S^{3}CCA: Smoothly Structured Sparse CCA For Partial Pattern Matching

Takumi Kobayashi
National Institute of Advanced Industrial Science and Technology
1-1-1 Umezono, Tsukuba, Japan
E-mail: takumi.kobayashi@aist.go.jp

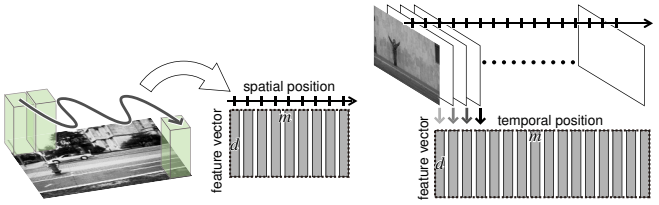
***Abstract*—The partial pattern matching is fundamental for pattern recognition to compare the pair of input patterns by exploiting the common features shared by those patterns while excluding the irrelevant ones. In this paper, for the pattern matching, we propose a novel method of smoothly structured sparse canonical correlation analysis, called S^{3}CCA. The proposed method works on the feature matrix composed of a (local) feature dimension and an array dimension. In the framework of CCA, the method provides map weights along the array dimension to depict the parts that exhibit the common/similar features across the pair of feature matrices. By introducing the appropriate regularization into CCA, the map weights are optimized so as to be both smooth and localized, i.e., structured sparse. Thereby, the common features are effectively detected by the smooth and well-localized weights to improve the matching performance. In the experiments on pattern matching as well as classification based on the matching, the proposed method produces the favorable performance compared to the other methods.**

I. Introduction

In pattern recognition, pattern matching is an important task to compare pair of data, such as for measuring the pair-wise similarity. The data, e.g., time-series sequences and two-dimensional images, usually contain the irrelevant and unessential patterns as backgrounds. Hence, in the pattern matching, it is demanded to extract the essential parts that are common/similar across the pair of pattern data to be compared, while excluding the other irrelevant parts. In that sense, we address the problem of *partial* pattern matching in this paper.

The conventional way for pattern matching is to compute the similarity between the pair of feature vectors based on a certain type of metric; the simplest one is Euclidean distance in the Gaussian kernel similarity [1]. And, the methods of metric learning [2], [3] including PCA and LDA [4] would provide the discriminative metric for the matching. Those methods assume that the input pattern data is represented in the form of a single feature *vector*, retaining the characteristics of the essential patterns while possibly eliminating the irrelevant ones. In order to extract such favorable feature vectors, it is required to roughly detect and segment out the essential part of interest by preprocessing. In contrast, we deal with the feature *arrays* that considerably contain the irrelevant patterns as well as the target one; the feature arrays that we use in this paper are formulated in a *matrix* form (two-way). Those feature matrices are found in many cases; for example, image pixels and local feature vectors extracted at spatio-temporal (grid) points in images/motion sequences as shown in Fig. 1.

The partial pattern matching is also related to the co-segmentation in the literature of image segmentation [5], [6]. The co-segmentation methods treat the local features extracted at (super-)pixels individually so that the discriminative classifier is applied to those feature vectors. In this study, we focus on the unsupervised matching between the (weighted) sum of the local feature vectors over the local region where the common/similar patterns are found in the pair of data. The operation of summation is typically employed in the histogram-based feature extraction, such as HOG [7] and GLAC [8], and the aggregated features are so discriminative (and salient) as to increase the similarity across the pair of patterns with robustness to noise.



(a) image (b) motion sequence

Fig. 1. Examples of the feature matrix. The local feature vectors are extracted from (a) an image and (b) a motion sequence to form the matrix.

In this paper, we propose the method of smoothly structured sparse canonical correlation analysis (S^{3}CCA) for partial pattern matching. The proposed method operates on the pair of feature matrices comprising feature dimension (row) and array dimension (column), and in the framework of CCA it produces map weights along the array dimension to indicate the sub-regions (parts) exhibiting the common/similar pattern shared by the pair of the features. The map weights are optimized so as to be both *smooth* and *localized*, i.e., structured sparse, by introducing the appropriate regularizations into the CCA framework, which is our main contribution. As described above (and in Fig. 1), the array dimension in the feature matrix is often related to the physical coordinates, such as the spatio-temporal positions. Hence, the parts shared by the feature matrices should be distributed *smoothly*, not scattered, and be well-localized. This fact leads to that the map weights are subject to both the smoothness and the structured sparseness regularizations as shown in Fig. 2; the smoothness would improve the false negatives by enhancing the consistency of near-by weights, while the structured sparseness would exclude the false positives due to well-localization.

From the viewpoint of matching via CCA, Conrad and Mester [9] applied the method of CCA to compute the pixel correspondence across the pair of images. The method exhaus-

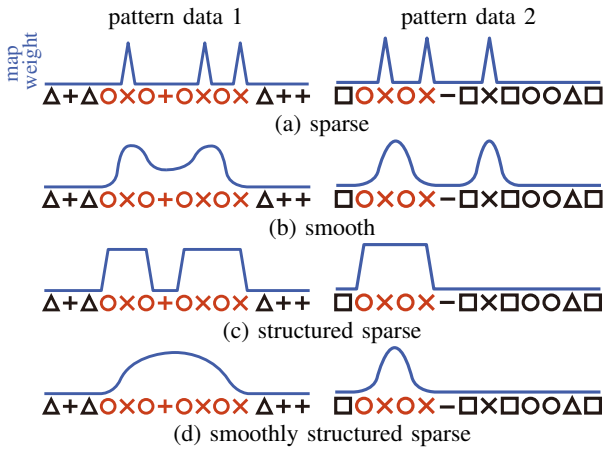


Fig. 2. Effect of smoothly structured sparseness. Each symbol represents the local feature and '+' is degraded from '×'. In this example, the shared pattern is '◯ × ◯ ⋯' marked by red color. (a) The simple sparse regularization would produce individual pattern matching. (b) The smoothness regularization corrects false negatives ('◯'), (c) while that of the structured sparseness enhances the locality with excluding false positives ('×'). (d) The combination of those two types of regularization can provide the smoothly localized weights enhancing the consistency. This figure is best viewed in color.

tively constructs the pixel correspondence through the CCA transformations, which is quite different from our method. Our proposed method can produce the map weights that are directly construable since the weights are activated on the shared parts due to the smooth localization. On the other hand, Chen et al. [10] have proposed the method of structured sparse CCA. It is closely related to our method in that the sparse CCA is developed, but the method deals with the specific problem for expression quantitative trait loci (eQTLs) mapping. Our method is formulated in the general form of CCA, including the method [10] as the special case.

II. Proposed Method

In this section, we detail the proposed method, *smoothly structured sparse CCA (S^3CCA)*, for partial pattern matching. We first formulate the partial pattern matching in the framework of CCA, and then extend the CCA by introducing the regularizations to produce the smoothly localized map weights which contribute to improve the partial matching.

A. Pattern Matching by Canonical Correlation Analysis

The problem of partial pattern matching that we consider in this paper is formulated as follows. Suppose the input pattern data are represented by the two-way feature arrays (matrices) of $\boldsymbol{X} \in \mathbb{R}^{d\times m}$ and $\boldsymbol{Y} \in \mathbb{R}^{d\times n}$ comprising d-dimensional local feature vectors in their columns with the number of features m and n as shown in Fig. 1. The task of partial pattern matching is to detect the regions (parts) in the arrays such that the features summed over the region are similar across the two arrays, which is simply described by

$$\boldsymbol{X}\boldsymbol{w}_X = \boldsymbol{x} \underset{\text{similar}}{\sim} \boldsymbol{y} = \boldsymbol{Y}\boldsymbol{w}_Y, \tag{1}$$

where $\boldsymbol{w}_X \in \mathbb{R}^m$ and $\boldsymbol{w}_Y \in \mathbb{R}^n$ are the map weights used for summation. Thus, the matching task is cast into the optimization of $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ so as to produce the feature vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ which are close to each other.

We measure the similarity between $\boldsymbol{x}$ and $\boldsymbol{y}$ based on the *angle*, inducing the following optimization problem with respect to $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$:

$$\min_{\boldsymbol{w}_X,\boldsymbol{w}_Y} \arccos \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|},\ s.t.\ \boldsymbol{x}=\boldsymbol{X}\boldsymbol{w}_X, \boldsymbol{y}=\boldsymbol{Y}\boldsymbol{w}_Y, \tag{2}$$

$$\Leftrightarrow \min_{\boldsymbol{w}_X,\boldsymbol{w}_Y} \frac{1}{2}\|\boldsymbol{X}\boldsymbol{w}_X - \boldsymbol{Y}\boldsymbol{w}_Y\|^2,\ s.t.\ \|\boldsymbol{X}\boldsymbol{w}_X\|^2=\|\boldsymbol{Y}\boldsymbol{w}_Y\|^2=1. \tag{3}$$

The formulation (3) is equivalent to CCA [11], [12] and further reduced into

$$\min_{\boldsymbol{w}} \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A}\boldsymbol{w},\ s.t.\ \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w} = 1, \tag{4}$$

where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{X} & -\boldsymbol{X}^\top\boldsymbol{Y} \\ -\boldsymbol{Y}^\top\boldsymbol{X} & \boldsymbol{Y}^\top\boldsymbol{Y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_X & -\boldsymbol{K}_{XY} \\ -\boldsymbol{K}_{XY}^\top & \boldsymbol{K}_Y \end{bmatrix}, \tag{5}$$

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{X}^\top\boldsymbol{X} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Y}^\top\boldsymbol{Y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_X & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{K}_Y \end{bmatrix},\ \boldsymbol{w} = \begin{bmatrix} \boldsymbol{w}_X \\ \boldsymbol{w}_Y \end{bmatrix}. \tag{6}$$

In this CCA, the map weights are optimized by solving the generalized eigenvalue problem of (4); $\boldsymbol{A}\boldsymbol{w} = \lambda_{min}\boldsymbol{B}\boldsymbol{w}$. Note that the method is straightforwardly kernelized by applying the kernel trick [13] to the Gram matrices $\boldsymbol{K}_X, \boldsymbol{K}_Y$ and $\boldsymbol{K}_{XY}$.

B. Smoothly Structured Sparse CCA (S^3CCA)

CCA (4) optimizes the map weights by greedily minimizing the angle, equivalently Euclidean distance, in disregard of any prior knowledge about the weights. As described in Sec.I, the feature arrays $\boldsymbol{X}$ and $\boldsymbol{Y}$ are often extracted from the physically regularized domain, such as spatio-temporal domain (Fig. 1). Thus, we propose the method to effectively incorporate the inherent physical structure into the above-defined framework by utilizing the two types of regularization; namely, the map weights are optimized so as to be both smooth (Sec.II-B1) and localized (structured sparse) (Sec.II-B2).

1) Smoothness Regularization: The map weights $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ are occasionally connected to physical coordinates; e.g., those weights work on spatio-temporal positions. In such cases, it is useful to take into account the physical relationships among the weight components as regularization in order to facilitate the partial matching. For instance, in the case of time-series signal sequences, the extracted local features are not independently drawn but naturally have continuity between adjacent features, expecting smooth weights $\boldsymbol{w}$ along the time axis. Such smooth weights would correct the false negatives in matching (Fig. 2). In this study, we introduce the smoothness regularization defined in the convex form as follows.

Smooth weights are generally obtained via minimizing the Laplacian,

$$l(\boldsymbol{w}) = \int |\Delta w(\boldsymbol{p})|^2 d\boldsymbol{p}, \tag{7}$$

where $\boldsymbol{p}$ denotes the physical position over which the weights are defined; e.g., for temporal sequence $\boldsymbol{p} = t$,

$$l(\boldsymbol{w}) = \sum_t |-w_{t-1} + 2w_t - w_{t+1}|^2, \tag{8}$$

and for 2-dimensional images $\boldsymbol{p} = [x, y]^\top$,

$$l(\boldsymbol{w}) = \sum_{x,y} | - w_{x-1,y-1} - 2w_{x-1,y} - w_{x-1,y+1} - 2w_{x,y-1} + 12w_{x,y} - 2w_{x,y+1} - w_{x+1,y-1} - 2w_{x+1,y} - w_{x+1,y+1}|^2. \quad (9)$$

These Laplacian costs are written in the convex form $l(\boldsymbol{w}) = \boldsymbol{w}^\top \boldsymbol{L}\boldsymbol{w}$ where the matrix $\boldsymbol{L}$ is determined according to the forms of Laplacian (8, 9). The smooth weights are obtained by introducing the smoothness regularization into (4) as

$$\min_{\boldsymbol{w}} \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A}\boldsymbol{w} + \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{L}\boldsymbol{w},\ s.t.\ \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w} = 1, \quad (10)$$

where the Laplacian regularization is represented by the matrix

$$\boldsymbol{L} = \begin{bmatrix} \eta_X \boldsymbol{L}_X & \boldsymbol{0} \\ \boldsymbol{0} & \eta_Y \boldsymbol{L}_Y \end{bmatrix}, \quad (11)$$

with the regularization parameters η_X and η_Y for $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$.

2) Structured Sparseness Regularization: The partial matching seeks for the *parts* which coincide across the pair of the feature arrays. That is, the optimized weights $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ are demanded to be well localized, assigning non-zeros to only the local parts in the weights and zeros to the others. Such kind of locality has been discussed in the structured sparseness [14].

Jenatton et al. [14] proposed the method to induce structured sparseness by considering overlapped mixed-norm (L_α/L_2, $0 < \alpha < 2$) regularizations in the form of

$$\Omega(\boldsymbol{w}) = \{\sum_{G\in\mathbb{G}} \|\boldsymbol{d}^G \circ \boldsymbol{w}\|_2^\alpha\}^{\frac{1}{\alpha}}, \quad (12)$$

where $\circ$ denotes the Hadamard product, $\boldsymbol{d}^G$ is the coefficient vector to describe the structured pattern G, and $\mathbb{G}$ indicates the set of those patterns. Note that the structured patterns G are overlapped to induce the structured sparseness as shown in Fig. 3; for details of $\boldsymbol{d}^G$, refer to [14]. To aggressively impose sparseness on the weights, we employ $\alpha = 0.5$ in this study, though it results in the non-convex optimization. While the conventional way to provide component-wise sparseness is given such as by L_1 regularization, the regularization (12) renders the sparseness that is localized in the structured form, e.g., rectangles or segments.

Finally, we obtain the formulation for S^{3}CCA by

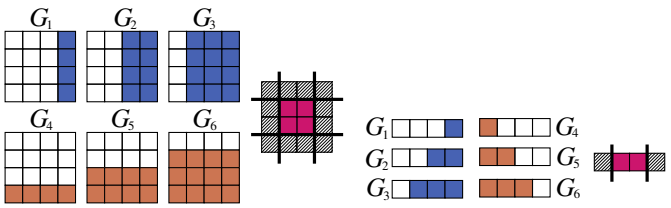
$$\min_{\boldsymbol{w}} \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{A}\boldsymbol{w} + \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{L}\boldsymbol{w} + \rho_X \Omega(\boldsymbol{w}_X) + \rho_Y \Omega(\boldsymbol{w}_Y), \quad (13)$$

$$s.t.\ \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w} = 1, \quad (14)$$

where ρ_X and ρ_Y are the regularization parameters regarding $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$, respectively. The proposed method produces the map weights $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ which are favorably smooth and localized; the combination of the smoothness and the structured sparseness enhances the locality of the weights suppressing the scattered weights (see Fig. 2). The efficient optimization approach for (13) is proposed in the next section.

III. Optimization

The main concern in the optimization (13) is how to deal with the sparseness regularization (12) as well as the norm constraint (14).



(a) 2-dimensional (b) 1-dimensional

Fig. 3. Examples of overlapped patterns for the structured sparseness.

A. Structured sparseness regularization

First, we address the regularization $\Omega(\boldsymbol{w})$ which is difficult to directly treat. For optimization, we resort to Lemma 3.1 in [15]: for any vector $\boldsymbol{y} \in \mathbb{R}^p$,

$$\|\boldsymbol{y}\|_\alpha = \min_{\boldsymbol{z}\in\mathbb{R}_+^p} \frac{1}{2}\sum_{j=1}^p \frac{y_j^2}{z_j} + \frac{1}{2}\|\boldsymbol{z}\|_\beta, \quad (15)$$

where the α-norm $\|\boldsymbol{y}\|_\alpha = \{\sum_i^p |y_i|^\alpha\}^{\frac{1}{\alpha}}$ and $\beta = \frac{\alpha}{2-\alpha}$. The optimizer is uniquely attained as

$$z_j = |y_j|^{2-\alpha}\|\boldsymbol{y}\|_\alpha^{\alpha-1},\ j = 1, .., p. \quad (16)$$

Based on (15), by introducing the auxiliary variables $\boldsymbol{z}_X \in \mathbb{R}_+^{|\mathbb{G}_X|}$ and $\boldsymbol{z}_Y \in \mathbb{R}_+^{|\mathbb{G}_Y|}$ where $|\mathbb{G}_X|$ and $|\mathbb{G}_Y|$ indicate the numbers of patterns in $\mathbb{G}_X$ and $\mathbb{G}_Y$, the formulation (13) can be transformed into

$$\min_{\boldsymbol{w},\boldsymbol{z}=[\boldsymbol{z}_X^\top,\boldsymbol{z}_Y^\top]^\top} \frac{1}{2}\boldsymbol{w}^\top(\boldsymbol{A}+\boldsymbol{L})\boldsymbol{w} + \frac{\rho_X}{2}\sum_{G_X\in\mathbb{G}_X}\frac{\|\boldsymbol{d}^{G_X}\circ\boldsymbol{w}_X\|_2^2}{z_X^{G_X}} + \frac{\rho_X}{2}\|\boldsymbol{z}_X\|_\beta + \frac{\rho_Y}{2}\sum_{G_Y\in\mathbb{G}_Y}\frac{\|\boldsymbol{d}^{G_Y}\circ\boldsymbol{w}_Y\|_2^2}{z_Y^{G_Y}} + \frac{\rho_Y}{2}\|\boldsymbol{z}_Y\|_\beta, \quad (17)$$

$$s.t.\ \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w} = 1, \quad (18)$$

which is further rewritten to

$$\min_{\boldsymbol{w},\boldsymbol{z}} \frac{1}{2}\boldsymbol{w}^\top\{\boldsymbol{A}+\boldsymbol{L}+\mathrm{diag}([\rho_X\boldsymbol{\zeta}_X^\top, \rho_Y\boldsymbol{\zeta}_Y^\top]^\top)\}\boldsymbol{w} + \frac{\rho_X}{2}\|\boldsymbol{z}_X\|_\beta + \frac{\rho_Y}{2}\|\boldsymbol{z}_Y\|_\beta, \quad (19)$$

$$s.t.\ \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w} = 1, \quad (20)$$

where $\zeta_{Xj} = \sum_{G_X\in\mathbb{G}_X, j\in G_X} \frac{(d_j^{G_X})^2}{z_X^{G_X}}$ $(j = 1, .., m)$, $\zeta_{Yj} = \sum_{G_Y\in\mathbb{G}_Y, j\in G_Y} \frac{(d_j^{G_Y})^2}{z_Y^{G_Y}}$ $(j = 1, .., n)$. The variables $\boldsymbol{w}$ and $\boldsymbol{z} = [\boldsymbol{z}_X^\top, \boldsymbol{z}_Y^\top]^\top$ are alternately optimized by fixing the other in an iterative manner.

B. Norm constraint

Under the norm constraint of (20), the objective cost (19) can be minimized via solving the generalized eigenvalue problem. It, however, suffers from large computation cost especially through the iterative optimization in which the generalized eigenvalue problem is solved many times. We circumvent it based on the fact that in (19) only the minimum eigenvalue/eigenvector is required at the optimum. Thus, we

employ the conjugate gradient (CG) method [16] for efficiently computing the extreme eigenvalue/eigenvector.

By fixing $\boldsymbol{z}$ (and $\boldsymbol{\zeta}$), the optimization (19) under (20) is transformed into the following unconstrained problem w.r.t $\boldsymbol{w}$;

$$\min_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{C}\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w}}, \tag{21}$$

where $\boldsymbol{C} = \boldsymbol{A} + \boldsymbol{L} + \mathrm{diag}([\rho_X \boldsymbol{\zeta}_X^\top, \rho_Y \boldsymbol{\zeta}_Y^\top]^\top)$. The method of CG efficiently solves such an unconstrained problem and it is noteworthy that in this optimization the step size is analytically computed [16] without applying the exhaustive line search such as Armijo rule [17]. The algorithm of the CG is shown in Algorithm 1. Note that the CG gives the global optimum corresponding the minimum eigenvalue/eigenvector.

As a result, the whole procedure for optimizing (13) is presented in Algorithm 2. In this procedure, the time consuming process is the optimization of (21) and it is efficiently performed by applying the CG (Algorithm 1) instead of the standard solver of the generalized eigenvalue problem.

C. Technical Tips

In the case that the scales (norms) of the feature arrays $\boldsymbol{X}$ and $\boldsymbol{Y}$ are highly biased, the regularization terms regarding both smoothness and sparseness are also biased; e.g., the map weights become small on the feature array that has a large norm, consequently diminishing the effects of those regularizations. For fairly imposing the regularizations on both $\boldsymbol{X}$ and $\boldsymbol{Y}$, we normalize the arrays by the matrix spectral norm; $\boldsymbol{K}_X \leftarrow \boldsymbol{K}_X / \|\boldsymbol{K}_X\|_S$, $\boldsymbol{K}_Y \leftarrow \boldsymbol{K}_Y / \|\boldsymbol{K}_Y\|_S$, $\boldsymbol{K}_{XY} \leftarrow \boldsymbol{K}_{XY} / \sqrt{\|\boldsymbol{K}_X\|_S \|\boldsymbol{K}_Y\|_S}$ where $\|\cdot\|_S$ indicates the spectral norm, i.e., the maximum singular value of the matrix.

Next, we mention the regularization parameters η_X, η_Y, ρ_X and ρ_Y. In the smoothness regularization, the parameters η_X and η_Y are balancing the matrices $\boldsymbol{K}_X, \boldsymbol{L}_X$ and $\boldsymbol{K}_Y, \boldsymbol{L}_Y$, respectively. To fairly consider the balance independently of the norm (scale) of those matrices, we transform the parameters to $\hat{\eta} = \frac{\eta \|\boldsymbol{L}\|_S}{\|\boldsymbol{K}\|_S}$; it actually results in $\hat{\eta} = \eta \|\boldsymbol{L}\|_S$ since the matrix $\boldsymbol{K}$ is normalized as described above, and in this study we set $\hat{\eta} = 1$ to equally balance those matrices $\boldsymbol{K}_X$ and $\boldsymbol{L}_X$ (or $\boldsymbol{K}_Y$ and $\boldsymbol{L}_Y$). As to the structured sparseness regularization, we can not estimate the scale of the regularization term in advance, since it is not simply described by the quadratic form (see (12)). Hence, we utilize the initial solution of $\boldsymbol{w}$ to roughly assess those scales. The initial solution $\boldsymbol{w}_0$ is obtained by applying Algorithm 1 to $\boldsymbol{C} = \boldsymbol{A} + \boldsymbol{L}$ under $\rho = 0$, and then the regularization parameters are transformed into $\hat{\rho} = \frac{\rho\{\Omega(\boldsymbol{w}_{X0}) + \Omega(\boldsymbol{w}_{Y0})\}}{\boldsymbol{w}_0^\top \boldsymbol{C}\boldsymbol{w}_0}$ and $\hat{\rho}$ is empirically determined, say $\hat{\rho} = 10$ in this study.

The above-mentioned technical tips are useful for facilitating the parameter setting which is an exhaustive procedure depending on the data.

IV. Experimental Results

We apply the proposed method to the pattern matching in static images (Fig. 1a) and motion sequences (Fig. 1b) as well as the classification of the motions.

Algorithm 1 : CG for $\min_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{C}\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w}}$

[Initialization] $\boldsymbol{w}_0$: random vector such that $\boldsymbol{w}_0^\top \boldsymbol{B}\boldsymbol{w}_0 = 1$,
$\lambda_0 = \boldsymbol{w}_0^\top \boldsymbol{C}\boldsymbol{w}_0$, $\boldsymbol{g}_0 = \boldsymbol{C}\boldsymbol{w}_0 - \lambda_0 \boldsymbol{B}\boldsymbol{w}_0$, $\boldsymbol{p}_0 = \boldsymbol{g}_0$, $i = 0$.
repeat
[step size δ]
$a = \boldsymbol{w}_i^\top \boldsymbol{C}\boldsymbol{p}_i$, $b = \boldsymbol{p}_i^\top \boldsymbol{C}\boldsymbol{p}_i$, $c = \boldsymbol{w}_i^\top \boldsymbol{B}\boldsymbol{p}_i$, $d = \boldsymbol{p}_i^\top \boldsymbol{B}\boldsymbol{p}_i$,
$\delta = \frac{\lambda_i d - b + \sqrt{(\lambda_i d - b)^2 - 4(bc - ad)(a - \lambda_i c)}}{2(bc - ad)}$.
[update]
$\hat{\boldsymbol{w}}_{i+1} = \boldsymbol{w}_i + \delta \boldsymbol{p}_i$, $\boldsymbol{w}_{i+1} = \frac{\hat{\boldsymbol{w}}_{i+1}}{\sqrt{\hat{\boldsymbol{w}}_{i+1}^\top \boldsymbol{B}\hat{\boldsymbol{w}}_{i+1}}}$.
$\lambda_{i+1} = \boldsymbol{w}_{i+1}^\top \boldsymbol{C}\boldsymbol{w}_{i+1}$, $\boldsymbol{g}_{i+1} = \boldsymbol{C}\boldsymbol{w}_{i+1} - \lambda_{i+1}\boldsymbol{B}\boldsymbol{w}_{i+1}$,
$\boldsymbol{p}_{i+1} = \boldsymbol{g}_{i+1} + \gamma \boldsymbol{p}_i$ where $\gamma = \frac{\boldsymbol{g}_{i+1}^\top \boldsymbol{g}_{i+1}}{\boldsymbol{g}_i^\top \boldsymbol{g}_i}$,
$i \leftarrow i + 1$.
until convergence
[Output] $\boldsymbol{w}_i$: the optimized weight vector,
λ_i: the minimum cost of $\frac{\boldsymbol{w}^\top \boldsymbol{C}\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w}}$.

Algorithm 2 : S^{3}CCA

[Initialization] $\boldsymbol{w}_0 = \arg\min_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top (\boldsymbol{A} + \boldsymbol{L})\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w}}$ by Algorithm 1,
$i = 0$.
repeat
[Optimize $\boldsymbol{z}$]
$z_X^{G_X} = \|\boldsymbol{d}^{G_X} \circ \boldsymbol{w}_{X_i}\|_2^{2-\alpha} [\Omega(\boldsymbol{w}_{X_i})]^{\alpha - 1}$ $(G_X \in \mathbb{G}_X)$,
$z_Y^{G_Y} = \|\boldsymbol{d}^{G_Y} \circ \boldsymbol{w}_{Y_i}\|_2^{2-\alpha} [\Omega(\boldsymbol{w}_{Y_i})]^{\alpha - 1}$ $(G_Y \in \mathbb{G}_Y)$.
[Optimize $\boldsymbol{w}$]
$\boldsymbol{\zeta}_X = \left\{ \sum_{G_X \in \mathbb{G}_X, j \in G_X} \frac{(d_j^{G_X})^2}{z_X^{G_X}} \right\}_{j=1}^m$,
$\boldsymbol{\zeta}_Y = \left\{ \sum_{G_Y \in \mathbb{G}_Y, j \in G_Y} \frac{(d_j^{G_Y})^2}{z_Y^{G_Y}} \right\}_{j=1}^n$,
$\boldsymbol{C} = \boldsymbol{A} + \boldsymbol{L} + \mathrm{diag}([\rho_X \boldsymbol{\zeta}_X^\top, \rho_Y \boldsymbol{\zeta}_Y^\top]^\top)$,
$\boldsymbol{w}_{i+1} = \arg\min_{\boldsymbol{w}} \frac{\boldsymbol{w}^\top \boldsymbol{C}\boldsymbol{w}}{\boldsymbol{w}^\top \boldsymbol{B}\boldsymbol{w}}$ by Algorithm 1.
$i \leftarrow i + 1$.
until convergence
[Output] $\boldsymbol{w}_i = [\boldsymbol{w}_X^\top, \boldsymbol{w}_Y^\top]^\top$: the optimized map weights.

A. Static Images

The task is to detect the partial patterns that are shared by the pair of input images. The images are defined in two dimensions (x-y) over which the map weights are defined, and thus the feature array is formulated as *feature-vs-position* by unfolding the two-dimensional position into one-dimensional sequence via the raster scan (Fig. 1a).

1) Synthetic Image: We synthesized the pair of image patterns of $15 \times 20 \times 500$ and $20 \times 30 \times 500$ pixels assuming that 500-dimensional features are extracted at each pixel in the images of 15×20 and 20×30 pixels; the patterns are shown in Fig. 4. Thus, we perform the partial matching between the feature arrays $\boldsymbol{X} \in \mathbb{R}^{500 \times 300}$ and $\boldsymbol{Y} \in \mathbb{R}^{500 \times 600}$. Fig. 5 shows the map weights optimized by various methods; conventional CCA ($\hat{\rho} = \hat{\eta} = 0$), smooth CCA (SCCA) ($\hat{\rho} = 0, \hat{\eta} = 1$), structured sparse CCA (S^{2}CCA) ($\hat{\rho} = 10, \hat{\eta} = 0$), and the proposed S^{3}CCA ($\hat{\rho} = 10, \hat{\eta} = 1$). We can see that S^{3}CCA produces the favorable map weights that are activated only in the region of the common patterns, while the other methods are highly affected by the noise; in particular, the weights by CCA are quite scattered, making no sense for

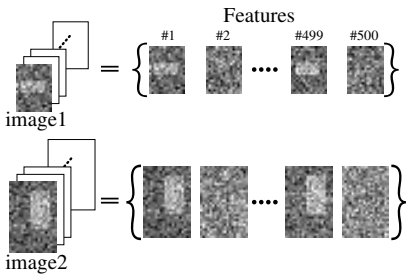


Fig. 4. Synthetic image data. The bright rectangles which are common patterns appear every two features with white noise. The feature matrix is constructed by unfolding each of image frame into a vector form.

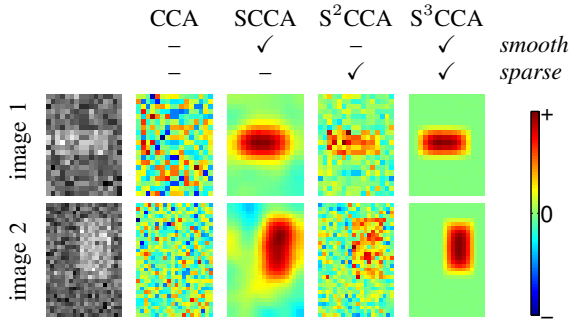


Fig. 5. Pattern matching on synthetic image data. The map weights $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ are depicted in pseudo colors. This figure is best viewed in color.

matching. By comparing S^{3}CCA to SCCA and S^{2}CCA, it is demonstrated that the combination of the smoothness and the structured sparseness works quite well for partial matching. As to computation time, in S^{3}CCA, the CG method (Algorithm 1) takes 0.4 sec, while the generalized eigenvalue solver requires 3.6 sec on Core-i7 2.9GHz PC.

2) Realistic Image: Next, we applied the proposed method to the partial matching between the realistic images (from UIUC car image dataset [18]) that contain *car* as common patterns. Those images are of 175×90 and 260×195 pixels. In this task, it should be noted that unlike the ordinary car detection, there is no prior knowledge; that is, we do not know what and where the target is in advance. The GLAC local image features [8] are extracted at dense grid points in 6 pixel step with the scale of 6 pixels, producing 364 and 1302 points in respective images; we follow the setting of the feature reported in [8] to obtain 324-dimensional GLAC feature vectors, the feature arrays of $\boldsymbol{X} \in \mathbb{R}^{324\times 364}$ and $\boldsymbol{Y} \in \mathbb{R}^{324\times 1302}$. The obtained map weights are shown in Fig. 6, demonstrating that S^{3}CCA can detect the *car* even though neither the target category (*car*) nor the detector is given. This is because the features belonging to the *car* regions are salient and shared by the feature arrays $\boldsymbol{X}$ and $\boldsymbol{Y}$.

B. Motion Sequences

We conducted two types of experiments using motion sequences in Weizmann action dataset [19]; one is the motion matching as in the image matching, and the other is the classification of motions based on the similarity computed via the matching. The Weizmann dataset [19] contains nine types of human action; *running*, *walking*, *jumping jacks*, *jumping forward*, *jumping in place*, *galloping sideways*, *waving two hands*, *waving one hand*, and *bending*, which are performed once by each of nine subjects. In these experiments, we extracted the frame-based motion features by CHLAC [20] from the motion sequence to form the feature matrix $\boldsymbol{X} \in \mathbb{R}^{251\times T}$ where T indicates the number of frames in the sequence.

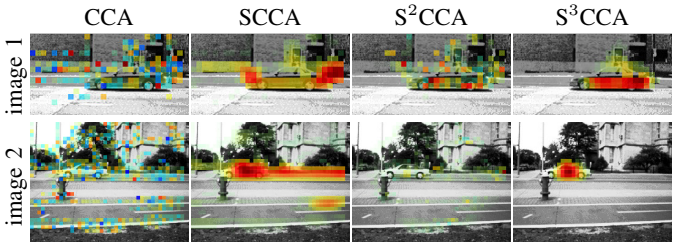


Fig. 6. Pattern matching on car image data. The map weights are depicted in pseudo colors and overlaid on the images. This figure is best viewed in color.

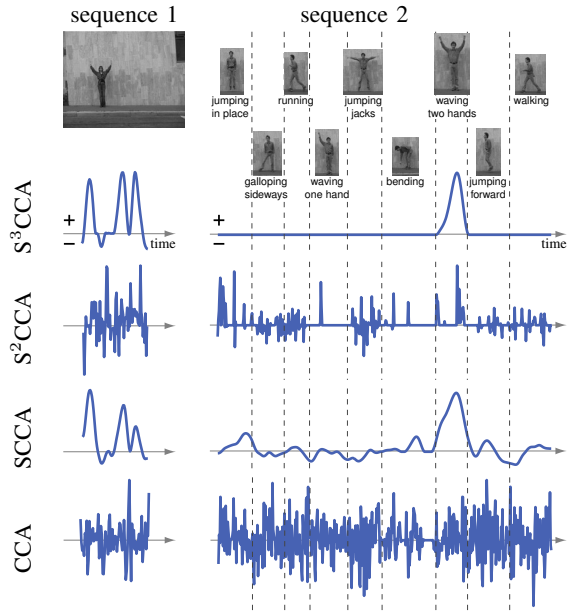


Fig. 7. Motion matching on Weizmann action dataset. The map weights $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ are shown as the time-series sequence.

1) Motion Matching: We concatenated all the motion sequences of the nine actions performed by the subject 2 and compared it to the sequence of the action of *waving two hands* by the subject 1. In this experiment, the map weights are obtained as the 1-dimensional time-series sequence and the results are shown in Fig. 7. The proposed S^{3}CCA produces the favorable weights that detect the common action of *waving two hands*. Note that the map weights are zeros on the irrelevant actions that are not shared in those two sequences.

2) Motion Classification: The proposed S^{3}CCA can be utilized for classification as follows. The two motion sequences are compared via the partial matching and their similarity is defined by the "canonical angle", $\theta = \cos^{-1}\{\boldsymbol{w}_X^\top \boldsymbol{K}_{XY} \boldsymbol{w}_Y\}$, where $\boldsymbol{w}_X$ and $\boldsymbol{w}_Y$ are the map weights produced by S^{3}CCA. Obviously, the sequences of the high affinity exhibit small angle $\theta \approx 0$. We classify the input sequence by k-NN, say $k = 3$, utilizing the canonical angles as the similarity measure.

For comparison, we also applied the methods of mutual subspace method (MSM) [21] and constrained mutual sub-

TABLE I. MOTION CLASSIFICATION PERFORMANCE ON WEIZMANN DATASET.

Method	acc. (%)
Dollar et al. [24]	86.7
Jhuang et al. [25]	98.8
Wang and Mori [23]	**100**
MSM [21]	80.47
CMSM [22]	73.51
S^3CCA	82.94
kernel MSM [21]	96.30
kernel CMSM [22]	97.53
kernel S^3CCA	**100**

space method (CMSM) [22], both of which employ the similar classification approach as described above; those methods are based on the canonical angles between the subspaces. In this experiment, we additionally applied the kernel-based S^3CCA that is developed by applying kernel tricks to the Gram matrices $\boldsymbol{K}_X, \boldsymbol{K}_Y$ and $\boldsymbol{K}_{XY}$ in (5, 6). We employed the Gaussian kernel $\mathsf{k}(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|^2}{2\sigma^2})$, where σ is the standard deviation computed in the dataset.

The performance is evaluated by leave-one-subject-out as in [23]; the sequences of eight subjects are used for training and those of the remaining subject are for test, which is repeated for all nine subjects. The averaged classification accuracy is measured and shown in Table I. The proposed method produces the perfect classification results as the other state-of-the-art work [23] does. Compared to MSM and CMSM using the same CHLAC features as ours, the proposed method can favorably improve the performance. The smoothly localized map weights exploit the essential motion patterns that are common in the pair of sequences, excluding the other irrelevant parts, to enhance the discriminativity of the similarities used in k-NN classification. It is demonstrated that the S^3CCA contributes to the classification as well as the partial matching.

V. CONCLUSION

We have proposed the method of smoothly structured sparse CCA (S^3CCA) for partial pattern matching. The proposed method works on the pair of feature arrays and produces the map weights which indicate the parts exhibiting the common pattern across those feature arrays. By introducing the appropriate regularizations, the map weights are optimized so as to be both smooth and localized, i.e., structured sparse, which effectively contribute to correct false negatives while suppressing the false positives. In the experiments on pattern matching for 2-dimensional static images and 1-dimensional motion sequences as well as the motion classification, the proposed method produces favorable performance compared to the other methods.

REFERENCES

[1] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[2] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell, "Distance metric learning, with application to clustering with side-information," in *Advances in Neural Information Processing Systems (NIPS)*, 2003, pp. 521–528.

[3] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research*, vol. 10, pp. 207–244, 2009.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Germany: Springer, 2007.

[5] Y. Chai, V. Lempitsky, and A. Zisserman, "Bicos: A bi-level co-segmentation method for image classification," in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2579–2586.

[6] A. Joulin, F. Bach, and J. Ponce, "Discriminative clustering for image co-segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 1–8.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 886–893.

[8] T. Kobayashi and N. Otsu, "Image feature extraction using gradient local auto-correlations," in *European Conference on Computer Vision (ECCV)*, 2008, pp. 346–358.

[9] C. Conrad and R. Mester, "Learning geometrical transforms between multi camera views using canonical correlation analysis," in *British Machine Vision Conference (BMVC)*, 2012, pp. 47.1–47.12.

[10] X. Chen, H. Liu, and J. G. Carbonell, "Structured sparse canonical correlation analysis," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012, pp. 199–207.

[11] T. Anderson, *An introduction to multivariate statistical analysis*. New York, NY, USA: Wiley, 1958.

[12] Å. Björck and G. H. Golub, "Numerical methods for computing angles between linear subspaces," *Mathematics of Computation*, vol. 27, no. 123, pp. 579–594, 1973.

[13] B. Scholkopf and A. Smola, *Learning with Kernels*. MIT Press, 2001.

[14] R. Jenatton, J.-Y. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," *Journal of Machine Learning Research*, vol. 12, pp. 2777–2824, 2011.

[15] R. Jenatton, G. Obozinski, and F. Bach, "Structured sparse principal component analysis," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009, pp. 366–373.

[16] Y. Feng and D. Owen, "Conjugate gradient methods for solving the smallest eigenpair of large symmetric eigenvalue problems," *International Journal For Numerical Methods in Engineering*, vol. 39, no. 13, pp. 2209–2229, 1996.

[17] J. Nocedal and S. Wright, *Numerical optimization*. New York, NY: Springer Verlag, 1999.

[18] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, 2004.

[19] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *International Conference on Computer Vision (ICCV)*, 2005, pp. 1395–1402.

[20] T. Kobayashi and N. Otsu, "A three-way auto-correlation based approach to motion recognition," *Pattern Recognition Letters*, vol. 30, no. 3, pp. 185–192, 2009.

[21] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition using temporal image sequence," in *International Conference on Automatic Face and Gesture Recognition (FG)*, 1998, pp. 318–323.

[22] K. Fukui and O. Yamaguchi, "Face recognition using multi-viewpoint patterns for robot vision," in *International Symposium of Robotics Research*, 2003, pp. 192–201.

[23] Y. Wang and G. Mori, "Human action recognition by semi-latent topic models," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1762–1774, 2009.

[24] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005, pp. 65–72.

[25] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action recognition," in *International Conference on Computer Vision (ICCV)*, 2007.