Classification of Hyperspectral Remote Sensing Images With Support Vector Machines

Farid Melgani, *Member, IEEE,* and Lorenzo Bruzzone, *Senior Member, IEEE*

***Abstract*—This paper addresses the problem of the classification of hyperspectral remote sensing images by support vector machines (SVMs). First, we propose a theoretical discussion and experimental analysis aimed at understanding and assessing the potentialities of SVM classifiers in hyperdimensional feature spaces. Then, we assess the effectiveness of SVMs with respect to conventional feature-reduction-based approaches and their performances in hypersubspaces of various dimensionalities. To sustain such an analysis, the performances of SVMs are compared with those of two other nonparametric classifiers (i.e., radial basis function neural networks and the K-nearest neighbor classifier). Finally, we study the potentially critical issue of applying binary SVMs to multiclass problems in hyperspectral data. In particular, four different multiclass strategies are analyzed and compared: the one-against-all, the one-against-one, and two hierarchical tree-based strategies. Different performance indicators have been used to support our experimental studies in a detailed and accurate way, i.e., the classification accuracy, the computational time, the stability to parameter setting, and the complexity of the multiclass architecture. The results obtained on a real Airborne Visible/Infrared Imaging Spectroradiometer hyperspectral dataset allow to conclude that, whatever the multiclass strategy adopted, SVMs are a valid and effective alternative to conventional pattern recognition approaches (feature-reduction procedures combined with a classification method) for the classification of hyperspectral remote sensing data.**

***Index Terms*—Classification, feature reduction, Hughes phenomenon, hyperspectral images, multiclass problems, remote sensing, support vector machines (SVMs).**

I. Introduction

REMOTE sensing images acquired by multispectral sensors, such as the widely used Landsat Thematic Mapper (TM) sensor, have shown their usefulness in numerous earth observation (EO) applications. In general, the relatively small number of acquisition channels that characterizes multispectral sensors may be sufficient to discriminate among different land-cover classes (e.g., forestry, water, crops, urban areas, etc.). However, their discrimination capability is very limited when different types (or conditions) of the same species (e.g., different types of forest) are to be recognized. Hyperspectral sensors can be used to deal with this problem. These sensors are characterized by a very high spectral resolution that usually results in hundreds of observation channels. Thanks to these channels, it is possible to address various additional applications requiring very high discrimination capabilities in the spectral domain (including material quantification and target detection). From a methodological viewpoint, the automatic analysis of hyperspectral data is not a trivial task. In particular, it is made complex by many factors, such as: 1) the large spatial variability of the hyperspectral signature of each land-cover class; 2) atmospheric effects; and 3) the curse of dimensionality. In the context of supervised classification, one of the main difficulties is related to the small ratio between the number of available training samples and the number of features. This makes it impossible to obtain reasonable estimates of the class-conditional hyperdimensional probability density functions used in standard statistical classifiers. As a consequence, on increasing the number of features given as input to the classifier over a given threshold (which depends on the number of training samples and the kind of classifier adopted), the classification accuracy decreases (this behavior is known as the Hughes phenomenon [1]).

Much work has been carried out in the literature to overcome this methodological issue. Four main approaches can be identified: 1) regularization of the sample covariance matrix; 2) adaptive statistics estimation by the exploitation of the classified (semilabeled) samples; 3) preprocessing techniques based on feature selection/extraction, aimed at reducing/transforming the original feature space into another space of a lower dimensionality; and 4) analysis of the spectral signatures to model the classes.

The first approach uses the multivariate normal (Gaussian) probability density model, which is a widely accepted statistical model for optically remotely sensed data. For each information class, such a model requires the correct estimation of *first-* and *second*-order statistics. In the presence of an unfavorable ratio between the number of available training samples and features, the common way of estimating the covariance matrix may lead to inaccurate estimations (that may make it impossible to invert the covariance matrix in maximum-likelihood (ML) classifiers). Several alternatives and improved covariance matrix estimators have been proposed to reduce the variance of the estimate for limited training samples [2], [3]. The main problem involved by improved covariance estimators is the risk that the estimated covariance matrices overfit the few available training samples and lead to a poor approximation of statistics for the whole image to be classified.

The second approach to overcome the Hughes phenomenon proposes to use in an iterative way the semilabeled samples obtained after classification in order to enhance statistics estimation and to improve classification accuracy. Samples are initially

Manuscript received November 4, 2003; revised May 16, 2004. This work was supported by the Italian Ministry of Education, Research and University (MIUR).

The authors are with the Department of Information and Communication Technologies, University of Trento, I-38050 Trento, Italy (e-mail: melgani@dit.unitn.it; lorenzo.bruzzone@ing.unitn.it).

Digital Object Identifier 10.1109/TGRS.2004.831865

0196-2892/04$20.00 © 2004 IEEE

classified by using the available training samples. Then, the classified samples, together with the training ones, are exploited iteratively to update the class statistics and, accordingly, the results of the classification up to convergence [4], [5]. The process of integration between these two typologies of samples (i.e., the training and the semilabeled samples) is carried out by the expectation–maximization (EM) algorithm, which represents a general and powerful solution to the problem of ML estimation of statistics in the presence of incomplete data [6], [7]. The main advantage of this approach is that it fits the true class distributions better, since a larger portion of the image (available with no extra cost) contributes to the estimation process. The main problems related to this second approach are two: 1) it is demanding from the computational point of view and 2) it requires that the initial class model estimated from the training samples should match well enough the unlabeled samples in order to avoid divergence of the estimation process and, accordingly, to improve the accuracy of the model parameter estimation.

In order to overcome the problem of the curse of dimensionality, the third approach proposes to reduce the dimensionality of the feature space by means of feature selection or extraction techniques. Feature-selection techniques perform a reduction of spectral channels by selecting a representative subset of original features. This can be done following: 1) a selection criterion and 2) a search strategy. The former aims at assessing the discrimination capabilities of a given subset of features according to statistical distance measures among classes (e.g., Bhattacharyya distance, Jeffries–Matusita distance, and the transformed divergence measure [8], [9]). The latter plays a crucial role in hyperdimensional spaces, since it defines the optimization approach necessary to identify the best (or a good) subset of features according to the used selection criterion. Since the identification of the optimal solution is computationally unfeasible, techniques that lead to suboptimal solutions are normally used. Among the search strategies proposed in the literature, it is worth mentioning the basic sequential forward selection (SFS) [10], the more effective sequential forward floating selection [11], and the steepest ascent (SA) techniques [12]. The feature-extraction approach addresses the problem of feature reduction by transforming the original feature space into a space of a lower dimensionality, which contains most of the original information. In this context, the decision boundary feature extraction (DBFE) method [13] has proved to be a very effective method, capable of providing a minimum number of transformed features that achieve good classification accuracy. However, this feature-extraction technique suffers from high computational complexity, which makes it often unpractical. This problem can be overcome by coupling with the projection pursuit (PP) algorithm [14], which plays the role of a preprocessor to the DBFE by applying a preliminary limited reduction of the feature space with (hopefully) an almost negligible information loss. An alternative feature-extraction method, whose class-specific nature makes it particularly attractive, was proposed by Kumar *et al.* [15]. It is based on a combination of subsets of (highly correlated) adjacent bands into fewer features by means of top-down and bottom-up algorithms. In general, it is evident that even if feature-reduction techniques take care of limiting the loss of information, this loss is often unavoidable and may have a negative impact on classification accuracy.

Finally, the approach inherited from spectroscopic methods in analytical chemistry to deal with hyperspectral data is worth mentioning. The idea behind this approach is that of looking at the response from each pixel in the hyperspectral image as a *one*-dimensional spectral signal (signature). Each information class is modeled by some descriptors of the shape of its spectra [16], [17]. The merit of this approach is that it significantly simplifies the formulation of the hyperspectral data classification problem. However, additional work is required to find out appropriate shape descriptors capable of capturing the spectral shape variability related to each information class accurately.

Other methods also exist that are not included in the group of the four main approaches discussed above. In particular, it is interesting to mention the method based on the combination of different classifiers [18] and that based on cluster-space representation [19].

Recently, particular attention has been dedicated to support vector machines (SVMs) for the classification of multispectral remote sensing images [20]–[22]. SVMs have often been found to provide higher classification accuracies than other widely used pattern recognition techniques, such as the maximum likelihood and the multilayer perceptron neural network classifiers. Furthermore, SVMs appear to be especially advantageous in the presence of heterogeneous classes for which only few training samples are available. In the context of hyperspectral image classification, some pioneering experimental investigations preliminarily pointed out the effectiveness of SVMs to analyze hyperspectral data directly in the hyperdimensional feature space, without the need of any feature-reduction procedure [23]–[26]. In particular, in [24], the authors found that a significant improvement of classification accuracy can be obtained by SVMs with respect to the results achieved by the basic minimal-distance-to-means classifier and those reported in [3]. In order to show its relatively low sensitivity to the number of training samples, the accuracy of the SVM classifier was estimated on the basis of different proportions between the number of training and test samples. As will be explained in the following section, this mainly depends on the fact that SVMs implement a classification strategy that exploits a margin-based "geometrical" criterion rather than a purely "statistical" criterion. In other words, SVMs do not require an estimation of the statistical distributions of classes to carry out the classification task, but they define the classification model by exploiting the concept of margin maximization. The growing interest in SVMs [27]–[30] is confirmed by their successful implementation in numerous other pattern recognition applications such as biomedical imaging [31], image compression [32], and three-dimensional object recognition [33]. Such an interest is justified by three main general reasons: 1) their intrinsic effectiveness with respect to traditional classifiers, which results in high classification accuracies and very good generalization capabilities; 2) the limited effort required for architecture design (i.e., they involve few control parameters); and 3) the possibility of solving the learning problem according to linearly constrained quadratic programming (QP) methods (which have been studied intensely in the scientific literature). However, a major drawback of SVMs is that, from a theoretical point of view, they were originally developed to solve binary classification problems. This drawback becomes even more evident when dealing with data acquired from hyperspectral sensors, since they are intrinsically designed to discriminate among a broad range of land-cover classes that may be very similar from a spectral viewpoint. The implementation of SVMs in multiclass classification problems can be approached in two ways [23], [24], [34], [35]. The first consists of defining an architecture made up of an ensemble of binary classifiers. The decision is then taken by combining the partial decisions of the single members of the ensemble. The second is represented by SVMs formulated directly as a multiclass optimization problem. Because of the number of classes that are to be discriminated simultaneously, the number of parameters to be estimated increases considerably in a multiclass optimization formulation. This renders the method less stable and, accordingly, affects the classification performances in terms of accuracy. For this reason, multiclass optimization has not been as successful as the approach based on the two-class optimization.

In this paper, we present a theoretical discussion and an accurate experimental analysis that aim: 1) at assessing the properties of SVM classifiers in hyperdimensional feature spaces and 2) at evaluating the impact of the multiclass problem involved by SVM classifiers when applied to hyperspectral data by comparing different multiclass strategies. With regard to the experimental part of the first objective, assessment of SVM effectiveness is carried out through two different experiments. In the first, we propose to compare the performances of SVMs with those of two other nonparametric classifiers applied directly to the original hyperdimensional feature space: the radial basis function neural network, which is another kernel-based classification method (like SVMs) that uses a different classification strategy based on a "statistical" rather than a "geometrical" criterion; and the K-nearest neighbors classifier, which is widely used in pattern recognition as a reference classification method. The second experiment consists of a comparison of SVMs with the classical classification approach adopted for hyperspectral data, i.e., a conventional classifier combined with a feature-reduction technique. This also allows to assess the performances of SVMs in hypersubspaces of various dimensionalities. As regards the second objective of this work, four different multiclass strategies are analyzed and compared. In particular, the widely used one-against-all and one-against-one strategies are considered. In addition, two strategies based on the hierarchical tree approach are investigated. The experimental studies were carried out on the basis of hyperspectral images acquired by the Airborne Visible/Infrared Imaging Spectroradiometer (AVIRIS) sensor in June 1992 on the Indian Pines area (Indiana) [36]. Different performance indicators are used to support our experimental analysis, namely, the classification accuracy, the computational time, the stability to parameter setting, and the complexity of the multiclass architecture adopted. Experimental results confirm the significant superiority of the SVM classifiers in the context of hyperspectral data classification over the conventional classification methodologies, whatever the multiclass strategy adopted to face the multiclass dilemma.

The rest of this paper is organized in four sections. Section II recalls the mathematical formulation of SVMs and discusses their potential properties in hyperspectral feature spaces. Section III describes different strategies that can be used to solve multiclass problems with binary SVMs and that are adopted in the experiments to assess the impact of the multiclass problem in a hyperdimensional context. Section IV deals with the experimental phase of the work. Finally, Section V summarizes the observations and concluding remarks to complete this paper.

II. SVM Classification Approach

A. SVM Mathematical Formulation

1) Linear SVM: Linearly Separable Case: Let us consider a supervised binary classification problem. Let us assume that the training set consists of N vectors from the d-dimensional feature space $\mathbf{x_i} \in \Re^d$ $(i = 1, 2, \ldots, N)$. A target $y_i \in \{-1, +1\}$ is associated to each vector $\mathbf{x_i}$. Let us assume that the two classes are linearly separable. This means that it is possible to find at least one hyperplane (linear surface) defined by a vector $\mathbf{w} \in \Re^d$ (normal to the hyperplane) and a bias $b \in \Re$ that can separate the two classes without errors. The membership decision rule can be based on the function $\mathrm{sgn}[\mathrm{f}(\mathbf{x})]$, where $\mathrm{f}(\mathbf{x})$ is the discriminant function associated with the hyperplane and defined as

$$\mathrm{f}(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b. \tag{1}$$

In order to find such a hyperplane, one should estimate $\mathbf{w}$ and b so that

$$y_i(\mathbf{w} \cdot \mathbf{x_i} + b) > 0, \qquad \text{with } i = 1, 2, \ldots, N. \tag{2}$$

The SVM approach consists in finding the optimal hyperplane that maximizes the distance between the closest training sample and the separating hyperplane. It is possible to express this distance as equal to $1/\|\mathbf{w}\|$ with a simple rescaling of the hyperplane parameters $\mathbf{w}$ and b such that

$$\min_{i=1,2,\ldots,N} y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1. \tag{3}$$

The geometrical margin between the two classes is given by the quantity $2/\|\mathbf{w}\|$. The concept of margin is central in the SVM approach, since it is a measure of its generalization capability. The larger the margin, the higher the expected generalization [27].

Accordingly, it turns out that the optimal hyperplane can be determined as the solution of the following convex quadratic programming problem:

$$\begin{cases} \text{minimize: } \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{subject to: } y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1, \qquad i = 1, 2, \ldots, N \end{cases}. \tag{4}$$

This classical linearly constrained optimization problem can be translated (using a Lagrangian formulation) into the following dual problem:

$$\begin{cases} \text{maximize: } \displaystyle\sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (\mathbf{x_i} \cdot \mathbf{x}_j) \\ \text{subject to: } \sum_{i=1}^{N} \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0, \qquad i = 1, 2, \ldots, N. \end{cases} \tag{5}$$

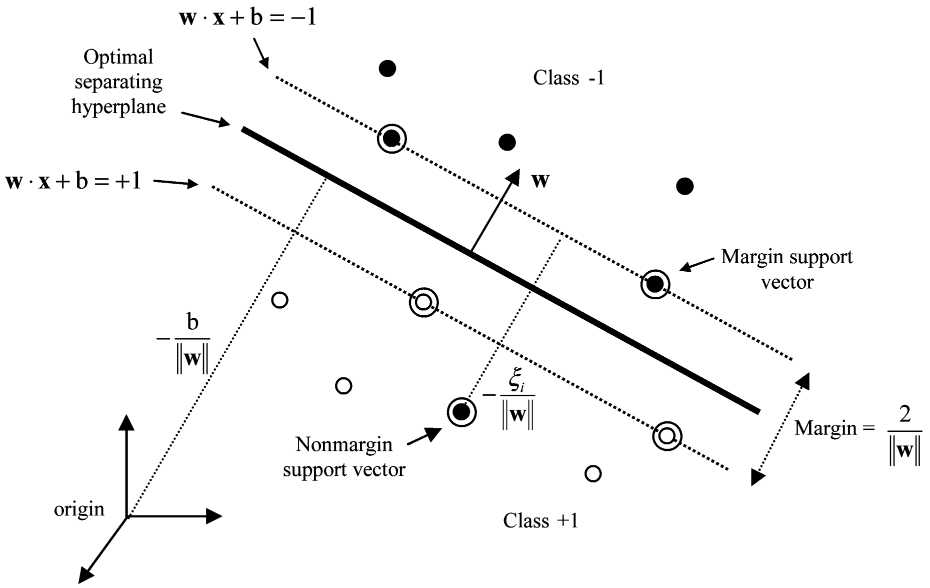
w · x + b = −1
Optimal separating hyperplane
Class -1
w · x + b = +1
w
Margin support vector
$-\frac{b}{\|\mathbf{w}\|}$
$-\frac{\xi_i}{\|\mathbf{w}\|}$
Margin = $\frac{2}{\|\mathbf{w}\|}$
Nonmargin support vector
origin
Class +1

Fig. 1. Optimal separating hyperplane in SVMs for a linearly nonseparable case. White and black circles refer to the classes "+1" and "−1," respectively. Support vectors are indicated by an extra circle.

The Lagrange multipliers α_i's $(i = 1, 2, \ldots, N)$ expressed in (5) can be estimated using quadratic programming (QP) methods [27]. The discriminant function associated with the optimal hyperplane becomes an equation depending both on the Lagrange multipliers and on the training samples, i.e.,

$$\mathrm{f}(\mathbf{x}) = \sum_{i \in S} \alpha_i y_i (\mathbf{x_i} \cdot \mathbf{x}) + b \tag{6}$$

where S is the subset of training samples corresponding to the nonzero Lagrange multipliers α_i's. It is worth noting that the Lagrange multipliers effectively weight each training sample according to its importance in determining the discriminant function. The training samples associated to nonzero weights are called *support vectors*. These lie at a distance exactly equal to $1/\|\mathbf{w}\|$ from the optimal separating hyperplane.

2) Linear SVM: Linearly Nonseparable Case: The SVM formulation described in the previous subsection holds only if data are linearly separable. Such an optimistic condition is difficult to satisfy in the classification of real data. In order to handle nonseparable data, the concept of optimal separating hyperplane has been generalized as the solution that minimizes a cost function that expresses a combination of two criteria: margin maximization (as in the case of linearly separable data) and error minimization (to penalize the wrongly classified samples). The new cost function is defined as

$$\Psi(\mathbf{w}, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + \mathrm{C}\sum_{i=1}^{N} \xi_i \tag{7}$$

where the ξ_i's are the so-called *slack variables* introduced to account for the nonseparability of data, and the constant C represents a regularization parameter that allows to control the penalty assigned to errors. The larger the C value, the higher the penalty associated to misclassified samples. The minimization of the cost function expressed in (7) is subject to the following constraints:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} + b) \geq 1 - \xi_i, \qquad i = 1, 2, \ldots, N \tag{8}$$

$$\xi_i \geq 0, \qquad i = 1, 2, \ldots, N. \tag{9}$$

It is worth noting that, in the nonseparable case, two kinds of support vectors coexist: 1) margin support vectors that lie on the hyperplane margin and 2) nonmargin support vectors that fall on the "wrong" side of this margin (Fig. 1).

3) Nonlinear SVM: Kernel Method: A natural way to improve further the separation between two information classes consists in generalizing the above method to the category of nonlinear discriminant functions. Accordingly, one may think of mapping the data through a proper nonlinear transformation $\Phi(\cdot)$ into a higher dimensional feature space $\Phi(\mathbf{x}) \in \Re^{d'}$ $(d' > d)$, where a separation between the two classes can be looked for following the method described in the previous subsections, i.e., by means of an optimal hyperplane defined by a normal vector $\mathbf{w} \in \Re^{d'}$ and a bias $b \in \Re$. To identify the latter, one should solve a dual problem such as the one defined in (5) for the linearly separable case by replacing the inner products in the original space $(\mathbf{x}_i \cdot \mathbf{x}_j)$ with inner products in the transformed space $[\Phi(\mathbf{x_i}) \cdot \Phi(\mathbf{x}_j)]$. At this point, the main problem consists of the explicit computation of $\Phi(\mathbf{x})$, which can prove expensive and at times unfeasible. The kernel method provides an elegant and effective way of dealing with this problem. Let us consider a kernel function that satisfies the condition stated in Mercer's theorem so as to correspond to some type of inner product in the transformed (higher) dimensional feature space [27, pp. 423–424], i.e.,

$$K(\mathbf{x_i}, \mathbf{x}) = \Phi(\mathbf{x_i}) \cdot \Phi(\mathbf{x}). \tag{10}$$

This kind of kernel function allows to simplify the solution of the dual problem considerably, since it avoids the computation of the inner products in the transformed space $[\Phi(\mathbf{x_i}) \cdot \Phi(\mathbf{x}_j)]$, i.e., as in

$$\begin{cases} \text{maximize: } \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x_i}, \mathbf{x}_j) \\ \text{subject to: } \sum_{i=1}^{N} \alpha_i y_i = 0 \quad \text{and} \quad 0 \le \alpha_i \le \mathrm{C}, \\ \qquad\qquad\qquad\qquad\qquad\qquad i = 1, 2, \ldots, N. \end{cases} \tag{11}$$

The final result is a discriminant function $\mathrm{f}(\mathbf{x})$ conveniently expressed as a function of the data in the original (lower) dimensional feature space

$$\mathrm{f}(\mathbf{x}) = \sum_{i \in S} \alpha_{\mathbf{i}} y_i K(\mathbf{x_i}, \mathbf{x}) + b. \tag{12}$$

The shape of the discriminant function depends on the kind of kernel functions adopted. A common example of kernel type that fulfills Mercer's condition is the Gaussian radial basis function

$$K(\mathbf{x_i}, \mathbf{x}) = \exp(-\gamma \|\mathbf{x_i} - \mathbf{x}\|^2) \tag{13}$$

where γ is a parameter inversely proportional to the width of the Gaussian kernel. Another extensively used kernel is the polynomial function of order p expressed as

$$K(\mathbf{x_i}, \mathbf{x}) = [\mathbf{x_i} \cdot \mathbf{x} + \mathbf{1}]^p. \tag{14}$$

It is worth underlining that the kernel-based implementation of SVMs involves the problem of the selection of multiple parameters, including the kernel parameters (e.g., the γ and p parameters for the Gaussian and polynomial kernels, respectively) and the regularization parameter C. Recently, two interesting automatic techniques have been developed to deal with this issue [37], [38]. They are based on the idea of estimating the parameter values so that: 1) they maximize the margin; and 2) they minimize the estimate of the expected generalization error. The latter is expressed in analytical form by the well-known leave-one-out (LOO) procedure. Optimization of the parameters is then carried out using a gradient descent search over the space of the parameters.

Since a detailed analysis of the theory of SVMs is beyond the scope of this paper, we refer the reader to [27]–[30] for greater detail on SVMs.

B. SVMs in Hyperspectral Feature Spaces

Unlike traditional learning techniques, SVMs do not depend explicitly on the dimensionality of input spaces. They solve classical statistical problems such as pattern recognition, regression, and density estimation in high-dimensional spaces [27]. In greater detail, as stated in the previous subsection, the input feature space is mapped by a kernel transformation into a higher dimensional space, where it is expected to find a linear separation that maximizes the margin between the two classes. In order to appreciate the potentialities of SVMs in high-dimensional spaces, it is useful to recall the statistical and geometrical properties of the data in such spaces.

First, in a hyperspectral space, normally distributed samples (a reasonable assumption for optically remotely sensed data) tend to fall toward the tails of the density function with virtually no samples falling in the central region [39]. This can be illustrated by a simple geometric example [40]. Let us consider the ratio R_V between the volume of a sphere of radius R and one of a cube defined in the interval $[-R, R]$ in the d-dimensional space. It is equal to

$$R_V = \frac{\pi^{d/2}}{d 2^{d-1} \Gamma(d/2)} \tag{15}$$

where $\Gamma(\cdot)$ represents the well-known gamma function. From (15), it easy to show that the higher the dimensionality of the space, the lower the volume ratio. Accordingly, the volume of a hypercube is almost concentrated in its corners. In other words, turning back to our classification problem, the increase in dimensionality makes the space almost empty and results in a "centrifuge" effect such that data have a tendency to concentrate close to the tails of the distribution where they are very likely to be in proximity of decision boundaries between the information classes. This statistical property is of interest potentially to pattern recognition approaches, such as SVMs, that define discriminant functions on the basis of samples situated near the decision boundaries, since the presence of a larger number of samples in this region allows to generate more accurate and reliable discriminant functions.

In the second place, it is well-known that as the dimensionality of the data increases, the distances between the samples (and consequently between the information classes) increase [41]. In this situation, local neighborhoods are almost certainly empty, requiring the bandwidth of estimation to be large and producing the effect of losing accuracy in density estimation for a statistical classifier [39]. On the contrary, the "geometrical" nature of SVMs results in a methodology that is not aimed at estimating the statistical distributions of classes over the entire hyperdimensional space. Indeed, SVMs are inspired by the following idea:

> *If you possess a limited amount of information to solve a problem, try solving it directly and never solve a more general problem as an intermediate step. The available information may be sufficient for a direct solution, though insufficient to solve a more general intermediate problem.* [27, p. 12]

In other words, SVMs do not involve a density estimation problem that can lead to the Hughes effect, but they directly exploit the geometrical behavior of data (space local emptiness) as they make it more likely to find a decision boundary between classes that results in a small classification error. The above-discussed properties (statistical and geometrical) render SVMs potentially less sensitive to the curse of dimensionality.

Another important aspect to be pointed out is the intrinsic good generalization capability of SVMs, which stems from the selection of the hyperplane that maximizes the geometrical margin between classes. In a hyperspectral context, the maximum margin solution allows to fully exploit the discrimination capability of the relatively few training samples available.

Accordingly, this solution deals with some of the major problems, such as the large spatial variability of the hyperspectral signature of each information class, in the best way in terms of generalization capability, given the limited information present in the training set. However, it is worth noting that to solve the problem of the spatial variability of the hyperspectral signature of classes effectively, good generalization properties of the classifiers should be coupled with other data analysis techniques.

III. SVMs: Multiclass Strategies

As stated in the previous section, SVMs are intrinsically binary classifiers. However, the classification of hyperspectral remote sensing data usually involves the simultaneous discrimination of numerous information classes. In this section, we describe four different strategies of combination of SVMs considered to evaluate the impact of the multiclass problem in the context of hyperspectral data classification. Let $\Omega = \{\omega_1, \omega_2, \ldots, \omega_T\}$ be the set of T possible labels (information classes) associated with the d-dimensional hyperspectral image $\mathbf{X}$ of the study area. In the multiclass case, the problem is to associate to each d-dimensional sample $\mathbf{x}$ the label of the set Ω that optimizes a predefined classification criterion. In order to carry out this task, the general approach adopted in strategies based on binary classifiers consists of: 1) defining an ensemble of binary classifiers; and 2) combining them according to some decision rules.

The definition of the ensemble of binary classifiers involves the definition of a set of *two*-class problems, each modeled with two groups Ω_A and Ω_B of classes ($\Omega_A \subset \Omega$ and $\Omega_B \subset \Omega$). Targets with values $+1$ and -1 are assigned to the samples of Ω_A and Ω_B, respectively, for each SVM. The selection of these subsets depends on the kind of approach adopted to combine the ensemble. Two main approaches can be identified: the "parallel" and the "hierarchical tree-based" approaches. In the following, we describe two multiclass strategies from each approach characterized by different classification complexity and computational cost properties.

A. Parallel Approach

1) One-Against-All Strategy: The one-against-all (OAA) strategy represents the earliest and most common multiclass approach used for SVMs [42]. It involves a parallel architecture made up of T SVMs, one for each class (Fig. 2). Each SVM solves a *two*-class problem defined by one information class (e.g., $\omega_i \in \Omega$) against all the others, i.e.,

$$\begin{cases} \Omega_A = \omega_i \\ \Omega_B = \Omega - \omega_i. \end{cases} \tag{16}$$

The "winner-takes-all" rule is used for the final decision, i.e., the winning class is the one corresponding to the SVM with the highest output (discriminant function value).

2) One-Against-One Strategy: The main problem of the OAA strategy is that the discrimination between an information class and all the others often leads to the estimation of complex discriminant functions. In addition, a problem with strongly unbalanced prior probabilities should be solved by each SVM. The idea behind the one-against-one (OAO) strategy is that of a different reasoning, in which simple classification tasks are made possible thanks to a parallel architecture made up of a large number of SVMs [23], [43]. The OAO strategy involves $T(T-1)/2$ SVMs, which model all possible pairwise classifications. In this case, each SVM carries out a binary classification in which two information classes ω_i and ω_j $(\omega_i \in \Omega, \omega_j \in \Omega, i \neq j)$ are analyzed against each other by means of a discriminant function $\mathrm{f}_{ij}(\mathbf{x})$. Consequently, the grouping becomes

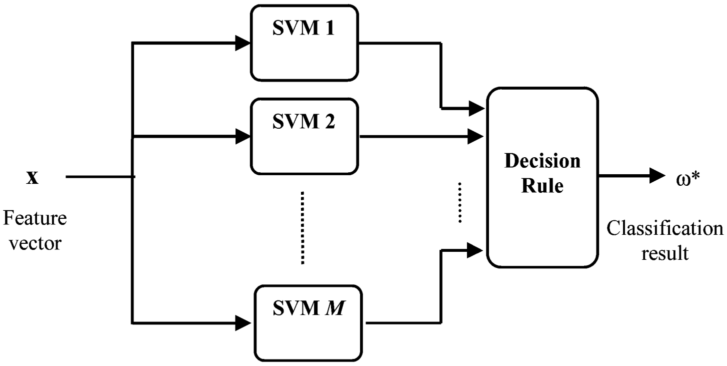

Fig. 2. Block diagram of a parallel architecture for solving multiclass problems with binary SVMs. In the OAA strategy, M is equal to T (i.e., the number of information classes). By contrast, the OAO strategy involves a larger number of SVMs and M is given by $T(T-1)/2$.

$$\begin{cases} \Omega_A = \omega_i \\ \Omega_B = \omega_j. \end{cases} \tag{17}$$

Before the decision process, it is necessary to compute for each class $\omega_i \in \Omega$ a score function $S_i(\mathbf{x})$, which sums the favorable and unfavorable votes expressed for the considered class

$$S_i(\mathbf{x}) = \sum_{\substack{j=1 \\ j \neq i}}^{T} \mathrm{sgn}\{\mathrm{f}_{ij}(\mathbf{x})\}. \tag{18}$$

The final decision in the OAO strategy is taken on the basis of the "winner-takes-all" rule, which corresponds to the following maximization

$$\omega^* = \arg\max_{i=1,\ldots,T} \{S_i(\mathbf{x})\}. \tag{19}$$

Sometimes, conflict situations may occur between two different classes characterized by the same score. Such ambiguities can be solved by selecting the class with the highest prior probability.

B. Hierarchical Tree-Based Approach

The idea of representing the data analysis process with a hierarchical tree is not new and has been under study in many pattern recognition application areas. Tree-based classifiers have represented an interesting and effective way to structure and solve complex classification problems [44]–[47]. The organization of information into a hierarchical tree allows to achieve a faster processing capability and, at times, a higher accuracy of analysis. This is mainly explained by the fact that the nodes of the tree carry out very focused tasks, meaningless when taken individually but meaningful when taken as a whole. Turning back to our problem, the binary hierarchical tree (BHT) can be seen as an alternative to the OAA and the OAO strategies, since it allows to reach a good tradeoff between the number of SVMs to be used and the complexity of the task assigned to each of them. Furthermore, the BHT does not implement a global decision scheme after evaluating the local decisions as in the OAA and OAO strategies. Indeed, the final decision is implicitly made after running through the tree and reaching one of its terminal nodes.

Many BHT strategies have been proposed in the literature. In this paper, we investigate two different binary tree hierarchies aimed at reducing the computational load required by the OAA and OAO strategies, especially in the operational classification phase (the off-line training phase is less critical from the viewpoint of the computational time). This can become particularly important when large hyperspectral images are considered. As described in the following, both trees exploit the prior probabilities of the classes to define the hierarchy of binary SVMs. It is worth noting that alternative strategies that also exploit the underlying affinities among the individual classes to define the binary trees (like in [46]) could be considered.

1) BHT-Balanced Branches Strategy: In the BHT-balanced branches (BHT-BB) strategy, the tree is defined in such a way that each node (SVM) discriminates between two groups of classes Ω_A and Ω_B with similar cumulative prior probabilities. Fig. 3(a) shows an example of tree that can be found with the BHT-BB strategy for a general T-class classification problem. The algorithm that implements the BHT-BB strategy is described as follows:

Step 0: **Root Node**

—Set level index $k = 0$

—Divide Ω into two groups $\Omega^k_{A,0}$ and $\Omega^k_{B,0}$ such that $\sum_{\omega_i \in \Omega^k_{A,0}} P(\omega_i) \approx \sum_{\omega_j \in \Omega^k_{B,0}} P(\omega_j)$

Step 1: **k-Level Branching**

—For $q = 0, \ldots, 2k - 1$ $(q \geq 0)$

- If $\mathrm{Card}\{\Omega^k_{A,q}\} \geq 2$, divide $\Omega^k_{A,q}$ into two groups $\Omega^{k+1}_{A,2q}$ and $\Omega^{k+1}_{A,2q+1}$ such that $\sum_{\omega_i \in \Omega^{k+1}_{A,2q}} P(\omega_i) \approx \sum_{\omega_j \in \Omega^{k+1}_{A,2q+1}} P(\omega_j)$
- If $\mathrm{Card}\{\Omega^k_{B,q}\} \geq 2$, divide $\Omega^k_{B,q}$ into two groups $\Omega^{k+1}_{B,2q}$ and $\Omega^{k+1}_{B,2q+1}$ such that $\sum_{\omega_i \in \Omega^{k+1}_{B,2q}} P(\omega_i) \approx \sum_{\omega_j \in \Omega^{k+1}_{B,2q+1}} P(\omega_j)$

—Set $k = k + 1$

Step 2: **Stop Condition**

—If $\exists \Omega^k_{A,q}$ or $\Omega^k_{B,2q}$ such that $\mathrm{Card}\{\Omega^k_{A,q}\} \geq 2$ or $\mathrm{Card}\{\Omega^k_{B,q}\} \geq 2$ with $(q = 0, \ldots, 2^k - 1)$, go to *Step 1.* Otherwise, Stop.

2) BHT-One Against All Strategy: The second binary tree-based hierarchy, called BHT-one against all (BHT-OAA), represents a simplification of the OAA strategy obtained through its implementation in a hierarchical context. To this end, we propose to define the tree in such a way that each node discriminates between two groups of classes Ω_A and Ω_B, where Ω_B represents the information class with the highest prior probability among those belonging to $\Omega_A \cup \Omega_B$. This kind of hierarchy leads to a tree with only one single branch as depicted in Fig. 3(b). The algorithm of the BHT-OAA strategy is drawn up in the following:

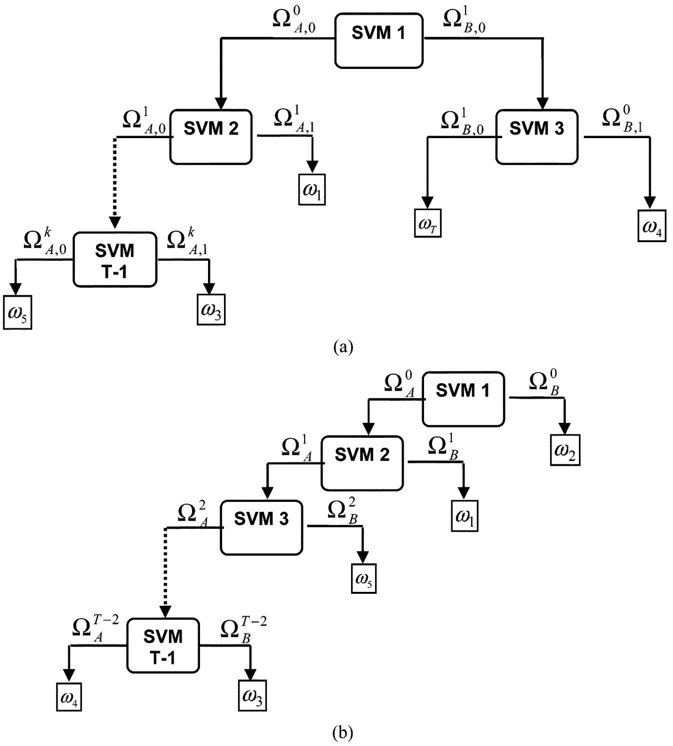

Fig. 3. Examples of BHTs for a T-class classification problem. (a) BHT-BB. (b) BHT-OAA.

TABLE I
Number of Training and Test Samples Used in the Experiments

CLASS	TRAINING	TEST
ω_1 – Corn-no till	742	692
ω_2 – Corn-min till	442	392
ω_3 - Grass/Pasture	260	237
ω_4 - Grass/Trees	389	358
ω_5 - Hay-windrowed	236	253
ω_6 - Soybean-no till	487	481
ω_7 - Soybean-min till	1245	1223
ω_8 - Soybean-clean till	305	309
ω_9 – Woods	651	643
Total	**4757**	**4588**

Step 0: **Root Node**

—Set level index $k = 0$

—Divide Ω into two groups Ω^k_A and Ω^k_B such that $P(\Omega^k_B)_{\Omega^k_B \in \Omega} = \max_{\omega_j \in \Omega}\{P(\omega_j)\}$ and $\Omega^k_A = \Omega - \Omega^k_B$

Step 1: **k-Level Branching**

—Divide Ω^k_A into two groups Ω^{k+1}_A and Ω^{k+1}_B such that $P(\Omega^{k+1}_B)_{\Omega^{k+1}_B \in \Omega^k_A} = \max_{\omega^k_{j \in \Omega_A}}\{P(\omega_j)\}$ and $\Omega^{k+1}_A = \Omega^k_A - \Omega^{k+1}_B$

—Set $k = k + 1$

Step 2: **Stop Condition**

—If $\mathrm{Card}\{\Omega^k_A\} \geq 2$, go to *Step 1.* Otherwise, Stop.

It is worth noting that both BHT strategies allow to reduce the number of required SVMs from T and $T(T-1)/2$, respectively, for the OAA and OAO strategies, to $T - 1$. Since the classification time depends linearly on the number of SVMs and since

TABLE II
BEST OVERALL AND CLASS-BY-CLASS ACCURACIES, AND COMPUTATIONAL TIMES ACHIEVED ON THE TEST SET BY THE DIFFERENT CLASSIFIERS IN THE ORIGINAL HYPERSPECTRAL SPACE

METHOD	**CLASSIFICATION ACCURACY [%]**										**TIME [s]**
	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	ω_9	**OA**	
SVM-Linear	89.02	69.13	94.51	98.60	100	75.47	83.48	83.17	99.22	**87.10**	**40342**
SVM-RBF	91.47	87.76	94.94	98.88	100	88.57	91.25	95.79	99.38	**93.42**	**2702**
K-nn classifier	96.73	61.16	86.59	80.46	99.60	98.88	90.72	65.82	74.42	**83.94**	**2618**
RBF classifier	98.44	74.11	88.47	79.83	99.21	98.04	91.98	73.72	80.06	**86.99**	**4743**

classification tasks of medium complexity are assigned to the SVMs of the tree, we expect a lower classification time required by the two BHT-based strategies with respect to both the OAO and, especially, the standard OAA strategies.

TABLE III
ANALYSIS OF THE STABILITY OF THE OVERALL CLASSIFICATION ACCURACY AND OF THE COMPUTATIONAL TIME VERSUS THE SETTING OF THE PARAMETERS OF THE DIFFERENT CLASSIFIERS

METHOD	**PARAMETER RANGE**	**OVERALL ACCURACY [%]**		**MEAN TOTAL TIME [s]**
		Mean	**Variance**	
SVM-Linear	$C \in [1, 100]$	85.38	4.94	20785
SVM-RBF	$C \in [1, 100]$; $\gamma = 1$	92.64	0.84	1695
SVM-RBF	$\gamma \in [0.1, 3]$; $C = 40$	92.51	0.50	2412
K-nn classifier	$K \in [1, 25]$	82.42	1.56	2630
RBF classifier	N° clusters $\in [20, 200]$	85.59	1.12	1505

IV. EXPERIMENTAL RESULTS

A. Dataset Description and Experiment Design

The hyperspectral dataset used in our experiments is a section of a scene taken over northwest Indiana's Indian Pines by the AVIRIS sensor in 1992 [36]. From the 220 spectral channels acquired by the AVIRIS sensor, 20 channels were discarded because affected by atmospheric problems. From the 16 different land-cover classes available in the original ground truth, seven were discarded, since only few training samples were available for them (this makes the experimental analysis more significant from the statistical viewpoint). The remaining nine land-cover classes were used to generate a set of 4757 training samples (used for learning the classifiers) and a set of 4588 test samples (exploited for assessing their accuracies) (see Table I). The experiments were run on a Sun Ultra 80 workstation.

The experimental analysis was organized into three main experiments. The first aims at analyzing the effectiveness of SVMs in classifying hyperspectral images directly in the original hyperdimensional feature space. A comparison with two other nonparametric classifiers is provided as well as an assessment of the stability of these three classification methods versus the setting of their parameters. In the second experiment, SVMs are compared with the classical approach adopted for hyperspectral data classification, that is a conventional pattern recognition system made up of a classification method combined with a feature-reduction technique. In these two experiments, we adopted the most popular multiclass strategy used for SVMs, that is the OAA strategy. Finally, the third experiment aims at analyzing and comparing the effectiveness of the different multiclass strategies described in the previous section, that is the OAA, OAO, BHT-BB, and BHT-OAA strategies.

B. Results of Experiment 1: Classification in the Original Hyperdimensional Feature Space

SVMs were compared with two widely used nonparametric classifiers: a radial basis functions (RBFs) neural network trained with the technique described in [48] and a conventional K-nearest neighbors (K-nn) classifier. The choice of the RBF classifier is motivated by the fact that it is a kernel-based method (like SVMs), which adopts a different strategy based on a "statistical" (rather than a "geometrical") criterion for defining the discriminant hyperplane in the transformed kernel space. The K-nn classifier was considered in our experiments, since it represents a reference classification method in pattern recognition. However, it is worth noting that we expect it to be sensitive to the curse of dimensionality. For both classifiers, different trials were carried out to determine empirically the best related parameters, namely, the number of nodes in the hidden layer and the variable K, respectively.

In the experiments, we considered two different kinds of SVMs: a linear SVM (SVM-Linear) which corresponds to an SVM without kernel transformation, and a nonlinear SVM based on Gaussian radial basis kernel functions (SVM-RBF). For both SVMs, the regularization parameter C must be estimated, since data are not ideally separable. In addition, the nonlinear SVM requires the determination of the width parameter γ of the Gaussian radial basis kernels, which tunes the smoothing of the discriminant function. For the considered dataset, the best values of the parameter C were 50 and 40 for the linear and nonlinear SVMs, respectively. The optimal kernel width parameter γ of the nonlinear SVM was found equal to 0.25. These values were estimated empirically on the basis of the available training samples.

The results in terms of classification accuracy and computational time provided by the different classifiers are summarized in Table II. The nonlinear SVM exhibited the best Overall Accuracy (OA), i.e., the best percentage of correctly classified pixels among all the test pixels considered, with a gain of 6.32%, 6.43%, and 9.48% over the linear SVM, the RBF, and the K-nn classifiers, respectively. In terms of class accuracies, the "corn-min till" class (ω_2) was the most critical. For this class, the nonlinear SVM still exhibited the best accuracy (87.76%), whereas the worst accuracy (61.16%) was obtained by the K-nn classifier. It is worth noting that, since the K-nn classifier is based on counting the number of nearest neighboring training samples, it requires the feature space to be filled in with a significant number of training samples to obtain reliable local estimates of the conditional posterior probabilities of classes. However, in the considered dataset, the small number of training samples (4757) is not sufficient to fill in a proper way the emptiness of the hyperdimensional feature space. This explains the relatively poor classification accuracies of the K-nn classifier. By contrast, SVMs exploit a discriminant model that is defined on the basis of a particular portion of the training samples (support vectors). As explained in Section II and confirmed by the obtained results, the behavior of the class distributions in hyperdimensional spaces makes it more effective to apply techniques that define discriminant functions on the basis of training samples located near the decision boundaries. Concerning computational cost, the nonlinear SVM exhibited a reasonable total computational time (given by the sum of the training and test times) compared to the other three classifiers. It is worth noting that the long computational time required by the linear SVM (40342 [s]) expresses the difficulties encountered by this kind of classifier in the training phase to find a reasonable linear separation between information classes.

In order to assess the robustness of each classifier to the parameter settings, we derived some statistics by looking at the overall accuracy (OA) and at the total computational time as random realizations obtained by varying the parameters in a predefined range of values. The results reported in Table III confirm the superiority of the nonlinear SVM in terms of both mean overall accuracy (92.64% and 92.51% by varying the parameters γ and C, respectively) and in terms of stability (it provided the lowest variances). It is worth noting that the nonlinear SVM is less sensitive to the choice of the kernel width value γ than to the regularization parameter C. The linear SVM showed the worst stability to the parameter C (overall-accuracy variance equals 4.94). This is explained by the fact that a linear separation between classes involves a large number of error samples, which lie on the wrong side of the separating hyperplane. This makes it more difficult to apply the regularization mechanism implemented in the SVM formulation, resulting in significant sensitivity of the classification accuracy to the value of the regularization parameter. Concerning the average total computational times, the obtained results confirm the conclusions drawn above on the basis of the total computational times obtained for the best parameter values of the four considered classifiers.

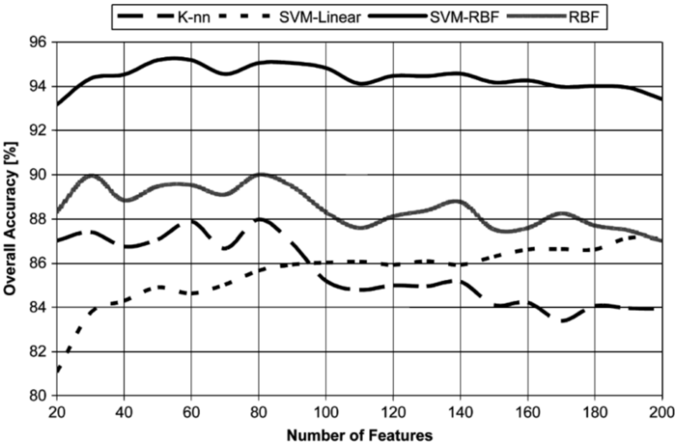

Fig. 4. Overall accuracy versus the number of features obtained on the test set by the four different classifiers considered in our investigation (i.e., linear and nonlinear SVMs, RBF and K-nn classifiers).

TABLE IV
FIRST- AND SECOND-ORDER STATISTICS OF THE OVERALL ACCURACIES OBTAINED ON THE TEST SET BY THE DIFFERENT CLASSIFIERS COMBINED WITH THE SA-BASED FEATURE-SELECTION PROCEDURE FOR A NUMBER OF FEATURES VARYING FROM 20 TO 200 (WITH A STEP OF 10)

METHOD	OVERALL ACCURACY [%]	
	Mean	Variance
SA + SVM-Linear	85.56	2.04
SA + SVM-RBF	94.38	0.30
SA + K-nn classifier	85.60	2.27
SA + RBF classifier	88.49	0.81

C. Results of Experiment 2: Feature Reduction and Classification

As already discussed in Section I, the traditional approach adopted to address the problem of the classification of hyperspectral data consists of two main phases: 1) reducing the dimensionality of the feature space; and 2) applying the resulting subset of features to a conventional classifier. In this experiment, we propose to assess the effectiveness of SVMs with respect to a traditional feature-reduction-based approach and to evaluate their performances in hypersubspaces of various dimensionalities. To this end, we used the Jeffries–Matusita (JM) interclass distance measure [8] and the steepest ascent (SA) search strategy [12] to reduce the original hyperdimensional space into spaces of a lower dimensionality (the number of features was varied from 20 to 200 with a step of 10). The SA technique formulates the problem of defining the subset of features that maximizes the JM distance as a discrete optimization problem in a d-dimensional space, which is viewed as a space of binary strings. It starts with a binary string randomly initialized, and performs an iterative local optimization of the adopted criterion function. At each iteration, the criterion is maximized over a neighborhood of the current solution under a predefined constraint. In our experiments, each subset of selected features was given as input to all four considered classifiers (i.e., linear and nonlinear SVMs, RBF neural networks, and the K-nn classifier). Fig. 4 plots the overall accuracy versus the number of selected features for the four considered classifiers. As can be seen, the obtained results still confirm the strong superiority of nonlinear SVMs over the other classifiers even in lower dimensional feature spaces, with a gain in overall accuracy (averaged over all the subsets of features) of $+8.82\%$, $+8.78\%$, and $+5.89\%$ with respect to the linear SVM, the K-nn, and the RBF neural network classifiers (see Table IV). In order to analyze the sensitivity of each classifier to the Hughes phenomenon, in the same table we reported the variance of the overall accuracy exhibited by each classification method when varying the number of features from 20 to 200. The lowest sensitivity was again obtained by the nonlinear SVM classifier with a sharp reduction of the variance with respect to those achieved by the K-nn, the linear SVM, and the RBF neural network classifiers.

TABLE V

Classification Accuracies Yielded on the Test Set by the Different Classifiers With the Subset of the Best 30 Features Selected According to the SA-Based Feature-Selection Procedure. The Difference in Overall Accuracy (DIFF-OA) for Each Classifier With Respect to the Accuracy Achieved in the Original Hyperdimensional Space Is Also Given

METHOD	CLASSIFICATION ACCURACY [%]										DIFF-OA [%]
	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	ω_9	OA	
SA + SVM-Linear	98.29	73.79	81.11	72.35	100	99.16	89.45	57.4	86.27	**83.74**	**-3.36**
SA + SVM-RBF	99.69	92.23	93.30	91.48	99.6	99.72	97.89	88.52	91.62	**94.35**	**0.93**
SA + K-nn classifier	80.93	71.43	94.52	99.44	99.61	87.53	88.06	71.20	96.42	**87.40**	**3.46**
SA + RBF classifier	83.24	77.30	93.25	97.77	98.03	86.08	90.03	77.67	98.29	**89.95**	**2.96**

Table V reports the overall and class-by-class accuracies obtained for the hypersubspace made up of the best 30 selected features. The choice of this subspace is motivated by the fact that it represents a good compromise between a low dimensionality of the feature space and a high classification accuracy achieved on average by the four classifiers. In particular, one can see the greater capacity of the nonlinear SVMs to recognize each information class, with a gain in the average of the class-by-class accuracies of $+10.69\%$, $+7.21\%$, and $+5.82\%$ with respect to the linear SVM, the K-nn, and the RBF neural network classifiers. In addition, the same table reports the difference in overall accuracy (DIFF-OA) for each classifier with respect to the accuracy achieved in the original hyperdimensional space. It is interesting to note the lower difference (associated with the expected lowest sensitivity to the problem of the curse of dimensionality) achieved by the SVM-RBF classifier (0.93%). The reduction in the number of features involved a decrease in accuracy of 3.36% for the linear SVM classifier. By contrast, significant increases in accuracy of 2.96% and 3.46% were obtained by the conventional K-nn and RBF classifiers, respectively, confirming a relatively high sensitivity to the curse of dimensionality.

In order to analyze the complexity of the decision boundaries produced by the nonlinear SVM classifier, we computed the number of SVs defined in each binary SVM of the OAA architecture in both the original hyperspace and the hypersubspace consisting of the best 30 selected features. These numbers are represented graphically in Fig. 5. It can be observed in general that the numbers of SVs are relatively small, except for the SVM associated with the class ω_3. This suggests that decision boundaries of moderate complexity were enough to discriminate accurately between the information classes. Furthermore, as discussed in Section II-B, an important property related to the "geometrical" nature of SVMs seems confirmed, i.e., that the classification complexity does not depend on the dimension of the feature space, since the number of SVs is almost similar in both the original and the reduced spaces.

D. Results of Experiment 3: SVM and Multiclass Strategies

The third (and last) experiment addressed the application of SVMs to the multiclass problem in the hyperdimensional space. The different multiclass strategies described in Section III (i.e., the OAA, OAO, BHT-BB, and BHT-OAA strategies) were designed and trained using nonlinear SVMs based on the Gaussian radial basis kernel functions. The trees of SVMs defined for the BHT-BB and the BHT-OAA strategies are illustrated in Fig. 6. The class prior probabilities necessary to obtain such trees were computed on the basis of the training set. After the training phase, the four strategies were analyzed and compared based on three parameters: 1) classification accuracy; 2) computational time; and 3) architecture complexity. The obtained results are reported in Tables VI and VII. From the viewpoint of the accuracy, all four strategies resulted in satisfactory results when compared with the two other nonparametric classifiers (i.e., the RBF neural networks and the K-nn classifier). In greater detail, the OAO strategy exhibited the best accuracy with a gain in overall accuracy of $+2.72\%$, $+1.72\%$, and $+0.54\%$ over the BHT-OAA, the BHT-BB and the OAA strategies, respectively. This suggests that the decomposition of the multiclass problem into an ensemble of *two*-class problems of very low-complexity represents an effective way of improving overall discrimination capability. The significant reduction in the complexity of the classification problem assigned to each SVM of the OAO architecture is shown by the very small average number of SVs that characterizes each SVM of the same architecture. Indeed, this number is 130 against 333, 334, and 424 for the BHT-OAA, BHT-BB and the OAA strategies, respectively (Table VII). These values explain also why the time required to train the SVMs of the OAO strategy is the shortest, despite the greater amount of SVMs required by the same strategy (212 [s] against 311 [s], 410 [s], and 2361 [s] to train the BHT-BB, BHT-OAA, and OAA strategies, respectively). It is worth noting that the smallest number of SVs was exhibited by the OAO strategy. Indeed, only nine SVs were necessary to discriminate between the fifth and ninth classes (hay-windrowed and woods, respectively) with an accuracy of 100%. On the other hand, the larger number of SVMs involved in the OAO strategy directly affects the computational time demanded during the classification of test samples (554 [s] against 125 [s], 155 [s] and 341 [s] for the BHT-BB, the

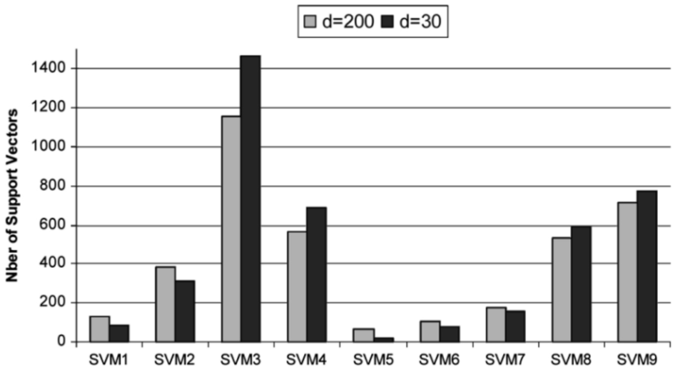

Fig. 5. Number of support vectors that characterize each binary SVM of the multiclass nonlinear SVM classifier (OAA strategy) in both the original hyperspace ($d = 200$) and the hypersubspace made up of the best 30 selected features ($d = 30$).

TABLE VI
OVERALL AND CLASS-BY-CLASS ACCURACIES OBTAINED ON THE TEST SET BY SVMS WITH THE DIFFERENT MULTICLASS STRATEGIES CONSIDERED

Multiclass Strategy	**CLASSIFICATION ACCURACY [%]**									
	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	ω_9	**OA**
OAA	91.47	87.76	94.94	98.88	100	88.57	91.25	95.79	99.38	**93.42**
OAO	90.32	89.54	94.51	99.72	100	88.36	93.54	94.82	99.38	**93.96**
BHT-BB	89.70	87.25	95.78	99.16	100	87.73	91.17	85.11	98.60	**92.24**
BHT-OAA	88.87	85.97	95.36	98.88	100	85.24	87.82	90.62	99.07	**91.24**

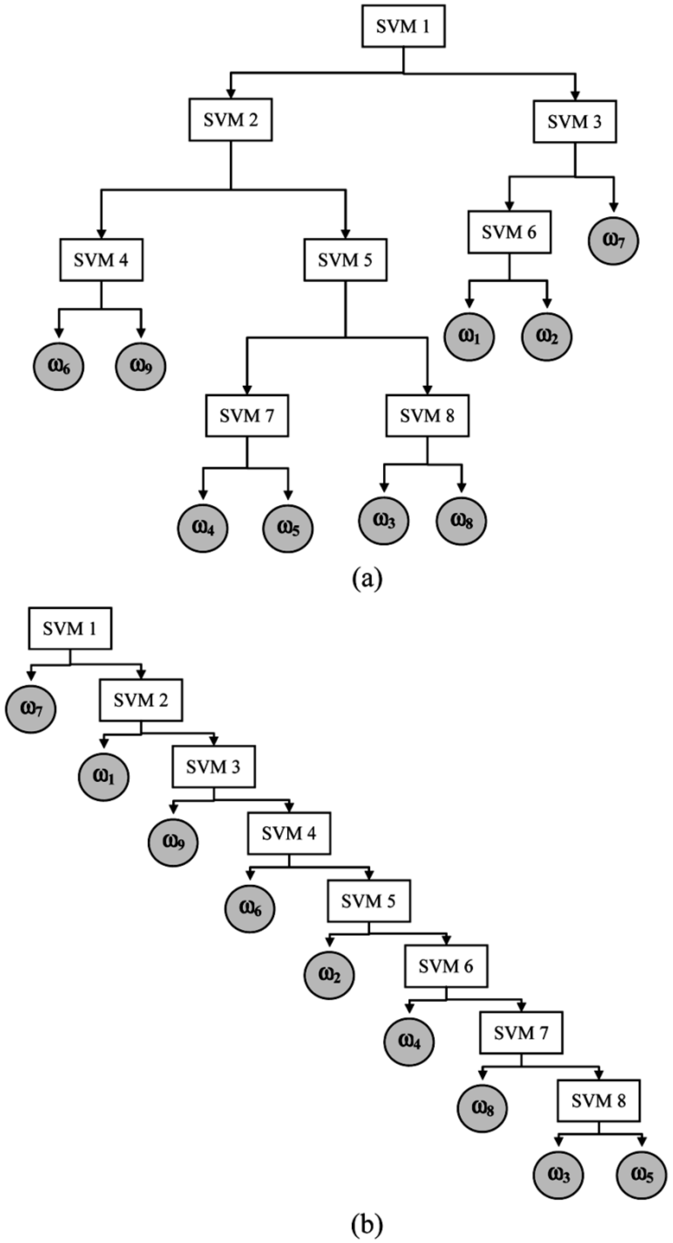

Fig. 6. Hierarchical trees obtained on the considered dataset by (a) the BHT-BB strategy and (b) the BHT-OAA strategy.

TABLE VII
COMPUTATIONAL TIME AND CLASSIFICATION COMPLEXITY ASSOCIATED TO THE DIFFERENT SVM MULTICLASS STRATEGIES CONSIDERED

Multiclass Strategy	**TIME [S]**		**NUMBER OF SVMS**	**number of Support Vectors**		
	Train	**Test**		**Min**	**Max**	**Average**
OAA	2361	341	9	62	1159	**424**
OAO	212	554	36	9	569	**130**
BHT-BB	311	125	8	31	977	**334**
BHT-OAA	410	155	8	45	1270	**333**

BHT-OAA, and the OAA strategies, respectively). Thanks to the small number of required SVMs and to the moderate complexity of the classification tasks assigned to each of them, the two BHT strategies seem particularly interesting in an operative phase involving the classification of large scale images. Table VIII shows the overall accuracies achieved by each SVM involved in both the BHT-BB and the BHT-OAA strategies. It is worth noting that the relatively low accuracy (93.77%) obtained by the first SVM of the BHT-OAA architecture (SVM1) combined with a significant depth of its associated tree (involving a higher risk of error propagation) may explain why this strategy was slightly less accurate than the BHT-BB strategy. In general, from a computational point of view, the two investigated BHT-BB and BHT-OAA strategies proved effective, resulting in a significant decrease in computational time.

V. DISCUSSION AND CONCLUSION

In this paper, we addressed the problem of the classification of hyperspectral remote sensing data using support vector machines. In order to assess the effectiveness of this promising classification methodology, we considered two main objectives. The first was aimed at assessing the properties of SVMs in hyperdimensional spaces and hypersubspaces of various dimensionalities. In this context, the results obtained on the considered dataset allow to identify the following three properties: 1) SVMs are much more effective than other conventional nonparametric classifiers (i.e., the RBF neural networks and the K-nn classifier) in terms of classification accuracy, computational time, and stability to parameter setting; 2) SVMs seem more effective than the traditional pattern recognition approach, which is based on the combination of a feature extraction/selection procedure and a conventional classifier, as implemented in this paper; and 3) SVMs exhibit low sensitivity to the Hughes phenomenon, resulting in an excellent approach to avoid the usually time-consuming phase required by any feature-reduction method. Indeed, as shown in the experiments, the improvement in accuracy obtained on the considered dataset by combining SVMs with a feature-reduction technique is definitely insufficient to justify the use of the latter.

The second objective of the work concerned the assessment of the effectiveness of strategies based on ensembles of binary SVMs used to solve multiclass problems in hyperspectral data. In particular, four different multiclass strategies were investigated and compared. These four strategies differ basically in

TABLE VIII
OVERALL ACCURACY YIELDED ON THE TEST SET BY EACH SINGLE SVM OF THE BHT-BB AND BHT-OAA STRATEGIES

Multiclass	OVERALL ACCURACY [%]							
Strategy	SVM1	SVM2	SVM3	SVM4	SVM5	SVM6	SVM7	SVM8
BHT-BB	95,66	99,25	94,1	100	99,57	98,62	100	98,53
BHT-OAA	93,77	97,74	99,7	99,16	98,9	99,57	99,12	100

the manner in which the classification problem complexity is distributed over the single members (SVMs) of the architecture. Compared with each other, the parallel architectures (OAA and OAO) showed a better discrimination capability than the hierarchical tree-based architectures (BHT-BB and BHT-OAA). This can be explained by the fact that the BHT strategies may involve the risk of propagation of errors, since the final decision is the result of several hierarchical exchanges of partial decisions that may accumulate errors. Accordingly, one may observe that the design of a BHT strategy should favor a large number of ramifications at the expense of a lower ramification depth, to attenuate such a risk. Another reason that justifies the lower discrimination capability of the two proposed BHT strategies can be found in the kind of information used to construct the tree. Indeed, the use of simple information, such as the class prior probabilities, cannot take into proper account the underlying affinities among individual classes (or metaclasses). However, from the viewpoint of computational time, the BHT-BB and BHT-OAA strategies proved the most effective. Consequently, depending on the considered application, the multiclass strategy should be selected according to a proper tradeoff between classification accuracy and computational time. As a final remark, it is important to point out that the classification accuracies exhibited by all four strategies suggest that the multiclass problem does not significantly affect the performances of SVMs in the analysis of hyperspectral data. Indeed, all the strategies exhibited accuracies sharply higher than those of the nonparametric classifiers considered in our experimental analysis.

ACKNOWLEDGMENT

Support by the Italian Ministry of Education, Research and University (MIUR) for this work is gratefully acknowledged. The authors would like to thank D. Landgrebe for providing the AVIRIS data and T. Joachims for supplying the software SVM$^{\text{light}}$ (http://svmlight.joachims.org/) used in the context of this work.

REFERENCES

[1] G. F. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 55–63, 1968.

[2] J. P. Hoffbeck and D. A. Landgrebe, "Covariance matrix estimation and classification with limited training data," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 763–767, July 1996.

[3] S. Tadjudin and D. A. Landgrebe, "Covariance estimation with limited training samples," *IEEE Trans. Geosci. Remote. Sensing*, vol. 37, pp. 2113–2118, July 1999.

[4] Q. Jackson and D. A. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set," *IEEE Trans. Geosci. Remote. Sensing*, vol. 39, pp. 2664–2679, Dec. 2001.

[5] B. M. Shahshahani and D. A. Landgrebe, "The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon," *IEEE Trans. Geosci. Remote. Sensing*, vol. 32, pp. 1087–1095, Sept. 1994.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc.*, vol. 19, pp. 1–38, 1977.

[7] T. K. Moon, "The expectation-maximization algorithm," *Signal Process. Mag.*, vol. 13, pp. 47–60, 1996.

[8] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis*, Berlin, Germany: Springer-Verlag, 1999.

[9] L. Bruzzone, F. Roli, and S. B. Serpico, "An extension to multiclass cases of the Jeffries–Matusita distance," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 1318–1321, Nov. 1995.

[10] J. Kittler, "Feature set search algorithm," in *Pattern Recognition and Signal Processing*, C. H. Chen, Ed. Alphen aan den Rijn, Netherlands: Sijthoff and Noordhoff, 1978, pp. 41–60.

[11] P. Pudil, J. Novovicova, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognit. Lett.*, vol. 15, pp. 1119–1125, 1994.

[12] S. B. Serpico and L. Bruzzone, "A new search algorithm for feature selection in hyperspectral remote sensing images," *IEEE Trans. Geosci. Remote Sensing*, vol. 39, pp. 1360–1367, July 2001.

[13] C. Lee and D. A. Landgrebe, "Feature extraction based on decision boundaries," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 388–400, Apr. 1993.

[14] L. O. Jimenez and D. A. Landgrebe, "Hyperspectral data analysis and feature reduction via projection pursuit," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 2653–2667, Nov. 1999.

[15] S. Kumar, J. Ghosh, and M. M. Crawford, "Best-bases feature extraction algorithms for classification of hyperspectral data," *IEEE Trans. Geosc. Remote. Sensing*, vol. 39, pp. 1368–1379, May 2001.

[16] J. P. Hoffbeck and D. A. Landgrebe, "Classification of remote sensing images having high-spectral resolution," *Remote Sens. Environ.*, vol. 57, pp. 119–126, 1996.

[17] F. Tsai and W. D. Philpot, "A derivative-aided hyperspectral image analysis system for land-cover classification," *IEEE Trans. Geosc. Remote. Sensing*, vol. 40, pp. 416–425, Feb. 2002.

[18] J. A. Benediktsson and I. Kanellopoulos, "Classification of multisource and hyperspectral data based on decision fusion," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 1367–1377, May 1999.

[19] X. Jia and J. A. Richards, "Cluster-space representation of hyperspectral data classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 40, pp. 593–598, Mar. 2002.

[20] L. Hermes, D. Frieauff, J. Puzicha, and J. M. Buhmann, "Support vector machines for land usage classification in landsat TM imagery," in *Proc. IGARSS*, Hamburg, Germany, 1999, pp. 348–350.

[21] F. Roli and G. Fumera, "Support vector machines for remote-sensing image classification," *Proc. SPIE*, vol. 4170, pp. 160–166, 2001.

[22] C. Huang, L. S. Davis, and J. R. G. Townshend, "An assessment of support vector machines for land cover classification," *Int. J. Remote Sens.*, vol. 23, pp. 725–749, 2002.

[23] J. A. Gualtieri and R. F. Cromp, "Support vector machines for hyperspectral remote sensing classification," *Proc. SPIE*, vol. 3584, pp. 221–232, 1998.

[24] J. A. Gualtieri, S. R. Chettri, R. F. Cromp, and L. F. Johnson, "Support vector machine classifiers as applied to AVIRIS data," in *Summaries 8th JPL Airborne Earth Science Workshop*, 1999, JPL Pub. 99-17, pp. 217–227. Online. [Available]: ftp://popo.jpl.nasa.gov/pub/docs/workshops/99_docs/toc.html.

[25] J. A. Gualtieri and S. Chettri, "Support vector machines for classification of hyperspectral data," in *Proc. IGARSS*, Honolulu, HI, 2000, pp. 813–815.

[26] F. Melgani and L. Bruzzone, "Support vector machines for classification of hyperspectral remote-sensing images," in *Proc. IGARSS*, Toronto, ON, Canada, 2002, pp. 506–508.

[27] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[28] C. B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. ACM Workshop Computational Learning Theory*, 1992, pp. 144–152.

[29] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discov.*, vol. 2, pp. 121–167, 1998.

[30] Set of tutorials on SVM's and kernel methods [Online]. Available: http://www.kernel-machines.org/tutorial.html.

[31] I. El-Naqa, Y. Yongyi, M. N. Wernick, N. P. Galatsanos, and R. M. Nishikawa, "A support vector machine approach for detection of microcalcifications," *IEEE Trans. Med. Imag.*, vol. 21, pp. 1552–1563, Dec. 2002.

[32] J. Robinson and V. Kecman, "Combining support vector machine learning with the discrete cosine transform in image compression," *IEEE Trans. Neural Networks*, vol. 14, pp. 950–958, July 2003.

[33] M. Pontil and A. Verri, "Support vector machines for 3D object recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 637–646, June 1998.

[34] D. J. Sebald and J. A. Bucklew, "Support vector machines and the multiple hypothesis test problem," *IEEE Trans. Signal Processing*, vol. 49, pp. 2865–2872, Nov. 2001.

[35] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Networks*, vol. 13, pp. 415–425, Mar. 2002.

[36] AVIRIS NW Indiana's Indian Pines 1992 data set [Online]. Available: ftp://ftp.ecn.purdue.edu/biehl/MultiSpec/92AV3C (original files) and ftp://ftp.ecn.purdue.edu/biehl/PC_MultiSpec/ThyFiles.zip (ground truth).

[37] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, pp. 131–159, 2002.

[38] K.-M. Chung, W.-C. Kao, T. Sun, L.-L. Wang, and C.-J. Lin, "Radius margin bounds for support vector machines with the RBF kernel," *Neural. Comput.*, vol. 15, pp. 2643–2681, 2003.

[39] L. O. Jimenez and D. A. Landgrebe, "Supervised classification in high-dimensional space: Geometrical, statistical, and asymptotic properties of multivariate data," *IEEE Trans. Syst., Man, Cybern. C*, vol. 28, pp. 39–54, Jan. 1998.

[40] M. G. Kendall, *A Course in the Geometry of n-Dimensions*. New York: Hafner, 1961.

[41] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic, 1990.

[42] L. Bottou, C. Cortes, J. Denker, H. Drucker, I. Guyon, L. Jackel, Y. LeCun, U. Muller, E. Sackinger, P. Simard, and V. Vapnik, "Comparison of classifier methods: A case study in handwriting digit recognition," in *Proc. Int. Conf. Pattern Recognition*, 1994, pp. 77–87.

[43] U. H.-G. Kreßel, "Pairwise classification and support vector machines," in *Advances in Kernel Methods: Support Vector Learning*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA: MIT Press, 1999, pp. 255–268.

[44] P. H. Swain and H. Hauska, "The decision tree classifier: Design and potential," *IEEE Trans. Geosci. Electron.*, vol. GE-15, pp. 142–147, 1977.

[45] B. Kim and D. A. Landgrebe, "Hierarchical classifier design in high-dimensional, numerous class cases," *IEEE Trans. Geosci. Remote Sensing*, vol. 29, pp. 518–528, July 1991.

[46] J. T. Morgan, A. Henneguelle, M. M. Crawford, J. Ghosh, and A. Neuenschwander, "Adaptive feature spaces for land cover classification with limited ground truth data," in *Proc. 3rd Int. Workshop on Multiple Classifier Systems—MCS 2002*, Cagliari, Italy, June 2002, pp. 189–200.

[47] M. Datcu, F. Melgani, A. Piardi, and S. B. Serpico, "Multisource data classification with dependence trees," *IEEE Trans. Geosci. Remote Sensing*, vol. 40, pp. 609–617, Mar. 2002.

[48] L. Bruzzone and D. F. Prieto, "A technique for the selection of kernel-function parameters in RBF neural networks for classification of remote-sensing images," *IEEE Trans. Geosci Remote. Sensing*, vol. 37, pp. 1179–1184, Mar. 1999.

Farid Melgani (M'04) received the State Engineer degree in electronics from the University of Batna, Batna, Algeria, in 1994, the M.Sc. degree in electrical engineering from the University of Baghdad, Baghdad, Iraq, in 1999, and the Ph.D. degree in electronic and computer engineering from the University of Genoa, Genoa, Italy, in 2003.

From 1999 to 2002, he cooperated with the Signal Processing and Telecommunications Group, Department of Biophysical and Electronic Engineering, University of Genoa. He is currently an Assistant Professor of telecommunications at the University of Trento, Trento, Italy, where he teaches pattern recognition, radar remote sensing systems, and digital transmission. His research interests are in the area of processing and pattern recognition techniques applied to remote sensing images (classification, multitemporal analysis, and data fusion). He is coauthor of more than 30 scientific publications.

Dr. Melgani served on the Scientific Committee of the SPIE International Conferences on Signal and Image Processing for Remote Sensing VI (Barcelona, Spain, 2000), VII (Toulouse, France, 2001), VIII (Crete, 2002), and IX (Barcelona, Spain, 2003) and is a referee for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

Lorenzo Bruzzone (S'95–M'99–SM'03) received the laurea (M.S.) degree in electronic engineering (summa cum laude) and the Ph.D. degree in telecommunications, both from the University of Genoa, Genoa, Italy, in 1993 and 1998, respectively.

He is currently Head of the Remote Sensing Laboratory in the Department of Information and Communication Technologies at the University of Trento, Trento, Italy. From 1998 to 2000, he was a Postdoctoral Researcher at the University of Genoa. From 2000 to 2001, he was an Assistant Professor at the University of Trento, where he has been an Associate Professor of telecommunications since November 2001. He currently teaches remote sensing, pattern recognition, and electrical communications. His current research interests are in the area of remote sensing image processing and recognition (analysis of multitemporal data, feature selection, classification, data fusion, and neural networks). He conducts and supervises research on these topics within the frameworks of several national and international projects. He is the author (or coauthor) of more than 100 scientific publications, including journals, book chapters, and conference proceedings. He is a referee for many international journals and has served on the Scientific Committees of several international conferences.

Dr. Bruzzone ranked first place in the Student Prize Paper Competition of the 1998 IEEE International Geoscience and Remote Sensing Symposium (Seattle, July 1998). He is the Delegate in the scientific board for the University of Trento of the Italian Consortium for Telecommunications (CNIT) and a member of the Scientific Committee of the India–Italy Center for Advanced Research. He was a recipient of the *Recognition of IEEE Transactions on Geoscience and Remote Sensing Best Reviewers* in 1999 and was a Guest Editor of a Special Issue of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING on the subject of the analysis of multitemporal remote sensing images (November 2003). He was the General Co-chair of the First and Second IEEE International Workshop on the Analysis of Multi-temporal Remote-Sensing Images (Trento, Italy, September 2001—Ispra, Italy, July 2003). Since 2003, he has been the Chair of the SPIE Conference on Image and Signal Processing for Remote Sensing (Barcelona, Spain, September 2003—Maspalomas, Gran Canaria, September 2004). He is an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. He is a member of the International Association for Pattern Recognition (IAPR) and of the Italian Association for Remote Sensing (AIT).