Encoding Invariances in Remote Sensing Image Classification With SVM

Emma Izquierdo-Verdiguier, *Student Member, IEEE*, Valero Laparra, Luis Gómez-Chova, *Member, IEEE*, and Gustavo Camps-Valls, *Senior Member, IEEE*

***Abstract*—This letter introduces a simple method for including invariances in support-vector-machine (SVM) remote sensing image classification. We design explicit invariant SVMs to deal with the particular characteristics of remote sensing images. The problem of including data invariances can be viewed as a problem of encoding prior knowledge, which translates into incorporating informative support vectors (SVs) that better describe the classification problem. The proposed method essentially generates new (synthetic) SVs from the obtained by training a standard SVM with the available labeled samples. Then, original and transformed SVs are used for training the virtual SVM introduced in this letter. We first incorporate invariances to rotations and reflections of image patches for improving contextual classification. Then, we include an invariance to object scale in patch-based classification. Finally, we focus on the challenging problem of including illumination invariances to deal with shadows in the images. Very good results are obtained when few labeled samples are available for classification. The obtained classifiers reveal enhanced sparsity and robustness. Interestingly, the methodology can be applied to any maximum-margin method, thus constituting a new research opportunity.**

***Index Terms*—Image classification, invariance, support vector machine (SVM).**

I. Introduction

THE support vector machine (SVM) classifier has been successfully used in remote sensing image classification during the last decade [1], [2]. The method has been widely exploited for contextual image classification [3], multitemporal classification and change detection [4], or spectral mixing analysis [5], [6]. Nowadays, the SVM is a standard tool for remote sensing data processing and image classification. In addition, during the last years, the basic SVM formulation has been modified to deal with unlabeled pixels in the images [7], [8] interacting with an expert user [9], and to combine different sources of information efficiently [3], [4], to name just a few.

All the previous developments can be seen as *blind* approaches to the general problem of including *prior knowledge* in the classifier. In semisupervised learning approaches, for instance, unlabeled samples may help in better defining the data support, which is typically poorly sampled in remote sensing applications. In the case of active learning, interaction with a user is the most naive form of incorporation knowledge as manual labeling corrects the posterior probability provided by a limited classifier. Finally, relying on the classical smoothness assumption of natural images, it is reasonable to regularize the solution by including contextual, multisource, multiangular, or multitemporal information. In this context, including prior knowledge is related to encoding *invariances* in the classifier and also to including a proper *regularization* scheme. Note that most of the aforementioned approaches concentrate on modifying the kernel function in SVMs, which essentially translates into enforcing smoothness via a particular regularizer (or prior function).

Classifiers should be robust to changes in the data representation. The property of such mathematical functions is called "invariance," and the algorithm is referred to as being "invariant," i.e., its decision function should be unaltered under transformations of data objects. The problem of encoding invariances in remote sensing and image processing applications is ubiquitous. An algorithm for biophysical retrieval estimation should be resistant (invariant) to illumination changes and to canopy spectral invariants. Similarly, a classifier should be invariant to rotations of patches, to changes in illumination and shadows, or to the spatial scale of the objects to be detected. The question raised here is how to include *any* kind of prior invariant behavior into a large margin classifier.

Different ways of incorporating invariances in an SVM were originally presented in [10]–[12]. Recently, other methods have been presented: Walder and Lovell [13] proposed a penalization of the variance of the decision function across similar class memberships, whereas in [14], the classifier is forced to be invariant to permutations of subelements within each sample. The work in [11] considers two main solutions to the invariance problem: designing particular kernel functions that encode local invariance under transformations or generating artificial examples from the selected support vectors (SVs) and training an SVM with them. The latter method is informally named virtual SVM (VSVM) and, because of its simplicity and effectiveness, is the one studied in this letter in the context of remote sensing image classification. Nevertheless, we will see that, by properly encoding the notion of data invariance, the method implicitly implements the ideas in [13] and [14].

The remainder of this letter is organized as follows. Section II briefly reviews the standard formulation of SVM and the proposed VSVM as a way to deal with the invariance problem. Section III shows experimental results on three problems:

Manuscript received June 20, 2012; revised September 10, 2012; accepted October 5, 2012. This work was supported by the Spanish Ministry of Economy and Competitiveness under Project AYA2008-05965-C04-03, Project CSD2007-00018, and Project TIN2012-38102-C03-01 (LIFE-VISION).

The authors are with Image Processing Laboratory, University of Valencia, 46980 Paterna, Spain (e-mail: emma.izquierdo@uv.es; valero.laparra@uv.es; luis.gomez-chova@uv.es; gustavo.camps@uv.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2012.2227297

encoding invariances to rotations and reflections of image patches for contextual classification, to the different scales of objects in the land cover detection, and to illumination changes to deal with shadows in the images. Section IV concludes the latter with some remarks and future work.

II. Invariances in SVM

The SVM is one of the most successful examples of kernel methods, being a linear classifier that implements maximum margin separation between classes in a high dimensional space. Given a labeled training data set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$, and given a nonlinear mapping $\mathbf{\Phi}(\cdot)$, the SVM classifier 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

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

$$\xi_i \geq 0 \qquad \forall i = 1, \ldots, n \tag{3}$$

where $\mathbf{w}$ and b define a maximum margin linear classifier in the feature space, and ξ_i are positive slack variables that are enabled to deal with permitted errors. An appropriate choice of nonlinear mapping $\mathbf{\Phi}(\cdot)$ guarantees that the transformed samples are more likely to be linearly separable in the feature space. Parameter C controls the complexity (regularization) of the model, and it must be selected by the user. The primary problem of (1) is to solve using its dual-problem counterpart [15], 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 are Lagrange multipliers found by optimization, and they have the useful property of being nonzero for the samples lying on the margin, which are called the SVs [15]. It is important to note here that the classifier is solely defined by the SVs and the corresponding weights. In the labeling (or classification) phase, one compares the new incoming example to all SVs in the solution.

The VSVM implements invariances in a very simple and intuitive way. The method basically consists of three steps.

1) Train the standard SVM with the available training data, and find the corresponding SVs.
2) Perturb the features of the found SVs to which the solution should be invariant. This procedure generates a set of virtual SVs (VSVs).
3) Train a new SVM with both SVs and VSVs.

The method is intuitively illustrated in Fig. 1. The VSVM is general enough to encode any prior knowledge about the invariance of the classifier to specific features. The method was originally applied for handwritten digit recognition applications, in which the classifier should be invariant to rotation of the digits [10], [11]. Nevertheless, as we will see in the following, encoding more challenging invariances may be harder in remote sensing image analysis.

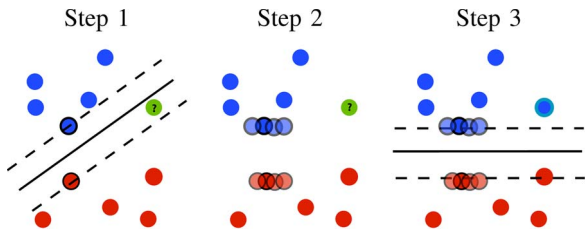

Fig. 1. VSVM in a binary toy example. The prior knowledge that we want to encode is that the classification function should be invariant to transformations of the horizontal feature. The first step of the algorithm gives a wrong prediction to the green test sample because it does not fulfill the invariance assumption since the sample should belong to the blue class. The second step generates a set of *VSVs* from the two found before. Intuitively, the idea is to construct a more meaningful hyperplane by forcing the presence of SVs in those regions to which the classifier should be invariant. By training an SVM again with both SVs and VSVs, a correct hyperplane is obtained in the third step.

III. Experimental Results

This section presents experimental evidence of the performance of the VSVM in three challenging remote sensing image classification problems that need the incorporation of different kinds of invariance.

A. Experimental Setup

In all the experiments, we compare the standard SVM and the VSVM, both using the standard radial basis function kernel with length scale σ. A tenfold cross-validation procedure was run to find the optimal SVM parameters $\sigma \in [10^{-2}, \ldots, 10^2]$, $C \in [1, \ldots, 10^3]$. In all cases, the one-versus-one multiclass scheme implemented in LibSVM [16] was used.[1] We report different figures of merit: overall accuracy (OA, in percentage), the estimated Cohen's kappa statistic κ, and the rate of SVs used after training both SVM and VSVM.[2] We train the models with different numbers of labeled examples *per* class, and show results in an independent test set. In particular, we compare the mean and standard deviation of a total of random training sample selections, and test for statistical significance of the differences between models. Depending on the invariance to be encoded, specific parameters are varied and discussed in the following. In addition, note that, in multiclass scenarios, the VSVs should be generated for the class that mainly encodes the invariance only. The Matlab source code is available at http://isp.uv.es/soft.htm for those interested readers.

B. Experiment 1: Invariance to Rotations

In the first experiment, we used a QuickBird image of a residential neighborhood of Zürich, Switzerland. The image was acquired in August 2002. Its size is 329 × 347 pixels and has a spatial resolution of 2.4 m. A total of 40 762 pixels were

[1] Available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[2] Note that comparing the total number of SVs would constitute an unfair measure because the VSVM will always use by definition more training samples, i.e., SVs plus VSVs.

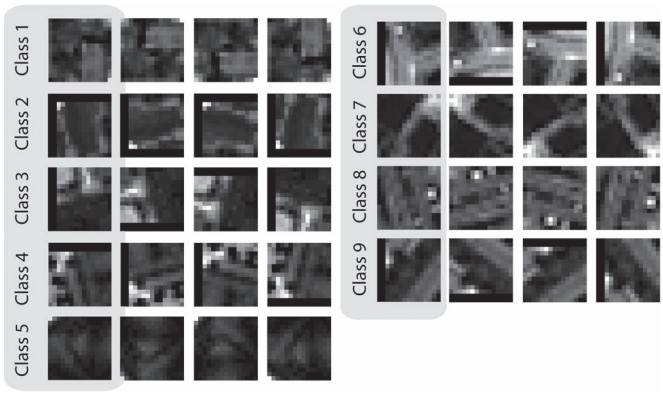

Fig. 2. Example for encoding invariance to rotation. The leftmost patches (shaded) are illustrative SVs of the nine different classes obtained in the first classification round by the SVM in the QuickBird image. From these SVs, we generate the three rightmost patches (called VSVs) by rotations and reflections.

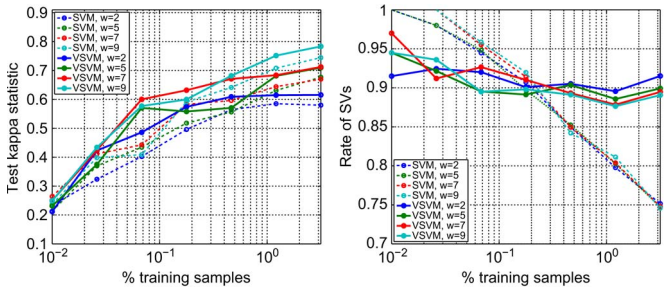

Fig. 3. (Left) Kappa statistic κ and (right) SV rate (in percentage) as a function of the number of training samples and window size w.

labeled by photointerpretation and were assigned to nine different land-use classes, such as soil, buildings, parkings, meadows, vegetation, roads, etc. Fig. 3 shows an RGB composite and the ground truth (GT) available.

To enhance the performance of classifiers in some particular classes, morphological top-hat features[3] were computed for the four bands and stacked to the multispectral bands before training the models. We performed patch-based classification: The image is divided into disjoint squared windows (patches) of size w, and each block is converted into a vector containing as features the pixels in the window. These vectors are used for classification, and its corresponding label is that of the center pixel in the patch. This is a very effective method to impose spatial smoothness in the classifier, and it is a procedure widely used in computer vision applications. Different patch sizes were considered, i.e., $w \in \{3, 5, 7, 9\}$. In this setting, the VSVs were generated by essentially rotating and reflecting the patches corresponding to the SVs, as shown in Fig. 2. We assume that the classifier should be invariant to the rotation or reflection of a patch, provided that the patch size contains enough information about the class characteristics.

Fig. 3 shows the performance of the methods as a function of the used number of training samples and window size. Accuracy results show that, in general, the VSVM performs better than the SVM for all window sizes, but the gain is slightly higher with larger window sizes. As window size increases,

[3]We used, as structural elements, squares and disks of sizes 5, 7, and 9, thus yielding a total of 24 spatial features.

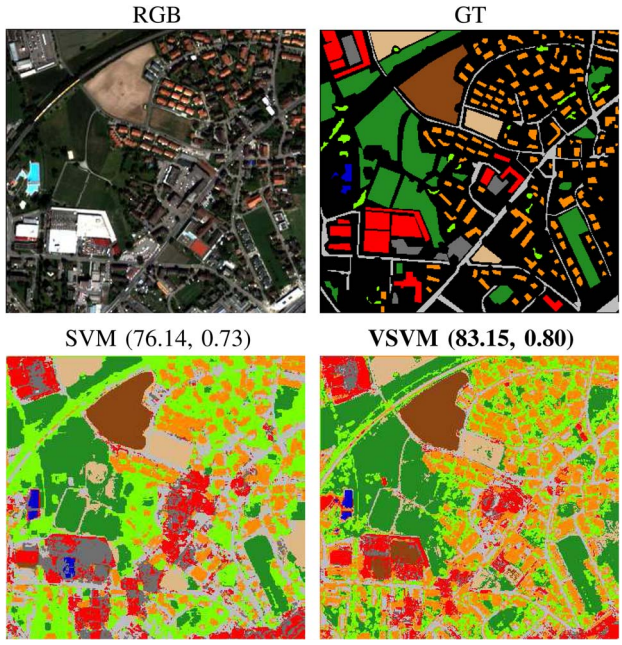

Fig. 4. (Top) True-color composite (RGB) and GT used in experiment 1 of patch-based classification. (Bottom) Classification maps of the experiments using a standard SVM and VSVM with 500 training samples selected spatially disjoint. Best results are shown in parentheses in the form of (OA (in percentage), κ).

pixels from different classes are included as features for the classifiers (and also used for generating VSVs), which can eventually lead to decreased performance. Similar trends for different window sizes are observed for the standard SVM, but the curves of the proposed VSVM cross each other as more samples are included, thus suggesting an optimal window size for encoding this type of invariance in this image. Another interesting observation is that the rate of SVs obtained with the VSVM is roughly constant for all training data set sizes, suggesting that the introduced virtual vectors are rich. However, the standard SVM in general leads to sparser models (remember that SVM uses a lower overall number of training vectors by definition). This turns to be even more noticeable with an increasing number of training samples.

Fig. 4 shows the accuracy results and classification maps obtained with the SVM and the VSVM for the specific case of using a total of 500 training patches and $w = 5$. Both classifiers show high classification scores, and the maps, in general, detect all major structures of the image. An improved numerical performance is obtained with the proposed VSVM (about $+7\%$ both in OA and κ). These results demonstrate the capabilities of the method to include invariances in the classifier but also show that properly encoding the invariance is of paramount importance.

C. Experiment 2: Invariance to Object Scales

In this experiment, we introduce invariance to object scale in SVM. This means that the same object with different sizes should be univocally classified. This illustrative example simply focuses on the binary problem of classifying image patches as "tree" or "bare soil." We used orthoimages of the Comunitat

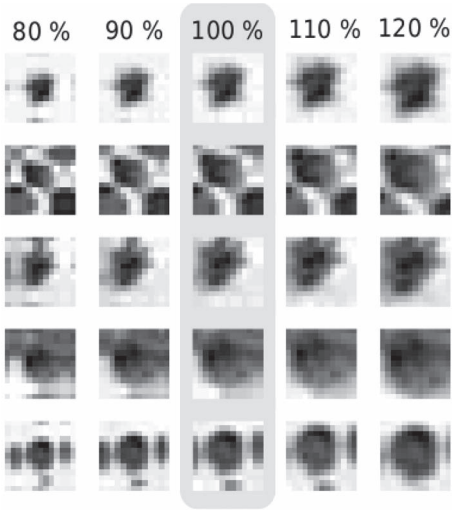

Fig. 5. Generation of VSVs by upscaling and downscaling five regular SVs (each one in the central shaded column).

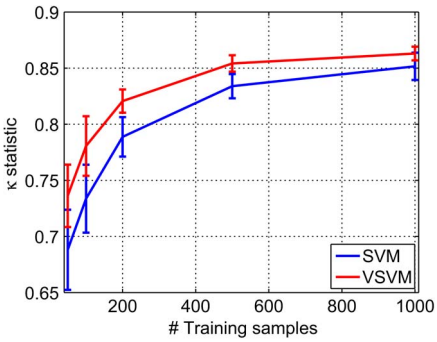

Fig. 6. Classification results for invariance to object scale for the (blue) standard SVM and (red) the VSVM as a function of the number of training samples. Error bars indicate confidence intervals at 95% over the average accuracy computed for 50% realizations.

Valenciana autonomous region (Spain) provided by the Instituto Cartográfico Valenciano. The images were acquired in 2007 using an airborne Vexcel UltraCam camera. The images have 0.5-m spatial resolution and four spectral bands (RGB and NIR). We generated a tree database with different classes (oranges, almond trees, olive trees, etc.) as well as uncultivated (bare soil) areas. Image patches of 13×13 pixels were used for classification.

The resizing operation was done by bicubic interpolation. Note that the VSVs must have the same dimensions as the original samples. In order to generate VSVs, we resized an SV (image) and then selected the central 13×13 pixels. When the output size is smaller than the original one (decreasing the object size), we increase first the size of the original image by using a mirror padding. A careful padding is needed to create realistic VSVs. It is also important to note that encoding the invariance can be done in this case for a particular feature (e.g., the image intensity) that alleviates the computational cost involved in the process. Fig. 5 shows examples of different SVs that give rise to generated VSVs.

Fig. 6 shows the results for different numbers of training samples for a fixed test set of 1000 different samples. We performed 50 realizations for each number of training samples.

TABLE I
NUMBER OF SVS, VSVS GENERATED, AND THE VSV FINALLY USED IN THE MODEL AS A FUNCTION OF THE NUMBER OF TRAINING SAMPLES

Training samples	50	100	200	500	1000
SVs	26	36	59	112	188
Generated VSVs	81	149	268	585	759
Selected VSVs	50	87	152	341	491

The scale limits was set to 50% and 120%, which are realistic values for trees. The amount of VSVs generated for each SV is different in each realization and has been tuned by cross-validation inside the training set. Results show that using the VSVs clearly improves the performance of the SVM. The average improvement with VSVMs is more noticeable with a low number of training samples, which suggests that the procedure helps in describing the class distributions properly. The average gain achieved is around $+5\%$ for all situations, and statistical significant differences between methods are observed (note that error bars showing confidence intervals at 95% do not generally overlap).

Table I shows the number of SVs and VSVs for each number of training samples. Reported results are the mean of the 50 realizations. Note how, although VSVM obtains better classification performance, the solution is less sparse.

D. Experiment 3: Invariance to Shadows

The third experiment deals with the segmentation of hyperspectral images. We used data acquired by an airborne ROSIS-03 optical sensor of the city of Pavia (Italy). The image consists of 102 spectral bands of size 1400×512 pixels with spectral coverage ranging from 0.43 to 0.86 μm. Spatial resolution of the scene is 1.3 m. Five classes of interest (buildings, roads, water, vegetation and shadows) have been considered, and a labeled data set of 206 009 pixels has been extracted by visual inspection. As in the previous examples, we performed patch-based classification. For this purpose, we only used 50 training patches of size $w = 5$. This example deals with a different, but rather common, problem in remote sensing images, i.e., the presence of shadows.

The study of the presence of shadows and how to remove them before image processing (e.g., biophysical parameter estimation or classification) has been long studied [17]. It is well known that the radiance ratio shadow/sunlit increases as the sunlight gets weaker, thus depending on the hour of the day; and the ratio is dependent on the wavelength, due to the direct and diffuse light proportions. The intensity of the shadows is also influenced by the spatial neighborhood. In [18], an exponential behavior of the ratio shadow/sunlit as a function of the wavelength was observed in the visible range of Quickbird. With these observations in mind, we encoded invariance to shadows by generating VSVs from exponentially modulated versions of the SVs, i.e., $\mathbf{x}_{\text{vsv}}(\lambda) = \mathbf{x}_{\text{sv}}(\lambda)\exp(-\gamma\lambda)$, where γ is a parameter that controls the impact of the spectral decay of the shadow/sunlit ratio as a function of the wavelength λ. We should note that only those SVs belonging to the class "shadow" were used to generate VSVs, resembling the invariant SVM in [14].

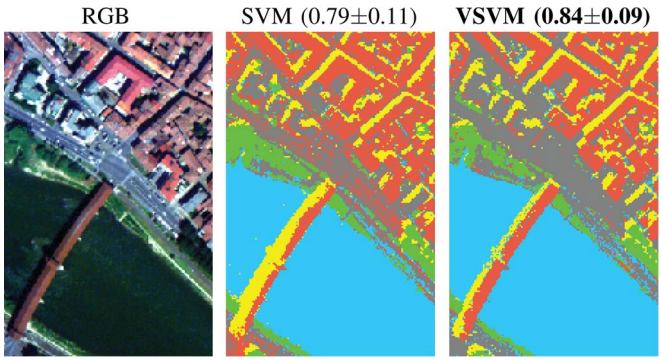

Fig. 7. True-color composite (RGB) used in the third experiment of patch-based classification and shadow invariance. Classification maps of the experiments using standard SVM and VSVM with 50 training pixels. Results are shown in parentheses in the form of (mean± standard deviation of κ in 20 realizations).

Numerical results and classification maps are reported in Fig. 7. Again, the VSVM leads to more accurate results than the standard SVM in this experiment. Essentially, we observe about $+5\%$ gain in κ and OA (not shown), and slightly more stable results for different realizations, even with a reduced number of examples. Looking at the classification maps in Fig. 7, it is however observed that encoding shadow invariance reports some improvements, particularly noticeable on the bridge and a more homogeneous classification on flat areas (see crossroads in the center of the image). The obtained numerical and visual results confirm the benefits of the invariance encoding in general. Nevertheless, we should note here that a statistical comparison between the solutions with a McNemar's test [19] did not show significant differences ($|z| < 1.96$). The marginal homogeneity assessed by McNemar's test (and many other statistical tests) assume independence between the pairs, which might not necessarily hold in this particular case: The SVs used in SVM are also included in the VSVM. We also used Wilcoxon's rank sum tests to assessed statistical differences, and results were similar to those obtained with the McNemar's test. Moreover, encoding shadow invariance in such a simple way may report some undesired effects in other classes, particularly on the class "vegetation" in this case. This effect and ways to impose cross invariances in SVMs will be studied in the future.

IV. Conclusion

We introduced a simple method to include data invariances in SVM remote sensing image classification. We illustrated the performance in relevant remote sensing problems: Invariance to rotations and reflections of image patches for contextual classification, object scale invariances, and including prior knowledge on the way shadows affect the acquired images. Good classification accuracy was obtained in general when few labeled samples were available for training the models. The obtained classifiers revealed in some cases enhanced sparsity and robustness properties. In the near future, we will study the impact of mislabeled samples on the VSVM. Other invariances, from translation to illumination and canopy spectral invariances, and other kernel methods, from regression to clustering methods, could be explored in the future. In all cases, the inclusion of physical-based models may lead to improved invariant classifiers. We consider that the promising results obtained here open a new and interesting research line.

Acknowledgment

The authors would like to thank Prof. P. Gamba from the University of Pavia, Italy for kindly providing both the ROSIS data, and Dr. D. Tuia from the Ecole Polytechnique Fédérale de Lausanne, Swizerland, for kindly providing the Brutisellen data.

References

[1] 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.

[2] G. Camps-Valls and L. Bruzzone, *Kernel Methods for Remote Sensing Data Analysis*. Hoboken, NJ: Wiley, 2009.

[3] 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.

[4] G. Camps-Valls, L. Gómez-Chova, J. Muñoz-Marí, J. Luis Rojo-Álvarez, and M. Martínez-Ramón, "Kernel-based framework for multi-temporal and multi-source remote sensing data classification and change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1822–1835, Jun. 2008.

[5] M. Brown, H. G. Lewis, and S. R. Gunn, "Linear spectral mixture models and support vector machines for remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 5, pp. 2346–2360, Sep. 2000.

[6] L. Wang and X. Jia, "Integration of soft and hard classifications using extended support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 3, pp. 543–547, Jul. 2009.

[7] L. Bruzzone, M. Chi, and M. Marconcini, "A novel transductive SVM for semisupervised classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3363–3373, Nov. 2006.

[8] L. Gómez-Chova, G. Camps-Valls, L. Bruzzone, and J. Calpe-Maravilla, "Mean map kernel methods for semisupervised cloud classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 1, pp. 207–220, Jan. 2010.

[9] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Muñoz-Marí, "A survey of active learning algorithms for supervised remote sensing image classifications," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 606–617, Jun. 2011.

[10] B. Schölkopf, C. Burges, and V. Vapnik, "Incorporating invariances in support vector learning machines," in *Proc. ICANN*, vol. 1112, *Lecture Notes in Computer Science*, 1996, pp. 47–52.

[11] D. DeCoste and B. Schölkopf, "Training invariant support vector machines," *Mach. Learn.*, vol. 46, no. 1–3, pp. 161–190, 2002.

[12] O. Chapelle and B. Schölkopf, "Incorporating invariances in nonlinear support vector machines," in *NIPS*. Cambridge, MA: MIT Press, 2002, pp. 609–616.

[13] C. J. Walder and B. C. Lovell, "Homogenized virtual support vector machines," in *Proc. DICTA*, Dec. 2005, pp. 57–63.

[14] P. K. Shivaswamy and T. Jebara, "Permutation invariant SVMs," in *Proc. 23rd ICML*, 2006, pp. 817–824.

[15] B. Schölkopf and A. Smola, *Learning With Kernels—Support Vector Machines, Regularization, Optimization and Beyond*. Cambridge, MA: MIT Press, 2002.

[16] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, Apr. 2011.

[17] G. D. Finlayson, S. D. Hordley, C. Lu, and M. S. Drew, "On the removal of shadows from images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 59–68, Jan. 2006.

[18] F. Yamazaki, W. Liu, and M. Takasaki, "Characteristics of shadow and removal of its effects for remote sensing imagery," in *Proc. IEEE Geosci. Remote Sens. Symp.*, Jul. 2009, vol. 4, pp. IV-426–IV-429.

[19] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, Jun. 1947.