CO-REFERENCED SUBSPACE CLUSTERING

Xiaobo Wang[1,2], *Zhen Lei*[*,1,2], *Hailin Shi*[1,2], *Xiaojie Guo*[3], *Xiangyu Zhu*[1,2], *Stan Z. Li*[1,2]

[1]CBSR&NLPR, Institute of Automation Chinese Academy of Sciences, Beijing, China, 100190
[2]University of Chinese Academy of Sciences, Beijing, China, 100049
[3]School of Computer Software, Tianjin University, Tianjin, China, 300350
{xiaobo.wang, zlei, hailin.shi, xiangyu.zhu, szli}@nlpr.ia.ac.cn, xj.max.guo@gmail.com

ABSTRACT

Subspace clustering refers to the problem of grouping data into their underlying groups. To address this task, spectral clustering based technique is arguably one of the most popular approaches, and its performance largely depends on the constructed similarity. However, most existing works merely employ the primary representation (*e.g.*, sparse or low-rank representation) as the similarity. In this paper, we propose to explore a high-level co-referenced similarity by employing the Hilbert-Schmidt Independence Criterion (HSIC). Moreover, geometry interpretation of the advantage of our co-referenced similarity is provided. Representation-induced kernels such as Mahalanobis metric, can also be easily embedded into the formulation. Extensive experiments on both synthetic and real-world data are conducted to show the superiority of the proposed method over the state-of-the-art alternatives.

Index Terms— Subspace clustering, similarity

1. INTRODUCTION

In real-world data analysis tasks, the data are usually of some certain structures. To characterize the given data according to the structure as different groups such that the data in the same group are highly similar to each other, the subspace is the most commonly used container. Subspace clustering technique has shown its significance as a theoretic foundation in many computer vision and machine learning tasks, such as face recognition [1], image representation and compression [2], motion segmentation [3] and saliency detection [4].

Over the past years, methods based on spectral clustering [5, 6, 7, 8, 9, 10, 11] have become dominant, whose framework can be summarized into two pipelined steps, *i.e.*, subspace learning and spectral clustering. Arguably, the first step (*i.e.*, learning representation) is of the most importance, as the success of the spectral clustering algorithms heavily depend on constructing an informative similarity. A majority of schemes on the market devote to learn a "good" representation. Concretely, Sparse Subspace Clustering (SSC) [5] seeks a sparse representation for each data. Low-Rank Representation (LRR) [6] is to find a global low-rank one. To simultaneously consider the global grouping effect and local sparsity of the representation, group sparse coding [8] and multi-subspace clustering [7] have been developed. While Least Squares Regression (LSR) [9] offers a much more efficient technique for computing the representation than LRR with a similar grouping effect. Moreover, some kernelized works [12, 13, 14, 15], which aim to find a good kernel on the original data $\mathbf{X}$ space to guide the representation learning, have also been developed in last decade. After obtaining the representation $\mathbf{Z}$, the above methods directly form the similarity as $\mathbf{S} = (|\mathbf{Z}| + |\mathbf{Z}^T|)/2$, then utilize a spectral clustering algorithm, *e.g.* Normalized Cuts [16], to partition the data into the underlying groups. However, sequentially dealing with each step may suffer from suboptimal performance due to the fact that these two pipelined steps highly depend on each other.

Recently, several works [17, 18, 11, 19] try to jointly optimize these two steps. Feng *et al.* [17] introduce a block diagonal constraint to the self-expressiveness model. Nie *et al.* [18] develop a more sophisticated method namely CAN to learn the similarity matrix by adaptively assigning neighbors to each data point based on the locality, and integrate the similarity learning into the following spectral clustering. Li *et al.* [11, 19] propose a Structured Sparse Subspace Clustering (S3C) model, which unifies subspace learning and spectral clustering by using a structured sparse norm.

Although the above approaches have achieved reasonable results, they share a common shortcoming, *i.e.*, only exploiting the primary representation as similarity. In this paper, we propose a novel method, namely CRSC/KCRSC, by exploring a high-level *co-referenced similarity* to boost the performance. The main contributions can be summarized as:

- We exploit a high-level co-referenced similarity based on the primary representation by adopting the Hilbert-Schmidt Independence Criterion (HSIC).

This work was supported by the National Key Research and Development Plan (Grant No.2016YFC0801002), the Chinese National Natural Science Foundation Projects #61473291, #61572501, #61502491, #61572536, and AuthenMetric R&D Funds.

* Corresponding author.

978-1-5386-1737-3/18/$31.00

- Geometry interpretation and representation-induced kernelized version of our high-level co-referenced similarity are provided.
- Extensive experiments on both synthetic data and real-world data are conducted to demonstrate the efficacy and the superior performance of the proposed algorithm over the state-of-the-art alternatives.

Notation: For a matrix $\mathbf{U}$, the i-th row and the j-th column of $\mathbf{U}$ are denoted by $\mathbf{U}^i$ and $\mathbf{U}_j$, respectively. $\mathbf{1}$ is the all-ones matrix with appropriate sizes.

2. PROBLEM FORMULATION

2.1. Structured Sparse Subspace Clustering (S3C [11])

As aforementioned, spectral clustering based methods involve two key factors, *i.e.*, subspace learning and spectral clustering. Given a set of data points $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_n] \in \mathbb{R}^{d\times n}$, where d is the feature dimension and n is the total amount of data points. Each sample $\mathbf{X}_i \in \mathbb{R}^d$ can be approximated by a linear combination of the reference samples $\mathbf{X}\mathbf{Z}_i$, where $\mathbf{Z}_i$ is the representation of sample $\mathbf{X}_i$. For all data points, the matrix form can be denoted as $\mathbf{X} \approx \mathbf{X}\mathbf{Z}$. Thus the subspace learning can be formulated as: $\min_{\mathbf{Z}} ||\mathbf{X} - \mathbf{X}\mathbf{Z}||_K + \lambda||\mathbf{Z}||_l$, where $||\cdot||_K$ and $||\cdot||_l$ are two properly chosen norms, λ is the trade-off parameter. After obtaining $\mathbf{Z}$, most existing works [5, 6, 13, 14] directly construct the similarity $\mathbf{S}$ as follows:

$$\mathbf{S} = (|\mathbf{Z}| + |\mathbf{Z}^T|)/2. \tag{1}$$

Then, spectral clustering is performed on similarity $\mathbf{S}$ as: $\min_{\mathbf{F}} \operatorname{tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F})$ s.t. $\mathbf{F}^T\mathbf{F} = \mathbf{I}$, where $\mathbf{F} \in \mathbb{R}^{n\times c}$ is the cluster indicator matrix. $\mathbf{L}_{\mathbf{S}}$ is the Laplacian matrix of $\mathbf{S}$. Considering the above two pipelined steps depend on each other, directly fusing them leads to the Structured Sparse Subspace Clustering (S3C) [11, 19] model:

$$\begin{aligned} \min_{\mathbf{Z},\mathbf{F}} \ & ||\mathbf{X} - \mathbf{X}\mathbf{Z}||_K + \lambda_1||\mathbf{Z}||_l + \lambda_2 \operatorname{tr}(\mathbf{F}^T\mathbf{L}_{\mathbf{S}}\mathbf{F}) \\ \text{s.t.} \ & \mathbf{F}^T\mathbf{F} = \mathbf{I}. \end{aligned} \tag{2}$$

2.2. Co-Referenced Subspace Clustering

Most works [5, 6, 11] can be viewed as a special case of model (2). However, the similarity in Eq. (1) mainly has two shortcomings. One is that *the representation may be negative*. As criticized in [20], it has already changed the meaning of similarity. The other one is that *it does not consider the high-level information*. [21] adopts the co-linkage to exploit a second-order random walk on the primary representation. However, it ignores the connection to the following spectral clustering.

In this paper, we also explore the high-level information of the representation. But differently, we adopt the Hilbert-Schmidt Independence Criterion (HSIC). Specifically, the target high-level representation should be maximal dependence with the primary one since both of them can be viewed as similarities. To measure such the dependence, we employ

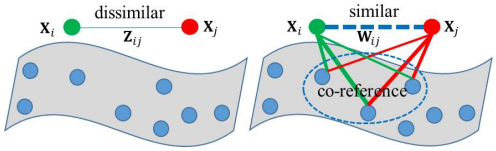


Fig. 1. Comparison of the primary similarity (left) and our high-level co-referenced similarity (right). The points $\mathbf{X}_i$ and $\mathbf{X}_j$ are dissimilar according to the primary representation $\mathbf{Z}_{ij}$ (solid line). But considering the co-reference information, they may be similar with a large probability (dotted line).

the HSIC [22]. For simplicity, we adopt the linear inner kernel for $\mathbf{Z} \in \mathcal{Z}$, say $\mathbf{K}_1 = \mathbf{Z}^T\mathbf{Z}$, and adopt the kernel $\mathbf{K}_2 = \mathbf{W}^{\frac{1}{2}T}\mathbf{W}^{\frac{1}{2}} = \mathbf{W}$, where $\mathbf{W} \succeq 0$ for the co-referenced representation $\mathbf{W} \in \mathcal{W}$. Accordingly, the similarity in the target representation $\mathbf{W}$ space is defined as $\mathbf{S} = (\mathbf{W} + \mathbf{W}^T)/2$. As aforementioned, the target high-level representation $\mathbf{W}$ should be maximal dependence with their primary $\mathbf{Z}$. Thus, we naturally introduce the following HSIC [22] constraint:

$$\max_{\mathbf{W}} \ \operatorname{HSIC}(\mathbf{Z}, \mathbf{W}) = \max_{\mathbf{W}} \ \operatorname{tr}(\mathbf{K}_1\mathbf{H}\mathbf{K}_2\mathbf{H}), \tag{3}$$

where $\mathbf{H} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T$ is a center matrix. Besides, we enforce $\mathbf{W}_i^T\mathbf{1} = 1$ to make $\mathbf{W}$ lie in a union of affine subspaces. Thus, $\mathbf{D}_{\mathbf{W}} = \operatorname{diag}(\sum_j \mathbf{W}_{ij}) = \mathbf{I}$, and $\operatorname{tr}(\mathbf{K}_1\mathbf{H}\mathbf{K}_2\mathbf{H}) = \operatorname{tr}(\mathbf{Z}^T\mathbf{Z}\mathbf{H}\mathbf{W}\mathbf{H}) = \operatorname{tr}(\mathbf{Z}^T\mathbf{Z}\mathbf{H}(\mathbf{W}-\mathbf{D}_{\mathbf{W}})\mathbf{H}) + \operatorname{tr}(\mathbf{H}\mathbf{Z}^T\mathbf{Z}\mathbf{H})$. Consequently, we minimize the following objective:

$$\min_{\mathbf{W}} \ \operatorname{tr}(\mathbf{Z}^T\mathbf{Z}\mathbf{H}\mathbf{L}_{\mathbf{W}}\mathbf{H}), \ \text{s.t.}\ \mathbf{W}_i^T\mathbf{1} = 1; \mathbf{W}_i \succeq 0, \tag{4}$$

where $\mathbf{W}_i \succeq 0$ is to guarantee the similarity value to be non-negative and $\mathbf{L}_{\mathbf{W}} = \mathbf{D}_{\mathbf{W}} - \mathbf{W}$.

Prior to formulating our algorithm, we analyze the advantage of our high-level similarity. Specifically, the objective (4) can be rewritten as:

$$\min_{\mathbf{W}} \operatorname{tr}(\mathbf{Z}^T\mathbf{Z}\mathbf{H}\mathbf{L}_{\mathbf{W}}\mathbf{H}) = \min_{\mathbf{W}} \sum_{ij} ||\overline{\mathbf{Z}}_i - \overline{\mathbf{Z}}_j||_2^2\mathbf{W}_{ij}, \tag{5}$$

where $\overline{\mathbf{Z}} = \mathbf{Z}\mathbf{H}$. This demonstrates that a smaller Euclidean distance of centered primary representation $||\overline{\mathbf{Z}}_i - \overline{\mathbf{Z}}_j||_2^2$ should be assigned a larger high-level $\mathbf{W}_{ij}$, and vice versa. In other words, if two points $\mathbf{X}_i$ and $\mathbf{X}_j$ employ almost the same reference samples during the subspace learning, the Euclidean distance of their primary representations will be smaller in general, thus their similarity will be larger. Accordingly, we call this high-level information as the **co-reference**. Compared with the primary similarity, which only adopts the linear self-representation model $\mathbf{X}_i = \mathbf{X}\mathbf{Z}_i$ and directly uses the representation $\mathbf{Z}$ as the similarity, the high-level one $\mathbf{W}$ is more powerful and robust due to the co-reference information exploited. Figure 1 shows their geometric comparison.

2.3. Kernelized Co-Referenced Subspace Clustering

In this part, we demonstrate that kernelizing the proposed co-referenced term (3) is straightforward. Different from existing

Algorithm 1: ADMM for solving problem (8)

Input: Matrices $\mathbf{X}$, $\mathbf{F}_0$,$\mathbf{Z}_0$, $\mathbf{Q}_0$ and $\mathbf{M}$, parameters λ_1, λ_2, k.
Initialize: $\mathbf{F} = \mathbf{F}_0$, $\mathbf{Z} = \mathbf{Z}_0$, $\mathbf{Q} = \mathbf{Q}_0$, $\mathbf{C} = \mathbf{0}$, $\rho = 1.1$.
while *not convergence* **do**
  Update $\mathbf{Z}$, $\mathbf{Q}$ via (9), (10), respectively;
  Update $\mathbf{W}_i$ via Eq. (12), where $i \in \{1, \ldots, n\}$;
  Update the multiplier $\mathbf{C}$ and the penalty μ via Eq. (14);
  Check the convergence condition $||\mathbf{X} - \mathbf{XZ}||_\infty \leq 10^{-6}$.
end
Output: $\mathbf{Z}$ and $\mathbf{W}$.

Algorithm 2: KCRSC Algorithm

Input: Data matrix $\mathbf{X}$, kernel matrix $\mathbf{M}$, cluster number c, parameters λ and k.
Initialize: $\mathbf{F} = \mathbf{0}, \lambda_1 = \lambda\alpha^t, \lambda_2 = \lambda, \alpha = 1.2, t = 0$.
while *not converged* **do**
  Given $\mathbf{F}$, solve problem (8) via Algorithm 1;
  Given $(\mathbf{Z}, \mathbf{W})$, solve problem (15);
  Check the convergence condition $||\mathbf{\Theta}^{(t+1)} - \mathbf{\Theta}^{(t)}||_\infty < 1$, where $\Theta_{ij} = \frac{1}{2}||\mathbf{F}^i - \mathbf{F}^j||_2^2$;
end
Output: Segmentation matrix $\mathbf{F}$.

works [12, 13, 15], which aim to find a kernel on the data $\mathbf{X}$, we embed a kernel in $\mathbf{Z}$. Suppose that Ψ is a kernel function on $\mathbf{Z}$: $\Psi : \mathbf{Z} \mapsto \Psi(\mathbf{Z})$. For simplicity, we employ the linear transformation, *i.e.*, $\Psi(\mathbf{Z}) = \mathbf{PZ}$. Thus, the co-referenced term (3) can be kernelized as:

$$\max_{\mathbf{W}} \operatorname{tr}(\Psi(\mathbf{Z})^T\Psi(\mathbf{Z})\mathbf{HWH}) = \max_{\mathbf{W}} \operatorname{tr}(\mathbf{Z}^T\mathbf{MZHWH}), \tag{6}$$

where $\mathbf{M} = \mathbf{P}^T\mathbf{P}$. Similarly, it measures the co-reference information by Mahalanobis distance $||\mathbf{P}\overline{\mathbf{Z}}_i - \mathbf{P}\overline{\mathbf{Z}}_j||_2^2$, rather than Euclidean distance $||\overline{\mathbf{Z}}_i - \overline{\mathbf{Z}}_j||_2^2$. In this paper, the main motivation does not focus on how to learn the kernel. We predetermine the representation-induced kernel $\mathbf{M}$. Now, putting every concern (Eqs. (2) and (6)) together leads to the Kernelized Co-Referenced Subspace Clustering (KCRSC) as[1]:

$$\begin{aligned}&\min_{\mathbf{Z},\mathbf{W},\mathbf{F}} ||\mathbf{X} - \mathbf{XZ}||_F^2 + \lambda_2 \operatorname{tr}(\mathbf{Z}^T\mathbf{MZHL_WH}) + \gamma||\mathbf{W}||_F^2 \\ &\lambda_1 \operatorname{tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}) \quad \text{s.t. } \mathbf{W}_i^T\mathbf{1} = 1; \mathbf{W}_i \succeq 0; \mathbf{F}^T\mathbf{F} = \mathbf{I}.\end{aligned} \tag{7}$$

where $\gamma||\mathbf{W}||_F^2$ is to avoid the trivial solution (*i.e.*, $\mathbf{W} = \mathbf{I}$), λ_1, λ_2 and γ are trade-off parameters. Obviously, when $\mathbf{M} = \mathbf{I}$, the kernelized version KCRSC degenerates into CRSC.

3. OPTIMIZATION

3.1. Update Representation Z and W

Given $\mathbf{F}$, we design an ADMM based algorithm [23] for conquering the subproblems of $\mathbf{Z}$ and $\mathbf{W}$. Specifically, we introduce an auxiliary variable $\mathbf{Q}$ to replace $\mathbf{M}^{\frac{1}{2}}\mathbf{Z}$ in the trace term of (7). Accordingly, $\mathbf{Q} = \mathbf{M}^{\frac{1}{2}}\mathbf{Z}$ acts as the additional constraint. The augmented Lagrangian function of (7) is:

$$\begin{aligned}&\mathcal{L}_{\{\mathbf{W}_i^T\mathbf{1}=1;\mathbf{W}_i\succeq 0\}}(\mathbf{Z}, \mathbf{Q}, \mathbf{W}) = ||\mathbf{X} - \mathbf{XZ}||_F^2 + \gamma||\mathbf{W}||_F^2 \\ &+ \lambda_1 \operatorname{tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}) + \lambda_2 \operatorname{tr}(\mathbf{Q}\overline{\mathbf{L_W}}\mathbf{Q}^T) + \Phi(\mathbf{C}, \mathbf{Q} - \mathbf{M}^{\frac{1}{2}}\mathbf{Z}),\end{aligned} \tag{8}$$

where $\overline{\mathbf{L_W}} = \mathbf{HL_WH}$ and $\Phi(\mathbf{C}, \mathbf{Y}) = \frac{\mu}{2}||\mathbf{Y}||_F^2 + \langle \mathbf{C}, \mathbf{Y}\rangle$. $\langle \cdot, \cdot \rangle$ denotes the matrix inner product, μ is a positive penalty scalar and, $\mathbf{C}$ is the Lagrangian multiplier.

[1] Empirically we choose $||\mathbf{X} - \mathbf{XZ}||_K$ as the Frobenius norm and dropout the $||\mathbf{Z}||_l$ (*e.g.*, $||\mathbf{Z}||_1$) for computational efficiency.

Z-subproblem:

$$\begin{aligned}\mathbf{Z} &= \operatorname*{argmin}_{\mathbf{Z}} ||\mathbf{X} - \mathbf{XZ}||_F^2 + \Phi(\mathbf{C}, \mathbf{Q} - \mathbf{M}^{\frac{1}{2}}\mathbf{Z}) \\ &= (2\mathbf{X}^T\mathbf{X} + \mu\mathbf{M})^{-1}(2\mathbf{X}^T\mathbf{X} + \mathbf{M}^{\frac{1}{2}^T}(\mu\mathbf{Q} + \mathbf{C})).\end{aligned} \tag{9}$$

Q-subproblem:

$$\begin{aligned}\mathbf{Q} &= \operatorname*{argmin}_{\mathbf{Q}} \lambda_2 \operatorname{tr}(\mathbf{Q}\overline{\mathbf{L_W}}\mathbf{Q}^T) + \Phi(\mathbf{C}, \mathbf{Q} - \mathbf{M}^{\frac{1}{2}}\mathbf{Z}) \\ &= (\mu\mathbf{M}^{\frac{1}{2}}\mathbf{Z} - \mathbf{C})(2\lambda_2\overline{\mathbf{L_W}} + \mu\mathbf{I})^{-1}.\end{aligned} \tag{10}$$

W-subproblem:

$$\begin{aligned}&\min_{\mathbf{W}} \lambda_1 \operatorname{tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}) + \lambda_2 \operatorname{tr}(\mathbf{QHL_W}(\mathbf{QH})^T) + \\ &\qquad \gamma||\mathbf{W}||_F^2 \quad \text{s. t. } \forall i, \quad \mathbf{W}_i^T\mathbf{1} = 1; \ \mathbf{W}_i \succeq 0.\end{aligned} \tag{11}$$

For each $\mathbf{W}_i$, the closed-form solution is [18, 20]:

$$\mathbf{W}_i = \left(\frac{1 + \sum_{j=1}^k \tilde{\mathbf{d}}_{ij}}{k}\mathbf{1} - \mathbf{d}_i\right)_+, \tag{12}$$

where $\mathbf{d}_i \in \mathbb{R}^n$ is a vector, the j-th element of which is $(\lambda_2||(\mathbf{QH})_i - (\mathbf{QH})_j||_2^2 + \lambda_1||\mathbf{F}^i - \mathbf{F}^j||_2^2)/(4\gamma)$. Please notice that the parameter $k \in \{1, ..., n\}$ is introduced to control the number of nearest neighbors $\mathbf{X}_j$ that could have chance to connect to $\mathbf{X}_i$. The elements of $\tilde{\mathbf{d}}_{ij}$ are those of $\mathbf{d}_{ij}$ but with the ascending order. The parameter γ is determined by [18]:

$$\gamma = \frac{1}{n}\sum_{i=1}^n (\frac{k}{2}\mathbf{d}_{i,k+1} - \frac{1}{2}\sum_{j=1}^k \mathbf{d}_{i,j}). \tag{13}$$

Multiplier:

$$\mathbf{C}^{(t+1)} = \mathbf{C}^{(t)} + \mu(\mathbf{Q} - \mathbf{M}^{\frac{1}{2}}\mathbf{Z}); \ \ \mu^{(t+1)} = \mu^{(t)}\rho, \ \rho > 1. \tag{14}$$

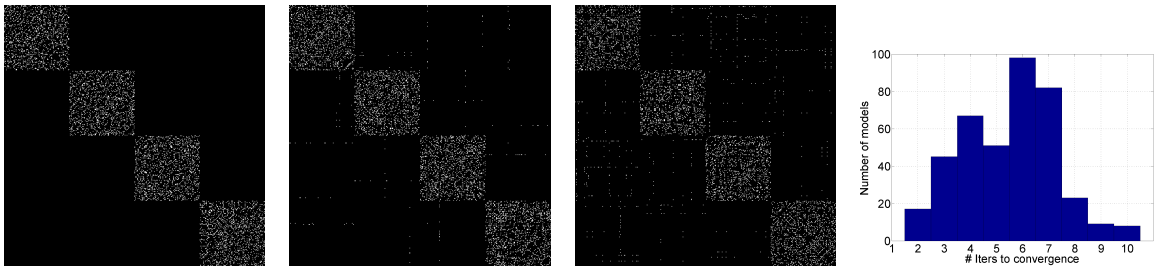
For initializations, $\mathbf{Z}$, $\mathbf{Q}$ and $\mathbf{W}$ are all zero matrices. The representation-induced metric $\mathbf{M}$ is identity matrix $\mathbf{I}$ (CRSC) or computed based on the S3C [11] (*i.e.*, Eq. (2)) by the work [24] (KCRSC).

3.2. Update Indicator F

The second step is to update the indicator matrix $\mathbf{F}$ by:

$$\min_{\mathbf{F}} \ \operatorname{tr}(\mathbf{F}^T\mathbf{L_S}\mathbf{F}) \ \ s.t. \ \mathbf{F}^T\mathbf{F} = \mathbf{I}. \tag{15}$$

The solution is the eigenvectors corresponding to the smallest c eigenvalues of $\mathbf{L_S}$ [25]. The rows of $\mathbf{F}$ are then used as the input to the k-means algorithm, which produces a clustering of the rows of $\mathbf{F}$ that can be used to produce a binary matrix



(a) level=0 (ACC=1.000) (b) level=0.1 (ACC=0.995) (c) level=0.3 (ACC=0.975) (d) Convergence

Fig. 2. **From Left to Right**: (a)-(c) The visual similarity of KCRSC on four-independent data with different levels of noise. (d) Convergence analysis. Most of the KCRSC models converge in 2∼8 iterations.

$\mathbf{F} \in \{0,1\} \in \mathbb{R}^{n\times c}$ such that $\mathbf{F1} = 1$. We summarize the whole scheme in Algorithm 1 and Algorithm 2.

4. EXPERIMENTS

In this section, we conduct experiments on both synthetic and real-world data to validate the superior performance of CRSC/KCRSC over the state-of-the-art alternatives including SSC [5], LRR [6], LSR [9], CAN and PCAN [18], LS3C [13], RSS [20] and the baseline S3C [11]. Three evaluation metrics including Accuracy (**ACC**), Normalized Mutual Information (**NMI**), **Purity** are utilized.

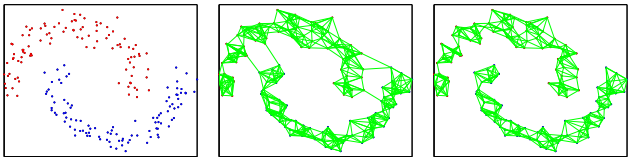
4.1. Synthetic Data

4.1.1. Mahalanobis vs. Euclidean

To visualize the advantages of the Mahalanobis distance, we randomly generate two-moon data. There are two clusters of data distributed in the two-moon shape. Our goal is to construct a similarity matrix to divide data points into exact two clusters. In Figure 3, we set the color of the two clusters to be red and blue separately and let the width of the connecting line denote the similarity weight of two corresponding points. In the similarity graph constructed by CRSC, several pairs of points from different clusters are connected. While in the learned similarity graph by the proposed KCRSC, there is not even a single line across the two clusters. In other words, the diagonal-block property is well preserved.

4.1.2. Robustness to Noise

Similar to [20], to verify the robustness to noise, we generate four independent subspaces $\{\mathcal{S}_i\}_{i=1}^4$ of four-dimensional data. There are 100 unit data points randomly sampled from each subspace, which are chosen to be corrupted with different levels of white Gaussian noise $\mathcal{N}(0,1)$. Figure 2 shows the diagonal-block property of the similarity matrix with different levels of noise. The left to right subfigures (2(a)-2(c)) show the visual variations of similarity matrix with respect to 0, 0.1 and 0.3 noise, respectively. With the increase of noise



(a) Original Data (b) Graph by CRSC (c) Graph by KCRSC

Fig. 3. The visual comparison of CRSC and KCRSC.

in a certain range, the diagonal-block of our method is nicely kept, which reveals the robustness of our method to noise.

4.1.3. Convergence Analysis

Similar to [11, 19], we show the convergence of our method empirically. Using the same synthetic four independent data, we independent run KCRSC 400 times, each time with randomly generated points, and show the histogram of the numbers of iterations (*i.e.*, the statistics of how many iterations has been taken when the algorithm meets the convergence condition.) for KCRSC to converge. As observed in Figure 2(d), our algorithm converges in $2 \sim 8$ iterations on average.

4.2. Real-World Data

4.2.1. Face and Handwritten Digits Clustering

We first evaluate the competitors on the Extended Yale-B[2] [26] and ORL[27] datasets. The Extended Yale-B database consists of 2,414 frontal face images of 38 individuals. While the ORL database contains 400 face images of 40 distinct subjects. The left part of Table 1 provides the quantitative comparison among the competitors on face datasets. The bold numbers in each column represent the best result of all the methods. On Extended Yale-B dataset, it can be observed that most of competitors achieve relatively low performances. The major reason is that the large variation of illumination involves in this dataset. Even so, our method still reaches significant improvements around 8.1%, 4.5% and 2.9% over the

[2]Downloaded from the work [20].

Table 1. Results of different methods for face and handwritten digits clustering.

	Extended Yale-B			ORL			USPS			MNIST		
	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
SSC [5]	0.454	0.528	0.608	0.483	0.568	0.757	0.723	0.732	0.851	0.404	0.525	0.693
LRR_1 [6]	0.565	0.373	0.461	0.692	0.783	0.690	0.744	0.575	0.665	0.564	0.158	0.895
LRR_{21} [6]	0.581	0.280	0.355	0.706	0.802	0.740	0.744	0.555	0.727	0.586	0.458	0.558
LSR_1 [9]	0.559	0.355	0.420	0.682	0.791	0.802	0.724	0.501	0.677	0.590	0.459	0.575
LSR_2 [9]	0.577	0.275	0.505	0.679	0.770	0.775	0.722	0.358	0.936	0.535	0.464	0.630
CAN [18]	0.382	0.401	0.751	0.570	0.766	0.785	0.716	0.730	0.901	0.578	0.556	0.832
PCAN [18]	0.409	0.456	0.635	0.600	0.748	0.787	0.743	0.716	0.942	0.582	0.624	0.870
LS3C [13]	0.534	0.498	0.524	0.598	0.679	0.741	0.745	0.602	0.617	0.543	0.420	0.684
$RSS_{\mathbf{S}}$ [20]	0.735	0.822	0.803	0.687	0.778	0.743	0.804	0.752	0.870	0.534	0.510	0.841
$RSS_{\mathbf{S}\odot\mathbf{Z}}$ [20]	0.730	0.836	0.788	0.687	0.809	0.740	0.810	0.795	0.876	0.538	0.564	0.750
S3C [11]	0.622	0.706	0.772	0.648	0.772	0.790	0.801	0.782	0.854	0.578	0.556	0.832
CRSC	**0.816**	0.876	**0.832**	0.716	0.841	0.805	0.846	0.816	**0.965**	0.623	**0.645**	0.918
KCRSC	0.814	**0.881**	0.831	**0.723**	**0.863**	**0.818**	**0.857**	**0.824**	0.948	**0.647**	0.643	**0.921**

Table 2. Motion segmentation on Hopkins155.

#Motions	2	3	total
S3C [11]	0.978	0.947	0.969
CRSC	0.980	0.954	0.972
KCRSC	**0.984**	**0.962**	**0.978**

Table 3. Different classes of Extended Yale-B.

#Class	2	5	10	20	30
S3C [11]	0.722	0.958	0.902	0.846	0.701
CRSC	1.000	0.997	0.966	0.937	0.826
KCRSC	**1.000**	**0.997**	**0.966**	**0.946**	**0.865**

most competitive method RSS in terms of ACC, NMI, and Purity, respectively. On ORL dataset, we notice that the performance of CAN, PCAN and SSC are relatively poor while LRR, LSR, RSS and S3C are attractive. However, due to the co-referenced similarity, our KCRSC achieves better performance over the best competitor LSR. Then, we attempt to test the abilities of different approaches on two challenging handwritten datasets, *i.e.* USPS [28] and MNIST [29]. The USPS is composed of 10 classes corresponding to 10 handwritten digits, $\{0, 1, \ldots, 9\}$. We randomly sample 100 examples from each subject. The original MNIST handwritten digits contains 70,000 samples from 10 subjects. We randomly sample 200 images from each digit as the dataset. From the right part of Table 1, on both the USPS and MNIST datasets, our KCRSC consistently outperforms the competitors owning to the exploited high-level co-referenced similarity. The experimental results indicate that at least 2% improvement over the others in terms of all three evaluation metrics has achieved.

4.2.2. Motion Segmentation

Due to the limited space, we report the accuracy of our CRSR, KCRSC, and the competitor S3C [11] on Hopkins155. Hopkins155 database consists of 155 video sequences with 2 or 3 motions in each video corresponding to 2 or 3 low-dimensional subspaces. The corresponding experimental results are presented in Table 2. From which, we can see that our CRSC algorithm outperforms S3C, however, due to the fact that the Hopkins155 database has a relatively low noise level, the improvement over S3C is relatively minor.

4.2.3. Comparison against different classes

To analyze the superiority of our co-referenced similarity against different number of classes, we report the performance of the baseline S3C and our CRSC/KCRSC on Extended Yale-B dataset with different numbers of subject, including 2, 5, 10, 20 and 30. The corresponding clustering accuracies are reported in Table 3. From the results, we can see that our CRSC/KCRSC remarkably outperform the competitor S3C.

4.2.4. Parameter Setting

Our method involves parameters λ_1, λ_2, and γ. Inspired by [18, 11], its convergence can be improved by using $\lambda_1 \leftarrow \lambda_1\alpha$. Moreover, for γ, it is determined by the parameter k according to Eq. (13). Therefore, we set $\lambda_1 = \lambda\alpha^{t-1}$, $\lambda_2 = \lambda$ to balance the corresponding terms, where $\alpha = 1.2$, t is the iteration index. In this way, we only have two parameters λ and k to tune. For simplicity, we tune the corresponding $\lambda \in \{0.1, 0.5, 1\}$, $k \in \{3, 5, 7\}$ for all the experiments. The detailed parameter settings (λ, k) are $(0.5, 5)$, $(1.0, 5)$, $(0.5, 3)$, $(0.5, 7)$ and $(0.1, 7)$ on Extended Yale-B, ORL, USPS, MNIST and Hopkins155, respectively.

5. CONCLUSION

This paper has proposed a novel subspace clustering model to explore a high-level co-referenced representation $\mathbf{W}$ by Hilbert-Schmidt Independence Criterion (HSIC) for spectral clustering. Geometry interpretation and a kernelized version

of the proposed constraint have been provided. Experimental results on several synthetic and real-world datasets have demonstrated the significant advantages of our method.

6. REFERENCES

[1] Xiaobo Wang, Xiaojie Guo, and Stan Z Li, "Adaptively unified semi-supervised dictionary learning with active points," in *ICCV*, 2015.

[2] Wei Hong, John Wright, Kun Huang, and Yi Ma, "Multiscale hybrid linear models for lossy image representation," *TIP*, 2006.

[3] Shankar Rao, Roberto Tron, Rene Vidal, and Yi Ma, "Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories.," *PAMI*, 2010.

[4] Congyan Lang, Zhenan Sun, Wei Jia, and Rongxiang Hu, "Saliency detection by multitask sparsity pursuit.," *TIP*, 2012.

[5] Ehsan Elhamifar and Rene Vidal, "Sparse subspace clustering.," *In: CVPR.*, 2009.

[6] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Jun Sun, Yong Yu, and Yi Ma, "Robust recovery of subspace structures by low-rank representation.," *PAMI*, 2013.

[7] Dijun Luo, Feiping Nie, and Heng Huang, "Multi-subspace representation and discovery.," *MLKDD*, 2011.

[8] Budhaditya Saha and and Svetha Venkatesh Ducson Pham, and Dinh Phung, "Sparse subspace clustering via group sparse coding.," *In: ICDM.*, 2013.

[9] Canyi Lu, Hai Min, Deshuang Huang, and Shuicheng Yan, "Robust and efficient subspace segmentation via least squares regression.," *In: ECCV.*, 2012.

[10] Xiaobo Wang, Xiaojie Guo, Zhen Lei, Changqing Zhang, and Stan Z. Li, "Exclusivity-consistency regularized multi-view subspace clustering.," *CVPR*, 2017.

[11] Chun-Guang Li, Rene Vidal, et al., "Structured sparse subspace clustering: A unified optimization framework.," in *CVPR*, 2015, pp. 277–286.

[12] Vishal M Patel and Rene Vidal., "Kernel sparse subspace clustering.," *In: ICIP*, 2014.

[13] Vishal M. Patel and Rene Vidal., "Latent space sparse subspace clustering.," *In: ICCV*, 2013.

[14] Chong You, Robinson Daniel, and Vidal. Rene, "Scalable sparse subspace clustering by orthogonal matching pursuit.," *In: CVPR.*, 2016.

[15] Ming Yin, Yi Guo, Junbin Gao, Zhaoshui He, and Shengli Xie, "Kernel sparse subspace clustering on symmetric positive definite manifolds," in *CVPR*, 2016, pp. 5157–5164.

[16] Jianbo Shi and Jitendra Malik, "Normalized cuts and image segmentation.," *PAMI*, 2000.

[17] Jiashi Feng, Zhouchen Lin, Huan Xu, and Shuicheng Yan, "Robust subspace segmentation with block-diagonal prior.," *In: CVPR.*, 2014.

[18] Feiping Nie, Xiaoqian Wang, and Heng Huang, "Clustering and projected clustering with adaptive neighbors.," *In: KDD.*, 2014.

[19] Chun-Guang Li, Chong You, and René Vidal, "Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework," *TIP*, 2017.

[20] Xiaojie Guo, "Robust subspace segmentation by simultaneously learning data representations and their affinity matrix.," *In: IJCAI.*, 2015.

[21] Liansheng Zhuang, Zihan Zhou, and Jingwen Yin, "Graph construction with label information for semi-supervised learning.," *arXiv.*, 2016.

[22] Arthur Gretton, Olivier Bousquet, and Alex Smola, "Measuring statistical dependence with hilbert-schmidt norms.," *In: ICALT.*, 2005.

[23] Zhouchen Lin, Risheng Liu, and Zhixun Su, "Linearized alternating direction method with adaptive penalty for low-rank representation.," *In: NIPS.*, 2011.

[24] Ian. Jolliffe, "Principal component analysis.," *John Wiley and Sons.*, 2002.

[25] Ky. Fan, "On a theorem of weyl concerning eigenvalues of linear transformations ii.," *In: Proceedings of the National Academy of Sciences.*, 1950.

[26] Kuang-Chih Lee, Jeffrey Ho, and David J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting.," *PAMI*, 2005.

[27] Deng Cai, Xiaofei He, Jiawei Han, and Hongjiang Zhang, "Orthogonal laplacianfaces for face recognition.," *TIP*, 2006.

[28] Deng Cai, Xiaofei He, and Jiawei Han, "Speed up kernel discriminant analysis.," *The International Journal on Very Large Data Bases*, 2011.

[29] Yann LeCun, Leon Botton, Yoshua Bengio, and Patrick Haffner., "Gradient-based learning applied to document recognition.," *Proceedings of the IEEE.*, 1998.