

Multi-kernel sparse subspace clustering on the Riemannian manifold of symmetric positive definite matrices[☆]

Sabra Hechmi*, Abir Gallas, Ezzeddine Zagrouba

University of Tunis EL Manar, Higher Institute of Computer Science of Tunis, LIMTIC Laboratory, Abou Raihane Bayrouni, Ariana 2080, Tunisia

ARTICLE INFO

Article history:

Received 22 March 2018

Available online 23 March 2019

Keywords:

Sparse subspace clustering (SSC)

Riemannian kernel

Multi-kernel SSC

Face clustering

ABSTRACT

Recently, clustering in the Riemannian manifolds has received a great interest. The main specificity of this kind of spaces is their ability to detect nonlinear forms in the real world data groups. Accordingly, numerous Euclidean clustering algorithms have been emerged to the Riemannian framework. One of them is the Sparse Subspace Clustering (SSC) method. Within this context, an interesting method named Kernel Sparse Subspace Clustering in the Riemannian manifold (KSSCR) has been proposed. In order to improve the clustering performance of the SSC method, KSSCR is based on the idea of data projection into a Reproducing Kernel Hilbert Space (RKHS) through a Riemannian kernel. This modification in terms of projection space has yielded to significant clustering results compared to the original SSC. This paper proposed a Multi-Kernel SSCR (MKSSCR) method. The idea is to create a linear combination of a set of Riemannian kernels. In fact, by using a single kernel, the faraway points will be neglected. Thus, the key motivation of this work is to use a mixture of kernels. The purpose here is to emphasize the closest data points without ignoring the faraway ones in the process of selecting neighbors for each data point. Several experiments carried out on varied face clustering data sets demonstrate the clustering accuracy improvement by the proposed method compared to other state-of-the-art methods.

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

Over the last few years, the Riemannian manifolds have received a particular interest in computer vision (Turaga and Srivastava, 2015). The main reason of switching from the conventional Euclidean geometry to the non-linear Riemannian geometry is that the real world data are often collected into special structures which Euclidean geometry fails to capture. Modeling data in such manifolds has lead to success in numerous computer vision fields such as human activity and interaction recognition [30], shape retrieval [22], visual object categorization and video based face recognition [12]. In the context of clustering, numerous attempts have been made to extend clustering methods into the Riemannian spaces. Such methods include the kernel K-means [13] and the non linear Mean shift [20] which have yielded better clustering results comparing to K-means [9] and the original Mean shift [5]. As a differential geometry, the Riemannian manifolds are endowed with special metrics. In fact, the similarity measure between two points in a Riemannian space is defined as the

length of the shortest curve connecting them and known as the geodesic distance [14]. Essentially two Riemannian manifolds can be cited. Both they have known significant attention in computer vision applications. The first is the Grassmann manifold [2] which presents the space of all m dimensional linear subspaces of $\mathbb{R}^d$. The second is the space of Symmetric Positive Definite (SPD) matrices [3]. In fact, for all nonzero element $x \in \mathbb{R}^d$, a real SPD matrix M , denoted as Sym_d^+ , is a matrix that verifies the property $x^T M x > 0$. This manifold forms a convex cone in the d^2 -dimensional Euclidean space. In practice, the SPD matrices are encountered in the forms of covariance region descriptors matrices modeled as points in this manifold. There are two principal similarity measures used on the SPD matrices space which are the affine-invariant distance and the Log-Euclidean distance [23]. Nevertheless, the calculation in these Riemannian topological spaces is extremely difficult and expensive, especially for massive amounts of data. To overcome this problem, the idea is to implicitly transform data in this manifold into a Reproducing Kernel Hilbert Space (RKHS) through a non linear map associated with a kernel function [12]. Indeed, a kernel in the geometry of Riemannian manifolds is obtained by replacing the Euclidean distance in the Gaussian kernel for example with the geodesic distance in the treated manifold. However, in almost all cases, the Riemannian kernels which are derived in this manner can not verify the Mercer condition (positive definite kernel).

[☆] Conflict of interest: None.

* Corresponding author.

E-mail addresses: hechmi_sabra@yahoo.fr (S. Hechmi), ezzeddine.zagrouba@fsm.rnu.tn (E. Zagrouba).

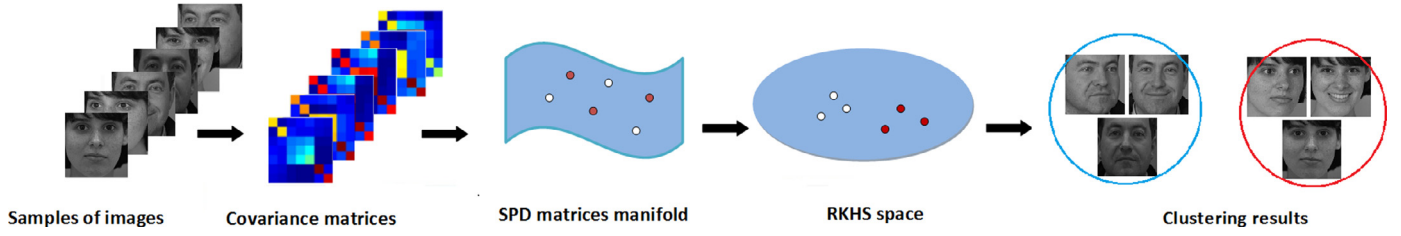


Fig. 1. Face clustering: Each face image is represented by a covariance matrix which is considered as a point in the SPD matrices manifold. By embedding these matrices into a RKHS space, an easy separation of the data points will be obtained.

Therefore, the kernel trick could be suitable to address the computational problem posed in non linear spaces. Moreover, the right choice of the kernel could significantly improve the clustering performance (Fig. 1).

On the other hand and within the context of high dimensional data, subspace clustering methods [25] have shown promising results. In fact, the goal of these methods is to uncover clusters of data according to their underlying subspaces. Explicitly, each method of them aims to collect a group of data by seeking for neighbors that are not close according to a predefined metric but rather to the fact of being in the same subspace or not. Based on the above-mentioned definition, an important number of algorithms have been proposed such as Generalized Principal Component analysis (GPCA) [26], Low Rank Representation (LRR) [15], Local Subspace Affinity (LSA) [28] and Sparse Subspace Clustering (SSC) [7]. More precisely, The SSC method is based on the idea that every data point in a union of subspaces can be written as a linear combination of a few other data points from its own local subspace. The resulting expression is known as the sparsest representation. This one is used in the next step to build a subspace clustering affinity matrix. Then, the segmentation of the data can be obtained by a spectral clustering technique. Because of its robust theoretical bases, the SSC method has succeeded to provide an exact neighborhood around each point. Besides, it ensured great clustering performance even for noisy data [7]. Motivated by its significant results, numerous attempts have been made to better improve the SSC method accuracy. For instance, Yin et al. [29] have proposed an extension of the SSC method on the SPD matrices space through a single Log-Gaussian kernel, chosen as the appropriate kernel function. This method, known as KSSCR, has shown interesting results especially for face clustering.

In this paper, the proposed approach is to define a multi-kernel SSC into the SPD matrices space. This is explained by the fact that the main goal of the SSC method is to search the exact local subspace for each data point. Nonetheless, by using this method, the nearest points could be only selected and we risk to neglect the faraway ones that can be in the same subspace as the query point. Indeed, the idea of integrating multiple kernels into classification algorithms, is not entirely new [8]. The principle of these attempts is to determine the appropriate number of kernels m forming a weighted linear combination which its coefficients are to be determined also. However all these initiatives remain in a theoretical and supervised framework. Therefore, the proposed idea is to use a mixture of two Log-Euclidean Gaussian kernels with two different sigma scales i.e. with high and low sigma values. In this case, the sparse optimization problem is reformulated by integrating a linear combination of two Log-Euclidean Gaussian kernels. Thus, the optimization of the new problem is aimed to find the optimal sparse coefficients as well as the linear coefficients associated to the two kernels. The rest of this paper is organized as follows: in Section 2 we present the classical SSC method and the Kernel Sparse Subspace Clustering on SPD manifold (KSSCR) and their basic theories. In Section 3, the proposed multi-kernel method is introduced. In Section 4, a deep experimental study is carried out on

several well-known data sets for face clustering to prove the efficiency of the proposed method. Finally, a conclusion and ideas for further work are presented in Section 5.

2. Related work

2.1. Sparse subspace clustering

The SSC is considered as one of the most successful subspace clustering methods [7]. The motivation here is to build a neighborhood around each data point based on the fact of belonging to the same subspace. Indeed, under some conditions, each data point lying in the union of subspaces can be expressed as a linear combination of few other data points from its own local subspace.

More specifically, let $Y = [y_1 y_2 \dots y_N]$ a noise-free data matrix that collects a set of points $\{y_i \in \mathbb{R}^D\}_{i=1}^N$ drawn from the union of n subspaces $S_1 \cup S_2 \cup \dots \cup S_n$ of dimensions $\{d_j\}_{j=1}^n$ in $\mathbb{R}^D$ such that, for each subspace S_j is associated a sub-matrix $Y_j \in \mathbb{R}^{D \times N_j}$ of rank d_j and a number of points N_j , such that $N_1 + N_2 + \dots + N_n = N$.

To find the sparse representation of data points, each data point has to be written with the minimum number of nonzero elements which corresponds efficiently to a d_j number of other data points from the data set Y . Therefore, the sparse optimization problem is written as:

$$y_i = Y c_i, \quad c_{ii} = 0, \quad \|c_i\|_0 \leq d_j \quad (1)$$

Where $c_i = [c_{i1} c_{i2} \dots c_{iN}]^T$ is the vector of linear coefficients associated to each y_i , $c_{ii} = 0$ eliminates the trivial solution of writing a data point as a linear combination of itself and $\|\cdot\|_0$ denotes the cardinality of nonzero elements of c_i . But since this problem is NP-hard, l_0 -norm is replaced by its l_1 -convex relaxation [6]. Thus, the optimizing problem becomes:

$$\min \|c_i\|_1 \quad \text{s.t.} \quad y_i = Y c_i \quad \text{and} \quad c_{ii} = 0 \quad (2)$$

This problem is expressed according to the matrix form as:

$$\min \|C\|_1 \quad \text{s.t.} \quad Y = YC \quad \text{and} \quad \text{diag}(C) = 0 \quad (3)$$

Where $C = [c_1 c_2 \dots c_N] \in \mathbb{R}^{N \times N}$ is the sparse coefficient matrix. To handle with the real world noisy data, the optimization problem is reformulated as follows:

$$\min \|C\|_1 + \frac{\lambda_z}{2} \|Y - YC\|_F^2 \quad \text{s.t.} \quad \text{diag}(C) = 0 \quad (4)$$

Where $\lambda_z > 0$ is a regularization term in the objective function, $\|\cdot\|_F$ designs the Frobenius norm selected as the appropriate norm to detect noise. After having the sparse coefficient matrix C , a similarity graph $W = |C| + |C|^T$ is constructed. Eventually, the data segmentation is obtained by applying a spectral clustering algorithm to the similarity graph W [17]. Compared to many subspace clustering methods, the SSC algorithm has shown interesting clustering results especially for face clustering and motion segmentation. Therefore, the SSC method has been widely studied in order to enhance its performance. One of many attempts is to exploit the Riemannian geometry. In this spirit, we define below the KSSCR method.

2.2. Kernel sparse subspace clustering on symmetric positive definite manifolds (KSSCR)

Recently, the Riemannian geometry has emerged as a powerful tool for many computer vision applications [23]. The main advantage of this geometry is the ability to support nonlinear forms contrary to the traditional Euclidean geometry. Some Riemannian manifolds have been widely used for data representation such as the manifold of symmetric positive definite SPD matrices which are usually encountered in the forms of covariance matrices [3]. This data representation descriptor is defined as follows: Given an image I of N pixels and let Φ defines a function that extracts a n -dimensional feature vector z_i from each pixel i in the image I such that:

$$\Phi(I, x_i, y_i) = z_i \quad (5)$$

where $z_i \in \mathbb{R}^n$ and (x_i, y_i) is the spatial coordinates of the i th pixel. The given image is represented by the $n \times n$ covariance matrix CM of the feature vectors $\{z_i\}_{i=1}^{|N|}$ of all pixels of the image. The feature vector z_i can be represented in several forms such as the pixel coordinates and the image gray level or color, edge magnitude and orientation, filter responses, etc [18,24].

Finally, the covariance matrix descriptor is given by:

$$CM = \frac{1}{N-1} \sum_{i=1}^N (z_i - \mu)(z_i - \mu)^T \quad (6)$$

where $\mu = \frac{1}{N} \sum_{i=1}^N z_i$ is the mean vector of the image I .

The covariance matrix is known as an efficient feature for image description. For this reason, it has been used in many computer vision applications including texture classification [21], object categorization and face recognition [23]. In fact, covariance matrices are considered as points on SPD manifolds. The geodesic distance between them is calculated essentially through two metrics which are the affine invariant and the Log-Euclidean distances. For two points C_1 and C_2 in the SPD space, the affine invariant metric is defined as:

$$dist_1(C_1, C_2) = \|\log(C_1^{-1/2} C_2 C_1^{-1/2})\|_F \quad (7)$$

and the Log-Euclidean metric is defined as:

$$dist_2(C_1, C_2) = \|\log(C_1) - \log(C_2)\|_F \quad (8)$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

To further improve clustering performance, kernels have been used on the Riemannian manifolds. In fact, SPD manifolds could be projected into RKHS via Mercer kernels. However, the resulting kernels are not necessarily positive definite for all metrics. For instance, the affine invariant metric does not yield a positive definite kernel. On the other hand, the Log-Euclidean kernels such as the Log-Euclidean Gaussian kernel and the Log-Euclidean polynomial kernel are valid Mercer kernels. For example, the Log-Euclidean Gaussian kernel is written as follows:

$$K(C_1, C_2) = \exp\{-\gamma \|\log(C_1) - \log(C_2)\|_F^2\} \quad (9)$$

Motivated by the above properties of the Riemannian geometry, numerous attempts have been made to generalize algorithms from Euclidean spaces to Riemannian manifolds in the goal to enhance their performance. One of them is the SSC method. Indeed, Yin et al. [29] have proposed a kernel sparse subspace clustering by embedding the SPD matrices into a RKHS space through the Log-Euclidean Gaussian kernel. Let Φ be a mapping function from the SPD matrices spaces to the reproducing kernel Hilbert space RKHS. The kernel Gram matrix K is defined as:

$$[K]_{i,j} = [\Phi(Y), \Phi(Y)]_{i,j} \text{ and } \Phi(y_i)^T \Phi(y_j) = k(y_i, y_j) \quad (10)$$

where k is the Log-Euclidean Gaussian kernel chosen as the appropriate kernel. Thus, by applying the kernel trick to the main sparse optimization problem, we obtain:

$$\min_C \|C\|_1 + \lambda \|\Phi(Y) - \Phi(Y)C\|_F^2 \text{ s.t. } \text{diag}(C) = 0 \quad (11)$$

Expanding the Frobenius norm, the optimization problem becomes:

$$\min_C \|C\|_1 - 2\text{tr}(KC) + \text{tr}(CKC^T) \text{ s.t. } \text{diag}(C) = 0 \quad (12)$$

Finally, to solve this problem via alternating direction method of multipliers (ADMM) [4], an auxiliary matrix A is introduced and the problem is written as follows:

$$\min_{C,A} \|C\|_1 - 2\text{tr}(KA) + \text{tr}(AKA^T) \text{ s.t. } A = C - \text{diag}(C) \quad (13)$$

As the SSC method, and after having the sparse coefficients, an affinity graph $W = |C| + |C|^T$ is built and a spectral clustering technique is applied to get the final data segmentation. By exploiting the interesting properties of the Riemannian geometry as well as the kernel trick, the KSSCR method has shown very significant results. This was proven specifically when tested on well-known data sets of texture and face clustering.

3. Contribution

We discuss in this paper the possibility to use a mixture of two Log-Euclidean Gaussian kernels as a linear combination of kernels. Bearing in mind that the goal of the optimization problem in the KSSCR method is to select, for each data point, a set of neighbors that belong to its own local subspace. However, by using a single kernel, the selection of the nearest points can be emphasized and the other ones would be neglected. To overcome this problem, the proposed method is to construct a sparse optimization problem where a linear combination of two kernels is integrated. In this case, the problem is to find the optimal sparse coefficients as well as the linear coefficients associated to the two kernels. The main condition to respect here, is the ability to provide the best balance between the selection of near and far data points.

Let suppose that we have two Log-Euclidean Gaussian kernels with high and low sigma values. Accordingly, there are two different RKHS spaces H_1 and H_2 associated to each kernel. To ensure that the combination of these kernels stills satisfying the Mercer condition, we consider an augmented Hilbert space $\tilde{H} = H_1 \oplus H_2$ which is constructed by the concatenation of two feature maps such that $\tilde{\phi}(x) = [\sqrt{w_1}\phi_1(x); \sqrt{w_2}\phi_2(x)]^T$ with two different weights $w_1, w_2 \geq 0$. Obviously, the clustering in the new space $\tilde{H}$ is equivalent to employing a combination of kernels $\tilde{K}(x, y)$ that gives rise also to a symmetric and positive semi-definite kernel matrix [31]. This kernel matrix is defined as:

$$\tilde{K}(x, y) = w_1 k_1(x, y) + w_2 k_2(x, y) \text{ with } w_1 + w_2 = 1, w_1, w_2 \geq 0. \quad (14)$$

In our case, the two kernels k_1 and k_2 defined above are two Log-Euclidean Gaussian kernels as:

$$k_1(x, y) = \exp\{-\gamma_1 \|\log(x) - \log(y)\|_F^2\}$$

$$k_2(x, y) = \exp\{-\gamma_2 \|\log(x) - \log(y)\|_F^2\}$$

After rewriting the sparse optimization problem, the obtained equation is the following:

$$\min_{\lambda} \lambda \|C\|_1 + \|\tilde{\phi}(Y) - \tilde{\phi}(Y)C\|_F^2 \text{ s.t. } \text{diag}(C) = 0 \quad (15)$$

Expanding the Frobenius norm and using the kernel trick, the objective function is:

$$\min_{\lambda} \lambda \|C\|_1 + \text{tr}(\tilde{K} - 2\tilde{K}C + C^T \tilde{K}C) \quad (16)$$

Where $tr(\cdot)$ is the trace operator of a given matrix. Integrating an auxiliary matrix A and replacing $\tilde{K}$ by $(w_1 k_1 + w_2 k_2)$, the optimization problem becomes as follows:

$$\begin{aligned} \min \lambda \quad & \|C\|_1 + tr(w_1 k_1 - 2w_1 k_1 A + A^T w_1 k_1 A) \\ & + tr(w_2 k_2 - 2w_2 k_2 A + A^T w_2 k_2 A) + \frac{\rho}{2} \|A - C + diag(C)\|_F^2 \\ & + \frac{\rho}{2} (w_1 + w_2 - 1)^2 \quad s.t. \quad A = C - diag(C), \\ & w_1 + w_2 = 1 \quad and \quad w_1, w_2 \geq 0. \end{aligned} \quad (17)$$

Thus, by using the ADMM method and adding the Lagrangian multipliers, the optimization steps of the above problem are described as:

1. Updating step for A : Let $\tilde{C} = C - diag(C)$. Thereby, a sub-problem is resulted as follows:

$$\begin{aligned} L(A) = & tr(w_1 k_1 - 2w_1 k_1 A + A^T w_1 k_1 A) + tr(w_2 k_2 - 2w_2 k_2 A \\ & + A^T w_2 k_2 A) + \frac{\rho}{2} tr[(A - \tilde{C})^T (A - \tilde{C})] + tr(\Delta^T (A - \tilde{C})) \end{aligned} \quad (18)$$

After setting the derivative of $L(A)$ to zero, we have:

$$A_{t+1} = (2w_1 k_1 + 2w_2 k_2 - \Delta_t + \rho \tilde{C}_t)(2w_1 k_1 + 2w_2 k_2 + \rho I)^{-1} \quad (19)$$

2. Updating step for C :

$$\begin{aligned} L(C) = & \min \lambda \quad \|C\|_1 + \frac{\rho}{2} tr[(A - C + diag(C))^T \\ & \times (A - C + diag(C))] + tr(\Delta^T (A - C + diag(C))) \end{aligned} \quad (20)$$

By fixing all other variables on zero, the solution of this sub-problem is given as:

$$C_{t+1} = J - diag(J), \quad J = \Gamma_{\frac{\lambda}{\rho}} \left(A_{t+1} + \frac{\Delta_t}{\rho} \right) \quad (21)$$

where $\Gamma_v(\cdot)$ is the shrinkage thresholding operator acting on each element of the given matrix, and is defined as $\Gamma_v(x) = (|x| + v)_+ \text{sgn}(x)$, where the operator $(\cdot)_+$ returns its argument if it is non-negative and returns zero otherwise.

3. Updating step for w_1 and w_2 : The optimization sub-problem is written as:

$$\begin{aligned} \min & tr(w_1 k_1 - 2w_1 k_1 A + A^T w_1 k_1 A) \\ & + tr(w_2 k_2 - 2w_2 k_2 A + A^T w_2 k_2 A) \\ & + \frac{\rho}{2} (w_1 + w_2 - 1)^2 \quad s.t. \quad w_1 + w_2 = 1 \quad and \quad w_1, w_2 \geq 0. \end{aligned} \quad (22)$$

Which the associated Lagrangian is:

$$\begin{aligned} L(w_1, w_2) = & tr(w_1 k_1 - 2w_1 k_1 A + A^T w_1 k_1 A) \\ & + tr(w_2 k_2 - 2w_2 k_2 A + A^T w_2 k_2 A) \\ & + \frac{\rho}{2} (w_1 + w_2 - 1)^2 + \delta (w_1 + w_2 - 1) \end{aligned} \quad (23)$$

After setting the derivative of $L(w_1, w_2)$ to zero with respect to each variable, we get:

$$w_{1(t+1)} = \frac{1}{\rho} (2k_1 A_t^T - k_1 - A_t k_1 A_t^T - \delta_t) - w_{2(t)} + 1 \quad (24)$$

$$w_{2(t+1)} = \frac{1}{\rho} (2k_2 A_t^T - k_2 - A_t k_2 A_t^T - \delta_t) - w_{1(t)} + 1 \quad (25)$$

4. Updating step for Δ and δ : By fixing A and C , the gradient ascent with step size of ρ is performed to update Δ and δ as:

$$\Delta_{t+1} = \Delta_t + \rho (A_{t+1} - C_{t+1} + diag(C_{t+1})) \quad (26)$$

$$\delta_{t+1} = \delta_t + \rho (w_{1(t)} + w_{2(t)} - 1) \quad (27)$$

These steps are repeated until $\|A_t - C_t\|_\infty \leq \epsilon$, $\|A_{t+1} - A_t\|_\infty \leq \epsilon$, $\|w_{1(t+1)} - w_{1(t)}\|_\infty \leq \epsilon$ and $\|w_{2(t+1)} - w_{2(t)}\|_\infty \leq \epsilon$. In Fig. 2, the main steps of the proposed method named Multi-Kernel Sparse Subspace Clustering on the Riemannian manifold or MKSSCR are illustrated.

4. Experimental results

In this section, an experimental study is carried out to verify the effectiveness of the proposed MKSSCR method compared to other state of the art subspace clustering methods essentially SSC and KSSCR. In order to do this, two data sets, which are commonly used in testing face clustering and recognition algorithms, are adopted. The first one is 'Feret' database [19] and the second is the YouTube Celebrities 'YTC' videos data set [10]. In order to quantify precisely the subspace clustering performance, two different criteria are adopted in this work:

1. The Rand Index (RI) [1] which is a popular cluster validation index defined as:

$$RI = \frac{TP + TN}{TP + FP + FN + TN} \quad (28)$$

Where TP is the true positive, TN shows the true negative, FP is the false positive and FN presents the false negative. The Rand Index value lies between [0, 1]. The RI value is equal to 1 when two solutions perfectly match.

2. The Normalized Mutual Information (NMI) [1] which is generally used to evaluate the performance of spectral clustering methods. Let T be the set of all true classes and V the set of all the clusters. We suppose that $p(T_i)$ and $p(V_j)$ are respectively the joint probability mass function of each class of T and each cluster of V . Thus, the NMI measure is defined as:

$$NMI(U, T) = \frac{I(T_i, V_j)}{[H(V_j) + H(T_i)]/2} \quad (29)$$

Where

$$H(V_j) = - \sum_j p(V_j) \log p(V_j) \quad (30)$$

$$I(T_i, V_j) = \sum_i \sum_j p(T_i \cap V_j) \log \frac{p(T_i \cap V_j)}{p(T_i)p(V_j)} \quad (31)$$

Here $I(T_i, V_j)$ is the mutual information between the true assigned class and the resulted cluster label. $H(V_j)$ denotes the entropy of cluster V_j when the information about T_i is available. Defined in the same way as $H(V_j)$, $H(T_i)$ denotes the entropy of cluster T_i when the information about V_j is known.

4.1. Experiments on Feret database

As a first application context, the performance of the proposed method MKSSCR compared to other state of the art methods, on the 'b' subset of 'Feret' database, is evaluated. This data set contains 1400 facial images of 200 persons with the size 80×80 pixels for each image taken under different illumination, pose and expression conditions. Precisely, each person has 7 frontal images noted as: 'ba', 'bd', 'be', 'bf', 'bg', 'bj' and 'bk' (Fig. 3).

In our experiments, each face image is represented as a covariance matrix of size 43×43 [18] using the feature vector: $[x, y, I(x, y), |G_{00}(x, y)|, \dots, |G_{07}(x, y)|, \dots, |G_{47}(x, y)|]^T$, where (x, y) is the pixel position, $I(x, y)$ is the intensity value of the current pixel and G_{uv} is the response of a 2D Gabor filter constructed by eight orientations ($u \in \{0, \dots, 7\}$) and five scales ($v \in \{0, \dots, 4\}$). Thus, for each pixel (x, y) there are 43 features composed by 40-dimensionality Gabor features, 2 spatial coordinates and an intensity value. In this test, 100 subjects of this data

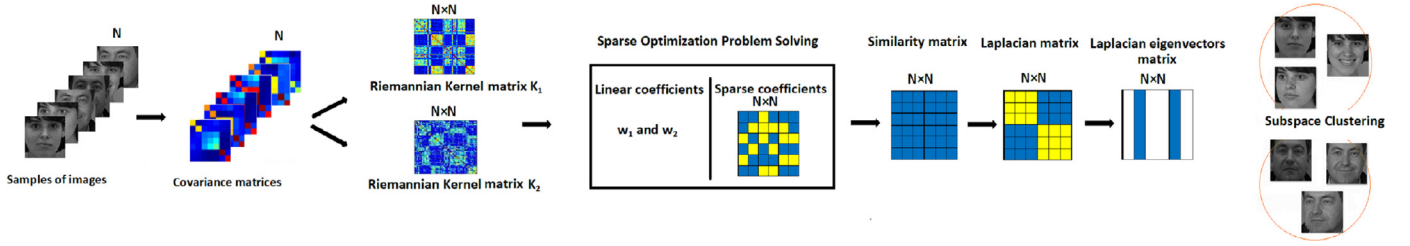


Fig. 2. Flowchart presenting the main steps of the proposed method MKSSCR.



Fig. 3. Facial images for one person from the Feret database.

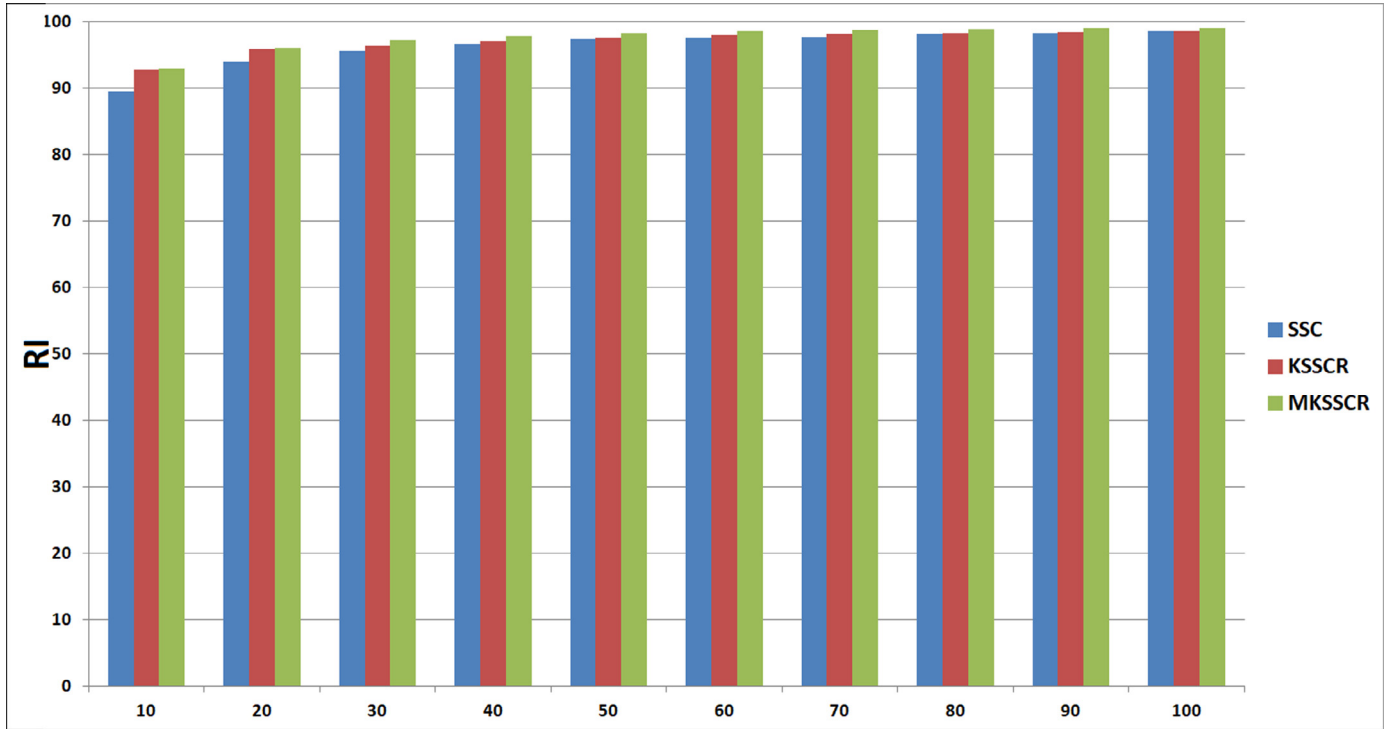


Fig. 4. RI score results on Feret data set.

set are chosen randomly. Thus, to see the effect of the number of subjects, all choices of $K \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$ are tested. In addition, for each number of subjects K , we choose to make 10 iterations such that each iteration represents a different combination of subjects except for $K = 100$ since we have a single combination. We apply all the subspace clustering methods for all K subjects with $\lambda_z = \frac{60}{\mu_z}$. For the KSSCR method γ is fixed to 2.10^{-3} and for the MKSSCR the values of γ_1 and γ_2 are fixed respectively to 1.10^{-3} and 2.10^{-3} . It is noted that these values of γ_1 and γ_2 are the ones giving the best clustering results.

In Fig. 4, the Rand Index (RI) Values are presented to prove the performance of the MKSSCR compared to SSC and KSSCR methods. The results recorded by MKSSCR are 92.81, 96.02, 97.16, 97.74, 98.15, 98.45, 98.61, 98.76, 98.92 and 98.98 for the number of sub-

jects 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 respectively. However, we have 92.75, 95.90, 96.30, 97.02, 97.42, 97.85, 98.04, 98.23, 98.40 and 98.53 for the KSSCR and we have 89.44, 93.83, 95.48, 96.63, 97.34, 97.46, 97.69, 98.01, 98.25 and 98.44 for the classic SSC. Thus, we have an average of RI of values 97.56, 97.04 and 96.25 for respectively, MKSSCR, KSSCR and SSC. Thereby, as it can be noticed, the proposed MKSSCR achieves high Rand Index rates.

Moreover, in Fig. 5, the NMI results are presented. We have recorded 75.23, 78.83, 79.74, 79.73, 80.55, 81.42, 81.16, 81.42, 82.17 and 81.54 for all the different choices of subject number (10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 respectively) for MSSCR. Nevertheless, the NMI scores of the KSSCR method are: 74.97, 78.13, 72.65, 72.62, 72.02, 72.81, 71.58, 71.36, 71.49 and 72.11 and we have 65.8, 66.64, 67.83, 71.69, 73.12, 71.12, 70.5, 71.44, 72.27, 72.52 for

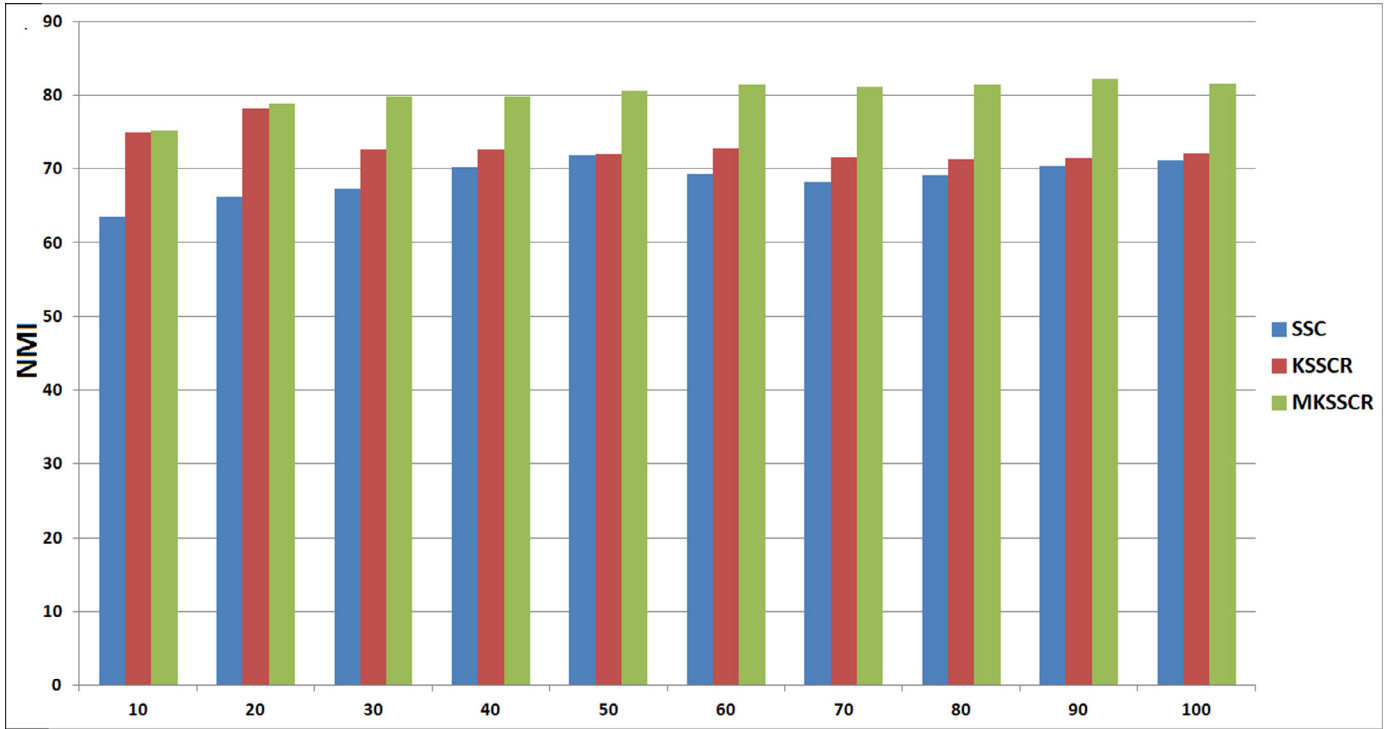


Fig. 5. NMI score results on Feret data set.



Fig. 6. Facial image frames from the YTC data set.

the classic SSC. Thus, we have an average of NMI values as: 80.17, 72.97 and 68.70 for respectively MKSSC, KSSCR and SSC. Finally, we can conclude that the proposed MKSSC method gives better results than those given by both KSSCR and SSC methods in terms of clustering quality. These results are explained by the impact of using a mixture of two kernels for the MKSSCR method instead of one kernel in the case of the KSSCR. In fact, this modification has succeeded in selecting different sets of neighbors that are supposed to be in the same final cluster of the query points and thus it improves the final clustering accuracy.

4.2. Experiments on YouTube Celebrities database

In this part, we choose to conduct our experiments on YouTube Celebrities (YTC) database. This data set compounds videos of celebrities making real life activities in various environments collected from YouTube. The YTC data set contains 1910 video sequences of 47 subjects (Fig. 6). This data set is divided into two subsets: Gallery and Probe. Each one has, for every person, 3 image sets and 6 image sets respectively. Furthermore, the face region in each image is resized into 20×20 intensity image according

Table 1

Clustering results on the YTC data set.

Algorithm	Gallery		Probe	
	RI	NMI	RI	NMI
KSSCR	96.67	71.56	96.49	61.23
MKSSCR	97.12	73.41	96.83	67.12

to the works [27] and [16]. For the set-based covariance matrix extraction, we follow the work of [11]. Finally, the process of random testing evaluation was repeated 10 times for each subset. We apply the KSSCR method and the proposed MKSSCR method for the number of subjects $K = 47$ with $\lambda_z = \frac{80}{\mu_z}$ for Gallery subset and $\lambda_z = \frac{20}{\mu_z}$ for Probe subset. For the KSSCR method γ is fixed to 2.10^{-4} and for the MKSSCR the values of γ_1 and γ_2 are fixed respectively to 2.10^{-4} and 3.10^{-4} .

Table 1 presents the clustering results in terms of Rand Index (RI) and Normalized Mutual Information (NMI). For the Gallery subset, RI values are respectively 96.67 and 97.12 for KSSCR and MKSSCR. For the same subset, NMI values are 71.56 and 73.41

respectively for KSSCR and MKSSCR. On the other hand, for the Probe subset, we have as RI values: 96.49 and 96.83 respectively for KSSCR and MKSSCR and NMI values: 61.23 and 67.12. These results show again the effectiveness of our proposed method (MKSSCR) as well as the effect of the use of mixture of kernels instead of one kernel in the process of neighbors selection.

5. Conclusion

In this paper, we have proposed a novel sparse subspace clustering method named MKSSCR. The main motivation of our method is using the kernels properties in a Riemannian geometry context in order to improve the clustering results. In fact, the idea is to use a mixture of two kernels in the process of neighbors selection for each data point. The experimental results on the ‘Feret’ database and ‘YTC’ video-based data set for face clustering, have revealed consistently the performance of the proposed MKSSCR compared to the traditional SSC and the KSSCR. As future work, we still focus on the process of neighbors selecting for each query point, as well as the minimization of the computational complexity in the sparse optimization problem. In order to achieve this goal, we will adopt the concept of coresets. The idea, in this case, is to choose particular points instead of the whole data set as an input of the sparse optimization problem.

References

- [1] A.K. Alok, S. Saha, A. Ekbal, Development of an external cluster validity index using probabilistic approach and min-max distance, *IJCISIM* 6 (2014) 494–504.
- [2] S. Anand, S. Mittal, P. Meer, Robust estimation for computer vision using grassmann manifolds, in: *Riemannian Computing in Computer Vision*, Springer, 2016, pp. 125–144.
- [3] V. Arsigny, P. Fillard, X. Pennec, N. Ayache, Geometric means in a novel vector space structure on symmetric positive-definite matrices, *SIAM J. Matrix Anal. Appl.* 29 (2007) 328–347.
- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Found. Trends Mach. Learn.* 3 (2010) 1–122.
- [5] K.G. Derpanis, Mean shift clustering, *Lecture Notes*, 2005.
- [6] D. Donoho, For most large underdetermined systems of equations, the minimal l_1 -norm near-solution approximates the sparsest near-solution.
- [7] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (2013) 2765–2781.
- [8] M. Gönen, E. Alpaydin, Multiple kernel learning algorithms, *J. Mach. Learn. Res.* 12 (Jul) (2011) 2211–2268.
- [9] J.A. Hartigan, M.A. Wong, Algorithm as 136: a k-means clustering algorithm, *J. R. Stat. Soc. Ser. C (Appl. Stat.)* 28 (1979) 100–108.
- [10] Z. Huang, R. Wang, S. Shan, X. Chen, Face recognition on largescale video in the wild with hybrid Euclidean-and-Riemannian metric learning, *Pattern Recognit.* 48 (2015) 3113–3124.
- [11] Z. Huang, R. Wang, S. Shan, X. Li, X. Chen, Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification, in: *International Conference on Machine Learning*, 2015, pp. 720–729.
- [12] S. Jayasumana, R. Hartley, M. Salzmann, Kernels on Riemannian manifolds, in: *Riemannian Computing in Computer Vision*, Springer, 2016, pp. 45–67.
- [13] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, M. Harandi, Kernel methods on the Riemannian manifold of symmetric positive definite matrices, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 73–80.
- [14] J.M. Lee, *Riemannian Manifolds: an Introduction to Curvature* 176, Springer Science and Business Media, 2006.
- [15] G. Liu, Z. Lin, Y. Yu, Robust subspace segmentation by low-rank representation, in: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 663–670.
- [16] J. Lu, G. Wang, P. Moulin, Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning, in: *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 329–336.
- [17] A.Y. Ng, M.I. Jordan, R.E. Weiss, On spectral clustering: analysis and an algorithm, in: *NIPS*, 2001, pp. 849–856.
- [18] Y. Pang, Y. Yuan, X. Li, Gabor-based region covariance matrices for face recognition, *IEEE Trans. Circuits Syst. Video Technol.* 18 (2008) 989–993.
- [19] P.J. Phillips, H. Wechsler, J. Huang, P.J. Rauss, The feret database and evaluation procedure for face-recognition algorithms, *Image Vis. Comput.* 16 (1998) 295–306.
- [20] R. Subbarao, P. Meer, Nonlinear mean shift over Riemannian manifolds, *Int. J. Comput. Vis.* 84 (2009) 1–20.
- [21] J.Y. Tou, Y.H. Tay, P.Y. Lau, Gabor filters as feature images for covariance matrix on texture classification problem, in: *International Conference on Neural Information Processing*, Springer, 2008, pp. 745–751.
- [22] P. Turaga, A. Veeraraghavan, R. Chellappa, Statistical analysis on Stiefel and Grassmann manifolds with applications in computer vision, in: *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, CVPR 2008, 2008, pp. 1–8.
- [23] P.K. Turaga, A. Srivastava, *Riemannian Computing in Computer Vision*, Springer, 2015.
- [24] O. Tuzel, F. Porikli, P. Meer, Pedestrian detection via classification on Riemannian manifolds, *IEEE Trans. Pattern Anal. Mach. Intell.* 30 (2008) 1713–1727.
- [25] R. Vidal, Subspace clustering, *IEEE Signal Process. Mag.* 28 (2011) 52–68.
- [26] R. Vidal, Y. Ma, S. Sastry, Generalized principal component analysis (gpca), *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (2005) 1945–1959.
- [27] R. Wang, H. Guo, L.S. Davis, Q. Dai, Covariance discriminative learning: a natural and efficient approach to image set classification, in: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, 2012, pp. 2496–2503.
- [28] J. Yan, M. Pollefeys, A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate, in: *European conference on computer vision*, Springer, 2006, pp. 94–106.
- [29] M. Yin, Y. Guo, J. Gao, Z. He, S. Xie, Kernel sparse subspace clustering on symmetric positive definite manifolds, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5157–5164.
- [30] Y. Yun, K. Fu, I.Y.H. Gu, H. Aghajan, J. Yang, Human activity recognition in images using svms and geodesics on smooth manifolds, in: *Proceedings of the International Conference on Distributed Smart Cameras*, ACM, 2014, p. 20.
- [31] H. Zeng, Y.M. Cheung, Feature selection and kernel learning for local learning-based clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (2011) 1532–1547.