Multiway Spectral Clustering with Out-of-Sample Extensions through Weighted Kernel PCA

Carlos Alzate, Johan A. K. Suykens

Abstract

A new formulation for multiway spectral clustering is proposed. This method corresponds to a weighted kernel principal component analysis (PCA) approach based on primal-dual least squares support vector machine (LS-SVM) formulations. The formulation allows the extension to out-of-sample points. In this way, the proposed clustering model can be trained, validated and tested. The clustering information is contained on the eigendecomposition of a modified similarity matrix derived from the data. This eigenvalue problem corresponds to the dual solution of a primal optimization problem formulated in a high-dimensional feature space. A model selection criterion called the Balanced Line Fit (BLF) is also proposed. This criterion is based on the out-of-sample extension and exploits the structure of the eigenvectors and the corresponding projections when the clusters are well-formed. The BLF criterion can be used to obtain clustering parameters in a learning framework. Experimental results with difficult toy problems and image segmentation show improved performance in terms of generalization to new samples and computation times.

Index Terms

Spectral clustering, kernel principal component analysis, out-of-sample extensions, model selection.

The authors are with the Department of Electrical Engineering ESAT-SCD-SISTA, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium (email: carlos.alzate@esat.kuleuven.be, johan.suykens@esat.kuleuven.be).

I. INTRODUCTION

Spectral clustering comprises a family of clustering methods that make use of the eigenvectors of some normalized affinity matrix derived from the data to group points that are similar. These techniques are known to perform well in cases where classical clustering methods such as k-means and linkage absolutely fail. Many spectral clustering applications can be found in fields like machine learning, computer vision and data analysis. Starting from graph theory, this family of algorithms [1]–[6] are formulated as relaxations of graph partitioning problems that are generally NP-complete. These relaxations take the form of eigenvalue problems involving a normalized affinity matrix containing pairwise similarities. When only two groups are required, the relaxed solution corresponds to a particular eigenvector. The cluster information can then be extracted by binarizing the relaxed solution. However, the clustering problem complicates when more than two clusters are required because converting the relaxed solutions to cluster information is not straightforward. Typical approaches to tackle this issue are recursive cuts and reclustering which were discussed in [1]. Recursive binary cuts are not optimal because only one eigenvector is used when other eigenvectors may also contain cluster information and it is not clear when to stop cutting. The reclustering approach consists of applying k-means over the eigenvectors. This approach works only when the cluster information contained in the eigenvectors has a spherical structure. More advanced techniques are based on minimizing a metric between the relaxed solutions and the set of allowed cluster indicators [7], [8] and on finding peaks and valleys of a criterion that measures cluster overlap [9].

One of the main drawbacks of spectral clustering is the fact that the clustering model is defined only for training data with no clear extension to out-of-sample points. The methods described in [10], [11] allow extensions to new points by approximating the implicit eigenfunction using the Nyström method [12], [13]. The underlying clustering model is usually not known and the parameter selection is done in a heuristic way.

Kernel principal component analysis (PCA) [14] corresponds to an unsupervised learning technique useful for nonlinear feature extraction, denoising and dimensionality reduction. This method is a nonlinear extension of PCA by using positive definite kernels. The objective is to find projected variables in a kernel induced feature space with maximal variance. Links between spectral clustering and kernel PCA have been discussed in [15], [16]. The methods show that

some forms of spectral clustering can be seen as a special case of kernel PCA. The method described in [16] shows that classical binary spectral clustering such as the normalized cut [1], the NJW algorithm [3] and the random walks model [6] are cases of weighted kernel PCA with different weighting matrices.

In this paper we propose a multiway spectral clustering model based on a weighted kernel principal component analysis scheme [16], [17]. The formulation is based on [16] which is a binary clustering technique and extended to the multiway clustering case with encoding and decoding schemes. The formulation fits into the least squares support vector machines (LS-SVM) framework [18], [19] by considering weighted versions. The proposed approach is cast in a constrained optimization framework with primal and dual insights. This interpretation allows the clustering model to be extended to out-of-sample data points without relying on approximation techniques such as the Nyström method. The eigenvectors of a modified similarity matrix derived from the data are shown to be the dual solutions to a primal optimization problem formulated in a high-dimensional feature space. These solutions contain clustering information and show a special structure when the clusters are well formed. The out-of-sample extension is obtained through projections onto these eigenvector solutions. We also propose a model selection criterion called the balanced line fit (BLF). This criterion exploits the structure of the eigenvectors and the corresponding projections and can be used to obtain model parameters.

This paper is organized as follows: Section II contains a review of existing spectral clustering techniques. Section III contains a description of the existing weighted kernel PCA approach to binary spectral clustering. In Section IV, we propose the multiway spectral clustering model. Section V contains the out-of-sample extension together with an algorithm. In Section VI, we discuss about model selection and propose the BLF criterion. Section VII contains the empirical results and in Section VIII we give conclusions.

II. Spectral Clustering

A. Graph Theory

Given a set of N data points $\{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$ and some similarity measure $s_{ij} \geq 0$ between each pair of points x_i and x_j, an intuitive form of representing the data is using a graph $G = (\mathcal{V}, \mathcal{E})$. The vertices $\mathcal{V}$ represent the data points and the edge $e_{ij} \in \mathcal{E}$ between vertices v_i, v_j has a weight determined by s_{ij}. If the similarity measure is symmetric then the graph is undirected.

The affinity matrix of the graph is the matrix S with ij-entry $S_{ij} = s_{ij}$. The degree of vertex v_i is the sum of all the vertex weights adjacent to v_i and is defined as

$$\deg_i = \sum_{j=1}^{N} s_{ij},$$

the degree matrix D is a diagonal matrix containing the vertex degrees $\deg_1, \ldots, \deg_N$ on the diagonal.

A basic problem in graph theory is the graph bipartitioning problem, that is to separate the graph G into two disjoint sets $\mathcal{A}, \mathcal{B}$ based on a cut criterion. The resulting sets should be disjoint: $\mathcal{A} \cap \mathcal{B} = \emptyset$ and $\mathcal{A} \cup \mathcal{B} = \mathcal{V}$. This problem has been extensively studied [2], [20] and several cut criteria have been proposed [1], [4], [20], [21]. The normalized cut *NCut* [1] is a common bipartitioning criterion and is defined as:

$$\text{NCut}(\mathcal{A}, \mathcal{B}) = \frac{\text{cut}(\mathcal{A}, \mathcal{B})}{\text{Vol}(\mathcal{A})} + \frac{\text{cut}(\mathcal{A}, \mathcal{B})}{\text{Vol}(\mathcal{B})}, \tag{1}$$

where $\text{Vol}(\mathcal{A}) \equiv \sum_{i \in \mathcal{A}} d_i$ denotes the volume of the set $\mathcal{A}$. The *NCut* can be extended to the general case in which the problem is to separate the graph into k disjoint sets $(k > 2)$:

$$\text{NCut}(\mathcal{A}_1, \ldots, \mathcal{A}_k) = \sum_{l=1}^{k} \frac{\text{cut}(\mathcal{A}_l, \overline{\mathcal{A}_l})}{\text{Vol}(\mathcal{A}_l)} \tag{2}$$

where $\overline{\mathcal{A}}$ denotes the complement of $\mathcal{A}$.

B. The Normalized Cut Relaxation

The *NCut* algorithm has become a very popular technique for spectral clustering and image segmentation. Its approximate solution follows from the second smallest eigenvector (also known as the Fiedler vector) of a generalized eigenvalue problem. The problem of minimizing the *NCut* can be written (see [1]) as:

$$\begin{aligned} \min_q \text{Ncut}(\mathcal{A}, \mathcal{B}) &= \frac{q^T L q}{q^T D q} \\ \text{such that} &\quad \begin{cases} q \in \{-c, 1\}^N \\ q^T D 1_N = 0 \end{cases} \end{aligned} \tag{3}$$

where $L = D - S$ is the unnormalized graph Laplacian and $c = \text{Vol}(\mathcal{A})/\text{Vol}(\mathcal{B})$. Unfortunately, minimizing the *NCut* is NP-complete [1]. However, an approximate solution can be found by relaxing the discrete constraint on q. If q can take real values then the objective in (3) corresponds to the Rayleigh quotient of the generalized eigenvalue problem

$$L\tilde{q} = \nu D\tilde{q} \tag{4}$$

where $\tilde{q} \in \mathbb{R}^N$. The remaining constraint $\tilde{q}^T D 1_N = 0$ is automatically satisfied in the generalized eigenproblem. Therefore, the relaxed solution to the *NCut* is the Fiedler vector.

C. Markov Random Walks

A Markov random walks view of spectral clustering was discussed in [6], [22]. A random walk on a graph consists of random jumps from vertex to vertex. This interpretation showed that many properties of spectral clustering methods can be expressed in terms of a stochastic transition matrix P obtained by normalizing the affinity matrix such that its rows sum to 1. The ij-th entry of P represents the probability of moving from node i to node j in one step. This transition matrix can be defined as $P = D^{-1}S$. The corresponding eigenvalue problem becomes

$$Pr = \xi r.$$

The eigenvalues of P are $1 = \xi_1 \geq \xi_2 \geq \ldots \geq \xi_N$ with corresponding eigenvectors $1_N = r^{(1)}, r^{(2)}, \ldots, r^{(N)}$. Consider the generalized eigenvalue problem of the *NCut*: $L\tilde{q} = \nu D\tilde{q}$, then premultiplying by D^{-1} leads to $(I_N - P)\tilde{q} = \nu\tilde{q}$. This shows that the eigenvectors of P are identical to the eigenvectors of the *NCut* algorithm and the eigenvalues of P are $\xi_i = 1 - \nu_i, i = 1, \ldots, N$. Minimizing the *NCut* can be interpreted as finding a partition of the graph in such a way that the random walk remains most of the time in the same cluster with few jumps to other clusters.

D. The k-way NCut Relaxation

Consider $f_p \in \{0,1\}^N$ as the cluster indicator vector for the p-th cluster such that f_p has a 1 in the entries corresponding to the data points in the p-th cluster. The cluster indicator matrix becomes $F = [f_1, \ldots, f_k] \in \{0,1\}^{N \times k}$. Defining $g_p = D^{1/2} f_p / ||D^{1/2} f_p||_2, p = 1, \ldots, k$ and

$G = [g_1, \ldots, g_k]$ leads to the k-way *NCut* criterion:

$$\text{NCut}(\mathcal{A}_1, \ldots, \mathcal{A}_k) = k - \text{tr}(G^T \hat{L} G)$$

where $\hat{L}$ is the normalized Laplacian defined as $\hat{L} = D^{-1/2} W D^{-1/2}$ and $G^T G = I_k$. The matrix G that minimizes the k-way *NCut* can be found by maximizing the trace of $G^T \hat{L} G$:

$$\begin{aligned} \underset{G}{\text{argmin}}\ \text{NCut}(\mathcal{A}_1, \ldots, \mathcal{A}_k) &= \underset{G}{\text{argmax}}\ \text{Tr}(G^T \hat{L} G) \\ \text{such that} \quad & G^T G = I_k. \end{aligned} \tag{5}$$

Relaxing the discrete constraint and allowing G to be a real-valued matrix leads to the k-way *NCut* relaxation [23]:

$$\underset{\tilde{G}}{\text{argmin}}\ \text{NCut}(\mathcal{A}_1, \ldots, \mathcal{A}_k) = \underset{\tilde{G}}{\text{argmax}}\ \text{Tr}(\tilde{G}^T \hat{L} \tilde{G}) \tag{6}$$

$$\text{such that} \quad \tilde{G}^T \tilde{G} = I_k. \tag{7}$$

This maximization problem is a special case of Fan's theorem [24]. The solution to this relaxed problem is

$$\tilde{G}^\star = A R_1$$

where $A = [a^{(1)}, \ldots, a^{(k)}] \in \mathbb{R}^{N \times k}$ is any orthonormal basis of the k-th principal subspace of $\hat{L}$ and $R_1 \in \mathbb{R}^{k \times k}$ is an arbitrary orthogonal matrix [7], [24].

Consider the eigenvalue problem of the normalized Laplacian $D^{-1/2} W D^{-1/2} a = \zeta a$, premultiplying by $D^{-1/2}$ leads to $Pu = \zeta u$, with $u = D^{-1/2} a$. Thus, the relation between the k-way *NCut* and the random walks model is $\zeta = \xi, a = D^{1/2} r$.

E. Perturbation Analysis and Piecewise Constant Eigenvectors

In the ideal case of a graph with k disconnected components, the corresponding affinity matrix shows a block diagonal structure composed of k blocks if the clusters are ordered. The transition matrix P in this case will have k eigenvalues equal to 1 and the remaining eigenvalues will be less than 1. The eigenvectors corresponding to these unitary eigenvalues are the indicator functions to the respective disconnected components and the clustering is trivial because each cluster is represented as a single point in $\mathbb{R}^k$. This also corresponds to the piecewise constant property of

eigenvectors discussed in [6], [25] where the spectral clustering problem is cast in a probabilistic framework using block stochastic matrices.

Definition 1 ***[6]*** *A piecewise constant vector* $\alpha = [\alpha_1, \ldots, \alpha_N] \in \mathbb{R}^N$ *relative to a partition of* $\Delta = \{\mathcal{A}_p\}_{p=1}^k$ *and a dataset* $\mathcal{D} = \{x_i\}_{i=1}^N$ *is defined as*

$$\alpha_j = c_p, \text{ for all } x_j \in \mathcal{A}_p, p = 1, \ldots, k,$$

where c_p *is a constant value corresponding to the* p*-th cluster.*

Perturbation analysis is the study of how the eigenvalues and eigenvectors of a symmetric matrix B change if an additive perturbation E is considered. The perturbed matrix is denoted by $\tilde{B} \equiv B + E$. The Davis-Kahan theorem [26] states that the distance between the top k eigenvalues and eigenvectors of the perturbed matrix $\tilde{B}$ and the top k eigenvalues and eigenvectors of B is bounded by $||E||_F/\delta$, where $||\cdot||_F$ denotes the Frobenius norm, $\delta = \tilde{\eta}_k - \tilde{\eta}_{k+1}$ corresponds to the additive eigengap and $\tilde{\eta}_k$ is the k-th ordered eigenvalue of $\tilde{B}$. Therefore, if the perturbation is small and the eigengap is large enough, the perturbed eigenvectors will be approximately piecewise constant and consequently easy to cluster [27]. This represents one of the fundamental ideas of spectral clustering.

F. From Eigenvectors to Clusters

The relaxed solutions provided by k-way spectral clustering methods are real-valued and do not provide cluster indicator matrices. Hence, it is necessary to convert these solutions in order to obtain indicators of the partitioning. In [1] two methods were proposed. The first one is based on applying binary cuts in a recursive way. The current partition is subsequently divided if the *NCut* is below some prespecified value. The process is repeated until k clusters have been found. This method is slow and not optimal due to the fact that only the second smallest eigenvector is used when other eigenvectors may contain useful information. The second method called *reclustering* is one of the most popular approaches for k-way spectral clustering and consists of computing the top k eigenvectors of the k-way *NCut* problem and then applying k-means onto them. This works for piecewise constant eigenvectors because every cluster $\mathcal{A}_i \in \Delta$ is mapped to a unique point in $\mathbb{R}^k$. A different approach called *rounding* was introduced in [28]. It is based on the minimization of a metric between the relaxed solution and the set of discrete

allowed solutions. This optimization problem can be solved either with respect to the partition resulting in a weighted k-means algorithm [29] or with respect to the similarity matrix yielding an algorithm for learning W given a particular partition. In [9], a method called linearized cluster assignment was discussed. This method transforms the problem of obtaining clusters from the eigenvectors into finding peaks and valley of a 1-D quantity called cluster crossing.

III. Kernel PCA and LS-SVMs

A. Classical Kernel PCA Formulation

A kernel-based algorithm for nonlinear PCA was described in [14]. This algorithm performs linear PCA in a kernel-induced feature space nonlinearly related to the original input space. Given a set of N data points $\{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$, kernel PCA finds directions in which the projected variables $w^T\varphi(x_i), i = 1, \ldots, N$ have maximal variance. Here $\varphi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^{d_h}$ corresponds to the mapping to a high dimensional feature space $\mathcal{F}$ of dimension d_h (which can be infinite dimensional). Assuming zero mean mapped data $\sum_{i=1}^N \varphi(x_i) = 0$, the covariance matrix in the feature space becomes

$$C \approx \frac{1}{N}\sum_{j=1}^{N} \varphi(x_j)\varphi(x_j)^T$$

with eigendecomposition $Cw = \lambda w$. The computation of $\varphi(\cdot)$ and C is complicated due to the high dimensionality of $\mathcal{F}$. However, by making use of the representer theorem [14], it is possible to avoid dealing with the mapped data explicitly. A positive definite kernel $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ is a positive definite function that corresponds to a dot product in a high dimensional feature space $K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$. The kernel PCA problem can be solved by means of the eigendecomposition of the kernel matrix $\Omega \in \mathbb{R}^{N\times N}$ which contains pairwise evaluations of the kernel function $\Omega_{ij} = K(x_i, x_j), i, j = 1, \ldots, N$. Typically, the data are preprocessed in the feature space by removing the mean. This step is accomplished by centering the kernel matrix: $\Omega_c = M_c\Omega M_c$ where $M_c = I_N - (1/N)1_N 1_N^T$ is the centering matrix.

The kernel PCA eigenvalue problem becomes

$$\Omega_c\beta = \lambda\beta.$$

The resulting eigenvectors have to be normalized in $\mathcal{F}$ by requiring $||\beta^{(j)}||_2 = 1/\sqrt{\lambda_j}, j = 1, \ldots, N$.

B. LS-SVM Formulation to Kernel PCA

An LS-SVM approach to kernel PCA was introduced in [19]. This approach showed that kernel PCA is the dual solution to a primal optimization problem formulated in a kernel induced feature space. The underlying loss function associated to kernel PCA was shown to be L_2. Given the training set $\{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$ sampled i.i.d. from an underlying distribution $\mathcal{P}$, the LS-SVM approach to kernel PCA is formulated in the primal as:

$$\begin{aligned} \min_{w,e_i,b} J_p(w,e_i) &= \frac{\gamma}{2N}\sum_{i=1}^{N} e_i^2 - \frac{1}{2}w^T w \\ \text{such that} \quad & e_i = w^T\varphi(x_i) + b, i = 1,\ldots,N, \end{aligned} \tag{8}$$

where b is a bias term.

Proposition 1 ***[17], [19]*** *Given a positive definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ with $K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$, the Karush-Kuhn-Tucker (KKT) conditions of the Lagrangian of (8) are satisfied by each eigenvector α of the dual problem:*

$$\Omega_c\alpha = \lambda\alpha \tag{9}$$

where Ω_c denotes the centered kernel matrix and $\lambda = N/\gamma$.

Remark 1 ***[19]*** *Note that including a bias term in* (8) *leads to centering the kernel matrix. The mapped points in the feature space have zero mean. The bias term becomes:*

$$b = -\frac{1}{N}1_N^T\Omega\alpha. \tag{10}$$

C. Weighted Kernel PCA

A weighted kernel PCA formulation based on the LS-SVM framework was discussed in [16], [17]. This formulation has been used for extending kernel PCA to general loss functions in order to achieve sparseness and robustness in an efficient way [17]. A different kind of weighting was used in [16] to link several binary spectral clustering algorithms with kernel PCA. Introducing a symmetric positive definite weighting matrix V (typically chosen to be diagonal) into (8) leads

to the following primal problem:

$$\begin{aligned} \min_{w,e} J_p(w,e) &= \frac{\gamma}{2N} e^T V e - \frac{1}{2} w^T w \\ \text{such that} & \quad e = \Phi w \end{aligned} \tag{11}$$

where $e = [e_1, \ldots, e_N]$ is the compact form of the projected variables $e_i = w^T \varphi(x_i), i = 1, \ldots, N$, $V = \text{diag}([v_1, \ldots, v_N])$ is the weighting matrix and $\Phi = [\varphi(x_1)^T; \ldots; \varphi(x_N)^T]$ is the $N \times n_h$ feature matrix.

Lemma 1 ***[16]*** *Given a positive definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ with $K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$, a symmetric positive definite weighting matrix V and a regularization constant $\gamma \in \mathbb{R}^+$, the Karush-Kuhn-Tucker (KKT) optimality conditions of the Lagrangian of (11) are satisfied by each eigenvector α of the dual problem:*

$$V\Omega\alpha = \lambda\alpha \tag{12}$$

where $\lambda = N/\gamma$ is the corresponding eigenvalue.

Remark 2 *Note that (11) is in general a non-convex problem. Hence, the KKT conditions are necessary but not sufficient. The eigenvectors α of $V\Omega$ correspond to stationary points of the Lagrangian and the objective function evaluated at all stationary points is equal to zero [17]. Therefore, each eigenvector/value pair can be seen as a candidate solution. The component selection problem can be solved by sorting the eigenvectors in decreasing order with respect to λ.*

Remark 3 *Given a set of N_{test} test points $\{x_m^{test}\}_{m=1}^{N_{test}}$ sampled i.i.d. from $\mathcal{P}$, the projection of these points (also called the score variables) over an eigenvector becomes*

$$z = \Phi_{test} \Phi^T \alpha$$

where $\Phi_{test} = \left[\varphi(x_1^{test})^T; \ldots; \varphi(x_{N_{test}}^{test})^T\right]$ is the $N_{test} \times n_h$ feature matrix. Applying the kernel trick leads to

$$z = \Omega_{test} \alpha$$

where $\Omega_{test} = \Phi_{test} \Phi^T$ is the $N_{test} \times N$ kernel matrix evaluated using the test points with entries

$\Omega_{test}^{mi} = K(x_m^{test}, x_i), m = 1, \ldots, N_{test}, i = 1, \ldots, N.$

D. Links with binary spectral clustering methods

The weighted kernel PCA framework has also been used to provide links with spectral clustering techniques such as the *NCut* [1], the approximate kernel alignment [30], [31], the NJW algorithm [3] and the random walks [6], emphasizing primal-dual insights in addition to out-of-sample extensions for binary clustering [16]. Table I shows the relation between weighted kernel PCA and some spectral clustering methods. The last column indicates which eigenvector of (12) is the relaxed solution to the original spectral method given V.

TABLE I
RELATION BETWEEN WEIGHTED KERNEL PCA AND SOME SPECTRAL CLUSTERING METHODS

Method	Original Problem	V	Relaxed Solution
Alignment	$\Omega q = \lambda q$	I_N	$N\alpha^{(1)}$
NCut	$Lq = \lambda Dq$	D^{-1}	$\alpha^{(2)}$
Random walks	$D^{-1}Wq = \lambda q$	D^{-1}	$\alpha^{(2)}$
NJW	$D^{-\frac{1}{2}}WD^{-\frac{1}{2}}q = \lambda q$	D^{-1}	$D^{\frac{1}{2}}\alpha^{(2)}$

IV. MULTIWAY FORMULATION TO SPECTRAL CLUSTERING

The multiclass problem in classification consists of obtaining n_c classes by using a set of n_y binary classification problems. The number of additional classifiers is an important issue and typically corresponds to different encoding/decoding schemes. A unique codeword $y_l \in \{-1, 1\}^{n_y}, l = 1, \ldots, n_c$ is assigned to each class. The *one-vs-all* coding scheme consists of setting n_e equal to the number of classes n_c and making binary decisions between each class and all the other classes. In *one-vs-one*, the number of binary classifiers is set to $n_c(n_c-1)/2$ and the discrimination is made for each possible pair of classes. The number of one-vs-one classifiers is usually higher than the number of classifiers in the one-vs-all case, but the resulting decision boundaries are typically simpler. Another encoding scheme is to use n_y problems to encode up to 2^{n_y} classes. The error-correcting output code (ECOC) [32] approach usually introduces redundancy $(n_y > n_c)$ and a careful design of codewords (e.g. good Hamming distance between

codewords, codewords motivated by known features). In order to classify a new point, the output variables are then compared with the set of codewords and the new point is assigned to the class of the closest codeword in terms of Hamming distance. This scheme has shown good results in terms of robustness against some errors induced by noise.

In this paper we extend the weighted kernel PCA model for binary spectral clustering to the k-way case by using some concepts of multiclass theory and introducing additional score variables as constraints. This extension is inspired in the multiclass formulation for LS-SVM models discussed in [18]. Introducing additional regularization constants, score variables, bias terms and using the inverse of the degree matrix as the weighting matrix leads to the following primal problem:

$$\min_{w^{(l)},e^{(l)},b_l} \quad \frac{1}{2N}\sum_{l=1}^{n_e}\gamma_l e^{(l)^T}D^{-1}e^{(l)} - \frac{1}{2}\sum_{l=1}^{n_e} w^{(l)^T}w^{(l)} \tag{13}$$

$$\text{such that} \quad \begin{cases} e^{(1)} = \Phi w^{(1)} + b_1 1_N \\ e^{(2)} = \Phi w^{(2)} + b_2 1_N \\ \vdots \\ e^{(n_e)} = \Phi w^{(n_e)} + b_{n_e} 1_N. \end{cases}$$

In this case, each score variables vector $e^{(l)}$ provides a binary clustering with cluster indicator $q^{(l)}$:

$$q^{(l)} = \text{sign}(e^{(l)}), l = 1, \ldots, n_e.$$

Lemma 2 *Given a positive definite kernel function $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ with $K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$, regularization constants $\gamma_l \in \mathbb{R}^+, l = 1, \ldots, n_e$ and the inverse of the degree matrix D^{-1} which is diagonal with positive elements, then the Karush-Kuhn-Tucker (KKT) optimality conditions of the Lagrangian of (13) are satisfied by the following eigenvalue problem:*

$$D^{-1}M_D\Omega\alpha^{(l)} = \lambda_l\alpha^{(l)} \tag{14}$$

where $\lambda_l = N/\gamma_l, l = 1, \ldots, n_e$ and

$$M_D = I_N - \frac{1}{1_N^T D^{-1} 1_N} 1_N 1_N^T D^{-1}. \tag{15}$$

Proof: Consider the Lagrangian of the problem (11):

$$\mathcal{L}(w^{(l)}, e^{(l)}, b_l; \alpha^{(l)}) = \frac{1}{2}\sum_{l=1}^{n_e} \gamma_l {e^{(l)}}^T D^{-1} e^{(l)} - \frac{1}{2}\sum_{l=1}^{n_e} {w^{(l)}}^T w^{(l)} - \sum_{l=1}^{n_e} {\alpha^{(l)}}^T \left(e^{(l)} - \Phi^{(l)} w^{(l)} - b_l 1_N\right) \tag{16}$$

with KKT optimality conditions:

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial w^{(l)}} = 0 \rightarrow w^{(l)} = \Phi^T \alpha^{(l)} \\ \frac{\partial \mathcal{L}}{\partial e^{(l)}} = 0 \rightarrow \alpha^{(l)} = \gamma_i D^{-1} e^{(l)} \\ \frac{\partial \mathcal{L}}{\partial b_l} = 0 \rightarrow 1_N^T \alpha^{(l)} = 0 \\ \frac{\partial \mathcal{L}}{\partial \alpha^{(l)}} = 0 \rightarrow e^{(l)} = \Phi w^{(l)} + b_l 1_N, \end{cases} \tag{17}$$

for $l = 1, \ldots, n_e$. The bias terms become:

$$b_l = -\frac{1}{1_N^T D^{-1} 1_N} 1_N^T D^{-1} \Omega \alpha^{(l)}, l = 1, \ldots, n_e. \tag{18}$$

Eliminating the primal variables $w^{(l)}, e^{(l)}, b_l$ leads to the eigenvalue problem (14). ■

The choice of D^{-1} as the weighting matrix is motivated by the random walks model [6] and the piecewise constant property of the eigenvectors when the clusters are well formed. When the weighting matrix is set to I_N, the eigenvalue problem (14) becomes kernel PCA which is known to lack discriminatory features for clustering.

Remark 4 *The score variables for the test points become*

$$z^{(l)} = \Omega_{test} \alpha^{(l)} + b_l 1_{N_{test}}, l = 1, \ldots, n_e. \tag{19}$$

Remark 5 *Note that including a bias term into the weighted kernel PCA formulation leads to a different kind of centering. The matrix M_D can be interpreted as a weighted centering matrix removing the weighted mean from each column of Ω. The weights are determined by $D^{-1}1_N$. The effect of this centering is that the eigenvector $\alpha^{(1)}$ corresponding to the largest eigenvalue λ_1 already contains information about the clustering. This is not the case in the random walks algorithm where the top eigenvector $r^{(1)}$ corresponds to 1_N and therefore does not contain any information about the clusters.*

Remark 6 *Consider the score variables of a test point x with the following centering in the*

feature space:

$$z^{(l)}(x) = w^{(l)^T}\big(\varphi(x) - \hat{\mu}_{\varphi_D}\big),$$

where $\hat{\mu}_{\varphi_D} = \sum_{j=1}^{N} D_{jj}^{-1}\varphi(x_j)/(1_N^T D^{-1} 1_N)$ *is the weighted mean and* $l = 1, \ldots, n_e$*. Applying the first optimality condition* (17) *leads to:*

$$z^{(l)}(x) = \sum_{i=1}^{N} \alpha_i^{(l)} K(x_i, x) + b_l, l = 1, \ldots, n_e.$$

This shows that the score variables for test points are also centered with respect to the weighted mean in the feature space.

Corollary 1 *Every entry in the score variables of training data has the same sign as the corresponding entry in the eigenvectors. This is formulated as:*

$$q_i^{(l)} = \mathit{sign}(\alpha_i^{(l)}), i = 1, \ldots, N, l = 1, \ldots, n_e$$

where $q_i^{(l)} = \mathit{sign}(e_i^{(l)})$.

Proof: Using (17), and taking into account that D is also diagonal leads to:

$$e_i^{(l)} = \lambda_l D_{ii} \alpha_i^{(l)}, i = 1, \ldots, N, l = 1, \ldots, n_e.$$

Since λ_l and the entries of D are always positive then the sign of $e_i^{(l)}$ depends only of the sign of $\alpha_i^{(l)}$. ■

A. Encoding / Decoding Scheme

In classification, the encoding scheme is chosen beforehand and enforced into the problem by using the labels. In clustering, as a representative of unsupervised learning there are no target values and therefore the codewords are obtained afterwards and given by the eigenvectors. If the eigenvectors are piecewise constant then every cluster $\mathcal{A}_p, p = 1, \ldots, k$ is represented with a unique codeword $c_p \in \{-1, 1\}^{n_e}$. The codebook $\mathcal{C} = \{c_p\}_{p=1}^{k}$ can then be obtained from the rows of the binarized eigenvector matrix $[\text{sign}(\alpha^{(1)}), \ldots, \text{sign}(\alpha^{(n_e)})]$ or from the clustering indicator matrix of training data $[q^{(1)}, \ldots, q^{(n_e)}]$ (Corollary 1). Introducing a bias term leads to centering the kernel matrix and obtaining zero mean eigenvectors. This is important for coding since the optimal threshold for binarizing the eigenvectors is automatically determined. The

first eigenvector already provides a dichotomy due to the centering imposed by the bias terms. Therefore, the number of score variables needed to encode k clusters is $n_e = k-1$. The decoding scheme consists of comparing the obtained cluster indicators with the codebook and selecting the nearest codeword in terms of Hamming distance. This scheme corresponds to the ECOC decoding procedure.

V. Out-of-Sample Extension

Spectral clustering methods provide a clustering only for the given training data without a clear extension to out-of-sample (test) points. This issue was first mentioned in [10] and applies also to several spectral algorithms for manifold learning such as local linear embedding (LLE) [33], isomap [34] and Laplacian eigenmaps [35]. A link between kernel PCA and manifold learning was presented in [15], where LLE, isomap and Laplacian eigenmaps and locality preserving projections (LPP) [36] are interpreted as kernel PCA with different kernel matrices. However, these techniques are more suited to data visualization than to clustering since the problem of obtaining clusters from the out-of-sample extensions is not clear. The out-of-sample method proposed in [10] consists of applying the Nyström method [12] in order to provide an embedding for the test points by approximating the underlying eigenfunction. Our proposed extension is based on the score variables which correspond to the projections of the mapped out-of-sample points onto the eigenvectors found in the training stage. The cluster indicators can be obtained by binarizing the score variables for out-of-sample points as follows

$$\text{sign}(z^{(l)}) = \text{sign}(\Omega_{\text{test}}\alpha^{(l)} + b_l 1_{N_{\text{test}}}), l = 1, \ldots, k-1. \tag{20}$$

This natural extension to out-of-sample points without relying on approximations like the Nyström method corresponds to the main advantage of the weighted kernel PCA framework. In this way, the clustering model can be trained, validated and tested in an unsupervised learning scheme.

In this section, we provide an implementation of the multiway formulation to spectral clustering together with an extension to out-of-sample data points. According to Fan's theorem [24], the relaxed solutions to the k-way normalized cut are arbitrary rotated eigenvectors. An intuitive approach to obtain the clusters from the eigenvectors is to find the rotation matrix that

makes the eigenvector matrix a cluster indicator matrix (i.e. only one non-zero entry per row). The spectral method discussed in [8] uses this approach and proposes an incremental gradient descent algorithm to obtain the optimal number of clusters. However, finding this rotation matrix increases the computational cost. The computation of the rotation matrix can be avoided using k-means because this method is invariant to rotated eigenvectors but is prone to local minima, requires good initialization and assumes spherical clusters.

Algorithm 1 Multiway Spectral Clustering with Out-of-Sample Extension

Input: Training set $\{x_i\}_{i=1}^{N}$, test set $\{x_m^{\text{test}}\}_{m=1}^{N_{\text{test}}}$, positive definite kernel function $K(x_i, x_j)$, number of clusters k.
Output: $\Delta = \{\mathcal{A}_1, \ldots, \mathcal{A}_k\}$, cluster codeset $\mathcal{C} = \{c_p\}_{p=1}^{k}$, $c_p \in \{-1, 1\}^{k-1}$.

1: Compute the eigenvectors $\alpha^{(l)}, l = 1, \ldots, k-1$ corresponding to the largest $k-1$ eigenvalues of $D^{-1}M_D\Omega$ where M_D is defined as in (15) and Ω is the training kernel matrix $\Omega_{ij} = K(x_i, x_j)$.
2: Binarize the eigenvectors: $\hat{\alpha}_i^{(l)} = \text{sign}(\alpha_i^{(l)}), i = 1, \ldots, N, l = 1, \ldots, k-1$.
3: Let $\text{sign}(\alpha_i) \in \mathbb{R}^{k-1}$ be the encoding vector for the training data point x_i, $i = 1, \ldots, N$.
4: Count the occurrences of the different encodings and find the k encodings with most occurrences. Let the codeset be formed by these k encodings: $\mathcal{C} = \{c_p\}_{p=1}^{k}, c_p \in \{-1, 1\}^{k-1}$
5: For all i, assign x_i to $\mathcal{A}_{p^\star}$ where $p^\star = \text{argmin}_p d_H(\text{sign}(\alpha_i), c_p)$ and $d_H(\cdot, \cdot)$ is the Hamming distance.
6: Compute the cluster indicators for test data $\text{sign}(q_m^{(l)}), m = 1, \ldots, N_{\text{test}}, l = 1, \ldots, k-1$ using (20).
7: Let $\text{sign}(q_m) \in \mathbb{R}^{k-1}$ be the encoding vector for the test data point x_m^{test}, $m = 1, \ldots, N_{\text{test}}$.
8: For all m, assign x_m^{test} to $\mathcal{A}_{p^\star}$ where $p^\star = \text{argmin}_p d_H(\text{sign}(q_m), c_p)$.

Proposition 2 *Consider a training dataset $\mathcal{D} = \{x_i\}_{i=1}^{N}, x_i \in \mathbb{R}^d$ sampled i.i.d. from an underlying distribution $\mathcal{P}$, a validation dataset $\mathcal{D}^v = \{x_m^v\}_{m=1}^{N_v}, x_m \in \mathbb{R}^d$ also sampled i.i.d. from $\mathcal{P}$ and an isotropic kernel function $K(x_i, x_j), i, j = 1, \ldots, N$. Let Ω be the kernel matrix with ij-entry $\Omega_{ij} = K(x_i, x_j)$ and let $\alpha^{(l)}, l = 1, \ldots, k-1$ be the eigenvectors of* (14)*. Consider the following assumptions:*

- *$\mathcal{P}$ contains k clusters denoted by $\Delta = \{\mathcal{A}_1, \ldots, \mathcal{A}_k\}$.*
- *The eigenvectors $\alpha^{(l)}, l = 1, \ldots, k-1$ are piecewise constant.*
- *$K(x_i, x_j) \to 0$ when x_i and x_j are in different clusters.*

Then, validation points in the same cluster are collinear in the $k-1$ dimensional subspace spanned by the columns of Z^v, where $Z^v \in \mathbb{R}^{N_v \times (k-1)}$ is the validation score variables matrix

with ml*-entry* $Z^v_{ml} = z^{(l)}_m = \sum_{i=1}^N \alpha^{(l)}_i K(x_i, x^v_m)$ *and* $m = 1, \ldots, N_v, l = 1, \ldots, k-1$.

Proof: The eigenvectors $\alpha^{(l)}_i$ are piecewise constant, therefore:

$$\alpha^{(l)}_j = c^{(l)}_p, \text{for all } x_j \in \mathcal{A}_p,$$

$l = 1, \ldots, k-1$, $p = 1, \ldots, k$ where $c^{(l)}_p$ is the constant value corresponding to the p-th cluster in the l-th eigenvector. The projections become:

$$z^{(l)}_m = c^{(l)}_p \sum_{j \in \mathcal{A}_p} K(x_j, x^v_m) + \sum_{u \notin \mathcal{A}_p} \alpha^{(l)}_u K(x_u, x^v_m), \tag{21}$$

$l = 1, \ldots, k-1$, $p = 1, \ldots, k$, $m = 1, \ldots, N_v$. Note that $K(x_u, x^v_m)$ tends to zero when x_u and x^v_m are not in the same cluster, therefore the term $\sum_{u \notin \mathcal{A}_p} \alpha^{(l)}_u K(x_u, x^v_m)$ is very small compared to the term $c^{(l)}_p \sum_{j \in \mathcal{A}_p} K(x_j, x^v_m)$. In this case, validation points belonging to the same cluster are represented as collinear points in the score variables space. ■

The proposed encoding scheme is based on an arbitrary set of orthogonal hyperplanes induced by the eigenproblem. Due to the fact that the eigenvectors can be arbitrarily rotated, the hyperplanes can bisect data points in the same cluster if the data are not well represented. However, this is improbable for well represented data because in that case, data points belonging to the same cluster will be located in the same orthant.

VI. Model Selection

The results obtained using spectral clustering depend greatly on the choice of the affinity function and its parameters. The most used affinity function is the radial basis function (RBF) kernel $K(x_i, x_j) = \exp(-0.5||x_i - x_j||^2_2/\sigma^2)$ with tuning parameter σ^2. The selection of this parameter is not considered as part of the learning process and is commonly done in a manual way using grid search and minimizing some cluster distortion function [3]. However, counterexamples can be easily found in which clusters with small distortion do not correspond to the natural grouping of the data. The tuning method proposed in [8] introduces local scaling by selecting a σ_i parameter per data point x_i. The selection of this scaling parameter is done in a heuristic way using the distance of the point x_i to some nearest neighbor. Again, the results of this approach depend on the choice of the nearest neighbor. A different method that can be used to learn the parameters given a particular partition was proposed in [7]. In this case, the method corresponds

to a supervised setting and the clustering cost function characterizes the ability of the induced affinity matrix to produce clusters that fit the given partition. The selection of the number of clusters k is also a general problem for all clustering methods and several approaches have been proposed. Typical algorithms are the eigengap heuristic, the elbow criterion, the gap statistic [37] and the silhouette index [38]. The simplest approach is the eigengap heuristic which consists of choosing a value for k such that the eigengap $|\lambda_{k+1} - \lambda_k|$ is greater than some threshold value θ. The justification of this procedure relies on the fact that in the ideal case when the graph has k disconnected components, the multiplicity of the eigenvalue 1 is k. Thus, the $(k+1)$-th eigenvalue will be less than 1 resulting in some eigengap. In this ideal case, the threshold can be set to zero, but in practical situations this simple heuristic absolutely fails. The elbow criterion is a simple measure of the compactness of the clusters and the gap statistic together with the silhouette index are advanced cluster distortion indicators.

In this paper, we propose a criterion called the *balanced line fit* (BLF) for estimating suitable spectral clustering parameters, namely the RBF kernel parameter σ^2 (or the χ^2 kernel parameter σ_χ^2) and the number of clusters k. This criterion exploits the eigenstructure of piecewise constant eigenvectors and the corresponding projections. Consider the following average measure of collinearity of a validation dataset $\mathcal{D}^v$ with respect to a clustering $\Delta = \{\mathcal{A}_1, \ldots, \mathcal{A}_k\}, k > 2$

$$\text{linefit}(\mathcal{D}^v, \Delta, k) = \frac{1}{k}\sum_{p=1}^{k} \frac{k-1}{k-2}\left(\frac{\zeta_1^{(p)}}{\sum_l \zeta_l^{(p)}} - \frac{1}{k-1}\right) \tag{22}$$

where $\zeta_1^{(p)} \geq \ldots \geq \zeta_{k-1}^{(p)}$ are the ordered eigenvalues of the sample covariance matrix:

$$C_{\tilde{Z}}^{(p)} = \frac{1}{|\mathcal{A}_p|}\tilde{Z}^{(p)^T}\tilde{Z}^{(p)}, p = 1, \ldots, k, \tag{23}$$

$\tilde{Z}^{(p)} \in \mathbb{R}^{|\mathcal{A}_p| \times (k-1)}$ is the matrix representing the zero mean score variables for validation data assigned to the p-th cluster. The term $\zeta_1^{(p)} / \sum_l \zeta_l^{(p)}$ indicates how much of the total variance is contained on the eigenvector corresponding to the largest eigenvalue of $C_{\tilde{Z}}^{(p)}$. In this way, if the validation score variables for the p-th cluster are collinear, then all variance is contained on the first eigenvector and $\zeta_1^{(p)} / \sum_l \zeta_l^{(p)}$ equals 1. On the other hand, if the total variance is evenly distributed on the eigenvectors then $\zeta_1^{(p)} / \sum_l \zeta_l^{(p)}$ equals $1/(k-1)$. The additional terms in (22) fix the criterion such that the linefit equals 0 when the score variables are distributed spherically (i.e. the eigenvalues are identical) and equals 1 when the score variables are collinear.

Note that the linefit is defined only for $k > 2$. When $k = 2$ only one eigenvector is needed to obtain a binary clustering. Consider the matrix $\breve{Z}^v \in \mathbb{R}^{N_v \times 2}$ where the m-th entry in the first column corresponds to the score variables $z_m = \sum_{i=1}^{N} \alpha_i K(x_i, x_m^v)$ and the m-th entry in the second column to $\sum_{i=1}^{N} K(x_i, x_m^v)$. This modification allows the application of the linefit criterion when $k = 2$:

$$\text{linefit}(\mathcal{D}^v, \Delta, k) = \begin{cases} \sum_{p=1}^{2} \left(\frac{\breve{\zeta}_1^{(p)}}{\breve{\zeta}_1^{(p)} + \breve{\zeta}_2^{(p)}} - \frac{1}{2} \right) & \text{if } k = 2 \\ \frac{1}{k} \sum_{p=1}^{k} \frac{k-1}{k-2} \left(\frac{\zeta_1^{(p)}}{\sum_l \zeta_l^{(p)}} - \frac{1}{k-1} \right) & \text{if } k > 2 \end{cases} \tag{24}$$

where $\breve{\zeta}_l^{(p)}$ are the ordered eigenvalues of the matrix $\breve{Z}^{(p)} \in \mathbb{R}^{|\mathcal{A}_p| \times 2}$ representing the zero mean score variables for validation data assigned to the p-th cluster when $k = 2$.

In real-life problems, the balance of the obtained clusters also becomes important. Consider the following clustering balance measure:

$$\text{balance}(\mathcal{D}^v, \Delta, k) = \frac{\min\{|\mathcal{A}_1|, \ldots, |\mathcal{A}_k|\}}{\max\{|\mathcal{A}_1|, \ldots, |\mathcal{A}_k|\}}. \tag{25}$$

The balance index equals 1 when the clusters have the same number of elements and tends to 0 in extremely unbalanced cases. Combining the linefit with the balance index leads to the BLF:

$$\text{BLF}(\mathcal{D}^v, \Delta, k) = \eta \text{linefit}(\mathcal{D}^v, \Delta) + (1 - \eta)\text{balance}(\mathcal{D}^v, \Delta), \tag{26}$$

where η is a parameter controlling the importance given to the linefit with respect to the balance index and $0 \leq \eta \leq 1$.

VII. Empirical Results

In this section, some experimental results are presented to illustrate the proposed approach. All experiments reported are carried out in MATLAB on an Intel Dual Core, 3.0 GHz, 2GB RAM. All data have been separated into training, validation and test sets. The BLF criterion is used on validation data to find the number of clusters k and the kernel parameters and the η parameter is fixed to $\eta = 0.75$. Hence, more emphasis is given to the linefit. Simulations on toy data and image segmentation are presented. The adjusted Rand index (ARI) [39] is used as an external validation in order to compare the clustering results with some external cluster indicators. This clustering performance index ranges from 0 to 1 and takes the unitary value when the clustering

results matches perfectly the external cluster indicators. We empirically observed that for the used kernels, the learned bias terms are very small compared to the magnitude of the projections. This means that the score variables value is dominated by the kernel expansions.

We compare our proposed method in terms of performance and computation times with the spectral clustering scheme discussed in [11]. This method uses random subsampling to solve an eigenvalue problem of a small subset of training points and then extrapolates the eigenvectors to the complete set of data points using the Nyström approximation [12], [13]. The resulting approximated eigenvectors are not mutually orthogonal and additional orthogonalization steps are needed. The cluster indicators are obtained by applying k-means on the approximated orthogonalized eigenvectors. We used the one-shot implementation described in [11].

A. Toy Problems

The first toy problem consists of 3 Gaussian clouds in a 2D space. The total number of data points is 800. The RBF kernel parameter σ^2 was set to 0.08. The training set consisted of 200 randomly selected points. Figure 1 shows the binary clustering boundaries induced by the encoding scheme. The first eigenvector $\alpha^{(1)}$ of (14) groups the left and bottom Gaussian clouds into one cluster. The second eigenvector $\alpha^{(2)}$ groups the bottom and right clouds into one cluster. The cluster indicators for out-of-sample points were inferred using the out-of-sample extension and the ECOC decoding scheme.

The second toy problem shows the influence of increasing number of clusters k with small training set sizes. Figure 2 shows the results of the proposed approach and the Nyström algorithm [11]. This dataset consists of 10 Gaussian clouds for a total of $2,000$ data points. We performed 10 randomizations of the training set and the total number of training point was set to one-fifth of the total number of points. We used the average ARI over the randomizations to assess the performance with respect to the optimal clustering. The proposed method performs better than the Nyström algorithm in terms of ARI.

The third toy problem consists of 3 concentric rings in a 2D space. The training, validation and test sets contain 600, $1,200$ and 800 data points respectively. This problem is known to be difficult for spectral clustering because of its nonlinearity, the fact that every datapoint has few neighbors resulting in a very sparse affinity matrix and the multiscale nature of the rings. Figure 3 shows the training set and the model selection curves. In this particular case, the BLF and the

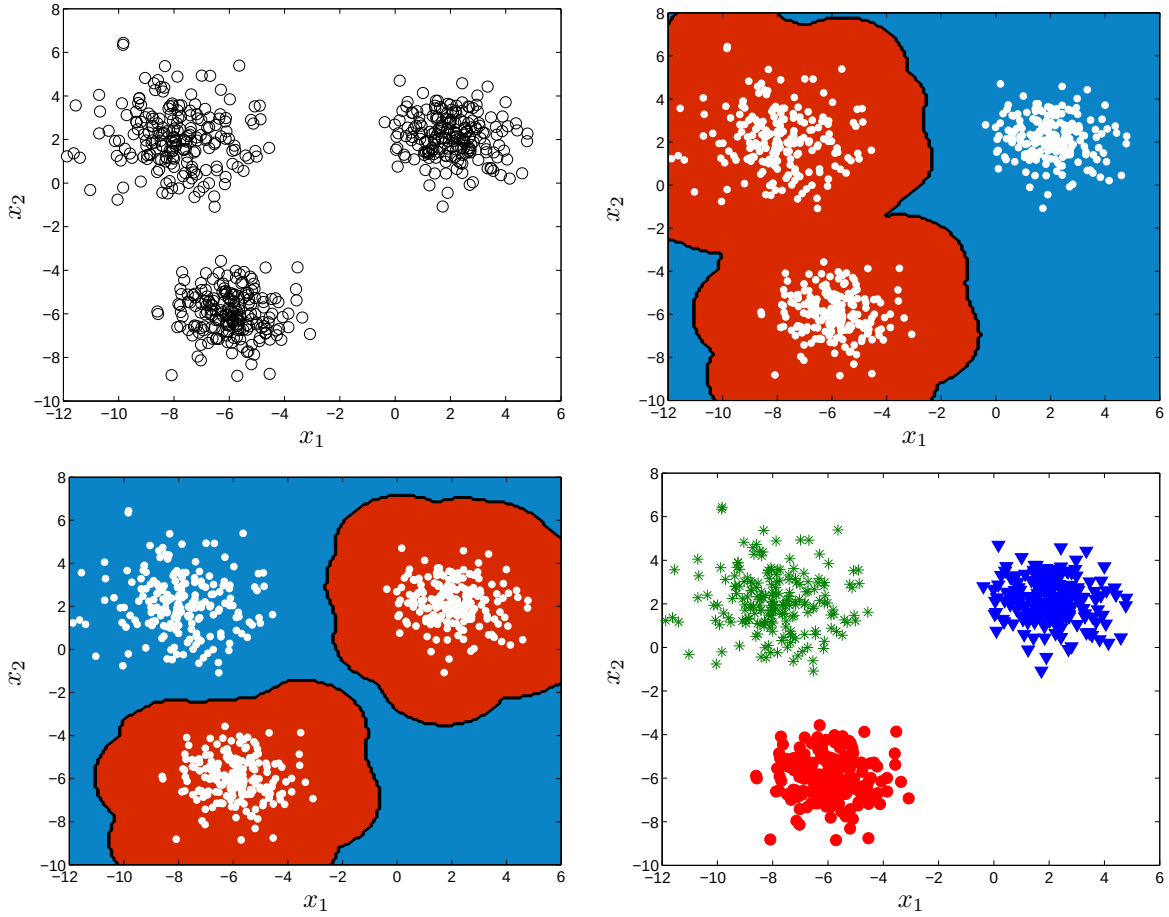

Fig. 1. Toy problem - Effect of the encoding scheme. **Top left:** Full dataset with 3 clusters and 800 data points. **Top right:** Bipartitioning induced by binarizing the first eigenvector $\alpha^{(1)}$ of (14). **Bottom left:** Bipartitioning induced by binarizing the second eigenvector $\alpha^{(2)}$ of (14). **Bottom right:** Cluster results after encoding the two binary partitions. The RBF kernel parameter was set to $\sigma^2 = 0.08$. The training set consisted of 200 randomly selected data points. The cluster indicators of the remaining data points were inferred using the out-of-sample extension and the ECOC decoding procedure.

silhouette index find the correct number of clusters, but for the selection of σ^2 the silhouette index selects a non reliable value. The BLF is very close to the adjusted Rand index yielding maxima in the same range of perfect performance. Note that the eigengap is very small in this experiment, making the eigendecomposition problematic. Despite this fact, the BLF succeeds in selecting optimal parameters. Figure 4 shows the score variables and the clustering results for an optimal value of σ^2 and for a value far away from the optimal range found by the BLF. The score variables in the optimal case are perfect lines resulting in an optimal clustering. In the non-optimal case, the score variables do not show the line structure leading to a wrong clustering.

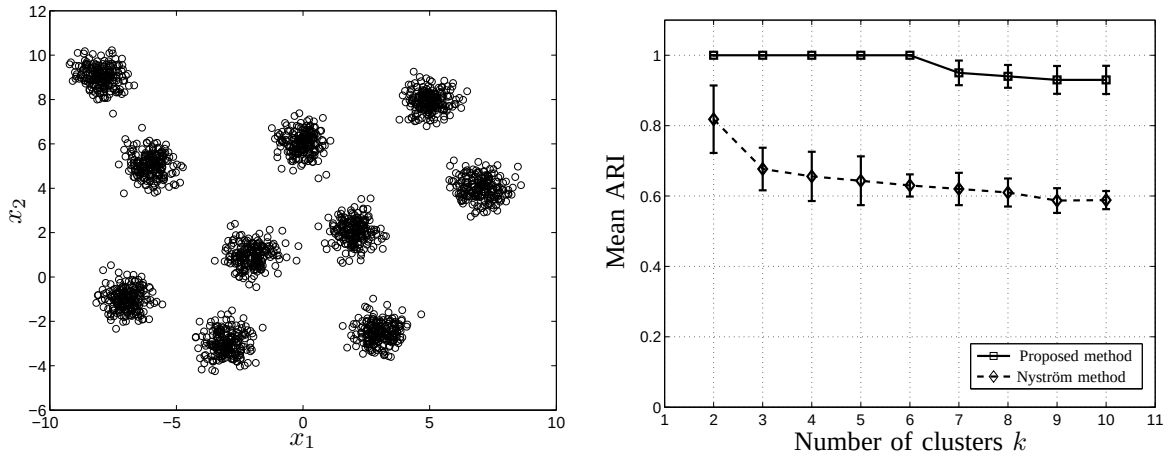

Fig. 2. Toy problem - ARI with increasing number of clusters. **Left:** Full dataset with 10 clusters and $2,000$ data points. **Right:** Mean ARI compared to the optimal clustering after 10 randomizations of the training set. The number of training data points was set to one-fifth of the total number of points. **Solid line:** Proposed approach. **Dashed line** Nyström approach [11]. The proposed method performs better than the Nyström method in terms of ARI with respect to an increasing number of clusters and small training set sizes.

Figure 5 shows comparisons of the proposed method with the Nyström method in [11]. We compute the ARI between cluster indicators obtained through different random subsamplings. This can be seen as a measure of pairwise clustering agreement between different randomizations of the training data. Comparisons with respect to the optimal clustering are also reported. The proposed method shows less variability with respect to random subsamplings compared to the Nyström method. This issue can be due to convergence problems of the reclustering step using k-means. In the proposed method, the agreement between cluster indicators tends to stabilize as the number of training points increases. Computation times are reported in Figure 6.

B. Image Segmentation

For the image segmentation experiments, we used 10 color images from the Berkeley image dataset[1] [40]. We computed a local color histogram with a 5×5 pixels window around each pixel using minimum variance color quantization of 8 levels. We used the χ^2 test to compute the distance between two local color histograms $h^{(i)}$ and $h^{(j)}$ [41]:

$$\chi^2_{ij} = \frac{1}{2} \sum_{b=1}^{B} \frac{(h_b^{(i)} - h_b^{(j)})^2}{h_b^{(i)} + h_b^{(j)}},$$

[1]http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

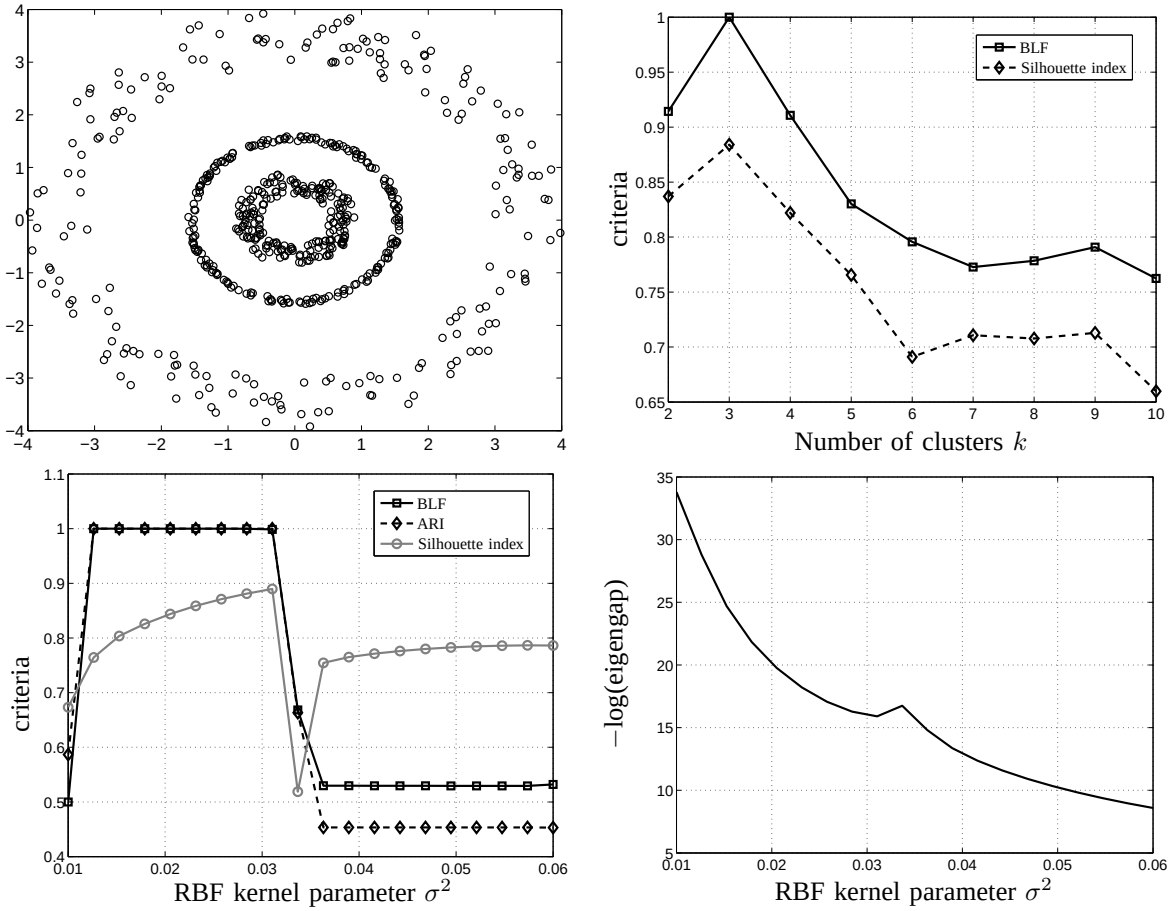

Fig. 3. Toy problem 1 - **Top Left:** Training set. **Top Right:** Maximum of the BLF (solid), maximum of the mean silhouette index applied to the score variables (dashed). Both criteria are maximal for $k = 3$. **Bottom Left:** Selection of σ^2. BLF for $k = 3$ (solid), adjusted Rand index (dashed), mean silhouette index for $k = 3$ (dotted). Note that the BLF is very similar to the adjusted Rand index. **Bottom Right:** Eigengap of $M_D\Omega$ as a function of σ^2 for $k = 3$. Note that the eigengap is very small, indicating a difficult eigenvalue problem.

where B is total number of quantization levels. The histograms are assumed to be normalized $\sum_{b=1}^{B} h_b^{(i)} = 1, i = 1, \ldots, N$. The χ^2 kernel $K(h^{(i)}, h^{(j)}) = \exp^{-\chi^2_{ij}/\sigma_\chi}$ with parameter $\sigma_\chi \in \mathbb{R}^+$ is positive definite and has shown to be a robust and efficient way of comparing the similarity between two histograms for color discrimination and image segmentation [11].

We performed model selection using the BLF for determining the optimal number of clusters k and the χ^2 kernel parameter σ_χ. The training scenario consisted of $1,000$ pixel histograms for training and $20,000$ pixel histograms for validation. The training and validation sets were randomized 20 times. Figure 7 shows the model selection results for image with ID 167062. The obtained tuned parameters are $k = 2, \sigma_\chi = 0.09$. Similar tuning plots were used to determine

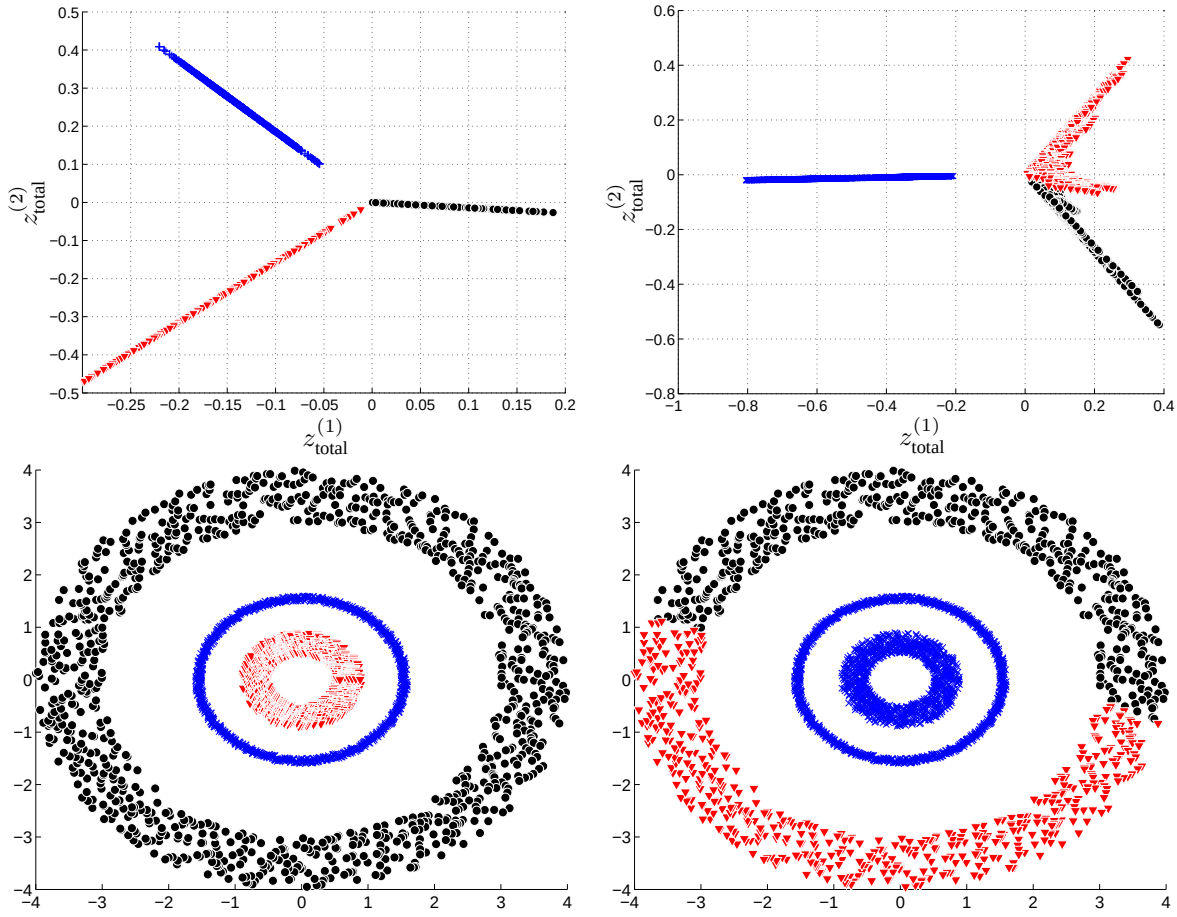

Fig. 4. Toy problem 1 - **Top Left:** Score variables for the full dataset with an optimal $\sigma^2 = 0.02$ according to the BLF on validation data. The triangles (red), crosses (blue) and circles (black) represent the clusters. Note that the line structure corresponds to the best solution. **Top Right:** Score variables for the full dataset with a non-optimal $\sigma^2 = 0.05$. **Bottom Left:** Clustering results with optimal σ^2. **Bottom Right:** Clustering results with a non-optimal $\sigma^2 = 0.05$

the optimal parameters of each of the 10 images used. Figure 8 shows the obtained clustering and the validation score variables for image with ID 145086 using tuned $k = 4, \sigma_\chi = 0.084$. Each cluster in the image is represented as a line in the score variables space. Figure 9 show the image segmentation results for 10 images. After tuning, the methods are trained with $1,000$ pixel histograms randomly chosen. The cluster indicators of the remaining pixel histograms are inferred using the out-of-sample extension in the proposed method and approximated using Nyström in the case of [11]. The clustering indicators represent a segment-label image. The edges are then computed by applying a Canny edge detector over the segment-label image. Table II shows the comparison between the two methods using a performance measure with respect to

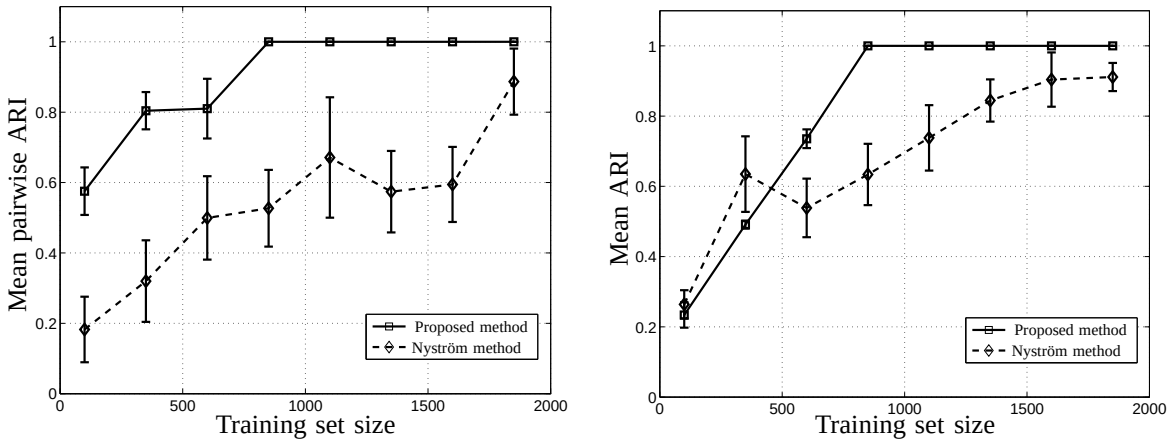

Fig. 5. Toy problem - Agreement between cluster indicators. **Left:** Mean adjusted Rand index between pairwise cluster indicators obtained through different random subsamplings. **Right:** Mean adjusted Rand index between cluster indicators and the optimal clustering. The σ^2 parameter was set to 0.02 and 20 random subsamplings were performed. Note that the Nyström method fails to stabilize even with increasing number of samples. On the other hand, the proposed method converges to the optimal clustering as the number of training samples increases.

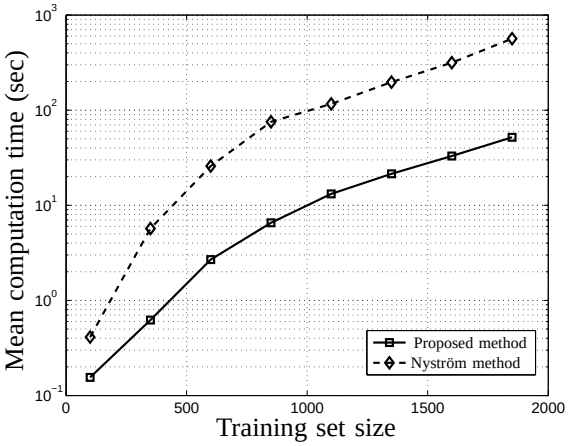

Fig. 6. Toy problem - Mean computation times. **Solid line:** Proposed method. **Dashed line:** Spectral clustering using Nyström [11]. The main bottleneck in the Nyström algorithm is the orthogonalization of the approximated eigenvectors.

human segmentation. The performance criterion used is the F-measure [42], [43]. This measure is known to be meaningful for evaluating image segmentation boundaries. The F-measure is closely related to the area under Receiver Operating Characteristic (ROC) curves and is defined as the harmonic mean between precision and recall. The proposed method outperforms the Nyström algorithm in terms of agreement with human segmentation. Mean computation times can be seen in Figure 11. The bottleneck of the Nyström method corresponds to the orthogonalization steps.

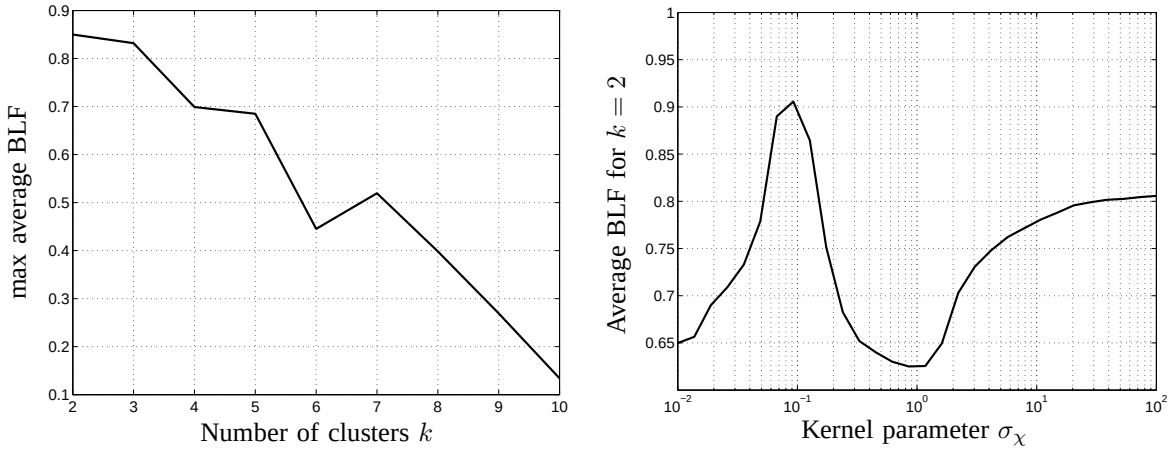

Fig. 7. Image Segmentation - Model selection using the BLF for image with ID 167062. **Left:** Parameter tuning for determining the number of clusters k. **Right:** Tuning the kernel parameter σ_χ. The model selection scenario consists of $1,000$ pixel histograms for training and $20,000$ pixel histograms for validation. The training and validation sets were randomized 20 times. Obtained tuned parameters are $k = 2, \sigma_\chi = 0.09$.

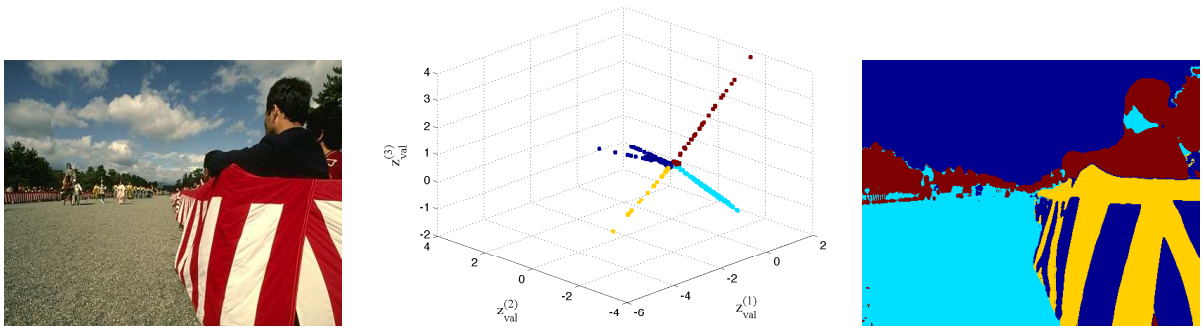

Fig. 8. Image Segmentation - Model selection using the BLF. **Left:** Original image. **Center:** Score variables for the validation set. The tuned parameters obtained through the BLF are $k = 4, \sigma = 0.084$. Note the strong line structures present on the score variables. **Right:** Segment-label image.

For training set sizes $N > 2,000$ the proposed methods runs faster than [11] by at least one order of magnitude.

VIII. Conclusions

A new formulation for multiway spectral clustering with out-of-sample extensions is proposed. This formulation is based on a weighted kernel PCA framework in which the clustering model can be extended to new points. The proposed approach is cast in a constrained optimization framework providing primal-dual insights in a learning scheme. In the case of well formed

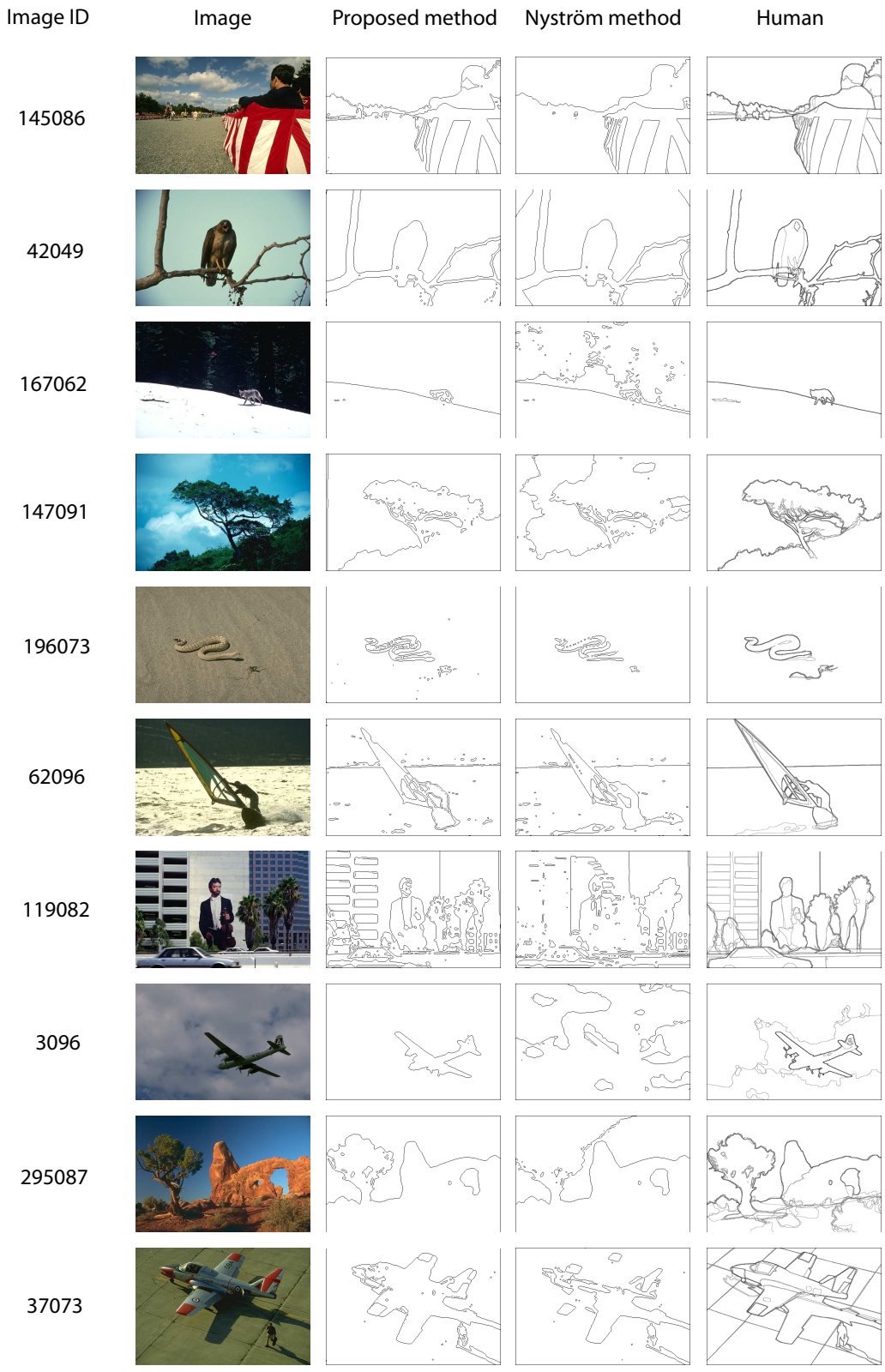

Fig. 9. Image segmentation results using $1,000$ randomly chosen pixel histograms. **First column:** Original image. **Second column:** Segmentation results using the proposed method. **Third column:** Segmentation results using [11]. **Fourth column:** Human segmentation. The proposed approach performs better than the Nyström algorithm in terms of agreement with human segmentation. The performance in terms of F-measure of the full test set can be seen in Table II.

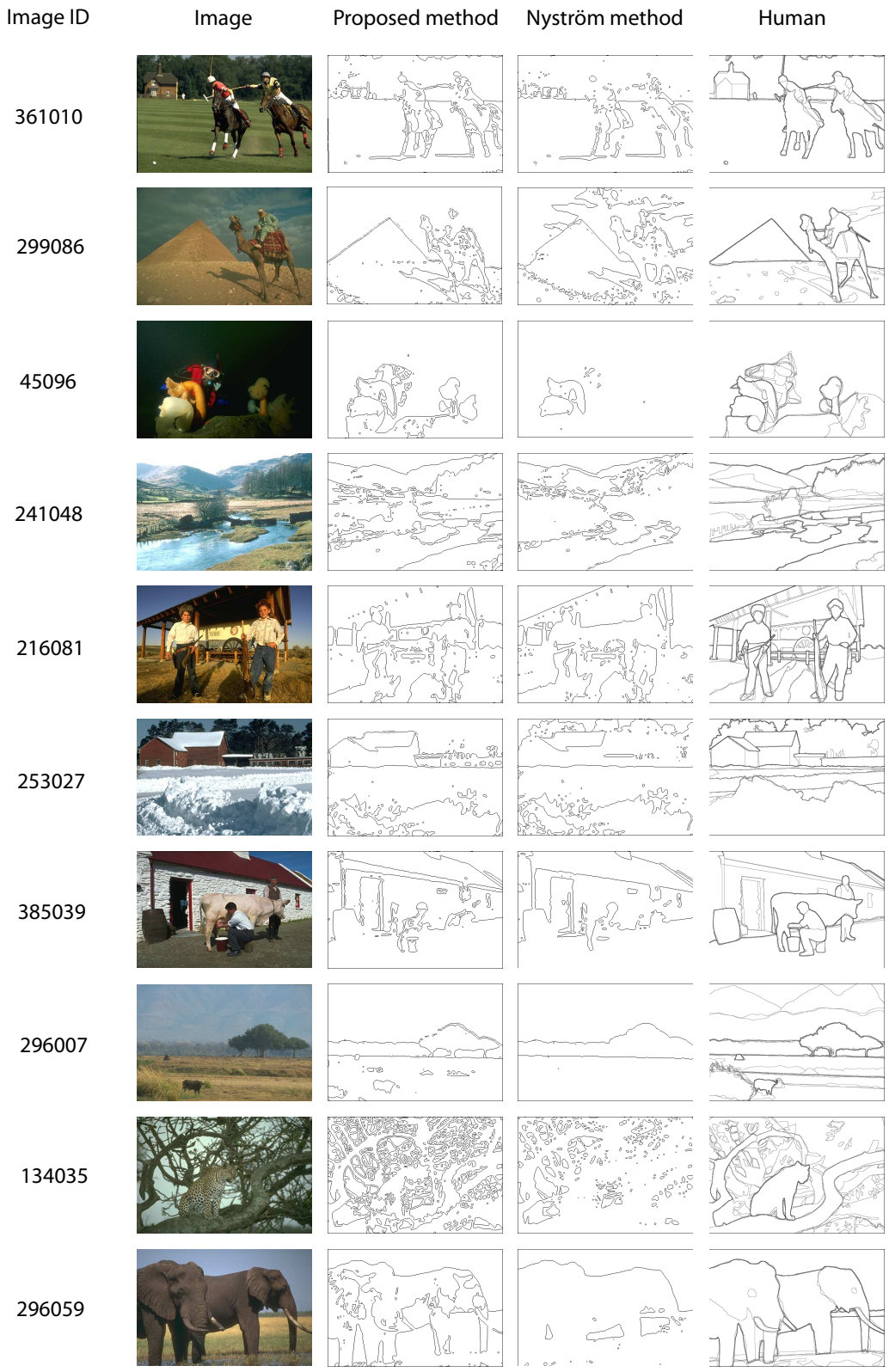

Fig. 10. Image segmentation results using $1,000$ randomly chosen pixel histograms. **First column:** Original image. **Second column:** Segmentation results using the proposed method. **Third column:** Segmentation results using [11]. **Fourth column:** Human segmentation. The proposed approach performs better than the Nyström algorithm in terms of agreement with human segmentation. The performance in terms of F-measure of the full test set can be seen in Table II.

TABLE II

PERFORMANCE COMPARISON OF THE FULL TEST IMAGES FROM THE BERKELEY IMAGE DATASET. **(A)** PROPOSED METHOD. **(B)** NYSTRÖM METHOD [11]. THE PROPOSED METHOD PERFORMS BETTER THAN THE NYSTRÖM ALGORITHM ON 83 OUT OF 100 TEST IMAGES IN TERMS OF AGREEMENT WITH RESPECT TO HUMAN SEGMENTATION.

Image ID	$F-$measure		Human
	(a)	(b)	
145086	**0.88**	0.78	0.85
42049	**0.88**	0.87	0.96
167062	**0.85**	0.46	0.95
147091	**0.8**	0.68	0.87
196073	**0.79**	0.74	0.85
62096	**0.78**	0.76	0.90
101085	0.68	**0.77**	0.93
69015	**0.77**	0.61	0.74
119082	**0.73**	0.62	0.80
3096	**0.72**	0.27	0.74
295087	**0.72**	0.60	0.82
37073	**0.72**	0.64	0.78
182053	0.65	**0.71**	0.75
101087	0.60	**0.71**	0.86
361010	**0.70**	0.58	0.91
299086	**0.70**	0.52	0.82
45096	**0.69**	0.35	0.77
241048	**0.69**	0.56	0.80
216081	0.66	**0.69**	0.86
253027	**0.68**	0.52	0.77
302008	**0.66**	0.59	0.86
385039	0.62	**0.66**	0.83
296007	**0.66**	0.47	0.83
285079	**0.66**	0.47	0.69
54082	**0.65**	0.47	0.68
189080	**0.65**	0.57	0.87
134035	**0.64**	0.61	0.54
126007	**0.64**	0.62	0.87
157055	**0.64**	0.56	0.78
21077	0.64	0.64	0.82
296059	**0.64**	0.53	0.88
41033	**0.64**	0.60	0.88
38092	0.62	**0.63**	0.87
219090	**0.63**	0.58	0.90
24077	0.63	0.63	0.79
306005	**0.63**	0.62	0.67
85048	**0.62**	0.52	0.85
271035	0.58	**0.62**	0.82
241004	**0.62**	0.60	0.94
66053	**0.61**	0.53	0.81
175032	**0.60**	0.55	0.73
42012	**0.60**	0.53	0.79
260058	**0.60**	0.37	0.73
143090	**0.60**	0.59	0.84
197017	**0.60**	0.56	0.89
208001	**0.60**	0.41	0.78
69020	**0.59**	0.49	0.86
148089	**0.59**	0.54	0.70
223061	**0.59**	0.44	0.60
76053	**0.59**	0.55	0.58

Image ID	$F-$measure		Human
	(a)	(b)	
376043	**0.59**	0.50	0.81
175043	**0.59**	0.56	0.50
58060	**0.59**	0.53	0.52
19021	**0.58**	0.52	0.80
159008	**0.58**	0.56	0.64
167083	**0.58**	0.44	0.75
14037	**0.58**	0.41	0.77
86000	**0.58**	0.52	0.73
351093	0.53	**0.57**	0.81
78004	**0.57**	0.54	0.81
300091	0.47	**0.57**	0.84
38082	0.56	0.56	0.74
103070	**0.56**	0.47	0.79
227092	**0.56**	0.49	0.65
106024	0.37	**0.55**	0.84
229036	**0.55**	0.47	0.86
148026	**0.55**	0.38	0.70
87046	**0.55**	0.47	0.82
65033	**0.54**	0.41	0.81
170057	**0.54**	0.32	0.76
291000	**0.53**	0.50	0.80
210088	**0.52**	0.45	0.54
102061	**0.52**	0.46	0.85
220075	**0.52**	0.42	0.78
97033	**0.51**	0.42	0.84
160068	**0.5**	0.42	0.91
41069	**0.5**	0.44	0.79
16077	**0.51**	0.43	0.87
236037	0.48	**0.49**	0.64
156065	**0.49**	0.43	0.69
89072	0.47	**0.49**	0.84
33039	**0.48**	0.42	0.75
304074	**0.48**	0.40	0.93
105025	**0.48**	0.36	0.77
109053	**0.48**	0.42	0.84
163085	**0.47**	0.40	0.73
12084	0.26	**0.47**	0.62
108005	**0.47**	0.41	0.81
253055	**0.47**	0.33	0.95
86016	**0.45**	0.41	0.77
55073	0.42	**0.45**	0.65
69040	0.38	**0.41**	0.50
86068	0.33	**0.41**	0.73
43074	**0.40**	0.34	0.72
304034	**0.39**	0.21	0.84
108082	**0.36**	0.16	0.90
130026	**0.36**	0.35	0.70
123074	**0.35**	0.30	0.87
108070	**0.33**	0.28	0.62
8023	**0.31**	0.29	0.81

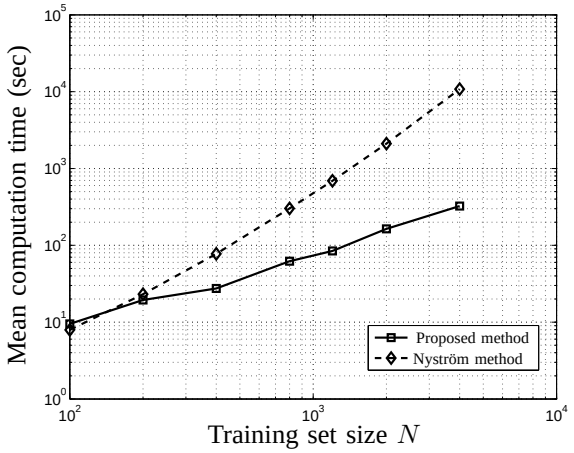

Fig. 11. Image segmentation - Mean computation times for 10 images of the Segbench dataset over 10 randomizations of the training set. **Solid line:** Proposed method. **Dashed line:** Spectral clustering using Nyström using the one-shot implementation described in [11]. For $N > 2,000$ the difference between the two methods is more than an order of magnitude. The main bottleneck in [11] is the orthogonalization of the approximated eigenvectors.

clusters, the eigenvectors of a modified similarity matrix derived from the data display a special structure. Training data points in the same cluster are mapped into a single point in the space spanned by the eigenvectors. In the case of validation data, points in the same cluster are collinear in the space spanned by the projected variables. This structure is exploited by the proposed model selection criterion to obtain useful model parameters leading to visually appealing and interpretable clusters. Experiments with known difficult toy examples and image segmentation show the applicability of the proposed multiway clustering model and model selection criterion.

ACKNOWLEDGMENT

This work was supported by grants and projects for the Research Council K.U.Leuven (GOA-Mefisto 666, GOA-Ambiorics, several PhD / Postdocs & fellow grants), the Flemish Government FWO: PhD / Postdocs grants, projects G.0240.99, G.0211.05, G.0407.02, G.0197.02, G.0080.01, G.0141.03, G.0491.03, G.0120.03, G.0452.04, G.0499.04, G.0226.06, G.0302.07, ICCoS, ANMMM; AWI;IWT:PhD grants, GBOU (McKnow) Soft4s, the Belgian Federal Government (Belgian Federal Science Policy Office: IUAP V-22; PODO-II (CP/01/40), the EU(FP5-Quprodis, ERNSI, Eureka 2063-Impact;Eureka 2419-FLiTE) and Contracts Research/Agreements (ISMC/IPCOS, Data4s, TML, Elia, LMS, IPCOS, Mastercard). Johan Suykens is a professor at the K.U.Leuven, Belgium. The scientific responsibility is assumed by its authors.

REFERENCES

[1] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[2] F. R. K. Chung, *Spectral Graph Theory*. American Mathematical Society, 1997.

[3] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, pp. 849–856.

[4] L. Hagen and A. Kahng, "New spectral methods for ratio cut partitioning algorithms," *IEEE Trans. Computer-Aided Design*, vol. 11, no. 9, pp. 1074–1085, 1992.

[5] P. K. Chan, M. D. F. Schlag, and J. Y. Zien, "Spectral k-way ratio-cut partitioning and clustering," *IEEE Trans. Computer-Aided Design*, vol. 13, no. 9, pp. 1088–1096, 1994.

[6] M. Meila and J. Shi, "A random walks view of spectral segmentation," in *Artificial Intelligence and Statistics AISTATS*, T. Jaakkola and T. Richardson, Eds., 2001.

[7] F. R. Bach and M. I. Jordan, "Learning spectral clustering, with application to speech separation," *Journal of Machine Learning Research*, vol. 7, pp. 1963–2001, 2006.

[8] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, pp. 1601–1608.

[9] C. Ding and X. He, "Linearized cluster assignment via spectral ordering," in *ICML '04: Proceedings of the twenty-first International Conference on Machine Learning*. New York, NY, USA: ACM Press, 2004, p. 30.

[10] Y. Bengio, J. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet, "Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. K. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004.

[11] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the Nyström method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 214–225, February 2004.

[12] C. Baker, *The numerical treatment of integral equations*. Clarendon Press, Oxford, 1977.

[13] C. Williams and M. Seeger, "Using the Nyström method to speed up kernel machines," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. MIT Press, 2001.

[14] B. Schölkopf, A. J. Smola, and K. R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299–1319, 1998.

[15] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A kernel view of the dimensionality reduction of manifolds," in *Proceedings of the twenty-first international conference on Machine learning*. New York, NY, USA: ACM Press, 2004, pp. 369–376.

[16] C. Alzate and J. A. K. Suykens, "A weighted kernel PCA formulation with out-of-sample extensions for spectral clustering methods," in *Proceedings of the 2006 International Joint Conference on Neural Networks (IJCNN'06)*, 2006, pp. 138–144.

[17] ——, "Kernel component analysis using an epsilon insensitive robust loss function," *IEEE Transactions on Neural Networks*, vol. 19, no. 9, pp. 1583–1598, 2008.

[18] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.

[19] J. A. K. Suykens, T. Van Gestel, J. Vandewalle, and B. De Moor, "A support vector machine formulation to PCA analysis and its kernel version," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 447–450, March 2003.

[20] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory," *Czech. Math. J.*, vol. 25, no. 100, pp. 619–633, 1975.

[21] A. Pothen, H. D. Simon, and K. P. Liou, "Partitioning sparse matrices with eigenvectors of graphs," *SIAM Journal on Matrix Analysis and Applications*, vol. 11, pp. 430–452, 1990.

[22] M. Hein, J.-Y. Audibert, and U. von Luxburg, "Graph Laplacians and their convergence on random neighborhood graphs," *Journal of Machine Learning Research*, vol. 8, pp. 1325–1370, 2007.

[23] M. Gu, H. Zha, C. Ding, X. He, and H. Simon, "Spectral relaxation models and structure analysis for k-way graph clustering and bi-clustering," Penn. State Univ, Computer Science and Engineering., Technical Report, 2001.

[24] K. Fan, "On a theorem of Weyl concerning eigenvalues of linear transformations." *Proc. Natl. Acad. Sci. USA*, vol. 35, no. 11, pp. 652–655, 1949.

[25] M. Meila, "Multiway cuts and spectral clustering," University of Washington, Technical Report, 2003.

[26] C. Davis and W. M. Kahan, "The rotation of eigenvectors by a perturbation. III," *SIAM Journal on Numerical Analysis*, vol. 7, no. 1, pp. 1–46, 1970.

[27] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[28] F. R. Bach and M. I. Jordan, "Learning spectral clustering," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. K. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004.

[29] H. Zha, C. Ding, M. Gu, X. He, and H. Simon, "Spectral relaxation for k-means clustering," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2002.

[30] N. Cristianini, J. Shawe-Taylor, and J. Kandola, "Spectral kernel methods for clustering," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, pp. 649–655.

[31] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola, "On kernel-target alignment," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002.

[32] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Machine Learning Research*, vol. 2, pp. 263–286, 1995.

[33] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.

[34] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.

[35] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, pp. 585–591.

[36] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004.

[37] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a dataset via the gap statistic," *Journal of the Royal Statistical Society B*, vol. 63, pp. 411–423, 2001.

[38] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, no. 1, pp. 53–65, 1987.

[39] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, pp. 193–218, 1985.

[40] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, 2001, pp. 416–423.

[41] J. Puzicha, T. Hofmann, and J. Buhmann, "Non-parametric similarity measures for unsupervised texture segmentation and image retrieval," in *Computer Vision and Pattern Recognition*, 1997, pp. 267–272.

[42] C. Van Rijsbergen, *Information Retrieval*. Dept. of Computer Science, Univ. of Glasgow, 1979.

[43] D. Martin, C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 1–20, 2004.