, followed by the *descriptor-based* scheme and the *spectral-enhanced descriptor-based* scheme. Specifically, the best techniques are again those based on *k-means* as codebook generator algorithm. In this case, the best techniques are, in particular, *k-means + BoW* and *k-means + VLAD*.

Table 10. Classification results in terms of OA (%), QD (%), AD (%), and execution times (seconds) obtained by the techniques detailed in Table 4 for the *Galicia dataset* images and using a ELM classifier. Fifteen percent of the superpixels are used for training. The best OA results are in a gray background.

	Oitavén				Mestas			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	71.90 ± 0.72	4.23 ± 1.19	2.90 ± 0.64	8 ± 0	84.66 ± 0.23	1.97 ± 0.63	0.54 ± 0.07	6 ± 0
<i>k-means + VLAD</i>	74.32 ± 0.57	3.19 ± 0.80	2.66 ± 0.89	248 ± 3	87.31 ± 0.15	0.74 ± 0.04	0.34 ± 0.02	178 ± 1
<i>k-means + BoW</i>	80.01 ± 0.86	2.54 ± 0.33	1.22 ± 0.92	248 ± 4	91.95 ± 0.16	1.38 ± 0.08	0.94 ± 0.06	179 ± 1
<i>GMM + FV</i>	78.16 ± 0.75	1.33 ± 0.52	1.98 ± 0.82	383 ± 4	89.46 ± 0.57	0.83 ± 0.04	0.74 ± 0.90	408 ± 2
<i>SIFT + k-means + VLAD</i>	78.66 ± 0.70	2.67 ± 0.58	1.21 ± 0.01	289 ± 2	84.71 ± 0.52	2.94 ± 0.57	1.35 ± 0.78	503 ± 1
<i>SIFT + GMM + FV</i>	63.55 ± 0.76	5.08 ± 0.32	3.71 ± 0.22	275 ± 2	89.18 ± 0.41	1.43 ± 0.67	0.02 ± 0.13	271 ± 1
<i>DSIFT + k-means + VLAD</i>	73.39 ± 0.36	3.85 ± 0.08	2.58 ± 0.13	1287 ± 7	85.09 ± 0.99	1.70 ± 0.81	0.86 ± 0.11	1469 ± 7
<i>DSIFT + GMM + FV</i>	76.84 ± 0.85	3.93 ± 0.68	1.62 ± 0.88	1153 ± 4	91.04 ± 0.59	1.36 ± 0.90	0.37 ± 0.07	2274 ± 3
<i>LIOP + k-means + VLAD</i>	81.07 ± 0.57	2.64 ± 0.29	2.36 ± 0.61	75 ± 1	89.43 ± 1.69	1.13 ± 0.08	1.26 ± 0.06	36 ± 1
<i>LIOP + GMM + FV</i>	70.16 ± 0.92	5.88 ± 0.61	4.09 ± 0.11	31 ± 0	89.64 ± 0.04	1.21 ± 0.13	1.73 ± 0.05	28 ± 1
<i>HOG + k-means + VLAD</i>	72.91 ± 0.40	5.01 ± 0.70	3.57 ± 0.73	121 ± 2	89.35 ± 0.31	1.67 ± 0.49	0.29 ± 0.09	56 ± 1
<i>HOG + GMM + FV</i>	60.55 ± 0.26	12.02 ± 1.66	7.79 ± 1.16	150 ± 2	89.69 ± 0.05	1.56 ± 0.04	0.64 ± 0.27	54 ± 0
<i>SIFT + k-means + VLAD + Spec</i>	76.79 ± 0.69	5.86 ± 0.99	3.10 ± 1.25	1076 ± 3	90.20 ± 0.02	0.96 ± 0.04	0.39 ± 0.01	1176 ± 3
<i>SIFT + GMM + FV + Spec</i>	75.68 ± 0.02	5.42 ± 1.82	3.46 ± 1.18	1142 ± 3	91.49 ± 0.01	0.16 ± 0.06	0.63 ± 0.07	1256 ± 2
<i>DSIFT + k-means + VLAD + Spec</i>	76.10 ± 0.76	5.96 ± 1.65	3.79 ± 1.84	1155 ± 4	91.85 ± 0.04	0.35 ± 0.04	0.36 ± 0.49	1701 ± 5
<i>DSIFT + GMM + FV + Spec</i>	77.42 ± 0.82	5.60 ± 0.86	3.19 ± 1.05	1835 ± 4	90.14 ± 0.03	0.47 ± 0.05	0.65 ± 0.06	2410 ± 4
	Ferreiras				Eiras			
	OA	QD	AD	Time	OA	QD	AD	Time
<i>Without Texture Features</i>	73.48 ± 0.82	7.68 ± 0.60	3.35 ± 0.49	12 ± 1	76.05 ± 0.60	6.15 ± 1.23	4.02 ± 1.33	19 ± 0
<i>k-means + VLAD</i>	80.93 ± 0.55	2.27 ± 1.97	2.75 ± 0.90	252 ± 3	85.98 ± 0.76	2.40 ± 0.52	1.80 ± 0.60	269 ± 3
<i>k-means + BoW</i>	83.27 ± 0.66	3.94 ± 0.97	2.72 ± 0.61	235 ± 1	84.68 ± 0.81	3.29 ± 0.97	1.60 ± 0.47	282 ± 2
<i>GMM + FV</i>	81.83 ± 0.66	3.33 ± 0.91	2.47 ± 0.75	722 ± 1	84.66 ± 0.42	2.03 ± 0.75	1.73 ± 0.52	925 ± 2
<i>SIFT + k-means + VLAD</i>	78.46 ± 0.28	3.40 ± 0.50	3.14 ± 0.25	701 ± 4	82.84 ± 0.17	3.91 ± 0.76	1.57 ± 0.47	573 ± 1
<i>SIFT + GMM + FV</i>	81.71 ± 0.91	3.93 ± 0.96	2.29 ± 0.47	630 ± 6	80.81 ± 0.51	4.82 ± 0.71	3.55 ± 0.52	536 ± 2
<i>DSIFT + k-means + VLAD</i>	82.82 ± 0.07	3.27 ± 0.44	1.41 ± 0.87	2254 ± 23	72.30 ± 1.48	6.20 ± 0.58	4.94 ± 0.42	2191 ± 1.70
<i>DSIFT + GMM + FV</i>	83.14 ± 0.64	4.21 ± 0.24	3.54 ± 0.88	2747 ± 52	80.29 ± 0.72	4.01 ± 0.44	3.34 ± 0.45	4201 ± 41
<i>LIOP + k-means + VLAD</i>	82.02 ± 0.55	3.29 ± 0.12	3.88 ± 0.53	238 ± 1	87.58 ± 0.37	3.39 ± 0.42	1.37 ± 0.48	106 ± 2
<i>LIOP + GMM + FV</i>	82.77 ± 0.23	5.01 ± 0.83	3.92 ± 0.63	72 ± 1	67.63 ± 0.65	7.70 ± 1.25	4.05 ± 0.04	96 ± 1
<i>HOG + k-means + VLAD</i>	81.12 ± 0.09	4.40 ± 1.09	2.29 ± 0.04	266 ± 1	83.70 ± 0.93	4.28 ± 1.07	2.78 ± 0.40	358 ± 1
<i>HOG + GMM + FV</i>	82.75 ± 0.18	3.67 ± 1.81	2.85 ± 0.50	296 ± 0	67.99 ± 0.47	6.99 ± 0.86	3.43 ± 0.84	358 ± 1
<i>SIFT + k-means + VLAD + Spec</i>	82.56 ± 0.48	4.68 ± 1.25	2.38 ± 0.31	1682 ± 7	82.33 ± 0.07	3.50 ± 0.29	1.77 ± 0.08	1857 ± 3
<i>SIFT + GMM + FV + Spec</i>	82.93 ± 0.28	3.40 ± 1.04	2.27 ± 0.09	1688 ± 0	84.08 ± 0.03	2.73 ± 0.95	1.54 ± 0.09	1835 ± 7
<i>DSIFT + k-means + VLAD + Spec</i>	83.88 ± 0.96	3.01 ± 1.10	2.30 ± 0.26	2674 ± 26	85.90 ± 0.66	2.80 ± 0.05	1.70 ± 0.59	2412 ± 14
<i>DSIFT + GMM + FV + Spec</i>	82.95 ± 0.78	3.11 ± 1.88	2.95 ± 0.39	2580 ± 476	83.48 ± 0.59	2.49 ± 0.60	1.41 ± 0.37	4345 ± 148

Table 11. Classification results for different percentages of training samples for the *Galicia dataset* images using a SVM classifier and varying the percentage of superpixels randomly selected for training. Accuracy values expressed in terms of OA (in %) indicating standard deviation values (±). The best results are in a gray background.

%	Technique	Oitavén	Mestas	Ferreiras	Eiras
10	<i>k-means + BoW</i>	76.01 ± 0.30	87.6 ± 0.20	79.52 ± 0.27	84.48 ± 0.21
	<i>GMM + FV</i>	75.24 ± 1.01	84.01 ± 3.14	80.23 ± 1.59	82.51 ± 2.00
	<i>LIOP + k-means + VLAD</i>	75.23 ± 0.82	86.24 ± 3.01	81.02 ± 1.01	83.31 ± 2.00
	<i>SIFT + k-means + VLAD + Spec</i>	75.83 ± 1.44	86.93 ± 3.08	81.22 ± 1.57	83.93 ± 2.00
15	<i>k-means + BoW</i>	81.94 ± 0.54	92.67 ± 0.41	84.93 ± 0.47	85.19 ± 0.81
	<i>GMM + FV</i>	79.62 ± 0.67	90.27 ± 0.25	82.77 ± 0.34	85.61 ± 0.41
	<i>LIOP + k-means + VLAD</i>	82.74 ± 0.47	90.24 ± 3.67	83.08 ± 0.41	88.21 ± 0.37
	<i>SIFT + k-means + VLAD + Spec</i>	77.86 ± 0.28	92.48 ± 0.02	84.08 ± 0.10	84.96 ± 0.78
20	<i>k-means + BoW</i>	83.12 ± 0.14	92.14 ± 0.19	85.07 ± 0.35	86.62 ± 0.27
	<i>GMM + FV</i>	81.93 ± 1.21	91.65 ± 0.66	84.42 ± 0.71	87.32 ± 1.22
	<i>LIOP + k-means + VLAD</i>	81.38 ± 2.04	91.22 ± 1.93	83.98 ± 0.75	86.77 ± 1.43
	<i>SIFT + k-means + VLAD + Spec</i>	82.39 ± 1.19	90.76 ± 2.03	83.99 ± 0.86	86.66 ± 0.88

Table 12. Classification accuracy results in terms of OA (%) for the *GID5* and *GID15* datasets.

Scheme	Technique	<i>GID5</i>	<i>GID15</i>
-	<i>Without Texture Features</i>	91.28	69.93
Codebook-based scheme	<i>k-means + VLAD</i>	93.84	79.63
	<i>k-means + BoW</i>	95.79	79.61
	<i>GMM + FV</i>	94.66	79.61
	<i>SIFT + k-means + VLAD</i>	92.46	76.21
Descriptor-based scheme	<i>SIFT + GMM + FV</i>	93.63	75.97
	<i>DSIFT + k-means + VLAD</i>	93.02	76.47
	<i>DSIFT + GMM + FV</i>	93.04	75.81
	<i>LIOP + k-means + VLAD</i>	94.85	77.32
	<i>LIOP + GMM + FV</i>	92.19	71.24
	<i>HOG + k-means + VLAD</i>	94.53	76.76
	<i>HOG + GMM + FV</i>	92.18	70.98
	Spectral-enhanced descriptor-based scheme	<i>SIFT + k-means + VLAD + Spec</i>	93.99
<i>SIFT + GMM + FV + Spec</i>		94.00	75.50
<i>DSIFT + k-means + VLAD + Spec</i>		93.91	75.47
<i>DSIFT + GMM + FV + Spec</i>		93.92	75.51

4. Discussion

In this work, different texture schemes based on BoW for vegetation classification using a superpixel approach were studied. We considered multi- and hyperspectral remote sensing images taken by UAVs and satellites. In all the presented schemes, the texture algorithms were computed inside superpixels, in contrast to most of the methods previously published in the literature, in which the vegetation textures are extracted from patches or objects. A detailed comparison of the different techniques was carried out in terms of classification accuracy for several land cover remote sensing datasets. In particular, the *Galicia dataset* contained five classes of vegetation (oak, meadows, autochthonous vegetation, eucalyptus, and pines), while in the *GID5* 5 classes were considered, three of them corresponding to vegetation, and in the *GID15*, 15 classes were considered, corresponding eight of them to vegetation. The best classification results for each image ranged from 81.07% to 93.77% for the *Galicia dataset*, and from 79.63% to 95.79% for the *Gaofen dataset*. The techniques and algorithms used in this work included several keypoint detectors and descriptors (HOG, LIOP, SIFT, and DSIFT), algorithms for codebook generation (k-means and GMM), algorithms for feature encoding (histogram-based, VLAD, and FV), and, finally, some algorithms for feature classification (SVM, RF, and ELM). Additionally, SLIC was used for superpixel generation and PCA for dimensionality reduction.

In previous works, studies were carried out to determine the most suitable set of parameters, including textures, to carry out the classification of vegetation in remote sensing images. However, in these works the only texture method considered was GLCM. Specifically, the authors of [10] present a classification scheme for the canopy cover mapping of spekboom in a large semiarid region in South Africa using multispectral imagery (red, green, blue, and near-infrared bands). Three classes were considered (spekboom, tree, and background) and the classification scheme is a decision tree with 47 features grouped into two broad categories: per-pixel (spectral information) and sliding window features (statistic of the pixels inside a small local neighborhood). The decision tree obtained a mean absolute canopy cover error of 5.85%. The authors of [14] present a crop classification method for hyperspectral images combining 40 spectral indices, spectral features (several class-pair distances), and GLCM texture information in an object-oriented approach. Eight classes were considered (chinese cabbage, japanese cabbage, lettuce, radish, pasture, pole bean, and forest) and the classification accuracy obtained was 97.84%. Finally, the authors of [16] performed the classification of savannah tree species from very high resolution images acquired by UAVs. Two flights capturing

multispectral imagery (red, green, blue, and near-infrared bands) were made to obtain image mosaics with longitudinal and lateral overlap. The method uses chromatic coordinates, spectral indices, the canopy height model, and GLCM texture measures in different window sizes. Nine classes of trees and shrubs (with an abundance of more than ten individuals within the samples) were considered and an outline of each single-stem individual was drawn onto the image. The overall accuracy obtained was of 77% on average.

For the detection of the extent of trees and shrubs, the canopy height model (CHM) is the one most commonly used. Information on height is obtained from different sources, in some cases through sensor fusion with LiDAR information. In other cases surface reconstruction from aerial images is the way of obtaining this complementary information [21,22]. In this paper, information on height was not considered as only single images are available in the dataset for the areas under study.

Other works use simple texture methods for vegetation classification, specifically LBP and GLCM. For example, the authors of [12] applied the LBP textures for the classification of tree species using hyperspectral data and an aerial stereo camera system. In the classification step, a pixel-based approach and a patch-based BoW approach were used. Four classes were considered (spruce, beech, mixed, and non-tree) and the classification accuracy obtained was approximately 60%. In [13], an UAV performs vegetation mapping in complex urban landscapes using ultra-high resolution color imagery acquired at low altitudes. A hybrid method combining Random Forest and GLCM texture analysis at nine different window sizes was used. Six typical land covers (out of which three are vegetated ones) were considered (grass, trees, shrubs, bare soil, impervious surface, and water) and the classification accuracies ranged from 86.2% to 91.8%. The authors of [15] propose an object-based GLCM texture extraction method for the classification of man-planted forests on mountainous areas using high resolution satellite data, including panchromatic and multispectral bands. The method used a multi-resolution segmentation algorithm to generate image objects and enhances the texture feature of objects using a 2D Gabor filter. Four classes were considered (non-vegetation, natural forest, rubber trees, and crops) and the classification accuracy obtained was 91.4%. The authors of [17] propose a method combining spectral measures and GLCM texture information for crop classification in time-series UAV images composed of three bands (green, red, and near-infrared). The object-oriented approach extracted meaningful objects via multi-resolution segmentation and classification was carried out on object units. Four vegetation classes (highland kimchi cabbage, cabbage, potato, and fallow) were considered. In six multi-temporal images, combining texture features with spectral information led to an increase of 7.72% in OA, compared to the classification result with spectral information only (from 83.13% to 90.85%).

For the classification of a image using textures it is necessary to delimit regions on which the texture features are computed. None of the works cited above uses superpixels to obtain the textures. Instead they use patches, segments or objects. Superpixels were used in [24], which proposes a scheme for natural roadside vegetation classification shooting on the ground (not remote sensing) using color cameras. Six classes were considered (brown grass, green, road, soil, tree, and soil) and the scheme learns two individual sets of BoW dictionaries from color and filter-bank texture features using the nearest Euclidean distance, which were aggregated into class probabilities for each superpixel. Experimental evaluations on a natural image dataset obtained 75.5% accuracy for classifying six objects.

For keypoint detection and description, we considered four algorithms: HOG, LIOP, SIFT, and DSIFT. Few works in the literature proposed descriptor-based methods for the classification of vegetation in images, although this approach is common in the classification of scenes. For example, the authors of [18] present a methodology for vegetation segmentation in cornfield images obtained by autonomous agricultural vehicles. A collection of outdoor color images, which were acquired under different illumination conditions and different plant growth state, were selected. The method focuses on finding an appropriate set of different color vegetation indices and local descriptors for vegetation characterization. Three different classes were considered (vegetation, light-brown soil, and dark-brown soil), and an accuracy value of 95.3% was achieved. In [19], the classification of weeds

growing among crops using a BoW model based on SIFT or SURF features is presented. In that work, a small-sized robot was developed for vision based precision control of volunteer potatoes (weed) in a sugar beet field. The highest classification accuracy (96.5%) was obtained using SIFT, Out-of-Row Regional Index (ORRI), and SVM. Finally, the authors of [20] study the application of SIFT to cropland mapping in the Brazilian Amazon based on vegetation index time series. It used a dense temporal SIFT BoW algorithm, which is able to capture temporal locality of the data. The dataset was thus made of 46 MODIS images acquired over two years. Five crop classes were considered (soybean, soybean + millet, soybean + maize, soybean + cotton, and cotton) with accurate detection of around 70% of the agricultural areas.

Based on the presented information, it can be concluded that the number of works in the literature that use descriptors to characterize textures in the topic of vegetation classification in images is very limited. To the best of our knowledge, no superpixel-based descriptors have been previously proposed for the classification of vegetation in multi and hyperspectral images. On the other hand, our classification results are comparable to other studies in the literature. However, an exact numerical comparison is difficult as it depends on the nature of the datasets, the number of classes, and the number of samples used for training. Comparable results in the literature are only available for GID5 [47]. In particular, the accuracy obtained by the CNN-based technique used in this reference was 95.74%, which is similar to the 95.79% obtained in the experiments shown in Table 12. However, the experimental conditions in [47] were different as disjoint sets of images were taken for training and testing. In our approach the same percentage of training and testing samples were selected over each one of the images.

5. Conclusions

In this paper, different texture extraction schemes at the superpixel level for the classification of vegetation species using multi- and hyperspectral imagery are proposed. These schemes, based on the classical BoW approach, are called *codebook-based*, *descriptor-based*, and *spectral-enhanced descriptor-based* schemes. Some of the following stages are considered for each one; superpixel extraction, keypoint detection and description, codebook generation, feature encoding, and dimensionality reduction. The relevant contributions of this paper are the use of a superpixel segmentation algorithm as a way of dividing an image into homogeneous regions previously to the texture extraction, and the adequate exploitation of the spectral information available in all the bands of the image. Superpixels are used in the keypoint detection and description, codebook generation, feature encoding and classification stages. Sixteen different texture-extraction techniques derived from the three proposed schemes are analyzed in detail in the paper and compared in terms of classification accuracy and execution time considering SVM, RF and ELM as supervised classification algorithms.

Three datasets consisting of real multi- and hyperspectral images containing vegetation classes were employed to test the proposed schemes. As the *standard dataset* does not focus on vegetation, the *Galicia* and *Gaofen* datasets were also considered. The best classification results for each image range from 81.07% to 93.77% for the *Galicia dataset* and from 79.63% to 95.79% for the *Gaofen dataset*. The techniques and algorithms used in this work include several keypoint detectors and descriptors (HOG, LIOP, SIFT, and DSIFT), algorithms for codebook generation (k-means and GMM), algorithms for feature encoding (histogram-based, VLAD and FV), and, finally, some algorithms for feature classification (SVM, RF, and ELM). Additionally, SLIC was used for superpixel generation and PCA for dimensionality reduction. The experimental results show that the best techniques are based on k-means as codebook generator. In particular, the highest OA values are offered by *k-means + BoW*, that is a representative of the *codebook-based* scheme, using BoW for feature encoding. The second best results on average are provided by *LIOP + k-means + VLAD*, which uses LIOP for keypoint detection and description and VLAD for feature encoding, as a representative of the *descriptor-based* scheme. These are also techniques that present reasonable computational cost according to our experiments.

As future work, we plan to analyze the performance of the best techniques with new multispectral images corresponding to vegetation. The desired properties of these images will be the abundance of vegetation and high spatial resolution. Several future research lines that would benefit from the current proposal have also been considered such as testing different algorithms for keypoint detection and description, for instance, robust and powerful techniques like SURF and KAZE. Moreover, the creation of schemes with a different structure from the three described is also projected as future work.

Author Contributions: Conceptualization, D.B.H. and F.A.; Experiments, S.R.B.; Project administration, D.B.H. and F.A.; Software, S.R.B.; Supervision, F.A.; Writing—original draft, S.R.B.; Writing—review and editing, D.B.H. and F.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Civil Program UAVs Initiative, promoted by the Xunta de Galicia and developed in partnership with the Babcock Company to promote the use of unmanned technologies in civil services. We also have to acknowledge the support by Ministerio de Ciencia e Innovación, Government of Spain (grant number PID2019-104834GB-I00), and Consellería de Educación, Universidade e Formación Profesional (ED431C 2018/19, and accreditation 2019-2022 ED431G-2019/04). All are cofunded by the European Regional Development Fund (ERDF).

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript.

UAV	Unmanned Aerial Vehicle
BoW	Bag of Words
LBP	Local Binary Pattern
GLCM	Gray-Level Co-occurrence Matrix
CHM	Canopy Height Model
SVM	Support Vector Machines
RF	Random Forest
ELM	Extreme Learning Machine
CNN	Convolutional Neural Network
GLCM	Gray-Level Co-occurrence Matrix
FNEA	Fractal Net Evolution Approach
NDVI	Normalized Difference Vegetation Index
DEM	Digital Elevation Model
SLIC	Simple Linear Iterative Clustering
ETPS	Efficient Topology Preserving Segmentation
SIFT	Scale-Invariant Feature Transform
HOG	Histogram of Oriented Gradients
DSIFT	Dense Scale-Invariant Feature Transform
LIOP	Local Intensity Order Pattern
GMM	Gaussian Mixture Modeling
VLAD	Vector of Locally Aggregated Descriptors
FV	Fisher Vectors
PCA	Principal Component Analysis
GID	Gaofen Image Dataset
ROSIS	Reflective Optics System Imaging Spectrometer
AVIRIS	Airborne Visible Infrared Imaging Spectrometer
RF	Random Forest
OA	Overall Accuracy
QD	Quantity Disagreement
AD	Allocation Disagreement

References

1. Ghamisi, P.; Maggiori, E.; Li, S.; Souza, R.; Tarabalka, Y.; Moser, G.; De Giorgi, A.; Fang, L.; Chen, Y.; Chi, M.; et al. New frontiers in spectral-spatial classification of hyperspectral images. *IEEE Geosci. Remote Sens. Mag.* **2018**, *6*, 10–43. [[CrossRef](#)]
2. Wagner, F.; Sanchez, A.; Tarabalka, Y.; Lotte, R.; Ferreira, M.; Aidar, M.; Gloor, M.; Phillips, O.; Aragão, L. Using convolutional network to identify tree species related to forest disturbance in a neotropical Forest with very high resolution multispectral images. *AGUFM* **2018**, *2018*, B33N–2861.
3. Zeng, Y.; Zhao, Y.; Zhao, D.; Wu, B. Forest biodiversity mapping using airborne LiDAR and hyperspectral data. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 3561–3562.
4. Liu, L.; Chen, J.; Fieguth, P.; Zhao, G.; Chellappa, R.; Pietikäinen, M. From BoW to CNN: Two decades of texture representation for texture classification. *Int. J. Comput. Vis.* **2019**, *127*, 74–109. [[CrossRef](#)]
5. Julesz, B. Textons, the elements of texture perception, and their interactions. *Nature* **1981**, *290*, 91–97. [[CrossRef](#)]
6. Csurka, G.; Dance, C.; Fan, L.; Willamowski, J.; Bray, C. Visual categorization with bags of keypoints. In Proceedings of the 8th European Conference on Computer Vision-ECCV 2004, Prague, Czech Republic, 11–14 May 2004; Volume 1, pp. 1–2.
7. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2012; pp. 1097–1105.
8. Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; Vedaldi, A. Describing textures in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 3606–3613.
9. Bormann, R.; Esslinger, D.; Hundsdoerfer, D.; Haegele, M.; Vincze, M. Texture characterization with semantic attributes: Database and algorithm. In Proceedings of the ISR 2016: 47th International Symposium on Robotics, VDE, Munich, Germany, 21–22 June 2016; pp. 1–8.
10. Harris, D.; Vlok, J.; van Niekerk, A. Regional mapping of spekboom canopy cover using very high resolution aerial imagery. *J. Appl. Remote Sens.* **2018**, *12*, 046022. [[CrossRef](#)]
11. Zhang, L.; Verma, B. Class-Semantic Textons with Superpixel Neighborhoods for Natural Roadside Vegetation Classification. In Proceedings of the IEEE 2015 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Adelaide, Australia, 23–25 November 2015; pp. 1–8.
12. Yuan, X.; Tian, J.; Cerra, D.; Meynberg, O.; Kempf, C.; Reinartz, P. Tree Species Classification by Fusing of Very Highresolution Hyperspectral Images and 3K-DSM. In Proceedings of the IEEE 2018 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), Amsterdam, The Netherlands, 23–26 September 2018; pp. 1–5.
13. Feng, Q.; Liu, J.; Gong, J. UAV remote sensing for urban vegetation mapping using random forest and texture analysis. *Remote Sens.* **2015**, *7*, 1074–1094. [[CrossRef](#)]
14. Zhang, X.; Sun, Y.; Shang, K.; Zhang, L.; Wang, S. Crop classification based on feature band set construction and object-oriented approach using hyperspectral images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 4117–4128. [[CrossRef](#)]
15. Yang, P.; Hou, Z.; Liu, X.; Shi, Z. Texture feature extraction of mountain economic forest using high spatial resolution remote sensing images. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 3156–3159.
16. Oldeland, J.; Große-Stoltenberg, A.; Naftal, L.; Strohbach, B. The potential of UAV derived image features for discriminating savannah tree species. In *The Roles of Remote Sensing in Nature Conservation*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 183–201.
17. Kwak, G.H.; Park, N.W. Impact of texture information on crop classification with machine learning and UAV images. *Appl. Sci.* **2019**, *9*, 643. [[CrossRef](#)]
18. Campos, Y.; Rodner, E.; Denzler, J.; Sossa, H.; Pajares, G. Vegetation segmentation in cornfield images using Bag of Words. In *International Conference on Advanced Concepts for Intelligent Vision Systems*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 193–204.

19. Suh, H.K.; Hofstee, J.W.; IJsselmuiden, J.; van Henten, E.J. Sugar beet and volunteer potato classification using Bag-of-Visual-Words model, Scale-Invariant Feature Transform, or Speeded Up Robust Feature descriptors and crop row information. *Biosyst. Eng.* **2018**, *166*, 210–226. [[CrossRef](#)]
20. Bailly, A.; Arvor, D.; Chapel, L.; Tavenard, R. Classification of MODIS time series with dense bag-of-temporal-SIFT-words: Application to cropland mapping in the Brazilian Amazon. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 2300–2303.
21. Dominik, W.A. Exploiting the redundancy of multiple overlapping aerial images for dense image matching based digital surface model generation. *Remote Sens.* **2017**, *9*, 490. [[CrossRef](#)]
22. Osińska-Skotak, K.; Bakula, K.; Jełowicki, Ł.; Podkowa, A. Using Canopy Height Model Obtained with Dense Image Matching of Archival Photogrammetric Datasets in Area Analysis of Secondary Succession. *Remote Sens.* **2019**, *11*, 2182. [[CrossRef](#)]
23. Fang, L.; Li, S.; Kang, X.; Benediktsson, J.A. Spectral–spatial classification of hyperspectral images with a superpixel-based discriminative sparse model. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 4186–4201. [[CrossRef](#)]
24. Zhang, X.; Chew, S.E.; Xu, Z.; Cahill, N.D. SLIC superpixels for efficient graph-based dimensionality reduction of hyperspectral imagery. In *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XXI*; International Society for Optics and Photonics: Bellingham, WA, USA, 2015; Volume 9472, p. 947209.
25. Li, J.; Zhang, H.; Zhang, L. Efficient superpixel-level multitask joint sparse representation for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 5338–5351.
26. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [[CrossRef](#)] [[PubMed](#)]
27. Beucher, S. Use of watersheds in contour detection. In *International Workshop on Image Processing*; CCETT: Rennes, France, 1979.
28. Yao, J.; Boben, M.; Fidler, S.; Urtasun, R. Real-time coarse-to-fine topologically preserving segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2947–2955.
29. Lazebnik, S.; Schmid, C.; Ponce, J. A sparse texture representation using local affine regions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1265–1278. [[CrossRef](#)] [[PubMed](#)]
30. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
31. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
32. Vedaldi, A.; Fulkerson, B. VLFeat: An Open and Portable Library of Computer Vision Algorithms. 2008. Available online: <http://www.vlfeat.org/> (accessed on 13 August 2020).
33. Wang, Z.; Fan, B.; Wu, F. Local intensity order pattern for feature description. In Proceedings of the IEEE 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 603–610.
34. Lazebnik, S.; Schmid, C.; Ponce, J. A sparse texture representation using affine-invariant regions. In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 18–20 June 2003; Volume 2, p. II.
35. Varma, M.; Zisserman, A. A statistical approach to texture classification from single images. *Int. J. Comput. Vis.* **2005**, *62*, 61–81. [[CrossRef](#)]
36. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [[CrossRef](#)]
37. Perronnin, F.; Sánchez, J.; Mensink, T. Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 143–156.
38. Jegou, H.; Perronnin, F.; Douze, M.; Sánchez, J.; Perez, P.; Schmid, C. Aggregating local image descriptors into compact codes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 1704–1716. [[CrossRef](#)]
39. Tong, X.; Xie, H.; Weng, Q. Urban land cover classification with airborne hyperspectral data: What features to use? *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2013**, *7*, 3998–4009. [[CrossRef](#)]

40. Ghamisi, P.; Yokoya, N.; Li, J.; Liao, W.; Liu, S.; Plaza, J.; Rasti, B.; Plaza, A. Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 37–78. [CrossRef]
41. Ghamisi, P.; Maggiori, E.; Li, S.; Souza, R.; Tarabla, Y.; Moser, G.; De Giorgi, A.; Fang, L.; Chen, Y.; Chi, M.; et al. New frontiers in spectral-spatial hyperspectral image classification: The latest advances based on mathematical morphology, Markov random fields, segmentation, sparse representation, and deep learning. *IEEE Geosci. Remote Sens. Mag.* **2018**, *6*, 10–43. [CrossRef]
42. Plaza, A.; Benediktsson, J.A.; Boardman, J.W.; Brazile, J.; Bruzzone, L.; Camps-Valls, G.; Chanussot, J.; Fauvel, M.; Gamba, P.; Gualtieri, A.; others. Recent advances in techniques for hyperspectral image processing. *Remote Sens. Environ.* **2009**, *113*, S110–S122. [CrossRef]
43. Ghamisi, P.; Plaza, J.; Chen, Y.; Li, J.; Plaza, A.J. Advanced spectral classifiers for hyperspectral images: A review. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–32. [CrossRef]
44. ROSIS. Hyperspectral Remote Sensing Scenes. 2013. Available online: http://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes (accessed on 13 August 2020).
45. Micasense RedEdge multispectral camera. Available online: <https://micasense.com/rededge-mx/> (accessed on 13 August 2020).
46. Bascoy, P.G.; Garea, A.S.; Heras, D.B.; Argüello, F.; Ordóñez, A. Texture-based analysis of hydrographical basins with multispectral imagery. In *Remote Sensing for Agriculture, Ecosystems, and Hydrology XXI*; International Society for Optics and Photonics: Bellingham, WA, USA, 2019; Volume 11149, p. 111490Q.
47. Tong, X.Y.; Xia, G.S.; Lu, Q.; Shen, H.; Li, S.; You, S.; Zhang, L. Learning transferable deep models for land-use classification with high-resolution remote sensing images. *arXiv* **2018**, arXiv:1807.05713.
48. He, L.; Li, J.; Liu, C.; Li, S. Recent advances on spectral-spatial hyperspectral image classification: An overview and new guidelines. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 1579–1597. [CrossRef]
49. Pontius Jr, R.G.; Millones, M. Death to Kappa: Birth of quantity disagreement and allocation disagreement for accuracy assessment. *Int. J. Remote Sens.* **2011**, *32*, 4407–4429. [CrossRef]
50. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 27:1–27:27. Available online: <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (accessed on 13 August 2020). [CrossRef]
51. Bradski, G. The OpenCV Library. *Dr. Dobb's J. Softw. Tools* **2000**, *25*, 120–125.
52. López-Fandiño, J.; Quesada-Barriuso, P.; Heras, D.B.; Argüello, F. Efficient ELM-based techniques for the classification of hyperspectral remote sensing images on commodity GPUs. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 2884–2893. [CrossRef]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).