Kernel-Based Framework for Multitemporal and Multisource Remote Sensing Data Classification and Change Detection

Gustavo Camps-Valls, *Senior Member, IEEE*, Luis Gómez-Chova, Jordi Muñoz-Marí, José Luis Rojo-Álvarez, *Member, IEEE*, and Manel Martínez-Ramón, *Senior Member, IEEE*

***Abstract*—The multitemporal classification of remote sensing images is a challenging problem, in which the efficient combination of different sources of information (e.g., temporal, contextual, or multisensor) can improve the results. In this paper, we present a general framework based on kernel methods for the integration of heterogeneous sources of information. Using the theoretical principles in this framework, three main contributions are presented. First, a novel family of kernel-based methods for multitemporal classification of remote sensing images is presented. The second contribution is the development of nonlinear kernel classifiers for the well-known difference and ratioing change detection methods by formulating them in an adequate high-dimensional feature space. Finally, the presented methodology allows the integration of contextual information and multisensor images with different levels of nonlinear sophistication. The binary support vector (SV) classifier and the one-class SV domain description classifier are evaluated by using both linear and nonlinear kernel functions. Good performance on synthetic and real multitemporal classification scenarios illustrates the generalization of the framework and the capabilities of the proposed algorithms.**

***Index Terms*—Change detection, information fusion, kernel methods, multisource, multitemporal classification, support vector (SV) domain description (SVDD), support vector machine (SVM).**

I. Introduction

THE PROBLEMS of multitemporal image classification and change detection are highly relevant in many domains [1], particularly in the field of remote sensing [2]–[4]. Typical applications consider updating digital remote sensing databases, following multiseasonal crop cover phenology or the automatic detection of growing urbanization. With the increasing multitemporal and multisensor data available from remote sensing platforms, the efficient fusion and exploitation of this unprecedented wealth of data is a critical issue at present. Many methods have been proposed to tackle the problem of multitemporal classification, in general, and of change detection, in particular. However, so far, there is no general methodological framework for combining different sources of information that involve different sensors, time instants, and spatial or contextual extracted features efficiently and with tunable complexity. This is the focus of this paper.

On the one hand, multitemporal classification algorithms classify pixels by learning the changing mapping between dates in a temporal sequence of images. When a labeled image dataset is available, supervised classifiers can yield improved performance over unsupervised approaches. Other advantages are their capability to explicitly detect land-cover transitions, robustness to different atmospheric and light conditions at the acquisition times, and their demonstrated ability to process multisensor/multisource images [5]. Many multitemporal supervised methods have been used during the last years, such as evidence reasoning [6], generalized least squares [7], or neural networks [8]–[10]. Nevertheless, several problems are identified in the presented strategies. First, classifiers are, in general, sensitive to the high dimension of pixels in hyperspectral images or to the high input space generated by putting together multisensor features at different temporal instants. This "stacked approach" increases the well-known "curse of dimensionality," which has lately been alleviated by using support vector (SV) machines (SVMs) in this setting [11], [12]. Second, classifiers can suffer from false-alarm detection rates when the contextual or textural information of the change is not considered. This is an important issue because in practice, the user is ultimately interested in very precisely detecting both the position and the spatial extent of the class(es) of interest. Multitemporal and multiband synthetic aperture radar (SAR) classification of urban areas using spatial analysis has been successfully addressed with both statistical and neural approaches [13] and at feature and pixel information levels [14]. Third, and very importantly, most methods do not consider the (potentially nonlinear) cross information among pixels (and among features) at different time instants. In fact, the learning paradigm is often violated because the classifier is trained and tested with data coming from different distributions due to differences in atmospheric and light conditions, sensor drifts, etc. To address this problem, several strategies have been presented. A dynamic approach to link hidden Markov random fields at different dates is used in [15], whereas in [16], scenes are classified by a fuzzy fusion of the spatial and spectral information, whereas the temporal information is obtained from transition probabilities. Last, but not least, it should be noted that, in most cases, only two dates are considered to illustrate

G. Camps-Valls, L. Gómez-Chova, and J. Muñoz-Marí are with the Departament d'Enginyeria Electrònica, Escola Tècnica Superior d'Enginyeria, Universitat de València, 46100 València, Spain (e-mail: gustavo.camps@uv.es).

J. L. Rojo-Álvarez is with the Departamento de Teorìa de la Señal y Comunicaciones, Universidad Rey Juan Carlos, 28943 Madrid, Spain (e-mail: joseluis.rojo@urjc.es).

M. Martínez-Ramón is with the Departamento de Teorìa de la Señal y Comunicaciones, Universidad Carlos III de Madrid, 28911 Madrid, Spain (e-mail: manel@tsc.uc3m.es).

method capabilities, and thus, the performance of the algorithms for long-term operational studies is unclear. In [17], a methodology that encompasses the use of both temporal and contextual information was presented for the classification of the long time series of satellite data. The method was based on krigging-integrated variograms and Gaussian maximum-likelihood classification and showed very good results. All the aforementioned shortcomings can be simultaneously alleviated with the adequate formulation of novel kernel methods [18], [19] that we will focus on in this paper.

On the other hand, change detection can be viewed as a particular case of the multitemporal image classification problem. Two main approaches are followed in the literature, namely: 1) postclassification comparison and 2) preclassification enhancement. In the first case, the images of two dates are independently classified and coregistered, and an algorithm is used to identify those pixels whose predicted labels change between dates. In the second case, a single classification is performed on the combined image dataset for the two dates. The postclassification approach can fail, considering that it relies on the accuracy of each independent classifier. Both approaches, however, inherit all the aforementioned problems of the multitemporal image classification scenario. Classical change detection techniques are based on multidate principal-component analysis, temporal image subtraction or ratioing, change-vector analysis, clustering, or cross-correlation analysis [2]. The main ideas underlying these techniques are visualizing, analyzing, or computing the differences among the sample distributions for two dates in a low-dimensional subspace (e.g., two principal components, few bands, etc.). If one detects changes in a (representative-enough) space, then, one can analyze the nature of the change by inspecting the spectral signatures involved in it. All these techniques are unsupervised in the sense that they do not require a labeled image at time t_1 to learn from and then to extrapolate to the subsequent image at time t_2. The early approaches considered intuitive threshold-based image differencing or ratioing; however, this was readily demonstrated to be inefficient. The selection of suitable thresholds under Bayesian criteria have been largely studied [5], [20]. Recently, the Kittler–Illingworth minimum-error thresholding algorithm attained good results for unsupervised SAR change detection [21], whereas a fuzzy hidden Markov chain model has been successfully used in combination with the ratio approach [22]. Furthermore, a full methodology for change detection based on the analysis of the difference vectors in the polar domain has been presented [23], and a method based on studying the evolution of the local statistics using the Kullback–Leibler divergence has been proposed with good results [24]. It is only recently that authors have turned to kernel-based methods for change detection. In [25], a semisupervised oil slick detection was proposed by using the SV domain description (SVDD) classifier in the wavelet domain of SAR images, and in [26], the SV classifier (SVC) for abrupt change detection was presented for detecting buried landmines from ground-penetrating radar data. Not only do these kernel methods allow large-margin classifications, but they also intrinsically match the well-known nonlinear nature of the change [27]. However, none of them has been particularly redesigned to consider cross relations between time instants or to efficiently include contextual and multisource data in the classifier.

In this scenario and attending to the previously identified problems, we present a novel methodological framework that allows us to develop a family of nonlinear classifiers for multitemporal, contextual, and multisource image classification and change detection. In particular, the methods are developed under the framework of kernel methods [18], [19], which has demonstrated good results in high-dimensional image classification [28]–[30]. Certainly, these are important characteristics of kernel methods, which become strictly necessary when multitemporal, multisensor, and contextual features are extracted and need to be combined. In addition, we derive specific formulations for dealing with the peculiarities of the change detection problem by proposing the "difference" and the "ratioing" of images in the kernel space. The proposed methodological framework also serves to efficiently integrate different information sources such as optical and SAR data.

The rest of this paper is organized as follows. Section II reviews both the framework of the kernel methods, paying special attention to their general properties, and the formulations of the standard SVC and SVDD supervised classifiers. Section III introduces the novel methodological framework for information fusion based on the kernels. The proposed family of kernels is used in Section IV to develop specific kernel classifiers for multitemporal classification and change detection. Section V exploits the presented framework to integrate the contextual, textural, and multisource information in the classifier. Section VI presents the experimental results in both long time series of simulated images and challenging real scenarios of multitemporal image classification and change detection. Finally, Section VII draws some concluding remarks.

II. Kernel Methods and Data Classification

Kernel methods offer a general framework for machine learning problems (classification, clustering, regression, density estimation, and visualization) with heterogeneous types of data, such as time series, images, strings, or objects [18], [19]. In this section, we briefly review the main properties of Mercer's kernels and the standard formulations for the binary SVC and for the one-class SVDD used in this paper, which have demonstrated excellent characteristics in problems with a low number of high-dimensional training samples [30], [31].

A. Background on Kernel Methods

When using linear algorithms, a well-established theory and efficient methods are often available. Kernel methods exploit this fact by embedding the dataset S defined over the input or attribute space $\mathcal{X}(S \subseteq \mathcal{X})$ into a higher (possibly infinite) dimensional Hilbert $\mathcal{H}$ or "feature" space, and then, they build a linear algorithm therein, resulting in an algorithm which is nonlinear with respect to the input data space. The mapping function is denoted as $\phi : \mathcal{X} \rightarrow \mathcal{H}$. Although linear algorithms will benefit from this mapping because of the higher dimensionality of the feature space, the computational load would dramatically increase because we should compute sample coordinates in that high-dimensional space. This computation is avoided through the use of the kernel trick by which, if an algorithm can be expressed with dot products in the input space, its (nonlinear) kernel version only needs the dot products among mapped samples. The kernel methods compute the similarity between

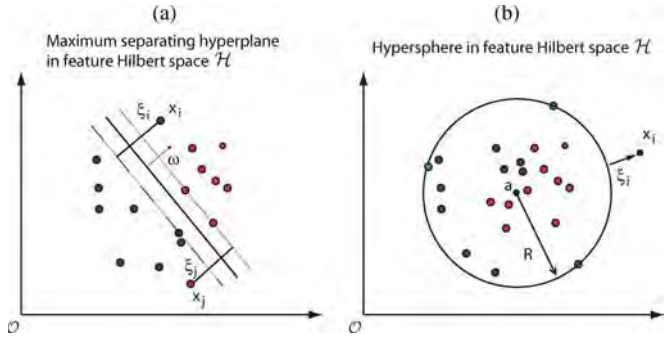

Fig. 1. Kernel classifiers. (a) SVC. Linear decision hyperplanes in a nonlinearly transformed space, where the slack variable ξ_i is included to deal with errors. (b) SVDD. The hypersphere containing the (colored) target data is described by center $\mathbf{a}$ and radius R. Samples in the boundary and outside the ball are (green) unbounded and bounded SVs, respectively.

training samples $S = \{\mathbf{x}_i\}_{i=1}^n$ using pairwise inner products between mapped samples, and thus, the so-called kernel matrix $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = \langle \boldsymbol{\phi}(\mathbf{x}_i), \boldsymbol{\phi}(\mathbf{x}_j) \rangle$ contains all the necessary information to perform many classical linear algorithms in the feature space.

B. SVC

Given a labeled training dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathbb{R}^N$ and $y_i \in \{-1, +1\}$, and given a nonlinear mapping $\boldsymbol{\phi}(\cdot)$, the SVC method solves

$$\min_{\mathbf{w},\xi_i,b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^n \xi_i \right\} \tag{1}$$

constrained to

$$y_i \left(\langle \boldsymbol{\phi}(\mathbf{x}_i), \mathbf{w} \rangle + b \right) \geq 1 - \xi_i \qquad \forall i = 1, \ldots, n \tag{2}$$
$$\xi_i \geq 0 \qquad \forall i = 1, \ldots, n \tag{3}$$

where $\mathbf{w}$ and b define a linear classifier in the feature space and ξ_i represents the positive slack variables which enable dealing with permitted errors [Fig. 1(a)]. The appropriate choice of nonlinear mapping $\boldsymbol{\phi}$ guarantees that the transformed samples are more likely to be linearly separable in the feature space [32]. The parameter C controls the generalization capabilities of the classifier, and it must be selected by the user. The primal problem (1) is solved by using its dual-problem counterpart [18], and the decision function for any test vector $\mathbf{x}_*$ is given by

$$f(\mathbf{x}_*) = \operatorname{sgn}\left(\sum_{i=1}^n y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_*) + b \right) \tag{4}$$

where α_i represents the Lagrange multipliers corresponding to the constraints in (2) and b can be easily computed from a few SVs, which are those training samples $\mathbf{x}_i$ with nonzero Lagrange multipliers α_i [18].

C. SVDD Classifier

A different problem statement for classification is given by the SVDD algorithm. Now, let $\{\mathbf{x}_i\}_{i=1}^n$ be a dataset belonging to a given class of interest. The purpose is to find a minimum-volume hypersphere in a high-dimensional feature space $\mathcal{H}$, with radius $R > 0$ and center $\mathbf{a} \in \mathcal{H}$, which contains most of these data objects [33] [Fig. 1(b)]. Considering that the training set may contain outliers, we introduce a set of slack variables $\xi_i \geq 0$, and the problem then becomes

$$\min_{R,\mathbf{a}} \left\{ R^2 + C\sum_{i=1}^n \xi_i \right\} \tag{5}$$

constrained to

$$\|\boldsymbol{\phi}(\mathbf{x}_i) - \mathbf{a}\|^2 \leq R^2 + \xi_i, \qquad \forall i = 1, \ldots, n \tag{6}$$
$$\xi_i \geq 0 \qquad \forall i = 1, \ldots, n \tag{7}$$

where parameter C controls the trade-off between the volume of the hypersphere and the permitted errors. One can define the rejection fraction parameter to be tuned as $\nu = 1/nC$, as noted in [34].

The dual functional leads to a quadratic programming problem that yields a set of α_i corresponding to the Lagrange multipliers of (6). These ones allow us to calculate the distance from a test point to the center $R(\mathbf{x}_*)$, i.e.,

$$R(\mathbf{x}_*) = K(\mathbf{x}_*, \mathbf{x}_*) - 2\sum_{i=1}^n K(\mathbf{x}_i, \mathbf{x}_*) + \sum_{i,j=1}^n K(\mathbf{x}_i, \mathbf{x}_j) \tag{8}$$

which is to be compared against the ratio R. Unbounded SVs are those samples $\mathbf{x}_i$ satisfying $0 < \alpha_i < C$, whereas bounded SVs are samples whose associated $\alpha_i = C$, and they are considered outliers.

D. Kernel Functions and Basic Properties

The bottleneck for any kernel method is the definition of a kernel mapping function $\boldsymbol{\phi}$ that accurately reflects the similarity among samples. However, not all metric distances are permitted. In fact, valid kernels are only those that fulfill Mercer's theorem [35], [36], and the most common ones are the linear $K(\mathbf{x}, \mathbf{z}) = \langle \mathbf{x}, \mathbf{z} \rangle$, the polynomial $K(\mathbf{x}, \mathbf{z}) = (\langle \mathbf{x}, \mathbf{z} \rangle + 1)^d$, $d \in \mathbb{Z}^+$, and the radial basis function (RBF) $K(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2 / 2\sigma^2)$, $\sigma \in \mathbb{R}^+$.

Some properties of Mercer's kernels that are relevant for this paper are the following.

Property 1: Let K_1 and K_2 be two Mercer's kernels on $S \times S$, $\mathbf{A}$ a symmetric positive semidefinite $n \times n$ matrix, and $\mu > 0$. Then, the following kernels

$$K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) + K_2(\mathbf{x}, \mathbf{z}) \tag{9}$$
$$K(\mathbf{x}, \mathbf{z}) = \mu K_1(\mathbf{x}, \mathbf{z}) \tag{10}$$
$$K(\mathbf{x}, \mathbf{z}) = \mathbf{x}^\top \mathbf{A} \mathbf{z} \tag{11}$$

are valid Mercer's kernels. □

Therefore, one can design kernels by summing up (weighted) valid kernels. This intuitive idea is formally expressed in the following sections under a general-purpose kernel-based framework for information fusion.

III. Kernel-Based Information Fusion

In this section, we present a novel methodological framework for information fusion based on kernels. As a general-purpose

TABLE I
FORMULATION OF THE COMPOSITE-KERNEL FRAMEWORK: INPUT VECTOR DEFINITION AND THE ASSOCIATED NONLINEAR KERNEL TRANSFORMATIONS

Kernel	Input vector	Feature vector	Kernel Eq.
Stacked	$\mathbf{x}_s \equiv \bigcup_{p=1}^{P} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x}_s)$	(13)
Direct summation	$\mathbf{x} = \bigcup_{p-1}^{P} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}) = \bigcup_{p-1}^{P} \left\{ \mathbf{A}_p \boldsymbol{\varphi}_p(\mathbf{x}^{(p)}) \right\}$	(15)
Weighted summation	$\mathbf{x} = \bigcup_{p=1}^{P} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}) = \bigcup_{p=1}^{P} \sqrt{\mu_p} \left\{ \mathbf{A}_p \boldsymbol{\varphi}_p(\mathbf{x}^{(p)}) \right\}$	(16)
Cross information	$\mathbf{x} = \bigcup_{p=1}^{P} \mathbf{x}^{(p)} \bigcup_{p=1}^{P} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}) = \bigcup_{p=1}^{P} \left\{ \mathbf{A}_p \boldsymbol{\varphi}_p(\mathbf{x}^{(p)}) \right\} \bigcup_{p=1}^{P} \left\{ \mathbf{B}_p \boldsymbol{\varphi}_p(\mathbf{x}^{(p)}) \right\}$	(17)
Difference	$\mathbf{x} = \bigcup_{p=t_0-1}^{t_0} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}^{(t_o)}) = \mathbf{A}_{t_o} \boldsymbol{\varphi}(\mathbf{x}^{(t_o)}) - \mathbf{A}_{t_o-1} \boldsymbol{\varphi}(\mathbf{x}^{(t_o-1)})$	(23)
Ratioing	$\mathbf{x} = \bigcup_{p=t_0-1}^{t_0} \mathbf{x}^{(p)}$	$\boldsymbol{\phi}(\mathbf{x}^{(t_o)}) = \frac{\{(\sqrt{\gamma}\mathbf{A}_{t_o} \boldsymbol{\varphi}(\mathbf{x}^{(t_o)}))^\top, (\mathbf{A}_{t_o-1} \boldsymbol{\varphi}(\mathbf{x}^{(t_o-1)}))^\top\}^\top}{\sqrt{\langle \mathbf{A}_{t_o} \boldsymbol{\varphi}(\mathbf{x}^{(t_o)}), \mathbf{A}_{t_o} \boldsymbol{\varphi}(\mathbf{x}^{(t_o)}) \rangle}}$	(24)

methodology, it constitutes a set of basic tools and associated properties for kernel-based model building, as follows.

Definition 1: Let the composite vector be a process characterized with P different information sources, each of which is given by a (possibly different dimensional) vector $\mathbf{x}^{(p)}$, $p = 1, \ldots, P$. The composite vector of the information sources is given by

$$\mathbf{x} = \bigcup_{p=1}^{P} \mathbf{x}^{(p)} \tag{12}$$

where the operator $\bigcup$ represents the concatenation to a single vector. □

Property 2—Stacked Composite Kernel: Given the composite vectors $\mathbf{x}$ and $\mathbf{z}$, and the nonlinear mapping $\boldsymbol{\phi}$, the associated kernel given by

$$K(\mathbf{x}, \mathbf{z}) = \langle \boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{z}) \rangle \tag{13}$$

is a Mercer's kernel. □

Definition 2—Composite Vector in the Feature Space: Given a composite vector $\mathbf{x}$, its generalized nonlinear transformation into a feature space is given by

$$\boldsymbol{\phi}(\mathbf{x}) = \bigcup_{p=1}^{P} \left\{ \mathbf{A}_p \boldsymbol{\varphi}_p \left(\mathbf{x}^{(p)} \right) \right\} \tag{14}$$

where $\boldsymbol{\varphi}_p$ represents the nonlinear mappings of each information component to (possibly different) Hilbert spaces, whose formal definition is included in Table I, and $\mathbf{A}_p$ represents positive definite matrices. Hence, the composite vector in the feature space is given by the concatenation of the nonlinearly transformed and matrix-scaled information sources. □

Property 3—Direct Summation Composite Kernel: Given a composite vector $\mathbf{x}$ and its corresponding vector in the feature space as in (14), the kernel given by

$$K(\mathbf{x}, \mathbf{z}) = \sum_{p=1}^{P} K_p \left(\mathbf{x}^{(p)}, \mathbf{z}^{(p)} \right) \tag{15}$$

is a Mercer's kernel, and its associated mapping $\boldsymbol{\phi}(\cdot)$ is given in Table I. □

Property 4—Weighted Summation Composite Kernel: Given the composite vectors $\mathbf{x}$ and $\mathbf{z}$, their vectors in the feature spaces can be redefined to take advantage of Property 1 (10), and then, the kernel becomes

$$K(\mathbf{x}, \mathbf{z}) = \sum_{p=1}^{P} \mu_p K_p \left(\mathbf{x}^{(p)}, \mathbf{z}^{(p)} \right) \tag{16}$$

which is a Mercer's kernel with associated mapping $\boldsymbol{\phi}$ defined in Table I. □

Property 5—Cross-Information Composite Kernel: Given a composite vector, its vector in the feature space can be conveniently redefined, and then, the kernel

$$K(\mathbf{x}, \mathbf{z}) = \sum_{p=1}^{P} K_p \left(\mathbf{x}^{(p)}, \mathbf{z}^{(p)} \right) + \sum_{p,p'=1}^{P} K_{p,p'} \left(\mathbf{x}^{(p)}, \mathbf{z}^{(p')} \right) \tag{17}$$

is a Mercer's kernel whose associated mapping $\boldsymbol{\phi}(\cdot)$ is defined in Table I. □

These composite-kernel basic tools were originally introduced in [37] for SVM-based nonlinear system identification problems, and they were recently extended to a Bayesian framework [38]. In [30], Camps-Valls *et al.* also exploited the same methodology to integrate the spatial (contextual or textural) and the spectral information in kernel-based hyperspectral image classifiers. Camps-Valls *et al.* [39] have also used the same idea under a semisupervised graph-based approach. In the following sections, we will deploy it for multitemporal image classification and for integrating multisensor information in the classifiers.

IV. MULTITEMPORAL CLASSIFICATION AND CHANGE DETECTION WITH KERNELS

First, this section introduces novel formulations for the problem of multitemporal data classification. Then, we formulate "kernelized" versions of the familiar difference and ratio methods for change detection.

A. Problem Statement

Let us start by formally revising the main differences between multitemporal classification and change detection. In the first case, one tries to classify the pixels of an image at the observation time t_o by using all available (instantaneous and/or previous) information $t \leq t_o$. In the second case, the aim is only to identify those pixels that have changed. Although,

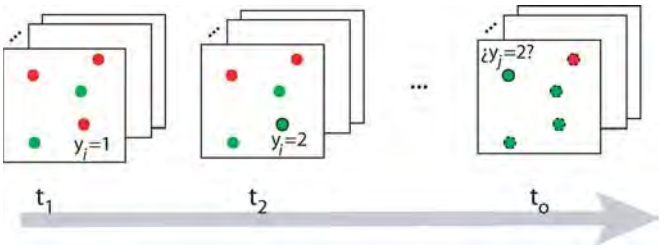

Fig. 2. Scheme for multitemporal classification and change detection. The problem of multitemporal classification consists of classifying a given pixel j at time t_o, $\mathbf{x}_j^{(t_o)}$, using all available information ($t \leq t_o$), while change detection tries to identify if the pixel's class label has changed or not.

intrinsically, very similar problems, the second one could require less effort [1]. In this paper, we follow a cascade strategy for classification, i.e., only the previous acquired information is used to classify a given image. This strategy differs from a mutual strategy in which the posterior images are also used. Therefore, under the supervised-learning framework, two main situations may arise, depending on the availability of information at the classification time $t = t_o$ (see Fig. 2).

1) Labeled information is available only for $t < t_o$. This is the most common scenario and discourages the use of classifiers, such as the SVC, that learn to discriminate classes at $t < t_o$ and then are used to extrapolate their predictions at $t = t_o$.
2) Labeled information is available for $t \leq t_o$. This much more advantageous situation makes the use of supervised classifiers, such as binary (SVC) and one-class (SVDD) schemes, more appropriate.

In both scenarios, we will use the SVC and SVDD classifiers, according to whether there is full information on the class labels or only on the class(es) of interest, respectively.

B. Multitemporal Classification

Let us assume a multitemporal set of labeled training samples (pixels) at a time t $\{\mathbf{x}_i^{(t)}\} \in \mathbb{R}^N$ and their corresponding output labels $\{y_i^{(t)}\} \in \mathbb{N}$, where $i = 1, \ldots, n$ and $t = 1, \ldots, t_o - 1$ or $t = 1, \ldots, t_o$, depending on the available data (aforementioned cases 1 and 2). An important assumption in the following is that the images at subsequent dates are coregistered so that, from a machine learning perspective, pixels $\{\mathbf{x}_i^{(t)}\}$ are different (temporal) views of the same object or pixel entity $\mathbf{x}_i$. In addition, let $\Omega = \{\omega_1, \ldots, \omega_{N_C}\}$ be the set of N_C classes that characterize the geographical area at any time, thus assuming that the spatial distribution of such classes changes but their number does not. This can be fairly assumed in standard and operational situations considering that the number of classes of interest is commonly prespecified by the user. If we now define $T = \max(t)$, generic multitemporal kernel-based classifiers can be formulated as described in the following.

1) Stacked Input-Vector Kernel: The most common approach to exploit the multitemporal information is to stack vectors at different time instants in order to predict the sample label at t_o. The composite input vector is given here by $\mathbf{x}_i^{(t_o)} \equiv \bigcup_{t=1}^{T} \mathbf{x}_i^{(t)}$, which yields the stacked kernel

$$K\left(\mathbf{x}_i^{(t_o)}, \mathbf{x}_j^{(t_o)}\right) = \left\langle \boldsymbol{\phi}\left(\mathbf{x}_i^{(t_o)}\right), \boldsymbol{\phi}\left(\mathbf{x}_j^{(t_o)}\right)\right\rangle. \quad (18)$$

However, although this straightforward approach to data merging can yield good performance with respect to previously proposed methods, it does not include explicit cross relations between samples at different time instants $\mathbf{x}_i^{(t)}$.

2) Direct Summation Kernel: A simple composite kernel combining the static available information comes from the concatenation of nonlinear transformations for each $\mathbf{x}_i^{(t)}$ according to (14), which can be easily computed as follows:

$$K\left(\mathbf{x}_i^{(t_o)}, \mathbf{x}_j^{(t_o)}\right) = \sum_{t=1}^{T} K_t\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t)}\right). \quad (19)$$

Note that this composite kernel basically sums the similarities of samples at each time instant individually, whereas stacking features is avoided. Once more, no temporal correlation among pixels in different images is included in the classifier.

3) Weighted Summation Kernel: By exploiting the kernel property in (10), a composite kernel that balances the temporal content in (19) can also be created as follows:

$$K\left(\mathbf{x}_i^{(t_o)}, \mathbf{x}_j^{(t_o)}\right) = \sum_{t=1}^{T} \mu_t K_t\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t)}\right) \quad (20)$$

where μ_t is a function assigning different weights to each time-dependent kernel and can either be estimated from the data or fixed by the user. A good choice is an exponential decay; this is $\mu_t = \lambda^{-(t_o - t)}$, $\lambda \in (0, 1)$.

4) Cross-Information Kernel: In order to account for the cross relationship among subsequent time instants, we can use the cross-information kernel in (17), which yields

$$K\left(\mathbf{x}_i^{(t_o)}, \mathbf{x}_j^{(t_o)}\right) = \sum_{t=1}^{T} K_t\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t)}\right) + \sum_{t,t'=1}^{T} K_{t,t'}\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t')}\right). \quad (21)$$

Note that this is a complex composite kernel that contains the cross information among all possible kernel matrices computed at different time instants. It is easy to show that this general equation can be simplified for the case of considering the correlation only for subsequent time instants t and $t+1$, and then, the composite kernel takes the form

$$K\left(\mathbf{x}_i^{(t_o)}, \mathbf{x}_j^{(t_o)}\right) = \sum_{t=1}^{T-1} \Big[K_t\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t)}\right) + K_t\left(\mathbf{x}_i^{(t+1)}, \mathbf{x}_j^{(t+1)}\right) + K_t\left(\mathbf{x}_i^{(t)}, \mathbf{x}_j^{(t+1)}\right)\Big]. \quad (22)$$

We should note that each term in the kernel summation can be of different type (RBF, polynomial, linear, etc.).

C. Change Detection

In this section, we present two novel kernel-based formulations to deal with the particular problem of change detection. The methods are inspired by the usual difference and ratioing operations; however, instead of being calculated in the input space, they are calculated in the feature space. The two main advantages of defining these operations in a high-dimensional feature space are the following: 1) it allows us to deal with the likely nonlinear nature of the relationships among samples and 2) the free parameters are learned from the data.

1) Image Difference in Feature Spaces: We can see change detection as a particular case of multitemporal classification, in which the only property to be detected is the presence of changes in the scene. In remote sensing, change identification traditionally subtracts the subsequent images and then applies a threshold that is tuned either heuristically or under other more sophisticated criteria [5], [20], [21], [23]. This difference vector can be formulated in the kernel feature space by defining a proper kernel mapping function. If we define the difference of the same sample in two subsequent images the following kernel is obtained (see Table I for the definition of the associated mapping ϕ to this kernel):

$$K\left(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)}\right)=K_{t_o}\left(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)}\right)+K_{t_o-1}\left(\mathbf{x}_i^{(t_o-1)},\mathbf{x}_j^{(t_o-1)}\right)$$
$$-K_{t_o,t_o-1}\left(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o-1)}\right)-K_{t_o-1,t_o}\left(\mathbf{x}_i^{(t_o-1)},\mathbf{x}_j^{(t_o)}\right). \quad (23)$$

Given that this difference kernel can only be used in the scenario where supervised information at time t_o is available, it can be seen as a particular case of the previous cross-information kernel in (21).

2) Image Ratioing in Feature Spaces: Another classical change detection method is the ratioing between images at two different dates, which helps to accommodate changes due to factors such as sun angle and shadows and is widely used for SAR data processing. By defining the ratio kernel mapping in Table I, we obtain the ratioing operation with kernels

$$K\left(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)}\right)=\gamma\boldsymbol{\delta}_{ij}+\frac{K_{t_o-1}\left(\mathbf{x}_i^{(t_o-1)},\mathbf{x}_j^{(t_o-1)}\right)}{K_{t_o}\left(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)}\right)} \quad (24)$$

where we have introduced the regularization parameter γ to make the matrix definitely positive. Otherwise, there would not be any warranty for this kernel to be a valid Mercer's kernel (see Table I for the formal definition of the associated mapping).

3) Remarks: The presented formulations are all valid for any kernel-based algorithm, such as the binary SVC and the one-class SVDD, because they ultimately rely on building a similarity (kernel) matrix among samples. The SVC will build the kernel among samples belonging to all labeled classes, whereas the SVDD will consider only samples belonging to the class of interest. Then, a multiclass strategy can be followed. Finally, note that solving the minimization problem in all kinds of composite kernels requires the same number of constraints as in the conventional algorithm, and hence, no additional computational efforts are induced in the size of their corresponding quadratic programming problems. In turn, by constructing dedicated kernels to process each information source, the problem of multicollinearity among features is alleviated, and the increase of the dimensionality of the training samples induced by stacking features is limited.

V. Contextual and Multisource Data Fusion With Kernels

In this section, we briefly review the use of the presented framework to incorporate the contextual or textural information in the kernel, and we also extend it to deal with multisource data.

A. Contextual Information

Let us now characterize the spectral content of a pixel at time t_o as $\boldsymbol{\omega}_i^{(t_o)}\in\mathbb{R}^{N_\omega}$, with N_ω being the number of its spectral bands. Now, let us perform some (local or global) feature extraction on the image, which yields the vector $\boldsymbol{s}_i^{(t_o)}\in\mathbb{R}^{N_s}$ associated to $\boldsymbol{\omega}_i^{(t_o)}$, with N_s representing the spatial (contextual or textural) features. The usual way to develop a kernel-based classifier that accounts for both the spectral and spatial features considers stacking both vectors $\mathbf{x}_i^{(t_o)}\equiv\{\boldsymbol{\omega}_i^{(t_o)},\boldsymbol{s}_i^{(t_o)}\}$ which are fed to a standard classifier. This is known as the stacked kernel, in which the kernel to be constructed is given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})$. However, by exploiting the previous composite methods, one can define several contextual kernel classifiers [30]: the direct summation kernel, which is given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})=K_s(\boldsymbol{s}_i^{(t_o)},\boldsymbol{s}_j^{(t_o)})+K_\omega(\boldsymbol{\omega}_i^{(t_o)},\boldsymbol{\omega}_j^{(t_o)})$, and the cross-information kernel, which is given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})=K_s(\boldsymbol{s}_i^{(t_o)},\boldsymbol{s}_j^{(t_o)})+K_\omega(\boldsymbol{\omega}_i^{(t_o)},\boldsymbol{\omega}_j^{(t_o)})+K_{s\omega}(\boldsymbol{s}_i^{(t_o)},\boldsymbol{\omega}_j^{(t_o)})+K_{\omega s}(\boldsymbol{\omega}_i^{(t_o)},\boldsymbol{s}_j^{(t_o)})$. Note that for the latter kernel, $\boldsymbol{s}_i^{(t_o)}$ and $\boldsymbol{\omega}_j^{(t_o)}$ need to have the same dimension ($N_\omega=N_s$) for this formulation to be valid. A possibility to enable its use is to extract a spatial feature per spectral band.

B. Multisource Data

Similarly, we can integrate multisensor information in the kernel itself in an elegant way. Now, if we have optical and radar information associated with the same coregistered pixel at time t_o, we can define the optical feature vector $\mathbf{o}_i^{(t_o)}$, the radar feature vector $\mathbf{r}_i^{(t_o)}$, and its concatenation $\mathbf{x}_i^{(t_o)}\equiv\{\mathbf{o}_i^{(t_o)},\mathbf{r}_i^{(t_o)}\}$. Working with these vectors forces us to develop the stacked kernel given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})$. Other kernels can be computed instead: the direct summation kernel, which is given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})=K_o(\mathbf{o}_i^{(t_o)},\mathbf{o}_j^{(t_o)})+K_r(\mathbf{r}_i^{(t_o)},\mathbf{r}_j^{(t_o)})$, and the cross-information kernel, which is given by $K(\mathbf{x}_i^{(t_o)},\mathbf{x}_j^{(t_o)})=K_o(\mathbf{o}_i^{(t_o)},\mathbf{o}_j^{(t_o)})+K_r(\mathbf{r}_i^{(t_o)},\mathbf{r}_j^{(t_o)})+K_{or}(\mathbf{o}_i^{(t_o)},\mathbf{r}_j^{(t_o)})+K_{ro}(\mathbf{r}_i^{(t_o)},\mathbf{o}_j^{(t_o)})$. Here, note once more that $\mathbf{o}_i^{(t_o)}$ and $\mathbf{r}_j^{(t_o)}$ need to have the same dimension ($N_r=N_o$) for this formulation to be valid, which is not a common situation.

C. Remarks

Note that at this point, the general problem of multitemporal classification can be decomposed in many constituents (temporal, spectral, spatial, source, etc.) which are mapped into different feature spaces and combined there implicitly through composite kernels. This will report some advantages, such as working with dedicated kernels to each information view, combining them linearly, and alleviating the problem of the curse of dimensionality because the stacking features are no longer necessary.

VI. Experimental Results

In this section, we analyze the performance of the proposed methodological framework based on kernels. Extensive

comparison is conducted for all scenarios (partial or complete labeled information at the prediction time): for multitemporal classification and change detection, multisource information fusion, and for many composite kernel combinations. We first illustrate the performance of the classification framework in a long time series of synthetic images. Then, we tackle the challenging problem of multitemporal and multisource image classification and change detection with two additional real test sites.

A. Model Development and Free-Parameter Selection

In our experiments, we used the linear and the RBF kernels to construct the kernel matrices. The linear kernel is tested here in order to show the performance of the standard "linear" techniques revised in Section I. However, we should note here that this linear kernel classifier is a more sophisticated model than the common approaches used in the literature because it constitutes a maximum margin algorithm. In addition, these linear-kernel-based classifiers allow a fair comparison with the nonlinear RBF kernel because, following the same composite-kernel framework, the temporal, contextual, and multisource information are also included. Considering that the number of potentially useful combinations of (spatial, spectral, temporal, and multisource) composite kernels is very high, in this paper, we will restrict ourselves to present results on those combinations showing the best performance according to our previous experience and experiments [30], [31], [39], [40].

For the case of the linear kernel machines, only the penalization factor C had to be tuned. For the case of the nonlinear RBF kernel classifiers and depending on the composite kernel used, a different σ parameter was additionally tuned for each component of the composite kernel. The sum of kernels was normalized in feature spaces before training [19]. All RBF kernel widths were tuned in the $\sigma = \{10^{-3}, \ldots, 10^{3}\}$ range, the regularization parameter C for SVC was varied in $[10^{-1}, 103]$, and the rejection parameter ν for the SVDD method was tuned in the range $[10^{-3}, 100]$. In the case of the multitemporal weighted summation kernel, μ was varied in the [0, 1] range. An exhaustive search among all free parameters is computationally unfeasible. Therefore, a nonexhaustive iterative search strategy (τ iterations) was used here. At each iteration, a sequential search of the minimum ten-fold cross-validation estimated kappa statistic on each parameter domain was performed by splitting the range of the parameter in L points. The values of $\tau = 3$ and $L = 20$ exhibited good performance in our simulations. A one-against-one multiclassification scheme was adopted for the SVC classifier and the multiclass scheme presented in [41] for one-class SVDD classification (complementary material (MATLAB source code, synthetic data, and demos) is available at http://www.uv.es/gcamps/soft.htm for those interested readers).

B. Experiment 1: Multitemporal Hyperspectral Image Classification

The first battery of experiments is concerned with assessing the multitemporal image classification framework presented, and thus, contextual or multisource information is not processed here. For this purpose, we generated a long time series of 12 synthetic labeled hyperspectral images of 200 $\times$ 200 pixel size containing eight classes (forest, grassland, shallow water, bare soil, rural urban area, sand, winter crops, and summer crops) that vary along time.

1) Synthetic Data: For the generation of realistic synthetic hyperspectral images, we used data from the Compact High-Resolution Imaging Spectrometer (CHRIS), which is mounted onboard the European Space Agency (ESA) satellite "Project for On Board Autonomy" (PROBA). CHRIS sensor provides hyperspectral images in the spectral range from 400 to 1050 nm (62 spectral channels for acquisition Mode 1) [42]. The selected image was acquired in the Agricultural Bio-/Geophysical Retrievals from Frequent Repeat SAR and Optical Imaging (AgriSAR) 2006 campaign over the Demmin site (in Germany) [43]. This image was selected for the study in order to take into account different surface types, and spatial textures (soil, vegetation, water, urban areas, etc.).

First, we manually labeled the CHRIS/PROBA image. In multispectral image processing, the assumption that the distribution of image classes can be approximated as a mixture of normally distributed samples is widely accepted. Therefore, we considered each homogeneous land cover as a normal distribution and used the labeled regions of the CHRIS image to estimate the parameters of a Gaussian mixture model (GMM; mean, $\boldsymbol{\mu}$, and covariance matrix, $\boldsymbol{\Sigma}$, for each class). Once we had the ground truth with the areas covered by the different spectral classes, the parameters of the GMM and the priors, we generated a synthetic image, as follows: 1) the required number of samples for each class was randomly generated from the corresponding 62-D Gaussian distribution; 2) a proper texture of gray-level values was assigned to each region (or class) in the image; and 3) the generated spectra of each class were shortened depending on their brightness (intensity) and were iteratively assigned to the image location that presented the next higher value in the texture image.

Following this procedure, the final image preserved the specified textures; however, the spectral classes are not modified. In addition, in order to simulate the mixed pixels, which usually occur in the boundaries between classes, we included a gradual linear spectral mixture in the four pixels that are closest to the class boundaries.

When generating the time series of 12 synthetic labeled hyperspectral images, we can distinguish three kinds of changes in the spectral signature of a given pixel along time, as follows: 1) natural spectral variability of the class accounted by the covariance matrix and the random generation of the samples for the different dates; 2) changes of the class distributions between dates (e.g., due to illumination or atmospheric effects) simulated with a multiplicative factor over the distribution parameters ($\boldsymbol{\mu}_t = \delta_t \boldsymbol{\mu}$ and $\boldsymbol{\Sigma}_t = \delta_t^2 \boldsymbol{\Sigma}$, where $\delta_t = 0.01t + 0.94$, $t = 1, \ldots, 11$ for all classes); and 3) artificially generated changes in the ground truth. These latter class changes were only included at odd time instants (t_3, t_5, t_7, t_9, and t_{11}), which allows us to study the adaptation capabilities of the time-varying kernel classifiers (see Table II for details of the introduced changes).

2) Multitemporal Image Classification: In order to analyze the performance of the proposed methods under realistic ill-posed situations, we varied the number of training samples per class $(n = \{5, 10, 15, 20, 30, 40, 50\})$ and measured the

TABLE II
INTRODUCED CHANGES AT THE CORRESPONDING TIME INSTANTS

Time	Change	♯Pixels
$t_2 \rightarrow t_3$	Grassland (C2) to Bare soil (C4)	1361
$t_4 \rightarrow t_5$	Bare soil (C4) to Rural urban area (C5)	608
$t_6 \rightarrow t_7$	Sallow water (C3) to Sand (C6)	229
$t_8 \rightarrow t_9$	Bare soil (C4) to Winter crops (C7)	1470
$t_{10} \rightarrow t_{11}$	Sand (C6) to Rural urban area (C5)	305

overall accuracy (OA; in percent), the estimated kappa statistic (κ), and the complexity of the machines using the rate of SVs (in percent). The best composite kernels were selected according to the κ score in it. Average results for a number of ten realizations and over all time prediction instants are shown at the top part of Fig. 3. In all cases, the RBF kernel outperformed the linear kernel.

Several conclusions can be derived. First, as we increase the number of training samples, the accuracy and sparsity increase. Second, the best kernel classifier is constituted by the cross-terms kernel because it includes the temporal information of image evolution (22). The weighted and direct summation kernels produce very similar (but lower) accuracy and sparse solutions, and the stacked kernel approach produces the worst results in almost all the domain, which are probably due to the extremely high input-space dimensionality generated. It is also important to note that the proposed classifiers obtain results close to the maximum *a posteriori* (MAP) classifier[1] even when working with a reduced number of training samples. This can be explained because the composite kernel classifiers consider the temporal information in addition to the static spectral information.

3) Change Detection: The same experiment was conducted for the change detection problems. Average results are shown at the bottom part of Fig. 3. Once again, the RBF kernel yielded higher accuracy than the linear kernel. The best results were obtained by using the difference kernel (average improvement of $\sim$5%), which is closely followed by the summation and the stacked composite kernels. The ratio kernel produces competitive results; however, some instabilities are appreciated in the accuracy curves, which are mainly due to the difficulties in selecting the regularization parameter γ, which is a topic to be studied in the future. It should be noted that, in general, OA and κ measurements are unbalanced, suggesting that high values of false detections are produced. Finally, all classifiers have similar structural complexities (SVs; in percent); however, in general terms, the SVDD offers sparser solutions, which are slightly more significant for the proposed difference and ratio kernels.

4) Classification Maps: For further comparison of the classifiers, we focused on the case of 50 training samples for each class (last point in the curves of Fig. 3). The top part of Table III shows the sequence of synthetic images; their corresponding true maps; and the classification maps obtained with the optimal MAP classifier, the best composite-kernel SVC, and SVDD classifiers at each time instant. It can be noted that, again, the SVC works slightly better than the SVDD in almost all the cases. This can be due to the fact that distributions vary along time quite smoothly, and thus, pure inductive classifiers can yield good results in this experiment.[2] One can also notice that, in general, results are improved when enough temporal data are available to allow the classifiers to follow the dynamic changes. Accuracy reaches the maximum values starting at time t_7, getting closer to the MAP solution with relatively few training data.

The bottom part of Table III shows the results for the change detection maps obtained with the RBF, the linear difference, and ratio kernels at each time instant. Note that good results are obtained with the nonlinear RBF kernel classifier, which are both in terms of accuracy and estimated kappa figures, thus suggesting robust and stable models. However, significantly lower kappa statistics are obtained at t_5 and t_9 whenever the (C4) "bare soil" class changes to (C5) "rural urban areas" and (C7) "winter-crops," respectively. These heterogeneous areas of the real hyperspectral images were formed by both vegetated and bare-soil pixels, and hence, the learned changes are similar to the natural within-class variability of these two classes. In consequence and due to the random generation of the synthetic images, this fact produces a drastic increase of false detections. This problem is even more critical because of the low number of available training samples (50 for the change and 50 for the no-change class), where the linear kernel clearly fails in almost all scenarios.

As in the multitemporal classification problem, one can observe, however, that all the classification maps appear quite noisy, which is a direct consequence of not considering the contextual or textural information. In the following section, we will analyze the impact that contextual, textural, and multisource information have on the classification of real images.

C. Experiment 2: Multitemporal/Source Urban Monitoring

This battery of experiments extends the previous ones by exploiting the proposed composite-kernel framework to build classifiers that, in addition to the temporal information, also integrate contextual, textural, and multisource data. The experiments consider multitemporal classification and change detection in real images.

1) Data Collection and Feature Extraction: The images used in this experiment were collected in the Urban Expansion Monitoring (UrbEx) ESA project [44]. Results from the UrbEx project were also used to perform the analysis of the selected test sites and for validation purposes.[3] The considered test sites were Rome and Naples (in Italy), where images from European Remote Sensing 2 (ERS2) SAR and Landsat Thematic Mapper (TM) sensors were acquired in 1995 and 1999. In this set of experiments, only two time instants are available, and thus, the complexity of the classifiers reduces significantly as $t_o = 2$.

An external digital elevation model and a reference land-cover map provided by the Italian Institute of Statistics (ISTAT) were also available. The ERS2 SAR 35-day interferometric

[1]Note that the MAP classification constitutes an upper bound of model performance because the true distribution that generated the data is used to generate the classifier.

[2]This hypothesis was confirmed by noting that the optimal time window of images was three (results not shown).

[3]For further details, visit: http://dup.esrin.esa.int/ionia/projects/summaryp30.asp.

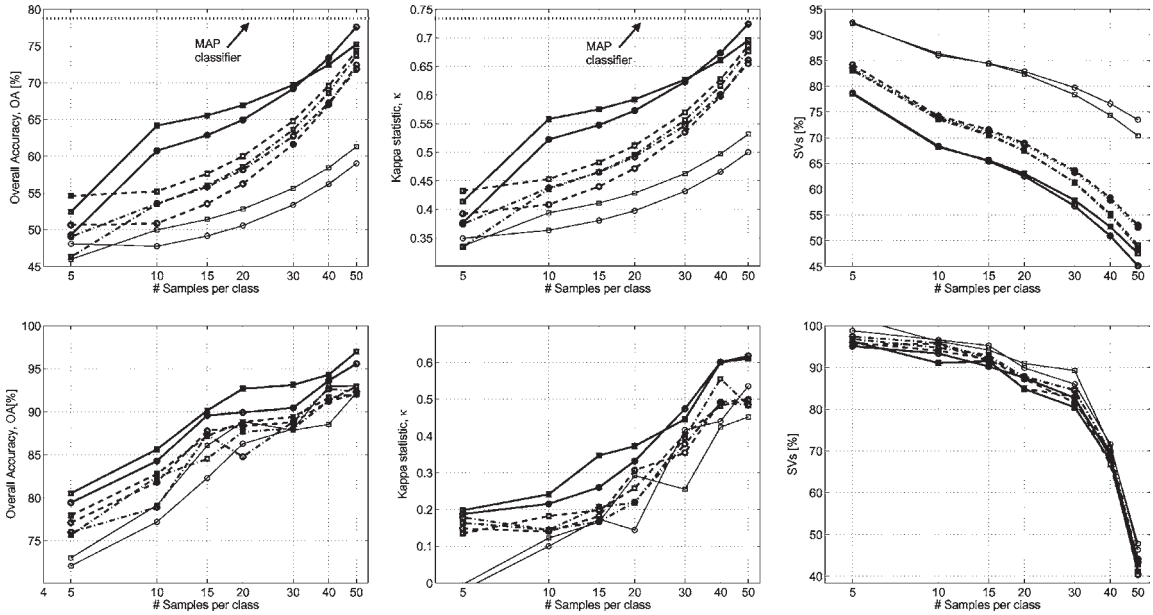

Fig. 3. Results for the (top) multitemporal classification and (bottom) change detection problems. (Left) Overall accuracy, (middle) kappa statistic, and (right) rate of SVs as a function of the number of training samples per class. Several kernel classifiers are shown by using (squares) SVC and (circles) SVDD with the RBF kernel function. (Thick solid lines) Cross-information and difference kernels, (dash-dotted lines) weighted and ratio kernels, (dashed lines) summation kernel, and (thin solid lines) stacked approach. Average results for all time instants and ten realizations are shown for all methods and the MAP classifier.

TABLE III

SEQUENCE OF SYNTHETIC IMAGES (RED–GREEN–BLUE (RGB) COMPOSITION, BANDS [19–12–2]) AND THEIR CORRESPONDING TRUE CLASSIFICATION MAPS. ONLY 50 TRAINING SAMPLES WERE USED FOR ALL CASES, AND THE ODD TIME INSTANT IMAGES ARE DEPICTED, CONSIDERING THAT CLASS CHANGES (BOUNDARIES HIGHLIGHTED IN RED) WERE ONLY INTRODUCED AT THESE DATES. ACCURACIES ARE INDICATED ON THE FORM (OA [IN PERCENT], κ)

	t_1	t_3	t_5	t_7	t_9	t_{11}
						
True Map						
MAP	(79.62%, 0.75)	(79.30%, 0.74)	(78.90%, 0.74)	(78.72%, 0.74)	(78.22%, 0.73)	(77.50%, 0.72)
RBF-SVC Cross kernel		(61.33%, 0.53)	(61.04 %,0.43)	(76.81%, 0.72)	(68.40%, 0.61)	(71.64%, 0.65)
RBF-SVDD Cross kernel		(49.15%, 0.39)	(51.35%,0.42)	(73.09%, 0.67)	(57.69%, 0.48)	(70.16%, 0.63)
RBF Difference Kernel		(97.04%,0.55)	(95.39%,0.11)	(94.33%,0.78)	(95.17%,0.24)	(99.41%,0.66)
Linear Difference Kernel		(67.78%,0.15)	(86.59%,0.03)	(94.78%,0.44)	(83.45%,0.18)	(97.47%,0.36)

pairs were selected with perpendicular baselines between 20 and 150 m in order to obtain the interferometric coherence from each complex SAR image pair. The available features were initially labeled as: L1–L7 for Landsat bands, In1–In2 for the SAR backscattering intensities (0–35 days), and Co for the coherence.

Considering that these features come from different sensors, the first step was to perform a specific processing and conditioning of optical and SAR data and to coregister all images. The seven bands of Landsat TM were coregistered with the ISTAT classification data and resampled to 30×30 m with the nearest neighbor algorithm. The registration for the multisource images was performed at a subpixel level, obtaining a root-mean-squared error of about 10 m, which potentially enables good urban classification abilities [45].

In the case of the optical images, the seven Landsat TM spectral bands (containing three visible, one near infrared (IR), two short-wave IR, and one thermal IR bands) were directly used $\mathbf{o}_i = \{\mathrm{L1}, \ldots, \mathrm{L7}\}$. In the case of the SAR images, the intensity and coherence were computed [46]. However, considering that speckle disturbs image interpretation, a multistage spatial filtering approach over coherence images was followed to increase the urban-area discrimination [47], which yielded the fourth radar input feature Co'. Therefore, in this case, we define $\mathbf{r}_i = \{\mathrm{In1}, \mathrm{In2}, \mathrm{Co}, \mathrm{Co}'\}_i$.

Once features were extracted from optical and SAR images, we analyzed their potential use for urban change detection. The high overlapping of change and no-change pixels indicated an extremely difficult change detection problem and suggested that non-linear methods should be deployed. We also computed spatial and textural features from these optical and SAR features. Specifically, the spatial features for the optical images were the average of all pixels in the surrounding 7×7 window, and the textural features for the SAR data were Gabor-filtered [48] versions of $\mathbf{r}_i$ at different scales $(\theta = 1, \ldots, 4)$ and orientations $\{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$, thus yielding the textural radar features.

In the following experiments, we selected subset images from the Rome and Naples scenes containing 200×200 pixels in areas with substantial urban changes. For the case of the Rome scene, 1392 pixels changed to "nonurban," 780 pixels changed to "urban," and 2978 changed to the "unknown" status in this four-year period. For the case of the Naples scene, 1826 pixels changed to "urban," 215 pixels changed to "nonurban," and 1973 changed to the "unknown" status. Pixels belonging to the unknown class were not considered, and hence, this becomes a classical binary problem of change versus no-change identification. In both cases, we randomly selected 25% of the changed pixels for training and used five-fold cross validation for free-parameter tuning. Then, we tested the built classifier on the whole image.

2) Multitemporal Image Classification: Table IV shows the results obtained by different supervised classifiers. Specifically, we compare SVC and SVDD under the (left) multitemporal classification and (right) change detection using different temporal, spatiospectral, and multisource composite kernels for the scenes of Rome and Naples. In the comparison, we also include the well-known multilayer perceptron (MLP) neural network, which has been a traditional approach for supervised multitemporal image classification and change detection [8]–[10], [13]. The MLP was trained using the Levenberg–Marquardt algorithm, which is more efficient in terms of computational cost than the standard back-propagation algorithm [49]. Different numbers of hidden neurons in the hidden layer were tested $n_h \in [2, 100]$, and the best architecture was selected by evaluating the averaged five-fold cross-validation kappa statistic. In order to include temporal, multisource, or contextual information in the MLP, we followed the traditional stacked approach because, this way, a nonlinear relationship among inputs can be modeled in the hidden layer. The best OA (in percent) and estimated κ values are provided in all cases. We also analyze class-by-class accuracies for particularly interesting cases and assess statistical differences among classifiers through Wilcoxon's rank-sum test at a 95% confidence interval.

The following conclusions can be obtained from this table. In all cases and scenes, it becomes obvious that the use of the RBF kernel provides much better results than the linear kernel which, in turn, constitutes an upper bound of the model's performance for (change detection) thresholding methods. The results yielded by the MLP are, in all cases, inferior to those provided by the RBF kernel approaches, which is probably due to the fact that the input dimension increases when stacking many features. Certainly, neural networks cannot efficiently cope with very high input dimensional spaces, as in our case. For the Rome scene (top part of Table IV) and in the case with unlabeled information of the prediction date image (1999), i.e., $t < t_o$, only the labeled samples for 1995 can be used to train a classifier and predict for 1999. In this complex situation, a purely supervised approach like the SVC yields poor solutions (OA (in percent); $< 70\%$ and $\kappa < 0.6$) because there is no information on the change. Contrarily, SVDD offers good results because, rather than building a separating hyperplane "urban"/"nonurban," the method tries to model the "urban" class accurately. In all cases, the best composite kernel for integrating the spatial and the different data sources was constituted by the summation kernel, i.e., dedicating separated kernels for the Landsat bands, SAR features, contextual Landsat features, and textural Gabor-filtered SAR features. This best method yielded a maximum accuracy of 84.2% but with biased classifications $(\kappa = 0.51)$, which was confirmed by looking at the individual class accuracies (90.3% for urban and 53.4% for nonurban). Finally, it is also worth noting that solutions are much sparser for the SVDD (average of 22% of SVs) than for the SVC (average of 59% of SVs). For the Naples scene, similar results are obtained (see the bottom part of Table IV). Again, when no information is available at time t_o, the SVDD constitutes a better approximation either with RBF or linear kernel embedding. These results are not only numerically different but, also, differences are statistically significant (see star symbols in Table IV; $p < 0.05$).

In the case with available labeled information for t_o, several composite kernels have been tested, drastically improving the results in both scenes. This is a clear consequence of using labeled samples from the t_o image. In these cases, the SVC classifiers show the best results; however, it can be appreciated that the SVDD classifiers also produce stable and robust outcomes, which confirms their suitability to application scenarios in which incomplete or partially complete information is available. Similar results have been lately observed in [41]. The same behavior is observed for the neural network, which provides inferior accuracies to the nonlinear SVC and SVDD

TABLE IV

RESULTS FOR THE (TOP) ROME AND (BOTTOM) NAPLES SCENES. OA (IN PERCENT) AND KAPPA STATISTIC (κ) FOR THE DIFFERENT SCENARIOS ($t < t_o$ AND $t \leq t_o$), TIME INTEGRATION, MULTISOURCE FUSION, AND CLASSIFIER ALGORITHMS. AVERAGE RESULTS OVER TEN REALIZATIONS ARE SHOWN FOR SVC, SVDD (USING BOTH LINEAR AND RBF KERNELS), AND MLP. (BOLD) BEST AND (ITALICS) SECOND-BEST RESULTS ARE HIGHLIGHTED FOR EACH COLUMN AND KERNEL TYPE. STATISTICALLY DIFFERENT RESULTS, EITHER IN TERMS OF OA (IN PERCENT; TESTED THROUGH PAIRED WILCOXON'S RANK-SUM TEST AT 95% CONFIDENCE INTERVAL) OR κ (CONSIDERING IT IS NORMALLY DISTRIBUTED) FROM THE BEST CLASSIFIER ARE MARKED WITH A STAR "$\star$"

			Multi-temp. class. $t < t_o$	Multi-temp. class. $t \leq t_o$			Change detection $t \leq t_o$	
	Spatio-Spectral	Multi-source	Summation Eq. (19)	Summation Eq. (19)	Cross-terms Eq. (21)	Weighted Eq. (20)	Kernel Diff. Eq. (23)	Kernel Ratio Eq. (24)
Rome scene								
SVC	Sum	Stack	55.0 (0.20)	83.2 (0.45)	68.2 (0.61)	70.4 (0.64)	81.1 (0.70)	80.2 (0.70)
LIN	Cross	Stack	54.1 (0.22)	81.4 (0.49)	69.2 (0.62)	71.4 (0.63)	**82.2 (0.68)**	*81.3 (0.71)*
	Sum	Sum	57.1 (0.31)	**84.1 (0.51)**	70.2 (0.63)	**73.4 (0.72)**	74.1 (0.72)*	79.4 (0.71)
SVDD	Sum	Stack	*58.1 (0.33)*	71.5 (0.52)*	*73.4 (0.63)*	68.1 (0.62)*	82.1 (0.71)	80.2 (0.71)
LIN	Cross	Stack	58.3 (0.34)	77.6 (0.55)	**74.1 (0.62)**	69.1 (0.54)*	81.3 (0.73)	78.3 (0.72)
	Sum	Sum	**66.1 (0.40)**	*78.2 (0.55)*	69.1 (0.62)	*78.3 (0.68)*	*82.2 (0.61)*	**83.1 (0.72)**
SVC	Sum	Stack	61.1 (0.51)*	91.4 (0.67)	83.1 (0.70)	89.5 (0.78)	95.3 (0.81)	95.1 (0.77)
RBF	Cross	Stack	66.5 (0.43)*	92.1 (0.69)	89.2 (0.71)	88.8 (0.77)	**96.1 (0.79)**	*95.8 (0.80)*
	Sum	Sum	68.3 (0.60)	**93.2 (0.77)**	**94.3 (0.78)**	**93.3 (0.81)**	94.1 (0.83)	93.3 (0.80)
SVDD	Sum	Stack	*77.3 (0.63)*	83.3 (0.66)*	85.3 (0.68)	88.1 (0.71)	92.3 (0.80)	91.1 (0.79)
RBF	Cross	Stack	75.1 (0.64)	88.7 (0.68)	84.5 (0.70)	79.0 (0.55)*	91.9 (0.82)	90.2 (0.79)
	Sum	Sum	**81.4 (0.70)**	*92.1 (0.75)*	*92.2 (0.75)*	*91.1 (0.79)*	*95.6 (0.80)*	**96.0 (0.81)**
MLP	Stack	Stack	58.7% (0.44)	83.5% (0.64)			86.4% (0.64)	
Naples scene								
SVC	Sum	Stack	58.3 (0.39)	**79.8 (0.48)**	*80.8 (0.51)*	81.0 (0.55)	85.1 (0.60)	81.3 (0.61)
LIN	Cross	Stack	56.2 (0.41)	77.1 (0.45)	80.3 (0.50)	*82.9 (0.55)*	*87.4 (0.64)*	*86.3 (0.62)*
	Sum	Sum	59.1 (0.41)	*77.8 (0.45)*	**81.3 (0.52)**	**86.1 (0.55)**	87.4 (0.64)	86.1 (0.62)
SVDD	Sum	Stack	60.1 (0.44)	77.8 (0.45)	71.8 (0.43)	71.1 (0.49)*	85.9 (0.60)	85.9 (0.61)
LIN	Cross	Stack	*61.4 (0.52)*	72.6 (0.45)	70.6 (0.41)*	72.8 (0.44)*	**87.8 (0.64)**	85.1 (0.62)
	Sum	Sum	**64.3 (0.54)**	75.8 (0.55)	75.8 (0.42)	76.0 (0.51)	84.5 (0.62)	**87.3 (0.62)**
SVC	Sum	Stack	70.1 (0.40)*	90.3 (0.71)	92.0 (0.59)	93.3 (0.60)	95.1 (0.71)	95.8 (0.75)
RBF	Cross	Stack	68.9 (0.45)*	94.1 (0.60)	88.1 (0.59)	94.4 (0.60)	**98.0 (0.75)**	96.3 (0.76)
	Sum	Sum	66.4 (0.55)*	*95.0 (0.73)*	**96.8 (0.64)**	**97.5 (0.66)**	96.9 (0.73)	*96.9 (0.76)*
SVDD	Sum	Stack	80.2 (0.50)	89.1 (0.57)	91.2 (0.56)	91.1 (0.59)	95.9 (0.70)	95.8 (0.71)
RBF	Cross	Stack	*82.6 (0.50)*	92.7 (0.55)	90.4 (0.60)	92.2 (0.57)	97.2 (0.74)	95.3 (0.71)
	Sum	Sum	**84.2 (0.51)**	**95.3 (0.67)**	*95.8 (0.61)*	*96.0 (0.65)*	*97.6 (0.71)*	**97.6 (0.71)**
MLP	Stack	Stack	66.1% (0.48)	88.4% (0.51)			80.1% (0.58)	

classifiers (both numerical and statistical). The best overall result was obtained by using simple summation kernels for integrating the spatio-spectral information and, in some cases, more complex cross-information kernels to process the temporal information. This type of classifier yielded a maximum OA $= 94.3\%$, a statistically compensated model ($\kappa = 0.78$), and good individual classification accuracies (97.1% for urban and 82.5% for nonurban, respectively) for the Rome image and yielded a maximum OA $= 96.8\%$ ($\kappa = 0.64$) and individual classification accuracies of 98.3% for urban and 83.3% for nonurban in the Naples image.

3) Change Detection: The right part of Table IV shows the results for the difference and ratio kernels for change detection. In these cases, labeled information for t_o is provided in the form of "change" versus "no-change" for the 1999 image; therefore, it can be considered as a supervised learning strategy. In general, a significant (both numerical and statistical) difference is observed by using RBF-based kernel classifiers (e.g., accuracy is about $+12\%$ higher). Note that the lower results offered by the MLP on the Naples image suggest the weakness of this method in dealing with high-dimensional, heterogeneous, and redundant input data. Note that in this case, even a linear regularized method yields significantly better OA ($+8\%$). The SVCs yield very good results in terms of accuracy (OA $> 90\%, \kappa > 0.7$); however, the SVDD provides better kappa values (although no significant statistical differences are appreciated), which indicates well-balanced classifications with reduced false detections. For the best SVC (SVDD) classifier, the individual accuracies were 97% (98%) for the change class and 69% (74%) for the unchanged class in the Rome image dataset. For the Naples dataset, results between SVC and SVDD did not differ significantly (98% versus 97% for the change

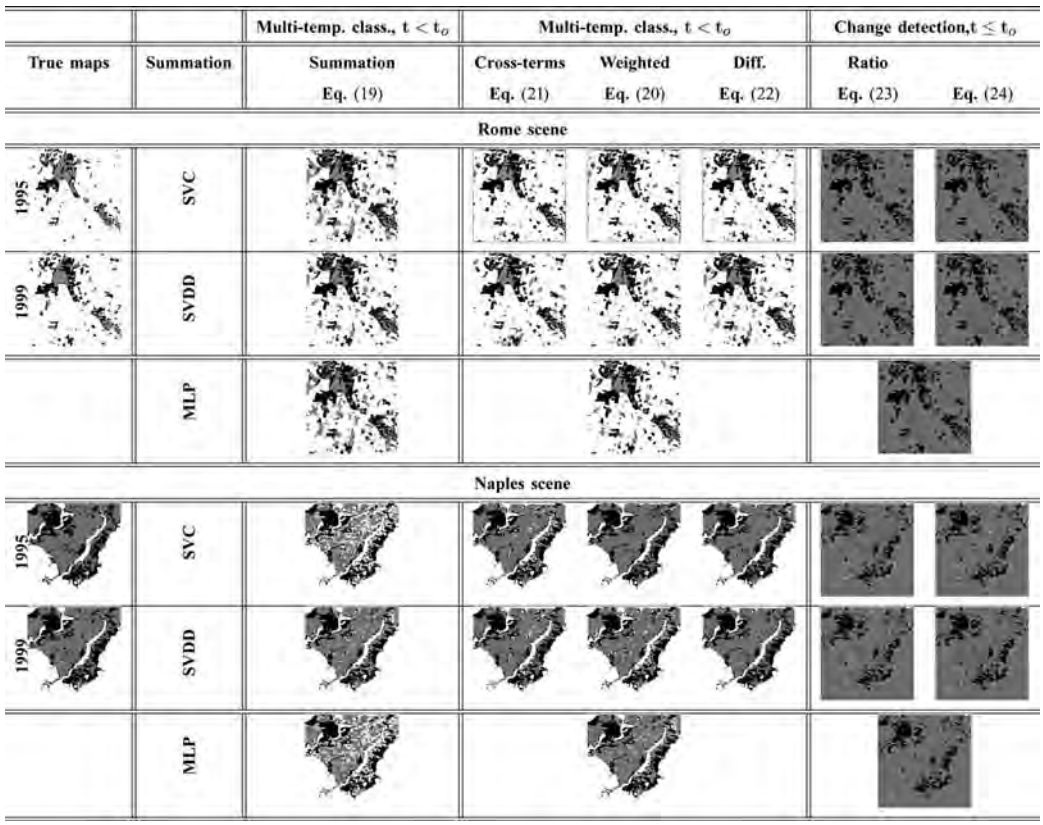

Fig. 4. Best classification maps as boldfaced in Table I for the (top) Rome and (bottom) Naples scenes. In the multitemporal classification maps, white pixels represent the class "nonurban," black pixels are "unknown class," and gray pixels are "urban." For the change detection maps, we have plotted the no-change classification in gray, the change in white, and the unknown class in black.

class and 75% versus 74% for the unchanged class), which are probably due to the fact that this dataset constitutes an easier problem and no particular guide to learn a specific class is included in the methods. Note, however, that this is a different (and much easier) experimental setup than the multitemporal approach, considering that the classifier only has to detect if the pixel's class label changed or not.

4) Classification Maps: The classification maps offered by the best methods (boldfaced in Table IV) are shown in Fig. 4 for (top) Rome and (bottom) Naples. The numerical results (which are already discussed before) are, in general, confirmed by the visual inspection. For instance, in the case of $t < t_o$, the results offered by the SVDD method are much better than the SVC (more homogenous areas and lower number of false detections), which are also observed in the case of using the difference or ratio kernels for change detection, although (slightly) better accuracies are obtained by using the SVC. In the case of $t \leq t_o$, the SVDD does not outperform the SVC, which is mainly because the SVDD poorly integrates the spatial/textural information (e.g., see the southern parts of the Rome scene or the middle eastern part of the Naples scene where evident changes occur). In addition, the neural network produces more noisy classification maps, which are particularly noticeable in change detection.

VII. Conclusion

In this paper, we have introduced a general methodological framework based on composite kernels for multitemporal classification of remote sensing images that simultaneously takes into account the spectral, spatial, and multisensor information. The main advantages of the proposed framework are the following. First, there is an alternative to stacking features for exploiting heterogeneous data sources, thus alleviating the curse of dimensionality. Second, the nonlinear relationship among pixels (and among features) at different time instants is treated with tunable flexibility. In this paper, we also introduced nonlinear kernel-based versions for the well-known difference and ratioing methods for change detection. In these cases, when a linear kernel is used, the approaches constitute an upper bound of their traditional counterparts.

As core learners, the binary SVC and the one-class SVDD classifier were used, and they were also benchmarked with neural networks in real scenarios. In general, neural networks show inferior results compared with nonlinear kernel classifiers, which is a direct consequence of their difficulties when working with very high dimensional input samples that are particularly important when stacking together other information sources, such as contextual or multitemporal. The binary SVC was more suitable than the SVDD when labeled data from the prediction instant were available or when the temporal dynamics of changes were slow. However, this is not the common situation, thus revealing the one-class SVDD as a particularly well-suited tool for learning the change detection problem.

We have illustrated the performance of the framework in both synthetic and real multitemporal images. The result of a long sequence of synthetic images, where a near optimal classification is obtained with a reduced number of training samples,

is particularly attractive. In general terms, the best composite kernel has been the cross-information kernel; however, we have also observed that simple composite kernels, such as the summation kernel, offered a very competitive performance. In the real-case scenario, results have also demonstrated that the use of basic composite kernels yields good results in the particular application domain of urban monitoring, outperforming the traditional stacked-vector approach in all cases. More sophisticated composite kernels that explicitly include cross relations between different information sources show better performance at the expense of increased computational burden.

Further work will consider the formulation of semisupervised kernel-based techniques in the multitemporal and multisource change detection framework. At present, good results have been obtained by using graph-based methods [39]; however, it is worth stressing that any kernel-based transductive or one-class learning algorithm could be adapted for this purpose. In addition, the future is tied to the study of optimizing methods of the linear summation of kernels [50] and of the formulation of adaptive (online) versions of the presented methodology.

ACKNOWLEDGMENT

The authors would like to thank the ESA for the availability of the image database and Dr. D. Fernandez-Prieto (Earth Observation Science and Applications Department, ESA–ESRIN) for the assistance provided.

REFERENCES

[1] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: A systematic survey," *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 294–307, Mar. 2005.

[2] T. M. Lillesand, R. W. Kiefer, and J. W. Chipman, *Remote Sensing and Image Interpretation*, 5th ed. New York: Wiley, 2004.

[3] A. Singh, "Digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, 2003.

[4] P. Coppin and M. Bauer, "Digital change detection in forest ecosystems with remote sensing imagery," *Remote Sens. Rev.*, vol. 13, no. 6, pp. 207–234, 1996.

[5] L. Bruzzone and S. Serpico, "An iterative technique for the detection of land-cover transitions in multitemporal remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 4, pp. 858–867, Jul. 1997.

[6] F. Wang, "A knowledge-based vision system for detecting land changes at urban fringes," *IEEE Trans. Geosci. Remote Sens.*, vol. 31, no. 1, pp. 136–145, Mar. 1993.

[7] J. T. Morisette and S. Khorram, "An introduction to using generalized linear models to enhance satellite-based change detection," in *Proc. Int. Conf. Geosci. Remote Sens.*, 1997, pp. 1282–1284.

[8] D. L Civco, "Artificial neural networks for land cover classification and mapping," *Int. J. Geogr. Inf. Syst.*, vol. 7, no. 2, pp. 173–186, 1993.

[9] D. Kushardono, K. Fukue, H. Shimoda, and T. Sakata, "Comparison of multi-temporal image classification methods," in *Proc. Int. Conf. Geosci. Remote Sens.*, 1995, pp. 1282–1284.

[10] S. Gopal and C. Woodcock, "Remote sensing of forest change using artificial neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 2, pp. 189–202, Mar. 1996.

[11] J. Li and R. M. Narayanan, "A shape-based approach to change detection of lakes using time series remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 11, pp. 2466–2477, Nov. 2003.

[12] D. Liu, M. Kelly, and P. Gong, "Classifying multitemporal Landsat TM imagery using Markov random fields and support vector machines," in *Proc. 3rd Int. Workshop Anal. Multi-temporal Remote Sens. Images*, 2005, pp. 225–228.

[13] T. M. Pellizzeri, P. Gamba, P. Lombardo, and F. Dell'Acqua, "Multitemporal/multiband SAR classification of urban areas using spatial analysis: Statistical versus neural kernel-based approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 10, pp. 2338–2353, Oct. 2003.

[14] P. Gamba, F. Dell'Acqua, and G. Lisini, "Change detection of multitemporal SAR data in urban areas combining feature-based and pixel-based techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pt. 1, no. 10, pp. 2820–2827, Oct. 2006.

[15] F. Melgani and S. B. Serpico, "A Markov random field approach to spatio-temporal contextual image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 11, pp. 2478–2487, Nov. 2003.

[16] F. Melgani, "Classification of multitemporal remote-sensing images by a fuzzy fusion of spectral and spatio-temporal contextual information," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 18, no. 2, pp. 143–156, Mar. 2004.

[17] A. Boucher, K. C. Seto, and A. G. Journel, "A novel method for mapping land cover changes: Incorporating time and space with geostatistics," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pt. 2, no. 11, pp. 3427–3435, Nov. 2006.

[18] B. Schölkopf and A. J. Smola, *Learning With Kernels*. Cambridge, MA: MIT Press, 2002.

[19] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[20] L. Bruzzone and D. Fernandez-Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 452–466, Apr. 2002.

[21] G. Moser and S. B. Serpico, "Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pt. 2, no. 10, pp. 2972–2982, Oct. 2006.

[22] C. Carincotte, S. Derrode, and S. Bourennane, "Unsupervised change detection on SAR images using fuzzy hidden Markov chains," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 432–441, Feb. 2006.

[23] F. Bovolo and L. Bruzzone, "A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 1, pp. 218–236, Jan. 2006.

[24] J. Inglada and G. Mercier, "A new statistical similarity measure for change detection in multitemporal SAR images and its extension to multiscale change analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, pt. 2, no. 5, pp. 1432–1445, May 2007.

[25] G. Mercier and F. Girard-Ardhuin, "Partially supervised oil-slick detection by SAR imagery using kernel expansion," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, pt. 1, no. 10, pp. 2839–2846, Oct. 2006.

[26] D. Potin, P. Vanheeghe, E. Duflos, and M. Davy, "An abrupt change detection algorithm for buried landmines localization," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 260–272, Feb. 2006.

[27] M. Carlotto, "Detection and analysis of change in remotely sensed imagery with application to wide area surveillance," *IEEE Trans. Image Process.*, vol. 6, no. 1, pp. 189–202, Jan. 1997.

[28] G. Camps-Valls, L. Gómez-Chova, J. Calpe, E. Soria, J. D. Martín, L. Alonso, and J. Moreno, "Robust support vector method for hyperspectral data classification and knowledge discovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1530–1542, Jul. 2004.

[29] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.

[30] G. Camps-Valls, L. Gómez-Chova, J. Muñoz-Marí, J. Vila-Francés, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.

[31] G. Camps-Valls, J. L. Rojo-Álvarez, and M. Martínez-Ramón, Eds., *Kernel Methods in Bioengineering, Signal and Image Processing*. Harshey, PA: Idea Group Inc., Jan. 2007.

[32] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with application in pattern recognition," *IEEE Trans. Electron. Comput.*, vol. 14, no. 3, pp. 326–334, Jun. 1965, IEEE Comput. Soc. Press. Reprinted in Artificial Neural Networks: Concepts and Theory, Los Alaminos, California, 1992, Eds. P. Mehra and B. Wah.

[33] D. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recognit. Lett.*, vol. 20, no. 11–13, pp. 1191–1199, Nov. 1999.

[34] B. Schölkopf, R. C. Williamson, A. Smola, and J. Shawe-Taylor, "Support vector method for novelty detection," in *Proc. Adv. Neural Inf. Process. Syst.*, Denver, CO, 1999, vol. 12.

[35] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philos. Trans. R. Soc. London A, Math. Phys. Sci.*, vol. CCIX, no. A456, pp. 215–228, May 1905.

[36] M. A. Aizerman, E. M. Braverman, and L. I. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Autom. Remote Control*, vol. 25, no. 6, pp. 821–837, 1964.

[37] M. Martínez-Ramón, J. L. Rojo-Álvarez, G. Camps-Valls, A. Navia-Vázquez, E. Soria-Olivas, and A. R. Figueiras-Vidal, "Support vector machines for nonlinear kernel ARMA system identification," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1617–1622, Nov. 2006.

[38] G. Camps-Valls, M. Martínez-Ramón, J. L. Rojo-Álvarez, and J. Muñoz-Marí, "Non-linear system identification with composite relevance vector machines," *IEEE Signal Process. Lett.*, vol. 14, no. 4, pp. 279–282, Apr. 2007.
[39] G. Camps-Valls, T. Bandos, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 2044–3054, Oct. 2007.
[40] G. Camps-Valls, L. Gomez-Chova, J. Muñoz-Marí, L. Alonso, J. Calpe-Maravilla, and J. Moreno, "Multitemporal image classification and change detection with kernels," in *Proc. SPIE Int. Symp. Remote Sens. XII*, Stockholm, Sweden, Sep. 2006, vol. 6365.
[41] J. Muñoz-Marí, L. Bruzzone, and G. Camps-Valls, "A support vector domain description approach to supervised classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 8, pp. 2683–2692, Aug. 2007.
[42] M. J. Barnsley, J. J. Settle, M. Cutter, D. Lobb, and F. Teston, "The PROBA/CHRIS mission: A low-cost smallsat for hyperspectral, multiangle, observations of the Earth surface and atmosphere," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1512–1520, Jul. 2004.
[43] I. Hajnsek, R. Bianchi, M. Davidson, and M. Wooding, "AgriSAR 2006—Airborne SAR and optics campaigns for an improved monitoring of agricultural processes and practices," *Geophys. Res. Abstr.*, vol. 9, 04085, 2007, Leuven, Belgium. SRef-ID: 1607-7962/gra/EGU2007-A-04085 European Geosciences Union 2007.
[44] P. Castracane, F. Iavaronc, S. Mica, E. Sottile, C. Vignola, O. Arino, M. Cataldo, D. Fernandez-Prieto, G. Guidotti, A. Masullo, and I. Pratesi, "Monitoring urban sprawl and its trends with EO data. UrbEx, a prototype national service from a WWF-ESA joint effort," in *Proc. 2nd GRSS/ISPRS Joint Workshop Remote Sens. Data Fusion Over Urban Areas*, 2003, pp. 245–248.
[45] L. Gómez-Chova, D. Fernández-Prieto, J. Calpe, E. Soria, J. Vila, and G. Camps-Valls, "Partially supervised hierarchical clustering of SAR and multispectral imagery for urban areas," in *Proc. SPIE Eur.*, Gran Canaria, Spain, Sep. 2004.
[46] A. Fanelli, M. Santoro, A. Vitale, P. Murino, and J. Askne, "Understanding ERS coherence over urban areas," in *Proc. ERSEnvisat Symp.*, 2000. In: ESA-SP-461 (Ed.), ERS-Envisat Symposium [CD-ROM].
[47] L. Gómez-Chova, D. Fernández-Prieto, J. Calpe, E. Soria, J. Vila, and G. Camps-Valls, "Urban monitoring using multitemporal SAR and multispectral data," *Pattern Recognit. Lett.*, vol. 27, no. 4, pp. 234–243, Mar. 2006.
[48] D. A. Clausi and B. Yue, "Comparing co-occurrence probabilities and Markov random fields for texture analysis of SAR sea ice imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 1, pp. 215–228, Mar. 2004.
[49] M. T. Hagan and M. Menhaj, "Training feed-forward networks with the Marquardt algorithm," *IEEE Trans. Neural Netw.*, vol. 5, no. 6, pp. 989–993, May 1994.
[50] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, no. 1, pp. 131–159, Mar. 2002.

Gustavo Camps-Valls (M'04–SM'07) was born in València, Spain, in 1972. He received the B.Sc. degree in physics in 1996, the B.Sc. degree in electronics engineering in 1998, and the Ph.D. degree in physics in 2002 from the Universitat de València, València, Spain.

He is currently an Associate Professor with the Departament d'Enginyeria Electrònica, Escola Tècnica Superior d'Enginyeria, Universitat de València, where he teaches electronics, advanced time series, and signal processing. He is the author and coauthor of several books and many published articles and is the editor of the book *Kernel Methods in Bioengineering, Signal, and Image Processing* (IGI, 2007). He is a Referee of many international journals. His research interests include neural networks and kernel methods for signal and image processing.

Dr. Camps-Valls currently serves on the Program Committees of SPIE Europe, the IEEE International Science and Remote Sensing Symposium, and IEEE International Conference on Image Processing.

Luis Gómez-Chova received the B.Sc. degree (first-class honors) in electronics engineering and the M.Sc. degree in electronic engineering in 2002 from the Universitat de València, València, Spain, where he is currently working toward the Ph.D. degree. His work is mainly related with pattern recognition and machine learning applied to remote sensing multispectral images.

Since 2000, he has been with the Departament d'Enginyeria Electrònica, Escola Tècnica Superior d'Enginyeria, Universitat de València, where he first enjoyed a research scholarship from the Spanish Ministry of Education and where he is currently a Lecturer.

Mr. Gómez-Chova was awarded by the Spanish Ministry of Education with the National Award for Electronic Engineering.

Jordi Muñoz-Marí was born in València, Spain, in 1970. He received the B.Sc. degree in physics in 1993, the B.Sc. degree in electronics engineering in 1996, and the Ph.D. degree in electronics engineering in 2003 from the Universitat de València, València, Spain.

He is currently an Associate Professor with the Departament d'Enginyeria Electrònica, Escola Tècnica Superior d'Enginyeria, Universitat de València, where he teaches analysis of circuits and linear systems, programmable logical devices, digital electronic systems, and electronic systems with microprocessors. His research interests include kernel methods for signal and image processing.

José Luis Rojo-Álvarez (M'01) received the Telecommunication Engineering degree from the University of Vigo, Vigo, Spain, in 1996 and the Ph.D. degree in telecommunication from the Polytechnical University of Madrid, Madrid, Spain, in 2000.

He is currently an Associate Professor with the Departmento de Teoría de la Señal y Comunicaciones, Universidad Rey Juan Carlos, Madrid, Spain. He is the author of many published articles on support vector machines and neural networks, robust analysis of time series and images, cardiac-arrhythmia mechanisms, and Doppler echocardiographic images for hemodynamic-function evaluation. His main research interests include statistical learning theory, digital signal processing, and complex system modeling, with applications both to cardiac signal and image processing and digital communications.

Manel Martínez-Ramón (M'01–SM'04) received the degree in telecommunications engineering from the Universitat Politècnica de Catalunya, Barcelona, Spain, in 1994 and the Ph.D. degree in telecommunications engineering from the Universidad Carlos III de Madrid, Madrid, Spain, in 1999.

He is currently with the Departmento de Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid. He is the author of several books on applications of support vector machines to antennas and electromagnetics and is the coauthor of many published articles. His research interests are the applications of statistical learning to signal processing, with emphasis in communications and brain imaging.