Target Detection with Semi-supervised Kernel Orthogonal Subspace Projection

Luca Capobianco, Andrea Garzelli, Senior Member, IEEE, and Gustavo Camps-Valls, Senior Member, IEEE

Abstract

The Orthogonal Subspace Projection (OSP) algorithm is substantially a kind of matched filter that requires the evaluation of a prototype for each class to be detected. The kernel OSP (KOSP) has recently demonstrated improved results for target detection in hyperspectral images. The use of the kernel method makes the method non-linear, helps to combat the high dimensionality problem and improves robustness to noise. This paper presents a semi-supervised graph-based approach to improve KOSP. The proposed algorithm deforms the kernel by approximating the marginal distribution using the unlabeled samples. Two further improvements are presented. First, a contextual selection of unlabeled samples is proposed. This strategy helps in better modeling the data manifold and thus improved sensitivity-specificity rates are obtained. Second, given the high computational burden involved, we present two alternative formulations based on the Nyström method and the Incomplete Cholesky Factorization to achieve operational processing times. The good performance of the proposed method is illustrated in a toy dataset, and two relevant hyperspectral image target detection applications: crop identification and thermal hot spot detection. A clear improvement is observed with respect to the linear and the non-linear kernel-based OSP, demonstrating good generalization capabilities when low number of labeled samples are available, which is usually the case in target detection problems. The relevance of unlabeled samples and the computational cost are also analyzed in detail.

Manuscript received September 2008;

LC and AG are with Università degli Studi di Siena, Siena, Italy. Information Engineering Department, {capobianco,garzelli}@di.unisi.it

GCV is with Dept. Enginyeria Electrònica. Escola Tècnica Superior d’Enginyeria. Universitat de València. C/ Dr. Moliner, 50. 46100 Burjassot (València) Spain. E-mail: gustavo.camps@uv.es, http://www.uv.es/gcamps

This work has been partly supported by Spanish Ministry of Education and Science projects AYA2008-05965-C04-03, and CONSOLIDER/CSD2007-00018.
Index Terms

Target detection, Orthogonal Subspace Projection (OSP), kernel method, KOSP, semi-supervised learning, graph, regularization, manifold learning.

I. INTRODUCTION

The field of remotely sensed hyperspectral image analysis comprises different learning paradigms, either supervised, unsupervised or semi-supervised. In the case of supervised classification, the user is given a number of labeled pixels belonging to different classes to develop a model that extrapolates well to unseen situations. Lately, high interest has been payed to 1) one-class classification, where one tries to detect one class and reject the others; 2) anomaly detection, where one tries to identify pixels differing significantly from the background; and 3) target detection, where the target spectral signature is assumed to be known (or available from spectral libraries), and the goal is to detect pixels that match the target.

Target detection from hyperspectral data is of great interest in many applications. Detecting targets in remote sensing images is typically described as a two-steps methodology, in which first an anomaly detector identifies spectral anomalies, and second a classifier is aimed at identifying whether or not the anomaly is a target or natural clutter. This step is only possible if the target spectral signature is known, which can be obtained from a spectral library or by using a spectral subspace matched filter learnt from the data. Several techniques have been proposed in the literature, such as the Reed-Xiaoli anomaly detector [1], the orthogonal subspace projection (OSP) [2], the Gaussian mixture model [3], the cluster-based detector [3], or the signal subspace processor [4].

All of them assume a parametric (linear or Gaussian mixture) model. Even though linearity is a convenient assumption, it is far from being realistic. Nonlinearity appears in different forms in remote sensing data [5], [6]: 1) the nonlinear scatter described in the bidirectional reflectance distribution function (BRDF); 2) the variable presence of water in pixels as a function of position in the landscape; 3) multiscattering and heterogeneities at subpixel level; and 4) atmospheric and geometric corrections that need to be done to deliver useful image products. These facts need to be encoded in the method either by allowing flexible non-linear mappings or by accurate description of the data manifold coordinates. In fact, the nonlinear relationships between different spectral bands within the target or clutter spectral signature need to be exploited in order to better
distinguish between target and background. However, most of the target detection algorithms are based on linear matched (subspace) filters where the spectral characteristics of a target or a target subspace representing target information is assumed to be known.

Non-linear supervised classifiers such as support vector machines (SVMs) [7], [8] excel in using the labeled information, being (regularized) maximum margin classifiers also equipped with an appropriate loss function [9], [10]. However, practical experience has shown that the design of a good classifier requires a sufficient amount of training data for each background class and that, usually, in operational remote sensing applications, a small number of labeled samples is typically available. Besides, in target detection applications, the main objective is to search a specific material (target), usually within very small number of pixels.

Conceptually, target detection can be viewed as a binary hypothesis testing problem, where each pixel is assigned a target or non-target label. In this way, target detection problems can be casted as ill-posed classification problems, thus being in the middle between canonical classification and target detection framework. For multiple targets, detection can be approached as a multiclass pattern recognition problems in which each class appears as a target ‘lying’ in the background. In the last years, many detection algorithms based on spectral matched (subspace) filters have been reformulated under the kernel methods framework: matched subspace detector (MSD), orthogonal subspace detector (OSD), spectral matched filter (SMF), and adaptive subspace detectors (ASD) [11]. Certainly, the use of kernel methods offers many advantages: they combat the high dimensionality problem in hyperspectral images, make the method robust to noise, and allow for flexible non-linear mappings with controlled (regularized) complexity [9]. Kernel methods in general, and kernel detectors in particular, rely on the proper definition of a kernel (or similarity) matrix among samples. So far, however, standard ‘ad hoc’ RBF or polynomial kernels have been used, and no attention has been payed to model the data marginal distribution that would be potentially helpful to design the kernel structural form. In addition, kernel detectors have only considered labeled information, and unlabeled samples from guard windows are merely used as contrast density in hypothesis testing.

In this paper, we introduce a semi-supervised technique that collectively incorporates labeled and unlabeled data in the target detection framework. In semi-supervised learning (SSL) [12], [13], the algorithm is provided with some available labeled information in addition to the unlabeled information, thus allowing to encode some knowledge about the geometry and the
shape of the dataset. This idea of exploring the shape of the marginal distribution in the dataset can be applied in kernel target detection in order to deform the ‘measure’ of distance in the kernel space according to the geometry of the neighboring pixels.

We propose a Semi-Supervised Kernel Orthogonal Subspace Projection (S$^2$KOSP), which introduces an additional regularization term on the geometry of both labeled and unlabeled samples by using the graph Laplacian [12]. The presented approach may constitute a useful framework to make semi-supervised other kernel-based target detection methods. Performance is illustrated in both a synthetic dataset and real hyperspectral images, paying special attention to ill-posed problems, where only few labeled samples are available.

The remainder of the paper is outlined as follows. Section 2 is dedicated to revise the canonical expression of linear and kernel-based OSP. Section 3 presents the proposed formulation of the S$^2$KOSP. Noting that the computational cost of the method is high when the number of unlabeled samples increases, two alternative formulations are proposed. Section 4 shows the experimental results in synthetic and real scenarios, analyzes the impact of the free parameters, the relevance of the unlabeled samples, and proposes a contextual method for unlabeled sample selection. The computational efficiency of the two algorithmic implementations is analyzed and some remarks on the use are advocated. Section 5 concludes and outlines the further work.

II. KERNEL ORTHOGONAL SUBSPACE PROJECTION ALGORITHM

This section reviews the orthogonal subspace projection (OSP) method and the non-linear kernel-based version proposed in [11].

A. Orthogonal Subspace Projection (OSP) Algorithm

In the standard formulation of the OSP algorithm [2], a linear mixing is assumed to model each $B$-bands pixel $r$ as follows:

$$r = M\alpha + n,$$

where $M$ is the matrix of size $(B \times p)$ containing the $p$ endmembers contributing to the mixed pixel $r$, $\alpha$ is a $(p \times 1)$ column vector of the coefficients that account for the spectral abundance of each endmember, and $n$ stands for an additive zero mean Gaussian noise vector.
In order to identify one particular signature in the images, and given its spectral signature \( \mathbf{d} \) with corresponding abundance measurements \( \alpha_p \), the above expression can be organized by rewriting the \( \mathbf{M} \) matrix in two submatrices \( \mathbf{M} = (\mathbf{U} : \mathbf{d}) \), so that

\[
\mathbf{r} = \mathbf{d}\alpha_p + \mathbf{U}\gamma + \mathbf{n}.
\]

The columns of \( \mathbf{U} \) represent the undesired spectral signatures (background), while the \( \gamma \) represents the abundance for the undesired spectral signatures.

The effect of the OSP algorithm on the data set can be summarized in two steps (see Fig. 1). First, an annihilating operator rejects the background signatures for each pixel, so that only the desired signature should remain in the spectral component of the data. This operator is given by the \((B \times B)\) matrix \( \mathbf{P}_U^\perp = \mathbf{I} - \mathbf{U}\mathbf{U}^\# \), where \( \mathbf{U}^\# \) is the right Moore-Penrose pseudoinverse of \( \mathbf{U} \). The second step of the OSP algorithm is represented by the filter \( \mathbf{w} \) that maximizes the SNR of the filter output, that is the matched filter \( \mathbf{w} = k\mathbf{d} \) where \( k \) is a constant [14], [15].

The OSP operator is given by \( \mathbf{q}_{OSP}^\top = \mathbf{d}^\top \mathbf{P}_U^\perp \), and the output of the OSP classifier is:

\[
\mathbf{D}_{OSP} = \mathbf{q}_{OSP}^\top \mathbf{r} = \mathbf{d}^\top \mathbf{P}_U^\perp \mathbf{r}.
\]

By using the singular value decomposition (SVD) of \( \mathbf{U} = \mathbf{B}\Sigma\mathbf{A}^\top \), the annihilating operator becomes \( \mathbf{P}_U^\perp = \mathbf{I} - \mathbf{B}\mathbf{B}^\top \), where the columns of \( \mathbf{B} \) are obtained from the eigenvectors of the covariance matrix of the background spectral samples.
B. From Linear to Non-linear Algorithms with Kernels

Kernel methods embed the data set \( S \) defined over the input or attribute space \( \mathcal{X} \) \((S \subseteq \mathcal{X})\) into a higher dimensional Hilbert space \( \mathcal{H} \), or feature space, and then a linear algorithm is built therein, resulting in an algorithm which is nonlinear with respect to the input data space. The mapping function is denoted as \( \Phi : \mathcal{X} \rightarrow \mathcal{H} \). If a given algorithm can be expressed in the form of dot products in the input space, its (non-linear) kernel version only needs the dot products among mapped samples.

Kernel methods compute the similarity between training samples \( S = \{x_i\}_{i=1}^n \) using pair-wise inner products between mapped samples, and thus the so-called kernel matrix \( K_{ij} = K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \) contains all the necessary information to perform many classical linear algorithms in the feature space.

The bottleneck for any kernel method is the proper definition of a kernel function that accurately reflects the similarity among samples. However, not all metric distances are permitted. In fact, valid kernels are only those fulfilling the Mercer’s Theorem [16], [17] and the most common ones are the linear \( K(x, z) = \langle x, z \rangle \), the polynomial \( K(x, z) = (\langle x, z \rangle + 1)^d, d \in \mathbb{Z}^+ \), and the Radial Basis Function (RBF), \( K(x, z) = \exp \left( -\|x - z\|^2/2\sigma^2 \right), \sigma \in \mathbb{R}^+ \).

Therefore, one can design kernels by direct sum or product of (weighted) valid kernels.

C. Kernel Orthogonal Subspace Projection (KOSP)

A non-linear kernel-based version of the OSP algorithm can be devised by defining a linear OSP in an appropriate Hilbert feature space where samples are mapped to through a kernel mapping function \( \Phi(\cdot) \). Similarly to the previous linear case, let us define the annihilating operator now in the feature space as \( P_{\Phi}^\perp = I_{\Phi} - B_{\Phi}B_{\Phi}^\top \). Then, the output of the OSP classifier in the feature space is now trivially given by

\[
D_{OSP_{\Phi}} = \Phi(d)^\top (I_{\Phi} - B_{\Phi}B_{\Phi}^\top)\Phi(x). \tag{4}
\]

The columns of matrix \( B_{\Phi} \), say \( b_\Phi^i \), are the eigenvectors of the covariance matrix of the undesired background signatures. Clearly, each eigenvector in the feature space can be expressed as a linear combination of the input vectors in the feature space transformed through the function \( \Phi(\cdot) \). Hence, one can write \( b_\Phi^i = X_{b_\Phi}^i \beta^I \), where the columns of \( X_{b_\Phi} \) are the vectors in the feature space corresponding to the background spectral signatures \( X_b \); \( \beta^I \) are the eigenvectors
of the (centered) Gram matrix $K(X_b, X_b)$ normalized by the square root of its corresponding eigenvalues.

The kernelized version of (4) is given by \[ D_{KOSP} = K(X_{bd}, d)^\top \Upsilon \Upsilon^\top K(X_{bd}, r) - K(X_b, d)^\top B B^\top K(X_b, r), \] (5)

where $K(X_b, r)$ and $K(X_b, d)$, referred to as the empirical kernel maps in the machine learning literature, are column vectors whose entries are $K(x_i, r)$ and $K(x_i, d)$ for $x_i \in X_b$ ($x_i \in \mathbb{R}^B$), $i = 1, \ldots, l$, being $l$ the number of labeled samples; $B$ is the matrix containing the eigenvectors $\beta^j$ described above; and $\Upsilon$ is the matrix containing the eigenvectors $\upsilon^j$, similar to $\beta^j$, but obtained from the centered kernel matrix $K(X_{bd}, X_{bd})$, where $X_{bd} = X_b \cup d$.

### III. Proposed Semi-Supervised KOSP

In this section, we pay attention to the appropriate definition of the kernel under semi-supervised criteria. Essentially, we propose to deform the kernel using the graph Laplacian. This idea, which was originally presented in [18] for inductive SVM, has been recently presented for one-class classification [19] and support vector regression [20]. Here, it is extended to the kernel OSP.

#### A. Learning a Suitable Kernel from Unlabeled Samples

The performance of any kernel method strongly depends on the adequate definition of the kernel structural form, which can be casted as a similarity (or distance) measure among samples. Traditionally, the kernel form has been chosen to be either linear, polynomial, or Radial Basis Function (RBF) to fulfil Mercer’s conditions. Particularly interesting is the RBF Gaussian kernel since it has less numerical difficulties and only the Gaussian width has to be tuned. In addition, the RBF kernel includes other valid kernels as particular cases [21].

However, despite the good performance offered, by imposing such ‘ad hoc’ signal relations, the underlying data structure is obviated. To properly define a suitable kernel, unlabeled information and geometrical relationships between labeled and unlabeled samples may be useful to ‘deform’ the core measure of distance, i.e. to estimate a likelihood kernel according to the unlabeled data structure that modifies the assumed prior kernel encoding signal relations. This is in the roots of
semi-supervised learning. Essentially, two different classes of SSL algorithms are encountered in the recent literature. Firstly, *generative models* involve estimating the conditional distribution by modeling the class-conditional distributions explicitly, such as expectation-maximization (EM) algorithms with finite mixture models [22], [23]. These clustering methods have been extensively applied in the context of remotely sensed image classification since the Gaussian nature of data is a reasonably observed feature [23], [24]. Secondly, *discriminative models*, in contrast to generative models, estimate the conditional distribution directly, and one does not need to specify the class-conditional distributions explicitly. Two subgroups of SSL algorithms can be distinguish within these models:

- *Low density separation algorithms* maximize the margin for labeled and unlabeled samples simultaneously, such as Transductive SVM (TSVM) [25], [26]; and
- *Graph-based methods*, in which each sample spreads its label information to its neighbors until a stable state is achieved on the whole dataset [27], [28].

In this paper, we focus on the later graph-based methods to define a kernel adapted to the geometry of the data marginal distribution. Essentially, we aim at exploiting the relation between labeled and unlabeled samples through the construction of a graph representation, where the vertices are the (labeled and unlabeled) samples, and the edges represent the similarity among samples in the dataset.

**B. Deforming the Kernel with the Graph Laplacian**

In order to include the geometry of the data distribution as a similarity measure in the model, let us first define a linear space $\mathcal{V}$ with a positive semi-definite inner product, and let

$$S : \mathcal{H} \rightarrow \mathcal{V}$$

be a bounded linear operator. We can now define $\tilde{\mathcal{H}}$ to be the space of functions from $\mathcal{H}$ with the modified inner product $\langle f, g \rangle_{\tilde{\mathcal{H}}} = \langle f, g \rangle_{\mathcal{H}} + \langle Sf, Sg \rangle_{\mathcal{V}}$. Semi-norm $\|S(f)\|_{\mathcal{V}}^2 = f^T N f$ is given by a symmetric semi-definite matrix $N$ and the decision function $f$ is given by

$$f = \text{sgn}(D_{\text{KOSP}} - \theta),$$

where $\theta$ is the decision threshold. If $u$ is the number of unlabeled samples available, the explicit form of the corresponding reproducing kernel function $\tilde{K}(x_i, x_j)$ can be explicitly defined over
Fig. 2. Graph classification on a toy graph. (a) The two shaded circles are the initially labeled vertices ($\pm 1$), while the white nodes represent unlabeled samples. The thickness of the edges represent the similarity among samples, easily computable with a proper distance measure. (b) Undirected graph methods classify the unlabeled samples according to the weighted distance, not just to the shortest path lengths, the latter leading to incorrectly classified samples. The two clusters (shaded) are intuitively correct, even being connected by (thin weak) edges.

labeled samples as [18]:

$$\tilde{K}(x_i, x_j) = K(x_i, x_j) - K_{x_i}^\top (I + NK)^{-1}NK_{x_j},$$

(6)

where $i, j \in \{1, \ldots, l + u\}$; $K$ is the (complete) kernel matrix; $I$ is the identity matrix; and $K_{x_i} = [K(x_1, x_i), \ldots, K(x_{l+u}, x_i)]^\top$. See Appendix A for detailed derivation of (6).

In this formulation the geometry of the data is included through $N$, usually defined proportional to the graph Laplacian [12], that is $N = \gamma L$, where $\gamma \in [0, \infty)$ is a free parameter that controls the ‘deformation’ of the kernel. To define $L$, let’s first define a graph $G(V, E)$ with a set of $n$ nodes, $V$, connected by a set of edges, $E$. The edge connecting nodes $i$ and $j$ has an associated weight, $W_{ij}$ [12]. In this framework, the nodes are the samples, and the edges represent the similarity among samples in the dataset (see Fig. 2). A proper definition of the graph is the key to accurately introduce data structure in the machine.

Two mathematical tools have to be introduced to understand how matrix $L$ is constructed [14], [27], [28]:

- $D$ is the degree matrix of size $n \times n$. Basically, $D$ is a diagonal matrix $D = [d_1, \ldots, d_n]$ containing the number of connections to a node (degree);
- $A$ is the adjacency matrix of size $n \times n$, where the nondiagonal entry is the number of connection from node $i$ to node $j$, and the diagonal entry is either twice the number of
loops at vertex $i$ or just the number of loops. In our case, it is a matrix containing only $(0, 1)$.

Finally, the Laplacian matrix $L$ is defined as $L = D - W$, where $W$ is obtained from $A$, the adjacency matrix, by assigning weights to each connection. Also, a normalized version of $L$ can be obtained as

$$L_{ij} = \begin{cases} 1 & \text{if } i = j \text{ and } d_j \neq 0 \\ -\frac{1}{\sqrt{d_id_j}} & \text{if } i \text{ and } j \text{ are adjacent} \\ 0 & \text{otherwise.} \end{cases}$$

where subscripts $i$ and $j$ stand for the row and column indexes as well as the edges as defined before.

Intuitively, $L$ measures the variation of the decision function $f$ along the graph built upon all (labeled and unlabeled) samples [12]. Note that, by fixing $\gamma = 0$, the original (undeformed) kernel is obtained. Therefore, a proper selection of this free parameter theoretically leads to better results than the pure supervised approach (the KOSP method in our case). By plugging (6) into (5), the semi-supervised KOSP, called $S^2$KOSP, can be written as

$$D_{KOSP}^{ss} = D_{KOSP} - [K_{x_i}(I + NK)^{-1}NK_{x_i}]_d,$$

where the subscript $d$ indicates the operation of extracting information relative to the desired target signature $d$.

C. Efficient Computation of the Deformed Kernel

Solving the new semi-supervised problem is computationally demanding. Firstly, two more parameters need to be adjusted: the number of neighbors in the graph Laplacian, $NN$, and the amount of introduced deformation, $\gamma$. Tuning the parameters implies tuning different values, and for each combination a matrix inversion of size $(l + u) \times (l + u)$ must be computed (see Eq. (6)). Note that this inversion scales exponentially with the number of used samples so one pays the cost of including more unlabeled samples to better model the data marginal distribution. We should stress that, with an adequate selection of (informative) labeled samples, the need of many unlabeled is limited. Also, there are techniques that make an approximate calculation of the inverse matrix at a lower cost. In this paper we present two alternative formulations to speed up the matrix inversion and make feasible both training and free parameter tuning for large scale datasets.
1) The Nyström Method: Rather than a direct matrix inversion, we propose to retain only the first largest \( p \) eigenvalues of the eigen-decomposition of matrix \( \mathbf{S} = \mathbf{L} \mathbf{K} = \mathbf{V} \mathbf{A} \mathbf{V}^\top \), where \( \mathbf{V} \) represents the unitary matrix of eigenvectors and \( \mathbf{A} \) is a diagonal matrix containing their associated eigenvalues. There are methods to find the first eigenvalues without explicitly solving the whole eigenproblem [29]. However, computational time is drastically reduced only when \( p \ll n \), being \( n = l + u \).

In order to reduce the computational cost involved, we introduce here the Nyström method [30]. The Nyström method is commonly used to produce an approximate matrix \( \tilde{\mathbf{S}} \) by randomly choosing \( m \) rows/columns of the original matrix \( \mathbf{S} \) and then making \( \tilde{\mathbf{S}}_{n,n} = \mathbf{S}_{n,m} \mathbf{S}_{m,m}^{-1} \mathbf{S}_{m,n} \), \( m \leq n \), where \( \mathbf{S}_{n,m} \) represents the \( n \times m \) block of \( \mathbf{S} \). As a result, the method simplifies the solution of the problem to computing an approximated eigen-decomposition of the low-rank kernel matrix \( \tilde{\mathbf{S}} = \tilde{\mathbf{V}} \tilde{\mathbf{A}} \tilde{\mathbf{V}}^\top \), involving \( O(mn^2) \) computational cost. See Appendix B for the full formulation of the Nyström method.

Therefore, if we approximate the normalized matrix \( \mathbf{S} \) with the Nyström method by expanding a small \( p \times p \) matrix, \( \tilde{\mathbf{S}} = \tilde{\mathbf{V}} \tilde{\mathbf{A}} \tilde{\mathbf{V}}^\top \), and substitute it into (6), we obtain:

\[
\tilde{\mathbf{K}}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) - \mathbf{K}_x^\top(\mathbf{I} + \gamma \tilde{\mathbf{V}} \tilde{\mathbf{A}} \tilde{\mathbf{V}}^\top)^{-1} \mathbf{N} \mathbf{K}_{x_j},
\]

Now, by using the Sherman-Morrison-Woodbury formula\(^1\) from linear algebra in our problem statement, it is straightforward to demonstrate:

\[
\tilde{\mathbf{K}}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) - \mathbf{K}_x^\top(\mathbf{I} - \tilde{\mathbf{V}}(\gamma^{-1} \mathbf{I} + \tilde{\mathbf{A}} \tilde{\mathbf{V}}^\top)^{-1} \tilde{\mathbf{A}} \tilde{\mathbf{V}}^\top)^{-1} \mathbf{N} \mathbf{K}_{x_j},
\]

which involves inverting a matrix of size \( p \times p \) (with \( p \leq m \leq n \)) and thus the computational cost is \( O(p^2n) \), i.e. linear with the number of \( (l+u) \)-samples.

2) Incomplete Cholesky Factorization: A positive semidefinite matrix \( \mathbf{K} \) can be factorized as \( \mathbf{G} \mathbf{G}^\top \), where \( \mathbf{G} \) is an \( n \times n \) matrix. This factorization can be found via direct Cholesky decomposition, which can be casted as a kind of Gaussian elimination. The objective of the proposed method is, nevertheless, to find an approximate matrix \( \tilde{\mathbf{G}} \) of size \( n \times m \) for a small \( m \) such that the difference \( \| \mathbf{K} - \mathbf{G} \mathbf{G}^\top \| < \eta \), where \( \eta \) is the tolerance or permitted error. The Incomplete Cholesky Factorization (ICF) differs from the standard Cholesky decomposition in

\(^1\)The Sherman-Morrison-Woodbury formula states that \( (\mathbf{C} + \mathbf{AB})^{-1} = \mathbf{C}^{-1} - \mathbf{C}^{-1} \mathbf{A}(\mathbf{I} + \mathbf{BC}^{-1} \mathbf{A})^{-1} \mathbf{BC}^{-1} \), where \( \mathbf{C} \) is an invertible \( n \times n \) matrix, \( \mathbf{A} \in \mathbb{R}^{n \times m} \) and \( \mathbf{B} \in \mathbb{R}^{m \times n} \).
that all pivots below a certain threshold are skipped. If $m$ is the number of non-skipped pivots, the lower triangular matrix $\tilde{G}$ with only $m$ nonzeros columns is obtained.

The proposed iterative method involves picking one column of $K$ at a time. This is done by maximizing a lower bound on the error reduction of the approximation. After $t$ iterations, a good approximation $K_t = \tilde{G}_t\tilde{G}_t^\top$ is obtained, where $\tilde{G}_t$ is $n \times t$. The ranking of the $n \times t$ vectors that might be added in the next step is done by comparing the diagonal elements of the remainder matrix $K - \tilde{G}_t\tilde{G}_t^\top$. Each of these elements requires $O(t)$ operations, and the update of $\tilde{G}_t$ has $O(tn)$ cost, so the overall complexity is $O(m^2n)$.

Once the factorization is obtained, the new formulation of the algorithm is obtained by first substituting $LK = \tilde{G}\tilde{G}^\top$ in (6), and then applying the Sherman-Morrison-Woodbury formula:

$$\tilde{K}(x_i, x_j) = K(x_i, x_j) - K_x^\top(I - \gamma \tilde{G}(I + \gamma \tilde{G}^\top\tilde{G})^{-1}\tilde{G}^\top)MK_{x_j}.$$  (10)

This method has several nice properties: 1) the time complexity is roughly the same as the Nyström method but only one free parameter is needed; 2) the only elements of $K$ to be kept in memory are the diagonal elements of the kernel, that are equal to one for the RBF, and the rest of the elements can be computed on demand; and 3) the number of $m$ can be chosen to fit the approximation as tight as desired.

D. Remarks, assumptions, and limitations

The key issue in semi-supervised learning is the assumption of consistency, which means that nearby points are likely to have the same label, and points on the same structure (typically referred to as ‘cluster’ or ‘manifold’) are likely to have the same label. In our case, nearby points are those pixels spectrally similar and thus the assumption is applied to the space of image pixels. Therefore, depending on the data complexity, either the cluster or manifold assumption may be not completely fulfilled, and thus no clear improvement may be eventually obtained with semi-supervised approaches [31]. We should nevertheless note that, by fixing $\gamma = 0$, the original (undeformed) kernel is obtained. Therefore, a proper selection of this free parameter theoretically leads to improved performance over supervised target detection.

IV. Experimental Results

This section shows the results obtained by the proposed $S^2$KOSP in both a 2D toy example and in two hyperspectral image target detection applications: crop identification and hot spot
detection.

A. Experiment 1: Toy Example

For illustration purposes, Fig. 3 shows the results obtained on the ‘two-moons’ toy dataset by applying the standard KOSP and the developed semi-supervised version, $S^2$KOSP, both with RBF kernels. By using the center of mass of each class as prototypes (white spots in Fig.3(a)), and choosing the maximum of KOSP output as classification criterion, the algorithm induces a separating linear hyperplane. Figure 3(b) shows instead how the structure of the dataset can deform the a priori distance metric by choosing as prototype the pixel closest to the center of mass, and thus allowing perfect classification.

![Fig. 3. (a) KOSP and (b) S$^2$KOSP results in the two-moons toy example.](image)

B. Experiment 2: Crop Detection

This section is concerned with a real target detection using hyperspectral images.

1) Dataset description: The used dataset comes from the AVIRIS instrument that acquired data over the Kennedy Space Center (KSC), Florida (USA), on March 23, 1996. AVIRIS acquires data in 224 bands of 10 nm width with wavelengths from 400-2500 nm. The data were acquired from an altitude of 20 km and has a spatial resolution of 18 m. After removing low SNR and water absorption bands, a total of 176 bands remains for analysis.
The wetlands of the Indian River Lagoon (IRL) system, located on the western coast of the KSC are a critical habitat for several species of waterfowl and aquatic life. The test site for this research consists of a series of impounded estuarine wetlands of the northern Indian River Lagoon that reside on the western shore of KSC. Detection of land cover for this environment is difficult due to the similarity of spectral signatures for certain vegetation types. The dataset contains 13 labeled classes representing the various land cover types of the environment. Figure 4 shows an RGB composition with the labeled classes highlighted. More information can be found in [http://www.csr.utexas.edu/](http://www.csr.utexas.edu/).

2) Experimental Setup: Among all available classes, we focus on the most complex class in the dataset. Many criteria of class separability exist in the literature, either relying on a specific metric, class PDF modeling, or based on information theory criteria of class independence. However, in our case, the interest is to evaluate the accuracy gain when the unlabeled information is used. For this purpose, we run the standard KOSP for all classes using only about 10 samples randomly chosen among the labeled pixels, and analyzed the obtained results. Table II shows the area-under-the-curve (AUC) computed from the receiver operating curve (ROC) for the target class. Note that the ROC curves are commonly used for evaluating target detection methods since they provide a quantitative comparison between the probability of correct detection, \( P_d \), and the false alarm rate \( F_a \). We selected class 8 (‘Graminoid Marsh’) since it constitutes a particularly
challenging problem, being typically misclassified with other similar marsh classes.

<table>
<thead>
<tr>
<th>Class</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graminoid marsh</td>
<td>0.826</td>
</tr>
<tr>
<td>Cattail marsh</td>
<td>0.832</td>
</tr>
<tr>
<td>Dark/Broadleaf</td>
<td>0.836</td>
</tr>
<tr>
<td>Salt marsh</td>
<td>0.872</td>
</tr>
<tr>
<td>Spartina marsh</td>
<td>0.873</td>
</tr>
<tr>
<td>Mud flats</td>
<td>0.884</td>
</tr>
<tr>
<td>Slash pine</td>
<td>0.884</td>
</tr>
<tr>
<td>Willow</td>
<td>0.891</td>
</tr>
<tr>
<td>CP/Oak</td>
<td>0.897</td>
</tr>
<tr>
<td>Scrub</td>
<td>0.902</td>
</tr>
<tr>
<td>CP Hammock</td>
<td>0.924</td>
</tr>
<tr>
<td>Hardwood swamp</td>
<td>0.946</td>
</tr>
<tr>
<td>Water</td>
<td>0.996</td>
</tr>
</tbody>
</table>

3) Model Development: The proposed method contains several free parameters to be tuned: \( k \) nearest neighbors to build the graph, \( \sigma \) for the kernel, and \( \gamma \) to trim the impact of unlabeled samples. In all cases, we used the RBF kernel to construct the similarity matrices, \( K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \). The RBF kernel width was tuned in the range \( \sigma = \{10^{-2}, \ldots, 10\} \), \( k \) was varied between 2-20, and \( \gamma \) in the range \( [10^{-2}, \ldots, 10^4] \). In order to select the values we followed a constrained optimization using pattern search [32]. A one-against-all target detection scheme was adopted. Complementary material (MATLAB source code, demos, and datasets) is available at [http://www.uv.es/gcamps/sskosp/](http://www.uv.es/gcamps/sskosp/) for those interested readers.

For the experiments, the 5211 labeled pixels were randomly split into three sets: training \( (l = 1300) \), cross-validation set \( (1311) \) for free parameter tuning, and the test set \( (2600) \). A number of unlabeled samples \( u = 120 \) was chosen. In the following, results are shown for OSP, KOSP and \( S^2 \)KOSP in terms of ROC curves. Several classifiers are developed with different rates of labeled \( (\{5\%, 10\%, 15\%, 20\%\}) \) and unlabeled \( (\{5\%, 25\%, 50\%, 100\%\}) \) samples. Selection of
unlabeled samples is a key issue in semi-supervised learning: in this work, we selected unlabeled samples as those being spatially near the available labeled samples. More theoretical details and implications are discussed in Section IV-D. From the ROC curves, quantitative accuracy measures are derived: area under the ROC curve (AUC), reflecting the overall detection probability, and the minimum squared distance from the ROC to the ideal point \([0, 1]\) (\(\Delta\)).

4) Numerical Results: Figure 5 shows the best ROC curves for different rates of training samples and considered models. In general, the proposed S\(^2\)KOSP outperforms the rest of the methods in all cases. Nevertheless, we should stress here that very good results are obtained with few labeled samples (Fig. 5(a)), which suggests that unlabeled information has been properly exploited.

All methods perform similar at very low false alarm rate or when very high sensitivities are requested. However, the proposed S\(^2\)KOSP systematically outperforms KOSP in the region \(F_a \in [0.1, 0.3]\), where the detection probability \(P_a\) is drastically improved. Note that, for a fixed value of \(F_a\) in this range, the semi-supervised KOSP gives 10\% – 30\% gain in detection accuracy \(P_a\). It is also worth noting that similar results are observed for different rates of training samples, but it is especially observed for the case of 25\% training samples (Fig. 5(e)).

These good robustness outcomes are confirmed by analyzing the AUC and \(\Delta\) scores in Table III. An average systematic gain of 5\% is observed compared to the KOSP algorithm in AUC. Also, the improvement is observed to be approximately constant with the number of unlabeled samples. Finally, we should remark that it is not only the amount of unlabeled samples but also the selection strategy what matters (see Section IV-D).

C. Experiment 3: Thermal Hot Spot Detection

The third experiment deals with the hot spot detection using an hyperspectral image. In this case, only visual performance can be assessed since no ground truth is available.

1) Dataset Description: The used dataset comes from the AVIRIS instrument that acquired data over the World Trade Center (WTC), New York, few days later the well-known collapse of the towers of September 11th, 2001. The data was acquired from an altitude of \(\sim 2\) km and has a spatial resolution of about 2 m. The accompanying maps available are false color images acquired over the same site on September 16th and 21st, which show the core affected area around the WTC (Fig. 6). More information and images are available at
Fig. 5. Best ROC curves obtained for (a) 5%, (b) 10%, (c) 15%, (d) 20% and (e) 25% of training samples.
TABLE II
OBTAINED RESULTS FOR KOSP AND $S^2$KOSP. IN BRACKETS, VALUES OF $\tau, \sigma_{ss}$ USED TO EVALUATE $l_i$ AND NEAREST NEIGHBORS (NN) USED TO BUILD THE GRAPH $L$ ARE GIVEN.

<table>
<thead>
<tr>
<th>% of $l$</th>
<th>% of $u$ ($\tau, \sigma_{ss}, NN$)</th>
<th>AUC</th>
<th>$\Delta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>KOSP</td>
<td>0.845</td>
<td>0.073</td>
</tr>
<tr>
<td></td>
<td>5 (1, 1, 5)</td>
<td>0.868</td>
<td>0.064</td>
</tr>
<tr>
<td></td>
<td>25 (4, 1, 1, 5)</td>
<td><strong>0.902</strong></td>
<td><strong>0.023</strong></td>
</tr>
<tr>
<td></td>
<td>50 (7, 1, 1, 7)</td>
<td>0.895</td>
<td>0.032</td>
</tr>
<tr>
<td></td>
<td>100 (10, 1, 5)</td>
<td>0.894</td>
<td>0.023</td>
</tr>
<tr>
<td>10</td>
<td>KOSP</td>
<td>0.852</td>
<td>0.072</td>
</tr>
<tr>
<td></td>
<td>5 (1, 1, 11)</td>
<td>0.876</td>
<td>0.044</td>
</tr>
<tr>
<td></td>
<td>25 (50, 0.75, 5)</td>
<td>0.880</td>
<td>0.035</td>
</tr>
<tr>
<td></td>
<td>50 (25, 0.85, 5)</td>
<td>0.900</td>
<td>0.025</td>
</tr>
<tr>
<td></td>
<td>100 (50, 0.85, 11)</td>
<td><strong>0.903</strong></td>
<td><strong>0.024</strong></td>
</tr>
<tr>
<td>15</td>
<td>KOSP</td>
<td>0.850</td>
<td>0.077</td>
</tr>
<tr>
<td></td>
<td>5 (1, 1, 12)</td>
<td>0.875</td>
<td>0.060</td>
</tr>
<tr>
<td></td>
<td>25 (10, 1, 5)</td>
<td>0.886</td>
<td>0.061</td>
</tr>
<tr>
<td></td>
<td>50 (12, 1, 8)</td>
<td><strong>0.892</strong></td>
<td><strong>0.041</strong></td>
</tr>
<tr>
<td></td>
<td>100 (8, 1, 8)</td>
<td>0.892</td>
<td>0.042</td>
</tr>
<tr>
<td>20</td>
<td>KOSP</td>
<td>0.842</td>
<td>0.073</td>
</tr>
<tr>
<td></td>
<td>5 (10, 0.9, 9)</td>
<td>0.797</td>
<td>0.083</td>
</tr>
<tr>
<td></td>
<td>25 (20, 0.9, 8)</td>
<td>0.859</td>
<td>0.052</td>
</tr>
<tr>
<td></td>
<td>50 (30, 1.05, 5)</td>
<td>0.894</td>
<td>0.038</td>
</tr>
<tr>
<td></td>
<td>100 (20, 1.05, 10)</td>
<td><strong>0.898</strong></td>
<td><strong>0.031</strong></td>
</tr>
<tr>
<td>25</td>
<td>KOSP</td>
<td>0.852</td>
<td>0.072</td>
</tr>
<tr>
<td></td>
<td>5 (5, 1.05, 11)</td>
<td>0.875</td>
<td>0.055</td>
</tr>
<tr>
<td></td>
<td>25 (10, 1.05, 5)</td>
<td>0.890</td>
<td>0.033</td>
</tr>
<tr>
<td></td>
<td>50 (5, 1.05, 9)</td>
<td>0.892</td>
<td>0.368</td>
</tr>
<tr>
<td></td>
<td>90 (5, 1.05, 5)</td>
<td><strong>0.904</strong></td>
<td><strong>0.015</strong></td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>0.898</td>
<td>0.037</td>
</tr>
</tbody>
</table>
Visual analysis of these data revealed a number of thermal hot spots on September 16th in the region where the buildings collapsed 5 days earlier. Analysis of the data indicates temperatures higher than 400°C. Different hot spots appear in the core zone, and those will be considered in the following as being the target for the detection.

2) Experimental Setup: There is no ground truth available for the World Trade Center dataset, so we carried out a pixel selection through visual photointerpretation of many false RGB compositions, taking advantage of the material available at http://speclab.cr.usgs.gov/wtc and reported in Figure 6. The WTC dataset is a urban area showing vegetated areas, roofs, streets and shadows but, however, it also shows presence of dust from different materials and debris due to the towers collapse. In order to keep the problem of detection as real as possible, the a priori information about the pixel components of the scene are not used, since it is supposed to be known only after performing a ‘in loco’ analysis. Therefore, in order to define a subspace as simple as possible—as an end-user without prior knowledge would do— we selected only a few pixels from roofs, trees, shadows, water and fire. As already explained, ‘fire’ is considered the target class in this experiment (see Fig. 6). As in previous experiment, the pixel closest to the
center of mass of each class was selected as the prototype of that class. Results for KOSP and S\textsuperscript{2}KOSP are shown by means of gray level images containing the outputs of the algorithms, and coloured images containing the thresholded outputs.

3) Model Development: As in the previous experiments, there are several parameters to be tuned for the model. Unfortunately, the reduced number of labeled samples available cannot be used to properly maximize a performance score, such as the area under the ROC curve. Instead, the parameters are tuned according to a ‘visual’ interpretation of the detection, as it will be better explained in the following subsection.

In the case of KOSP, the algorithm is performed by using windows of size $K_c \times K_c$, and the prototypes extracted from the ground truth. This approach is motivated by the fact that a few spatial coverage of the target class is expected. Therefore, it makes sense to define a maximum window where the target pixels may lie. This procedure not only reduces the computational cost of the algorithm but also improves the detection relying on the smooth spatial variation of the spectral signature.

In the semi-supervised version of the algorithm instead, the deformed kernel strictly depends on the choice of unlabeled samples and the distance between the samples of the image. Hence, to explore as much useful information as possible, the best choice would be performing the semi-supervised KOSP by processing the whole dataset at the same time. This would involve an inversion of a huge matrix that is not computationally possible at the moment (even with the provided fast versions of the S\textsuperscript{2}KOSP in Section III-C, the problem is intractable in a timely manner), so one have to choose a strategy to select the unlabeled samples to build the model. For this experiment, we consider the same block of $K_c \times K_c$ test samples used for the KOSP but, in addition, the deformed kernel is evaluated by considering also an overlapping of $K_u$ pixels between adjacent windows (see Fig. 7), in order to use ‘cross-contextual’ information between pixels belonging to neighbor windows. The importance of using a contextual selection of the pixels is detailed in Section IV-D. Contextual versions of KOSP have been recently proposed in [33].
Fig. 7. Window Technique used to Compute KOSP and $S^2$KOSP.

4) Results: In this section we compare the results of the canonical OSP and KOSP methods, and the proposed semi-supervised KOSP. Two versions of the proposed semi-supervised KOSP are presented: (1) the windowed approach explained before, and (2) a technique to select the unlabeled samples relying on the results obtained with the standard KOSP.

Figure 8 shows the (raw and thresholded) outputs obtained with KOSP and the two versions of $S^2$KOSP presented. From the KOSP predictions (top row), it can be seen that class ‘fire’ presents a variability that cannot be captured by the KOSP algorithm which offers poor results. Attending to the thresholded outputs (bottom row), there is only one pixel correctly classified by the KOSP (blue), but this is the selected prototype pixel. Pixels marked in yellow are all misclassified pixels with KOSP method, while red pixels are ‘false alarms’, or pixels erroneously detected by KOSP as belonging to class ‘fire’. Actually, a spectral inspection reveals that most of these pixels are very similar to the pixel correctly detected, and moreover, they are so spatially connected to the ground truth containing fire that they might contain fire as well.

The proposed $S^2$KOSP performs well in this case: the ground truth is completely detected (blue) and, in addition, all red pixels are not ‘false alarms’, except two in the lower part of the image. It has to be remarked that $S^2$KOSP is very sensitive to the choice of pixels used as unlabeled samples, so a smarter selection of unlabeled samples should be carried out. Two clear shortcomings of the proposed method are: (i) one can easily note the presence of a highly patched (and annoying) detection map, mainly due to the windowed technique used for detection; and (ii) by using the window technique, the computational time for a given image increases linearly.
with the number of windows, since for each block the evaluation of the inverse of the matrix \((I + NK)\) (see Eq. 6) is needed.

To solve these problems, an alternative semi-supervised version of the algorithm is introduced here. Note that, even though KOSP cannot capture the variability of the class, it detects pixels that are spectrally very similar to the target and, besides, it is fast and generally reliable. Therefore, we advocate that a proper selection of the unlabeled samples for the \(S^2\)KOSP can be accomplished in an easy way by performing a preliminary detection applying KOSP. In this way, the detected pixels by KOSP can be used as unlabeled samples for \(S^2\)KOSP. We want to remark that, even if there were pixels identifying false alarms, with a correct tuning of the \(S^2\)KOSP parameters, they would be moved away from the principal coordinates of the data manifold to not affecting the detection. Note that, with this alternative method, the inverse of the matrix \((I + NK)\) has to be evaluated only one time in the complete routine, making the computational time of KOSP and \(S^2\)KOSP similar. Detection maps offered by this method are now smoother (see last column in Fig. 8) while performance of the KOSP algorithm is improved: the patches due to the use of different sets of unlabeled samples per window have disappeared and the false alarm rate have been reduced.
D. Algorithm Analysis and Characterization

This section is devoted to deeply analyze and characterize the proposed semi-supervised algorithm in terms of the selection of unlabeled samples, the impact of unlabeled samples in method’s performance, and the computational cost of the proposed algorithmical alternatives. Finally, we illustrate how the kernel deformation takes place.

1) On the Selection of Unlabeled Samples: The choice of unlabeled samples is a key problem in semi-supervised learning, and it should be addressed in such a way that the unknown marginal distribution is properly sampled and fitted. A trivial approach consists in selecting unlabeled samples randomly from the whole image. In the third previous experiment we used as unlabeled samples the detected pixels by KOSP for computational convenience. However, these approximations lead to skewed modeling of the marginal pdf. We propose here a selection of the spatially closest samples to the target signature. Specifically, by means of sequential opening morphological operations, a contour containing two levels of nearby pixels are drawn for each connected set of the ground truth.

Note that this procedure resembles that of using dual window with guard band, and it is aimed at exploiting both the spatial and spectral variability among different materials. However, since the windowed samples are unlabeled, its exploitation turns to be theoretically more appealing. Note that some of the nearby pixels may lie on the tail of the marginal distribution and, along with the available labeled samples, may correctly model the geometry of the dataset. Some other pixels could lie spectrally close (far) to the labeled pixels, thus resulting in a strong (weak) deformation of the kernel. In both cases, however, with an adequate tuning of kernel parameters, these samples do not affect the performance. The method is a simple, yet effective way of limiting the impact of outlying samples when building the graph and avoids using more sophisticated and complex versions of Laplacian eigenmaps [34].

The effect of these sample selection procedure is illustrated in Fig. 9. Essentially, if a random sample selection is conducted (Fig. 9[left]), the information about the target contained in the selected samples is lower as they may differ both spatially and spectrally. When unlabeled samples are selected spatially near the target, the data distribution of the target class is more efficiently modeled thanks to the cluster assumption which typically holds in remote sensing data (Fig. 9[right]).
Fig. 9. Selection of the unlabeled samples to train the $S^2$KOSP can be done either randomly (left) or by spatial neighboring pixels (right).

Figure 10 shows the ROC curves for the linear OSP, its kernel version KOSP, and the semi-supervised KOSP using either random or contextual sample selection. A clear gain is observed with the contextual-based proposal. Note that, in addition, a random sample selection strategy leads to better results than KOSP, but it is only noticeable for low sensitivity ratios. In the case of the contextual selection, the gain is clearly observed for the whole sensitivity–specificity ROC range.

Fig. 10. Results in the case of OSP, KOSP, and $S^2$KOSP with either random or contextually-based unlabeled sample selection.

2) On the Relative Importance of Labeled and Unlabeled Samples: In the previous sections, performance of the supervised and semi-supervised KOSP was analyzed. In the experiments,
we explored the robustness of the classifiers to the number of labeled samples available during the training process; from ill-posed situations with only 5 labeled samples up to well-posed supervised cases with 100 labeled samples per class. For the semi-supervised methods, the number of unlabeled samples used in the training of the models was fixed to $u = 800$. However, in the case of semi-supervised learning, it is also interesting to analyze methods performance as a function of the number of unlabeled samples.

Fig. 11 shows the AUC surface of the proposed method as a function of the rate of labeled ($l$) and unlabeled ($u$) samples used in the training phase. The AUC surface and shows its dependence on the number of labeled samples, especially when the number of unlabeled samples are not available. The more supervised information is available, the more accurate is the classification. However, a clear saturation effect is readily observed. Besides, a similar curve is observed as we increase the number of unlabeled information. The method uses both the labeled samples to fix a support for the class distribution, and the unlabeled samples to characterize (parametrize) the data marginal distribution.

In general, semi-supervised learning methods require a higher number of unlabeled samples than labeled to provide a noticeable improvement in the classification accuracy, as suggested by [35]–[37]. However, the usage of a high number of unlabeled samples in the formulation of kernel methods is not straightforward and usually implies a high computational cost. This
problem can be mitigated with the use of the fast versions provided here.

3) Computational Efficiency: In this paper we have introduced two novel formulations based on the Nyström method and the ICF in order to make it feasible the inversion of a big matrix. For the Nyström method, two free parameters need to be tuned; \( m \) is the number of samples used to compute the approximate decomposition of the kernel matrix, and \( p \) is the number of demanded largest eigenvalues (and corresponding eigenvectors). For the ICF method, only an error tolerance value \( \eta \) has to be fixed. In this section, we analyze the trade-off between the accuracy of the approximation and the computational cost. In the following, we use \( l = 25\% \) and \( u = 100\% \) for illustration purposes. We used the optimal values of \( \sigma = 1.05 \), \( \gamma = 5 \), and \( k = 5 \), computed the corresponding kernel matrix, the graph Laplacian, and run five different inversion methods: (1) direct inversion of the \( 5000 \times 5000 \) matrix; (2) solve the linear system instead through standard Cholesky factorization; (3) use a fast implementation with the ARPACK method [29], in which only the largest \( p \) eigenvalues and corresponding eigenvectors are returned; and the proposed (4) Nyström and (5) ICF algorithms.

![Graph showing error and time](image)

Fig. 12. Analysis of matrix inversion methods to accelerate the semi-supervised KOSP method. (a) RMSE and (b) CPU time [s].

Figure 12 shows the Frobenius norm of the committed error in the matrix inversion (zero-error is considered for the direct inversion and the linear system solution), and the computational cost involved in calculating the inverse matrix (in seconds). Results show that the best trade-off between the accuracy and the computational cost is given by the ICF that is 42 times faster than the direct matrix inversion and 4.7 times faster than the Nyström method while reduced inverse
errors are obtained ($\sim 10^{-9}$). Additionally the computational cost and the approximation error can be easily trimmed with a single intuitive parameter $\eta$.

4) **Visualizing the Kernel Deformation**: An interesting possibility consists of visualizing the capabilities of the proposed method to capture non-linearities in the data. For such purpose, we subsampled the data and used 5 classes and 3 uncorrelated features. The data distribution and the decision boundaries are depicted in Fig. 13 for both the standard KOSP and the proposed semi-supervised version. In both cases, we used the RBF kernel and tuned the free parameters through cross-validation to avoid skewed conclusions. Note that, while for some classes, the decision boundaries are roughly undeformed (blue and green classes), other decision functions try to capture the data distribution (yellow and red classes). This versatility is easily tuned with parameter $\gamma$: the higher is the $\gamma$, the higher deformation is introduced.

V. CONCLUSIONS

A semi-supervised version of the KOSP algorithm was proposed for target detection applications. The information from unlabeled samples is included in the standard KOSP by means of the graph Laplacian with a contextual unlabeled sample selection mechanism. In addition, two fast versions of the method has been proposed.

The good results obtained suggest that unlabeled information is properly exploited, and that the data manifold can be modeled with the unlabeled surrounding samples. Additionally, the
proposed methodology can be useful to easily extend other kernel methods in general, and for target detection in particular.

ACKNOWLEDGMENTS

The authors would like to acknowledge Drs. Melba Crawford at Purdue University (USA) and Antonio Plaza at Universidad de Extremadura (Spain) for kindly sharing the datasets used in this work.

APPENDIX A

THE DEFORMING KERNEL

Let \( \mathcal{H} \) be a complete Hilbert space \( \mathcal{H} \) with inner product \( \langle \cdot, \cdot \rangle_{\mathcal{H}} \), and the reproducing symmetric positive semidefinite kernel \( K(x, z) = \langle K(x, \cdot), K(z, \cdot) \rangle_{\mathcal{H}} \) built from \( f(x) = \langle f, K(x, \cdot) \rangle_{\mathcal{H}} \). Let \( \mathcal{V} \) be a linear space with a positive semi-definite inner-product and \( S: \mathcal{H} \rightarrow \mathcal{V} \) be a bounded linear operator. Then, given the space of functions \( \tilde{\mathcal{H}} \) from \( \mathcal{H} \) with the modified inner product \( \langle f, g \rangle_{\tilde{\mathcal{H}}} = \langle f, g \rangle_{\mathcal{H}} + \langle Sf, Sg \rangle_{\mathcal{V}} \), the functional point evaluation is bounded in both Hilbert spaces.

The semi-norm \( \|Sf\|_{\mathcal{V}}^2 = f^T N f \) on \( \mathbb{R}^n \) is given by a symmetric positive semi-definite matrix \( N \) and the Hilbert space \( \tilde{\mathcal{H}} \) can be decomposed as \( \tilde{\mathcal{H}} = \text{span}\{\tilde{K}(x_1, \cdot), \ldots, \tilde{K}(x_n, \cdot)\} \oplus \tilde{\mathcal{H}}^\perp \), where \( \tilde{\mathcal{H}}^\perp \) consists of functions vanishing at all data points. For any \( f \in \tilde{\mathcal{H}}^\perp \), \( Sf = 0 \) and therefore \( \langle f, g \rangle_{\tilde{\mathcal{H}}} = \langle f, g \rangle_{\mathcal{H}} \), and thus \( [K(x, \cdot) - \tilde{K}(x, \cdot)] \in \text{span}\{\tilde{K}(x_1, \cdot), \ldots, \tilde{K}(x_n, \cdot)\} \).

For any point \( x_i \in \mathbb{R}^n \) and point-wise function \( f \in \tilde{\mathcal{H}}^\perp \), the induced kernel \( K(x_i, \cdot) \in (\tilde{\mathcal{H}}^\perp)^\perp \), and thus \( \text{span}\{K(x_i, \cdot)\}_{i=1}^n \subseteq \text{span}\{\tilde{K}(x_i, \cdot)\}_{i=1}^n \), thus leading to \( \tilde{K}(x, \cdot) = K(x, \cdot) + \sum_{j} \beta_j(x)K(x_j, \cdot) \), where the coefficients \( \beta_j \) depend on \( x \), and can be implicitly evaluated with the inner product \( K(x, x) = \langle K(x_1, \cdot), \tilde{K}(x, \cdot) \rangle_{\tilde{\mathcal{H}}} \), which gives rise to a system of linear equations \( (I + NK)\beta(x) = -NK_x \), where \( K \) is the matrix with components \( K(x_i, x_j) \) and \( k_x \) denotes the vector \( [K(x_1, x), \ldots, K(x_n, x)]^T \). Plugging this into \( \tilde{K}(x, \cdot) \), we obtain \( \tilde{K}(x, z) = K(x, z) - K_x^T (I + NK)^{-1}NK_x \).

APPENDIX B

THE NYSTRÖM METHOD

Given a Gram matrix \( S = (S_{ij}) \), and \((i_1, \ldots, i_m)\) a set of randomly chosen \( m \) indexes such as \( 1 \leq i_1 < i_2 < \cdots < i_m \leq l \), define matrices \( S_{m,m} \) and \( S_{l,m} \) such as \( S_{m,m}(j, k) = S(i_j, i_k) \)
for $1 \leq j, k \leq m$ and $S_{l,m}(i,k) = S(i,i_k)$ for $1 \leq i \leq l$ and $1 \leq k \leq m$. If $\lambda_i^{(m)}$ and $v_i^{(m)}$ are the $i$th largest eigenvalue and its corresponding eigenvector, exploiting the eigendecomposition principle, we can approximate $S$ with: 

$$\tilde{S} = \sum_{i=1}^{p} \tilde{\lambda}_i^{(l)} v_i^{(l)} (v_i^{(l)})^\top,$$

where $\tilde{\lambda}_i^{(l)} \equiv (l/m) \lambda_i^{(m)}$ and $\tilde{v}_i^{(l)} \equiv \sqrt{1/m(1/\lambda_i^{(m)})} S_{l,m} v_i^{(l)}$, which define the $l \times p$ matrix $\tilde{V}$ formed by column vectors $\tilde{v}_i^{(l)}$ ($i = 1, \ldots, p$), and the $p \times p$ (diagonal) matrix $\tilde{\Lambda}$ whose $(i,i)$ components are $\tilde{\lambda}_i^{(l)}$.

REFERENCES


