

Neither global nor local: A hierarchical robust subspace clustering for image data

Maryam Abdolali, Mohammad Rahmati*

Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran



ARTICLE INFO

Article history:

Received 22 May 2019

Revised 12 November 2019

Accepted 16 November 2019

Available online 18 November 2019

Keywords:

Sparse subspace clustering

Grassmann manifold

Group lasso

Spectral clustering

Graph fusion

ABSTRACT

In this study, we consider the problem of subspace clustering in the presence of spatially contiguous noise, occlusion, and disguise. We argue that self-expressive representation of data, which is a key characteristic of current state-of-the-art approaches, is severely sensitive to occlusions and complex real-world noises. To alleviate this problem, we highlight the importance of previously neglected local representations in improving robustness and propose a hierarchical framework that combines the robustness of local-patch-based representations and the discriminative property of global representations. This approach consists of two main steps: 1) A top-down stage, in which the input data are subject to repeated division to smaller patches and 2) a bottom-up stage, in which the low rank embedding of representation matrices of local patches in the field of view of a corresponding patch in the upper level are merged on a Grassmann manifold. This unified approach provides two key pieces of information for neighborhood graph of the corresponding patch on the upper level: cannot-links and recommended-links. This supplies a robust summary of local representations which is further employed for computing self-expressive representations using a novel weighted sparse group lasso optimization problem. Numerical results for several data sets confirm the efficiency of our approach.

© 2019 Elsevier Inc. All rights reserved.

1. Introduction

Detecting low-dimensional structures in high-dimensional data is essential for many applications in different areas including computer vision and image processing. In many of these applications, data often have fewer degrees of freedom than the original high ambient dimension. This observation has led to the development of several classical approaches to find the low-dimensional representation of the data [1]. A large family of these approaches, including the well-known Principal Component Analysis (PCA) method [2], assumes that the data lie in a *single* low-dimensional subspace/manifold. However, in many applications, data points are distributed around several subspaces and should be better represented by *multiple* low-dimensional subspaces. Modeling data with a collection of subspaces has led to a more general problem that is often referred to as *Subspace Clustering* [3]:

Definition 1 (Subspace Clustering): Let $X = \{x_i \in \mathbb{R}^D\}_{i=1}^N$ be a collection of data points from n unknown subspaces $S_1, S_2, \dots, S_n$ with intrinsic dimensions $d_1, d_2, \dots, d_n$ (where $d_i < D$). The goal of subspace clustering is to segment the data points according to their underlying subspaces and identify the parameters of each subspace.

* Corresponding author.

E-mail addresses: mabdolali@aut.ac.ir (M. Abdolali), rahmati@aut.ac.ir, rahmati@ieee.org (M. Rahmati).

This problem has received considerable attention over the past decade and many attempts have been made to address it [3–5]. Among them, a family of approaches based on recent developments in low-rank and sparse representation have achieved state-of-the-art results [6–8]. These algorithms have two major key steps: 1) Constructing an affinity matrix (neighborhood graph) using the self-expressiveness property of data [6,9], which assumes that each data point can be written as a linear combination of other data points, and 2) Obtaining the data clusters by applying spectral clustering on the affinity matrix. In particular, the affinity matrix in these algorithms is constructed by optimizing the following problem:

$$\begin{aligned} \min_{C \in \mathbb{R}^{N \times N}, E \in \mathbb{R}^{D \times N}} f(C) + \lambda g(E) \\ \text{subject to } X = XC + E \quad \text{and } C_{i,i} = 0 \text{ for all } i, \end{aligned} \quad (1)$$

where $X \in \mathbb{R}^{D \times N}$ is the data matrix (with D -dimensional data points as columns and N the number of data points), $C \in \mathbb{R}^{N \times N}$ is the coefficient matrix and $E \in \mathbb{R}^{D \times N}$ is the error matrix. $C_{i,i}$ indicates the diagonal elements of matrix C and the constraint $C_{i,i} = 0$ (for all i) eliminates the trivial solution of writing data points as a linear combination of themselves. The two functions $f(\cdot)$ and $g(\cdot)$ denote the regularizations on the C and E matrices, respectively. The three main representative methods among these algorithms, namely Sparse Subspace Clustering (SSC) [6], Low Rank Representation (LRR) [8] and Least Square Regression (LSR) [10], differ in the function of $f(C)$. In particular, $f(C)$ is the ℓ_1 norm for SSC which prioritizes sparse solutions, nuclear norm (the sum of the singular values of the input matrix) for LRR which prioritizes low rank solutions and Frobenius norm for LSR. Once the problem in (1) is optimized, the affinity matrix is obtained by symmetrizing the coefficient matrix C via $|C| + |C^T|$. The clusters are next calculated by applying spectral clustering on the affinity matrix.

Although advances in sparse and low-rank representation literature have significantly contributed to the development of elegant subspace clustering algorithms, they are based on *global* self-expressive representation of data in which the *entire* data point is expressed as a linear combination of other data points. However, clustering based on global data representation can be easily affected by occlusions and severely corrupted noisy blocks [11]. The conventional ℓ_1 or Frobenius norms that are often utilized to regularize the error matrix are incapable of modeling the complexity of real-world data corruptions [12]. The vulnerability of the self-expressive property to occlusion and disguise was previously noticed in related sparse-representation-based classification literature [13,14]. As discussed in these works, self-expressive representations are sensitive to occlusions and continuous corruptions that are similar to the regions of other samples from different categories. In particular, the robustness of sparse representations is determined by the neighborliness of the symmetric convex hull of data points and for the amount of occlusions which exceeds a threshold related to this quality, the holistic-based methods are expected to fail [13]. In the supervised algorithms, block-wise (local) representations might be paired with majority voting to improve the performance of classification [13]. However, although block-wise (local) representations tend to be more robust in response to occlusions and contiguous noises, they naturally contain less-discriminant information and hence combining the clustering results of the local patches via a trivial majority voting scheme might drastically be affected by non-informative patches.

To overcome this major shortcoming, we propose an efficient hierarchical framework that combines the advantages of *local* representations with global self-expressive representation of data. Instead of independently clustering the local representations and then applying majority voting to the obtained cluster outputs (which as shown here does not typically improve clustering accuracy), we construct a multi-layer graph based on local representations. We merge the individual layers such that it captures the similarities of individual representations and preserves the information that the majority of layers agree on. Specifically, in our approach, each data point is divided into a collection of blocks/patches and the local connectivities of these patches are summarized using the corresponding low-dimensional embeddings on a Grassmann manifold [15]. This local connectivity information is then used to guide the global self-expressive representation by providing prior knowledge on 1) which connections should be avoided using weighted sparse regularization and 2) which connections are recommended by the local representations using group sparse regularization [16].

The benefits of global and local representations are rather complementary and a trade-off between robustness and discriminant information of patches exists. To eliminate the effect of local patch size on performance, a hierarchical structure is developed such that the robustness of local representations gradually spreads over the self-expressive computation of the larger patches. This hierarchical framework attempts to balance complementary advantages of finer and coarser patches while reducing the sensitivity of the clustering performance to patch size.

The main contributions of this study are to 1) shed light on the importance of local representations for improving the robustness of subspace clustering approaches, which is neglected in the previous methods, and 2) propose a novel framework, Locally-Guided SSC (LG-SSC), to bridge the gap between discriminative global representation and robust local alternatives to achieve a robust discriminant representation for subspace clustering. The proposed framework includes of two key steps: 1) combining diverse characteristics of local patches using an efficient hierarchical low-dimensional analysis on Grassmann manifolds and 2) computing global sparse self-expressive representation of data under the guidance of obtained local representations using group lasso and weighted sparse regularizations. As is shown, our approach is more robust in response to occlusions, illumination effects and continuous block noises.

This paper is an extension of the work originally presented in [17]. Inspired by the significant increase in clustering accuracy using the self-expressive representation of local patches in our preliminary work, we further utilized this information in a different hierarchical approach. In particular, in this paper, the local information fusion at each level is fed back to guide the calculation of the coefficient matrix in the upper level to achieve a more robust representation. This approach, which

can be considered as *coefficient calculation-level* fusion, shows stronger robustness and clustering accuracy compared to that of the previous *subspace estimation-level* fusion.

The remainder of the paper is organized as follows: In Section 2, the related works are briefly reviewed. After presenting the motivation behind our proposed approach in Section 3, a detailed explanation of the two suggested major steps is elaborated in Section 4. We evaluate the performance of the proposed approach in Section 5 and finally conclude our work in Section 6.

2. Related works

With many applications in computer vision and signal processing, subspace clustering has drawn considerable attention over the past two decades and many algorithms have been designed to address this problem. These algorithms are typically divided into four main categories[3]: iterative, statistical, algebraic and spectral clustering-based methods.

Iterative methods, such as k-subspaces [18], were among the earliest and probably simplest approaches proposed for subspace clustering. In these intuitive methods, the clustering algorithm iterates between assigning points to clusters and fitting a subspace to each cluster. Despite being simple and intuitive, iterative methods are sensitive to initialization and outliers. Moreover, the number of subspaces and dimension of each subspace should be known in advance.

In statistical approaches, with Mixture of Probabilistic PCA (MPPCA)[5] being the most well-known, the distribution of data in each subspace is assumed to be Gaussian. Hence, they cast the clustering problem into fitting a mixture of Gaussian distributions to the data. The methods in this category are subject to the same drawbacks of the iterative approaches. In the third category, namely algebraic methods, such as Generalized Principal Component Analysis (GPCA), a set of polynomials are fitted to the data [4]. High computational complexity, sensitivity to noise and outliers are some of the shortcomings of the methods in this category.

The celebrated spectral clustering algorithm [9] initiated the development of a new wave of clustering techniques based on calculating the spectrum of a predefined similarity matrix of the data. The main difference between the methods in this category lies in the construction of the similarity matrix; interestingly, this matrix plays a critical role in the success of the clustering as well. Although this matrix can be obtained using the traditional K-nearest neighbors (KNN) method, the fundamental recent findings in low rank and sparse approximation literature dramatically revolutionized neighborhood graph construction for the subspace clustering problem. Three main current state-of-the-art methods [6,8,10] are based on the so-called *self-expressive* property which assumes each data point in the data set can be written as a linear combination of other data points from the same subspace.

Self-expressive based subspace clustering methods have broad theoretical guarantees for data that are perfectly drawn from a collection of low-dimensional subspaces [6,19]. Recently, these findings have been extended to noisy cases as well [20]. An intuitive simple means for modeling errors was first proposed in [6] where the self-expressiveness term was relaxed to $X = XC + E$. The elements of the error matrix E were generally modeled by independent Laplacian or Gaussian distributions using ℓ_1 or Frobenius norms. Although this has the advantage of simplicity and maintaining the convexity of the optimization problem, in practice, the error matrices are generated by more complicated variations and specifically the independence assumption between the error elements in these models is too restrictive.

To further improve robustness, other distributions were used to model the complex noise, including Cauchy [21], Mixture of Gaussian [22] and finite mixture of exponential power distribution[23]. Recently, correntropy induced subspace clustering has been utilized to address non-Gaussian and complex noises [24,25]. [26] directly considers the errors in the original space by proposing a non-convex formulation in which it is assumed that the corrupted data are generated by a linear combination of the error matrix and clean data matrix and the self-expressiveness holds for the clean data. Although these approaches show some improvements in addressing practical noise, they still assume that the elements of the error matrix are independent. However, the spatially continuous error caused by occlusion, illumination effects and disguise does not follow this assumption.

There are several other approaches that have attempted to improve robustness by improving the quality of the coefficient matrix. Lu et al. [27] explicitly enforced the Laplacian matrix (corresponding to the coefficient matrix) to be block diagonal using a block-diagonal-inducing regularizer. This encourages the neighborhood graph to contain exactly n connected components which might reduce the number of wrong connections. Some works have emphasized the importance of the grouping effect on better clustering performance. Seminal works in this category include the use of the relatively new Trace Lasso norm for regularizing the coefficient matrix [28], enforcing smooth representation for the coefficient matrix [29] and construction of an affinity graph using the nearest neighbor of each data point in spherical distance [30]. In the iterative approaches, the quality of the coefficient matrix was iteratively improved based on some criterion. For example, in the approach in [31], a linear transformation was learned such that low-rank structures for data points from the same subspaces were restored and a maximally separated structure for data from different subspaces was obtained. In another iterative method [32], a unified optimization framework for learning both the coefficient matrix and the segmentation was presented. In this iterative approach, at each iteration, a segmentation matrix was constructed to help re-weight the representation matrix to avoid certain connections.

Although the aforementioned approaches can enhance the robustness in some cases, an important shortcoming is that they are based on global representation and hence can be easily affected by severely corrupted regions in the data points. Patch-based image representations were often used to increase the robustness in response to continuous noises in sparse



Fig. 1. Few samples corresponding to three individuals of AR data base.

representation literature for classification [14,33]. Multi-scale patch-based regularization using nuclear norm for modeling the error has shown effectiveness for face classification [12] and single low-rank subspace estimation [34]. To the best of our knowledge, our previous approach, Multi-Graph based SSC (MG-SSC) [17], is the first that explicitly emphasized the importance of local representations in improving robustness of subspace clustering algorithms in the presence of occlusion and complex noise. In this work, a multi-layer graph was constructed by dividing each data into patches of different sizes and independently computing the sparse affinity matrix corresponding to each collection of patches (using SSC). A summarized low-dimensional representation of this multi-layer graph was computed using a weighted subspace analysis of individual graphs on a Grassmann manifold. In the current study, the advantages of global and local (patch-based) representations are combined for the problem of subspace clustering in a novel different framework.

Notably, the proposed clustering approach does not build a hierarchy of clusters but rather a hierarchy of patches. Hence, this approach should not be confused with methods in hierarchical clustering analysis in which the clusters are merged or split through the hierarchy. [35–37].

3. Importance of local representations in robust subspace clustering

In many practical subspace clustering cases, the data can be partially occluded or corrupted. Even in these cases, detecting the low-dimensional structures may still be possible using the redundancy that is often present in high-dimensional data. However, corrupted samples might severely affect the neighborhood graph such that they lead to a completely erroneous understanding of the data structure.

In this section, we illustrate the effect of occlusions and corruptions on self-expressive representation using a real-world example. We consider facial images of three subjects from AR database [38] (more information on this database is provided in the experimental section). A few samples from the selected images are shown in Fig. 1. As can be seen, several varieties of corruptions are present in these images, including disguises using sun glasses, scarves and illumination variations. It is typically assumed that facial images of a subject under different lighting conditions can be approximated by a nine-dimensional linear subspace [39]. Hence, a collection of facial images of several subjects lies on the union of nine dimensional subspaces [6].

We select three subsets $\{X^{(i)}\}_{i=1}^3$ from these images: $X^{(1)}$ that includes images with neither an illumination effect nor disguise, $X^{(2)}$ that includes images with no disguise but under different illumination effects, and $X^{(3)}$ that includes the entire images under both variations. Among three major representative methods for self-expressive based subspace clustering (SSC, LRR and LSR), SSC has the strongest theoretical guarantees and hence the SSC algorithm is selected and applied to these subsets of images. The normalized spectral embedding of the three corresponding coefficient matrices $\{C^{(i)}\}_{i=1}^3$ are plotted in Fig. 2. In particular, for each coefficient matrix $C^{(i)}$, the corresponding Laplacian matrix $L^{(i)}$ is calculated as $L^{(i)} = I - D^{(i)-\frac{1}{2}}C^{(i)}D^{(i)-\frac{1}{2}}$, where $D^{(i)}$ is the diagonal matrix with its (k, k) th entry defined as $D_{kk}^{(i)} = \sum_j C_{kj}^{(i)}$. It is already shown that the normalized eigenvectors corresponding to the second and third smallest eigenvalues of the Laplacian matrix provide a two-dimensional representation for the data affinity (from three clusters) [9]. The representations for three individuals are plotted in three different colors (red, blue and black). Comparing the three embedded data corresponding to $\{C^{(i)}\}_{i=1}^3$ in Fig. 2 (a) - (c), it is clear that illumination and particularly occlusion can affect the *global* self-expressive coefficients. The well-separated embedded data in Fig. 2 (a) confirms the efficiency of the obtained representation matrix of SSC for clean data. Applying a classical k-means on the embedded data can lead to perfect segmentation. However, in the presence of illumination (Fig. 2 (b)), the previous well-separated clusters vanish and the clustering accuracy decreases. The effect of occlusion (sun-glasses and scarves) is shown in part (c) of Fig. 2; it is clear that occlusion further ruins the separated clusters shown in part (a) and makes the detection of the three subspaces impossible.

In contrast to global representations, patch-based representations can be more robust in response to severely corrupted regions. However, the individual local representations might be compromised by lack of discriminant information which is

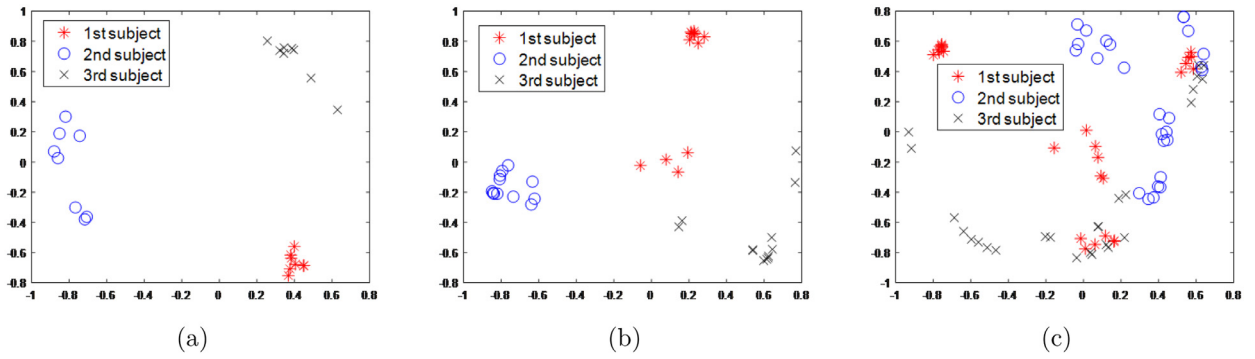


Fig. 2. SSC based embedded vectors of samples (a) without any illumination variations or occlusions (b) with illumination variations but without occlusions (c) with both illumination variations and occlusions.

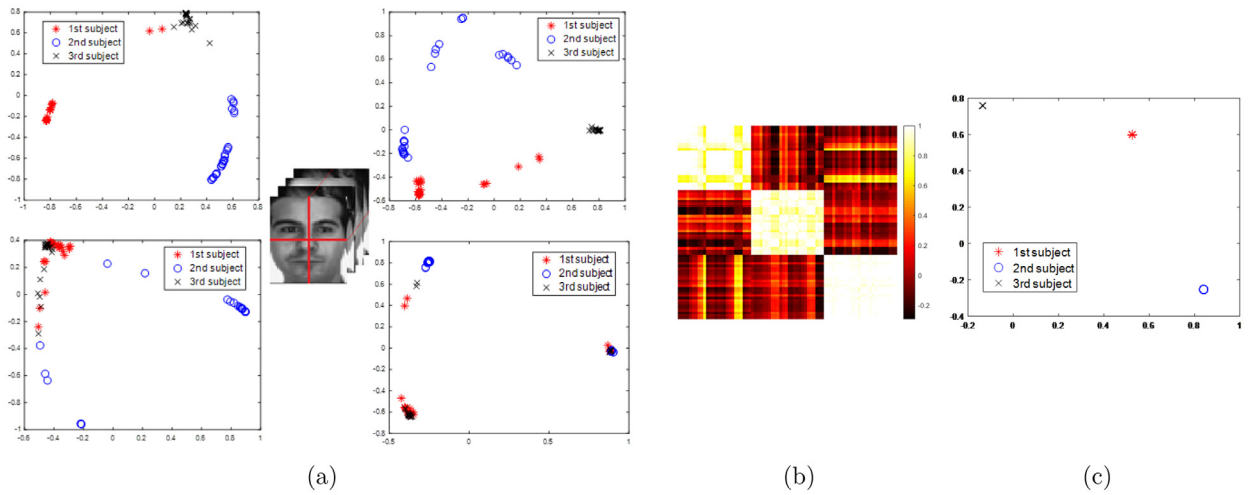


Fig. 3. The effect of local representations. (a) The embedded vectors of samples corresponding to 4 patches, (b) the obtained summary representation (spectral kernel) and (c) the final embedded vectors of samples using proposed framework.

critical for accurate clustering. In this study, we argue that an efficient combination of local representations can highlight the connections that the majority of these representations agree on. To illustrate this, we divide each facial image in $X^{(3)}$ (which contains all images of the three selected individuals) into four patches and plot the spectral embedding of the four patches immediately next to the corresponding patch in Fig. 3.¹ As shown, each local representation provides some information regarding the connectivities within the data points; however, in nearly none are the three clusters perfectly separated. Hence, independently considering each representation and applying a simple majority voting on the clustering outputs of these representations does not necessarily improve the clustering accuracy. Our proposed framework, based on effective multi-layer graph fusion, obtains a (spectral kernel) summary representation from these local representations (as shown in Fig. 3 (b) for completion). This combined representation is a $N \times N$ matrix with values between -1 and 1 whose entries quantify the possibility of two corresponding data points belonging to the same subspace according to the local representations. The values that are nearer 1 indicate the higher possibility. This information is used to improve the quality of the global representation by adding constraints on possible connections for the neighborhood graph (the details are presented in the next section). The final global low-dimensional embedding corresponding to our proposed LG-SSC is plotted in Fig. 3 (c). It can be seen that not only the three clusters are perfectly separated but also all the samples within each cluster have the exact same embedding in this case. This suggests that our proposed framework can improve within cluster connectivity along with overall clustering accuracy.

4. Proposed Framework: Locally-Guided SSC

The proposed hierarchical framework, LG-SSC, consists of a top-down stage and a bottom-up stage. LG-SSC has two major ingredients in each level of this hierarchy: 1) dividing the data into local patches (during the top-down stage) and

¹ The spectrum of each patch is obtained using the proposed algorithm by dividing it into four more patches. We explain this in the next section.

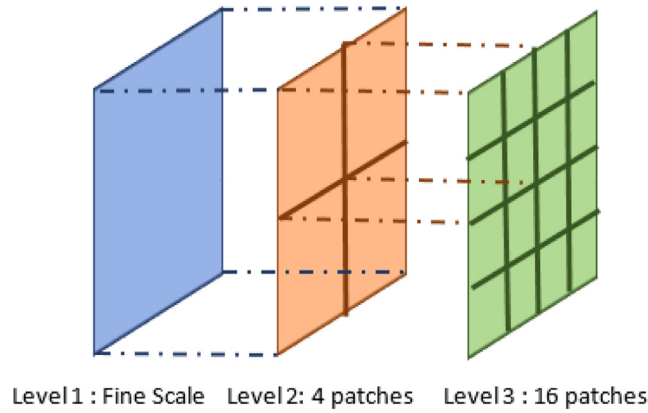


Fig. 4. Dividing sample x_i (in 2d image form) using the hierarchical structure with $s = 3$ (3 levels) and $p = 4$ (4 patches). The field of view of two patches in level 1 and level 2 are illustrated for clarity as well.

summarizing the connectivity information between them (during the bottom-up stage) and 2) guiding the detection of low-dimensional structures using this information (during the bottom-up stage). In this section, details of these two key steps are presented.

4.1. Division and local information summarization

There is a natural trade-off between robustness and the discriminant information contained in each patch of sample. Generally, the smaller patches contain less information but show more robustness. To take the benefits of patches of different sizes and to reduce the sensitivity of the clustering accuracy to patch size, we build a hierarchical framework from data. Given the data matrix $X = [x_1, x_2, \dots, x_N]$, where each column corresponds to a vectorized image sample, we construct a hierarchical structure composed of s levels. At the top of this structure is the given data matrix. At the next level, the image data are divided into p patches (typically four or nine)² The division is conducted for each image data x_i in its (2d) matrix form. This process is repeated for each obtained patch in the current level. The created p patches indicate the *field of view* of the corresponding patch from the upper level. This procedure is shown in Fig. 4 for $s = 3$ and $p = 4$ for one image data. Let the whole gallery of patches at level i be represented by $X^{(i)} = \{X_1^{(i)}, \dots, X_T^{(i)}\}$ where $T = p^{(i-1)}$ is the number of patches that are generated in the level of i and $X_j^{(i)}$ ($j = 1, \dots, T$) is the collection of local patches (at the i th level) with the same coordination (position) from all images. Clearly, the patches in the higher-numbered levels are smaller in size and generally contain less discriminant information regarding the entire image data. However, they are more robust in response to corruptions.

Suppose that in the i th level, the collection of patches $X_j^{(i)}$ (where $j \in \{1, \dots, p^{i-1}\}$) are further divided into p patches. Hence, at the $(i + 1)$ th level, p coefficient matrices can be obtained corresponding to the set of p collected patches. We denote these patches and their corresponding coefficient representations by $\{X_{\ell_j^{(i+1)}}^{(i+1)}\}_{k=1}^p = \{X_{\ell_j^{(i+1)}}^{(i+1)}, \dots, X_{\ell_j^{(i+1)}}^{(i+1)}\}$ and $\{C_{\ell_j^{(i+1)}}^{(i+1)}\}_{k=1}^p = \{C_{\ell_j^{(i+1)}}^{(i+1)}, \dots, C_{\ell_j^{(i+1)}}^{(i+1)}\}$, respectively. ℓ_j^k ($k = 1, \dots, p$) indicates the indices of the collection of patches that are generated by dividing the j th patch in the upper level. These patches are in the field of view of the collection of patches in $X_j^{(i)}$. The set $\{C_{\ell_j^{(i+1)}}^{(i+1)}\}_{k=1}^p$ contains p different types of relationships between the data points. With no prior knowledge, it is highly non-trivial to choose the best representation among them. In fact, the obtained different coefficient matrices can be interpreted as different views from the larger patches at the finer level in which each coefficient matrix corresponds to an affinity matrix of a graph. Hence, to combine the complementary information of the different coefficient matrices, we use a multi-layer graph fusion approach [40] by transforming the information of each representation into a subspace on the Grassmann manifold.

The utilized graph fusion approach [40] is briefly presented as follows: Given p different graph affinity matrices with a common vertex set representing the data points, to summarize the intrinsic relationships between the data points (the

² A possible rule of thumb for selecting the value of p is to set it as four if the patches in corners contain discriminant information or if a symmetrical structure is present in the image (for example for facial images), and to set it as nine if the patches near the center contain the majority of discriminant information (for example for object images, where the object appears in the center of image). The sensitivity of LG-SSC with respect to this parameter is evaluated in Section 5.4 (Table 6). In fact, a major motivation behind the hierarchical structure of LG-SSC is to reduce this sensitivity. As will be shown, the clustering accuracy is quite stable with respect to the patch size. Typically, there is no need to set p to higher values, as these values can be considered by adding more levels to the hierarchy (for example $p = 16$ could be covered by setting $s = 3$ and $p = 4$).

vertices of the graph), the problem can be mapped to the problem of finding a low-dimensional representative subspace such that it preserves information of each affinity matrix in a meaningful manner.

As mentioned in Section 2, for each affinity matrix ($A_{\ell_j^k}^i \in \mathbb{R}^{N \times N}$), there is a corresponding normalized spectral embedding ($U_{\ell_j^k}^i \in \mathbb{R}^{N \times n}$) that can be considered as the low-dimensional representation of the affinity matrix. This spectral embedding can be obtained by simply calculating the n eigenvectors corresponding to the n smallest eigenvalues of the Laplacian matrix $L_{\ell_j^k}^i$ (obtained from the affinity matrix). Using the collection of subspace representations of the p available affinity matrices, one can naturally define a summary representation of multiple affinity matrices in the level of i by optimizing the following problem [40]:

$$\begin{aligned} V_j^{(i-1)} = \arg \min_{V \in \mathbb{R}^{N \times n}} & \sum_{k=1}^p \text{tr}(V^T L_{\ell_j^k}^i V) \\ & - \alpha \sum_{k=1}^p \text{tr}(V V^T U_{\ell_j^k}^i U_{\ell_j^k}^{i T}) \\ \text{s.t. } & V^T V = I, \end{aligned} \quad (2)$$

where α is the regularization parameter. $\{L_{\ell_j^k}^i\}_{k=1}^p$ is the collection of Laplacian matrices, corresponding to p coefficient matrices $\{C_{\ell_j^k}^i\}_{k=1}^p$ in the level of i . $\{U_{\ell_j^k}^i\}_{k=1}^p$ indicates the individual normalized eigenvectors corresponding to the Laplacian matrices $\{L_{\ell_j^k}^i\}_{k=1}^p$. The first term in the aforementioned optimization problem finds a summary representation V from the collection of Laplacian matrices $\{L_{\ell_j^k}^i\}_{k=1}^p$ such that the connectivity information of each individual affinity matrix is preserved. The second term is the sum of the squared projection distances between V and individual subspaces $\{U_{\ell_j^k}^i\}_{k=1}^p$ on a Grassmann manifold. This term forces the summary representation to be near other subspace representations on the Grassmann manifold. This optimization problem has a closed form solution based on Rayleigh-Ritz theorem [9]. The summary representation $V_j^{(i-1)}$ can be obtained by calculating the smallest n eigenvectors of the following matrix:

$$(L_j^{(i)})_{\text{summary}} = \sum_{k=1}^p L_{\ell_j^k}^i - \alpha \sum_{k=1}^p U_{\ell_j^k}^i U_{\ell_j^k}^{i T}. \quad (3)$$

$V_j^{(i-1)}$ indicates the summarized representation for the collection of patches that are obtained by dividing the j th patch of all images in the level of $i - 1$. Each row of the matrix $V_j^{(i-1)}$ is normalized to have a unit ℓ_2 norm. As we shall see, the subspace representation of the p local patches in the i th level can be used to more efficiently calculate the coefficient matrix of the corresponding patch in the upper level ($i - 1$).

4.2. From local summarization to global representations

Following the division of the image data into local patches at different scales, to obtain the global coefficient matrix, a bottom-up process should be carried out. To clarify, suppose we have only two levels ($s = 2$) and each data point (an image) is divided into four patches ($p = 4$). The overview of LG-SSC for $p = 4$ and $s = 2$ is illustrated in Fig. 5. In particular, we start from the coarsest level (the second level here) and the regular SSC algorithm is applied to the collection of patches at this level. Hence, four sets of affinity matrices $\{C_{\ell_1^k}^2\}_{k=1}^4$ are obtained.

Using the methodology described in the previous subsection, a summary representation ($V_1^1 \in \mathbb{R}^{N \times n}$) from the four local representations can be efficiently calculated (by optimizing the problem 2). We define the spectral kernel of the summary representation in the embedded space as follows:

$$K_1^1 = (V_1^1)(V_1^1)^T, \quad (4)$$

The matrix $K_1^1 \in [-1 : 1]^{N \times N}$ provides information regarding the similarity between samples in the embedded space (note that the rows of matrix V_1^1 are normalized). The entries with low values indicate the connectivities that local representations highly agree to avoid. Therefore, we can indicate the *cannot-links* by adding an extra penalty onto the coefficient matrix when the values of K_1^1 are low. In particular, we define the matrix $\Theta_1^1 \in [0 : 2]^{N \times N}$ corresponding to K_1^1 as follows:

$$\Theta_1^1 = (\mathbf{1}_{N \times N} - K_1^1),$$

where $\mathbf{1}_{N \times N}$ is a $N \times N$ matrix with all entries equal to one. Hence, the entries of the matrix Θ_1^1 have high values when the corresponding entries in K_1^1 have low values. Using this matrix, we add restrictions to the entries of the coefficient matrix C_1^1 at the fine level (global representation in this example) by optimizing the following problem:

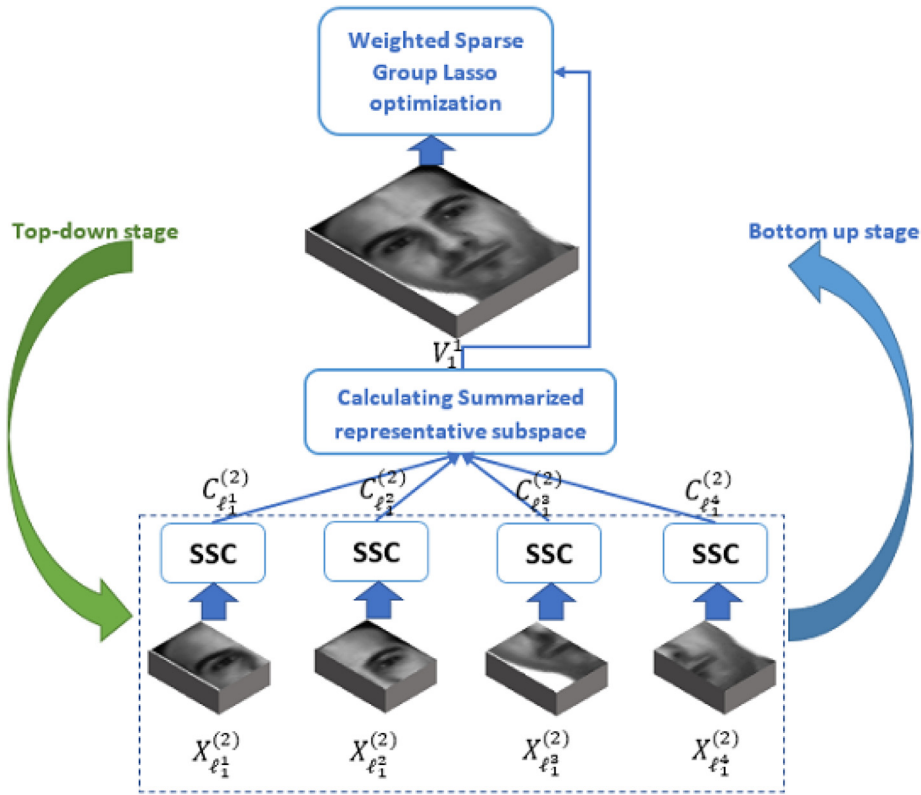


Fig. 5. Illustration of steps of LG-SSC with $p = 4$ and $s = 2$. In the top-down stage, each image is divided into p patches to obtain $\{X_{\ell_k}^{(2)}\}_{k=1}^4$ in the second level. In bottom-up stage, SSC is applied on each collection of patches and the obtained coefficient matrices are merged using subspace analysis on Grassmann manifold. The summarized information is employed to guide the calculation of final global coefficient matrix in the fine scale.

$$C_1^1 = \arg \min_{C \in \mathbb{R}^{N \times N}} \|C\|_1 + \lambda_1 \|\Theta_1^1 \odot C\|_1 + \frac{\mu}{2} \|X - XC\|_F^2$$

s.t. $\text{diag}(C) = 0$,

(5)

where λ_1 and μ are regularization parameters and $\odot$ denotes the element-wise (Hadamard) production. $\text{diag}(C) \in \mathbb{R}^N$ is the vector consisting of the diagonal elements of matrix C . This is in fact a weighted lasso optimization problem, in which we are adding an extra penalty to the entries of the matrix C when the corresponding entries in Θ_1^1 are high. This can be validated by rewriting the first two terms in problem (5) as follows:

$$\|C\|_1 + \lambda_1 \|\Theta_1^1 \odot C\|_1 = \sum_{i,j} |C_{ij}| (1 + \lambda_1 (\Theta_1^1)_{ij})$$
(6)

where $(\cdot)_{ij}$ denotes the (i,j) th entry of the input matrix. In other words, when the values of the entries in Θ_1^1 are high, an extra penalty is added to the corresponding entries in C . Hence, these entries are forced to be nearer zero. This is equal to adding constraints and restrictions on possible edges/links in the neighborhood graph based on the Θ_1^1 matrix. Thus, high values of Θ_1^1 indicate the cannot-links that are specified by the local representations.

Remark 1. The elements of the matrix K_j^i (here $i = j = 1$) can be interpreted as a quantified measure of the possibility that two corresponding samples belong to the same subspace according to the obtained summary representation. In particular, the nearer the values of these elements are to one, the higher the possibility of the two samples belonging to the same subspace. Hence, we can drop the additional penalty on entries with high values, specifically those higher than a predefined threshold. In practice, this threshold can be set to any value within the range $[0.7 - 0.9]$ as the proposed framework is not sensitive to this value.

However, the summary representation V_1^1 not only indicates the “cannot-links” constraints but also contains information regarding which connections are recommended by local views. By applying the k-means algorithm on the normalized rows of V_1^1 , an initial grouping of data points based on the summary of the local representations is achieved. This grouping information denotes the connections that the local representations recommend. Let G be the set that contains the indices of

the data points in each cluster according to V_1^1 . This information is added to the problem (5) as follows:

$$\begin{aligned} C_1^1 = \arg \min_{C \in \mathbb{R}^{N \times N}} & \|C\|_1 + \lambda_1 \|\Theta_1^1 \odot C\|_1 \\ & + \lambda_2 \sum_{j=1}^N \sum_{g \in G} \|(C_{:,j})_g\|_2 + \frac{\mu}{2} \|X - XC\|_F^2 \\ \text{s.t. } & \text{diag}(C) = 0, \end{aligned} \quad (7)$$

where the third term is the group lasso regularization [16]. Here $(\cdot)_{:,j}$ denotes the j th column of the input matrix and $(\cdot)_g$ indicates the input matrix's rows within the set g . Group lasso [16] is the generalization of the standard lasso in which the entries of the coefficient matrix corresponding to the indicated groups are either included or excluded together from the regression model. Using this regularization, we encourage the samples to connect to the samples within the recommended groups (improving the connectivity) and simultaneously to disconnect from the groups that are recommended to avoid. Hence, the clustering output of the summarized local representations shows the groups that are recommended by the individual representations. Adding the group lasso regularization enforces the connections that are within the samples in each obtained group to *occur together*. Thus, the group lasso term indicates the recommended links by the local representations at the lower level.

This procedure can be easily extended to cases with a higher number of levels ($s > 2$). In general, at the bottom level of this hierarchical structure, the typical SSC is applied on the set of patches $\{X_j^s\}_{j=1}^{p^{s-1}}$. In the upper levels, the explained method (with two key steps) is individually applied on the gallery of patches to obtain coefficient matrices. This procedure is continued until the root level outputs the final *global* coefficient representation and the corresponding segmentation result.

The proposed framework not only computes the segmentation based on the global discriminant representation of the data but also takes advantage of robust local knowledge at each level of the hierarchy, leading to a robust discriminant representation.

4.3. Optimization

In this section, we introduce the optimization of the proposed convex problem in (7) using Alternating Direction Method of Multipliers (ADMM) [41]. First, we introduce auxiliary matrix $Z \in \mathbb{R}^{N \times N}$ and consider the following problem (The superscripts and subscripts in the optimization problem are not shown for ease of reading):

$$\begin{aligned} \min_{C \in \mathbb{R}^{N \times N}, Z \in \mathbb{R}^{N \times N}} & \|C\|_1 + \lambda_1 \|\Theta \odot C\|_1 \\ & + \lambda_2 \sum_{j=1}^N \sum_{g \in G} \|(C_{:,j})_g\|_2 + \frac{\mu}{2} \|X - XZ\|_F^2 \\ \text{s.t. } & Z = C - \text{diag}(C). \end{aligned} \quad (8)$$

Next, using penalty parameter $\beta > 0$ and matrix $\Delta \in \mathbb{R}^{N \times N}$ of Lagrange multipliers for the equality constraint, we obtain the following problem:

$$\begin{aligned} \mathcal{L}(C, Z, \Delta) = & \|C\|_1 + \lambda_1 \|\Theta \odot C\|_1 + \lambda_2 \sum_{j=1}^N \sum_{g \in G} \|(C_{:,j})_g\|_2 \\ & + \frac{\mu}{2} \|X - XZ\|_F^2 + \frac{\beta}{2} \|Z - (C - \text{diag}(C))\|_F^2 \\ & + \text{tr}(\Delta^T (Z - C + \text{diag}(C))), \end{aligned}$$

where $\text{tr}(\cdot)$ is the trace operator. Based on ADMM, this problem is optimized using the following iterative procedure:

With an abuse of notation, let $(C^{(k)}, Z^{(k)})$ be the optimization variables at iteration k and $\Delta^{(k)}$ be the Lagrangian multiplier at the same iteration:

- Find $Z^{(k+1)}$, by minimizing $\mathcal{L}$ with respect to Z , while the remaining variables are fixed by setting the derivative of $\mathcal{L}$ with respect to Z to zero as follows:

$$(\mu X^T X + \beta I) Z^{(k+1)} = \mu X^T X + \beta C^{(k)} - \Delta^{(k)}.$$

The matrix $Z^{(k+1)}$ is obtained by solving the aforementioned system of linear equations using approaches such as the conjugate gradient method.

- Find $C^{(k+1)}$, by minimizing $\mathcal{L}$ with respect to C , while the other variables are fixed. Note that we can rewrite this as follows:

$$\min_C ||W \odot C||_1 + \lambda_2 \sum_{j=1}^N \sum_{g \in G} ||(C_{:j})_g||_2 + \frac{\beta}{2} ||Z - C||_F^2 + \text{tr}(\Delta^T (Z - C)),$$

where $W \in \mathbb{R}^{N \times N}$ is a matrix such that its (i, j) th entry is defined by $W_{i,j} = 1 + \lambda_1 \Theta_{i,j}$. We can further simplify it as follows:

$$\min_C ||W \odot C||_1 + \lambda_2 \sum_{j=1}^N \sum_{g \in G} ||(C_{:j})_g||_2 + \frac{\beta}{2} ||C - (Z + \frac{\Delta}{\beta})||_F^2.$$

The constraint of $\text{diag}(C) = 0$ can be considered by projecting the solution of the aforementioned problem on this constraint. This problem is group-separable; thus, we consider the following $N \times n$ separate problems:

$$\min_{(C_{:j})_g} ||(W_{:j})_g \odot (C_{:j})_g||_1 + \lambda_2 ||(C_{:j})_g||_2 + \frac{\beta}{2} ||(C_{:j})_g - ((Z_{:j})_g + \frac{(\Delta_{:j})_g}{\beta})||_2^2, \quad \text{for } j = 1, \dots, N \text{ \& } g \in G.$$

By taking the (sub)gradient of the above problem, we obtain the following equation:

$$(W_{:j})_g \odot (S_{:j})_g + \lambda_2 \frac{(C_{:j})_g}{|| (C_{:j})_g ||_2} + \beta \left((C_{:j})_g - \left((Z_{:j})_g + \frac{(\Delta_{:j})_g}{\beta} \right) \right), \quad (9)$$

where $S \in \mathbb{R}^{N \times N}$ is a matrix that is defined as follows:

$$[(S_{:j})_g]_k = \begin{cases} -1 & [(C_{:j})_g]_k < 0 \\ [-1, 1] & [(C_{:j})_g]_k = 0 \\ 1 & [(C_{:j})_g]_k > 0 \end{cases}$$

$[(S_{:j})_g]_k$ denotes k -th element of the vector $(S_{:j})_g$. By setting (9) to zero,

$$(C_{:j})_g + \frac{\lambda_2}{\beta} \frac{(C_{:j})_g}{|| (C_{:j})_g ||_2} = (Z_{:j})_g + \frac{(\Delta_{:j})_g}{\beta} - \frac{(W_{:j})_g \odot (S_{:j})_g}{\beta}.$$

Hence, matrix C can be updated as follows:

$$(J_{:j})_g = \mathcal{T}_b \left((Z_{:j})_g + \frac{(\Delta_{:j})_g}{\beta} - \frac{(W_{:j})_g \odot (S_{:j})_g}{\beta}, \frac{\lambda_2}{\beta} \right), \quad \text{for } j \in [1, \dots, N] \text{ \& } g \in G$$

$$C^{(k+1)} = J - \text{diag}(J) \quad (10)$$

where:

$$\mathcal{T}_b(x, \rho) = \frac{x}{||x||_2} \max\{0, ||x||_2 - \rho\}.$$

Using this definition, (10) can be further summarized as follows:

$$(J_{:j})_g = \mathcal{T}_b \left(\mathcal{T} \left((Z_{:j})_g + \frac{(\Delta_{:j})_g}{\beta}, \frac{(W_{:j})_g}{\beta} \right), \frac{\lambda_2}{\beta} \right), \quad \text{for } j \in [1, \dots, N] \text{ \& } g \in G$$

$$C^{(k+1)} = J - \text{diag}(J). \quad (11)$$

Here $\mathcal{T}$ is the shrinkage-thresholding operator and is defined as follows:

$$\mathcal{T}(x, \rho) = (|x| - \rho)_+ \text{sign}(x)$$

In fact, this is equal to consecutively performing two shrinkage mappings corresponding to the weighted ℓ_1 and group lasso norms. This is consistent with the findings in [42].

- Find $\Delta^{(k+1)}$ by gradient ascent:

$$\Delta^{(k+1)} = \Delta^{(k)} + \beta(Z^{(k+1)} - C^{(k+1)})$$

These steps are repeated until convergence is met. Note that the problem in (8) is convex and consists of two separable blocks of variables; hence, the solution obtained by ADMM is guaranteed to be optimal. Algorithm 1 summarizes the updating rules for the ADMM implementation.

Algorithm 1: Optimizing (8) via ADMM.

Require: $X \in \mathbb{R}^{D \times N}$, G as the indices of data points in each cluster, parameters λ_1 , λ_2 , μ .

Ensure: Coefficient matrix C .

1: Initialization: $k = 0$, Initialize $C^{(0)}$, $Z^{(0)}$ and $\Delta^{(0)}$ to zero.

2: **while** some convergence criterion is not met **do**

3: Update $Z^{(k+1)}$ by solving the following system of linear equations:

$$(\mu X^T X + \beta I)Z^{(k+1)} = \mu X^T X + \beta C^{(k)} - \Delta^{(k)}$$

4: Update $C^{(k+1)}$ as:

$$(J_{:j})_g = \mathcal{T}_b(\mathcal{T}((Z_{:j}^{(k+1)})_g + \frac{(\Delta_{:j}^{(k)})_g}{\beta}, \frac{(W_{:j})_g}{\beta}), \frac{\lambda_2}{\beta}),$$

$$\text{for } j \in [1, \dots, N] \quad \&g \in G$$

$$C^{(k+1)} = J - \text{diag}(J).$$

5: Update $\Delta^{(k+1)}$ as:

$$\Delta^{(k+1)} = \Delta^{(k)} + \beta(Z^{(k+1)} - C^{(k+1)})$$

6: **end while**

5. Experiments

In this section, we demonstrate the effectiveness of LG-SSC in the presence of illumination effects, occlusion and disguise using three publicly available databases for face and object clustering: The Extended Yale B [43], AR [38] and COIL-20 [44] databases. The algorithm and all the experiments are implemented in MATLAB and run on a computer with an Intel Core i7-3770 CPU, 3.40 GHZ, 16GB RAM.

Evaluation metrics: To evaluate the quality of clustering, we use three well-known metrics, namely Accuracy (ACC), Normalized Mutual Information (NMI) [45] and Adjusted Rand Index (ARI) [46]. The accuracy of the clustering algorithm is calculated by the following formula:

$$\text{ACC}(\%) = \frac{\text{\#of correctly classified points}}{\text{total\#of points}} \times 100.$$

In NMI, the mutual information between the segmentation result and ground-truth clusters is calculated and scaled between 0 and 1. ARI computes the Rand Index Score and corrects it for chance. We multiply the values of NMI and ARI by 100 to facilitate comparison to ACC.

Compared algorithms: The performance of the proposed LG-SSC algorithm is compared to the following six subspace clustering algorithms: SSC [6], LRR [8], EDSC [47], S^3C [32], LRSC [26] and MG-SSC [17] (our preliminary work). The parameters of each algorithm are tuned for the best result and are reported in Table 1 for each database.

5.1. Extended yale b face data set

The Extended Yale B database [43] contains 2414 frontal-face images of 38 humans. There are 64 images, each 192×168 pixels in size, per individual. The face images were captured under various lighting conditions. Similar to [6], the images were downsampled to 48×42 pixels. For LG-SSC, we set $p = 4$ and $s = 2$. To study the effect of the number of clusters on the clustering performance, we implement two different settings of experiments: 1) We follow the setting utilized in [6] that has been implicitly specified as the general setting for reporting the performance of subspace clustering algorithms on this database over the past few years. In particular, for $n \in \{2, 3, 5, 8, 10\}$ clusters, the images of 38 subjects are divided into four groups of [1–10], [11–20], [21–30] and [31–38]. For $n \in \{2, 3, 5, 8\}$ clusters, all choices of possible different trials for each group is considered. For $n = 10$ subjects, only the first three groups are considered. The subspace clustering algorithms

Table 1
Parameters of the compared approaches.

Approach	Parameters		
	Extended Yale B	AR	Coil 20
LGSSC	$\alpha = 20, \lambda_1 = 10, \lambda_2 = 1$ $p = 4, s = 2$	$\alpha = 100, \lambda_1 = 10, \lambda_2 = 5$ $p = 4, s = 3$	$\alpha = 20, \lambda_1 = 10, \lambda_2 = 2$ $p = 4, s = 2$
MG-SSC	$\alpha = 20, p = 4, s = 3$	$\alpha = 100, p = 9, s = 3$	$\alpha = 20, p = 4, s = 2$
SSC	$\alpha = 20$	$\alpha = 100$	$\alpha = 20$
LRR	$\lambda = 0.009$	$\lambda = 0.095$	$\lambda = 0.0092$
EDSC	$\lambda_1 = 0.06, \lambda_2 = 0.01$ $\dim = 10, \alpha = 4$	$\lambda_1 = 0.06, \lambda_2 = 0.01$ $\dim = 10, \alpha = 4$	$\lambda_1 = 0.06, \lambda_2 = 0.01$ $\dim = 12, \alpha = 8$
S ³ C	$\gamma = 1, \alpha = 20$	$\gamma = 1, \alpha = 100$	$\gamma = 1, \alpha = 20$
LRSC	$\tau = 0.045, \alpha = 10^5$	$\tau = 0.07, \alpha = 0.1$	$\tau = 0.045, \alpha = 0.07$

Table 2
Average performance (expressed in %) on the Extended Yale B data set with different number of subjects. The best performance is indicated in bold.

Algorithm	LG-SSC	MG-SSC	SSC	LRR	EDSC	S ³ C	LRSC
2 subjects							
ACC	99.92	99.91	98.14	89.69	97.35*	99.48*	96.23
NMI	99.54	99.38	93.16	66.69	–	–	82.05
ARI	99.70	99.66	94.29	68.13	–	–	86.23
3 subjects							
ACC	99.42	99.87	96.70	79.09	96.35*	99.11*	93.55
NMI	99.04	99.43	92.75	59.61	–	–	81.06
ARI	98.99	99.62	92.61	53.22	–	–	82.61
5 subjects							
ACC	99.35	99.78	95.68	65.46	94.89*	98.49*	90.46
NMI	99.02	99.32	91.56	54.53	–	–	80.74
ARI	98.85	99.46	90.17	39.07	–	–	78.99
8 subjects							
ACC	99.41	99.72	94.13	59.02	93.93*	97.69*	76.36
NMI	98.54	99.35	90.58	56.34	–	–	70.71
ARI	98.65	99.38	86.44	36.27	–	–	59.15
10 subjects							
ACC	99.68	99.68	92.60	60.42	92.76*	97.19*	66.56
NMI	99.33	99.33	89.37	59.79	–	–	66.27
ARI	99.31	99.31	82.72	38.13	–	–	49.23

Table 3
Performance (expressed in %) on the Extended Yale B data set with different number of subjects. The best performance is indicated in bold.

#subjects	Metric	LG-SSC	MG-SSC	SSC	LRR	EDSC	S ³ C	LRSC
15	ACC	99.47	100	78.81	64.30	86.44	88.24	68.75
	NMI	99.10	100	79.14	66.18	88.97	91.25	71.57
	ARI	98.87	100	60.89	41.18	80.13	84.94	51.65
20	ACC	98.73	98.65	73.61	68.07	88.51	85.73	71.08
	NMI	98.05	98.02	76.67	70.68	90.79	91.15	75.49
	ARI	97.31	97.04	54.50	42.99	81.98	82.79	52.90
30	ACC	98.69	92.43	74.66	71.50	87.22	84.91	71.24
	NMI	98.09	94.57	77.69	75.35	91.22	90.48	75.19
	ARI	97.27	88.35	51.20	43.90	79.46	80.63	52.90
38	ACC	93.37	90.27	70.67	66.28	85.29	78.71	70.17
	NMI	94.91	91.47	75.44	72.19	90.08	86.78	75.19
	ARI	86.03	78.99	40.52	45.99	72.67	68.16	52.46

are applied to the corresponding subsets of images and the average ACC, NMI and ARI values over these subsets are reported in Table 2. The numbers with an * are taken from the corresponding papers. 2) For $n \in \{15, 20, 30, 38\}$, we simply select the first n images of the database and apply the subspace clustering algorithms. The values of the three metrics ACC, NMI and ARI for each subspace clustering algorithm are reported in Table 3.

The following observations were made:

- LG-SSC and MG-SSC significantly outperform other approaches in all cases. Specifically, for $n \geq 15$, the accuracy of LG-SSC is greater than 20% higher than that of SSC which is the basic foundation of this approach. The results indicate the efficiency of the hierarchical structure of LG-SSC in addressing severe illumination effects.

Table 4

Performance on the AR data set with different number of subjects. The best accuracy is indicated in bold.

#subjects	Metric	LG-SSC	MG-SSC	SSC	LRR	EDSC	S ³ C	LRSC
5	ACC	100	100	76.92	82.31	75.38	76.92	63.08
	NMI	100	100	62.79	71.57	64.74	64.04	47.87
	ARI	100	100	48.32	62.33	56.17	48.55	35.55
10	ACC	100	100	66.54	68.08	73.46	71.15	67.69
	NMI	100	100	65.34	72.27	81.29	73.19	68.64
	ARI	100	100	41.97	55.13	65.89	50.13	52.11
20	ACC	100	90.00	59.42	80.58	66.16	60.00	72.69
	NMI	100	92.82	68.86	86.14	75.61	69.25	77.42
	ARI	100	86.38	40.41	73.89	51.78	37.84	55.68
50	ACC	96.15	86.15	67.31	87.31	65.76	58.08	69.15
	NMI	97.71	90.33	78.21	91.79	79.95	72.90	79.76
	ARI	94.28	75.09	47.21	75.85	51.56	31.95	56.30
75	ACC	94.97	87.08	67.64	84.26	67.69	61.49	69.49
	NMI	96.93	91.23	82.11	91.39	83.69	78.17	81.55
	ARI	92.69	70.93	53.34	75.85	55.32	38.81	59.13
100	ACC	90.00	83.27	68.15	79.92	67.54	60.54	67.23
	NMI	93.74	90.98	82.87	90.01	82.77	79.89	82.34
	ARI	83.06	73.02	52.52	71.06	52.29	43.09	57.25

- MG-SSC performs slightly better than LG-SSC when the number of clusters is low. However, by increasing the number of clusters to 20, LG-SSC outperforms MG-SSC. This confirms that by gradually feeding summarized information of local patches into a hierarchical framework, the robustness increases, particularly in more challenging cases.
- The performance of LG-SSC is quite stable with respect to the number of clusters.
- The performance of SSC, LRR, S³C and LRSC significantly decreases as the number of clusters increases.
- Sparse-based approaches, in particular SSC and S³C, perform generally better compared to the low-rank-based approaches of LRR and LRSC.
- EDSC benefits from a specific post-processing step that tends to be different from the other six approaches. This post-processing of the affinity matrix plays a major role in the clustering accuracy. We noted that without this post-processing step, the quality of the obtained coefficient matrix of EDSC is similar to that of LRR and LRSC. This makes sense as EDSC regularizes the coefficient matrix using the Frobenius norm which shows similar characteristics to the nuclear norm in subspace clustering.

5.2. AR face data set

The AR database [38] contains frontal face images for 100 individuals (50 men and 50 women). There are 26 colored pictures collected for each person. The images include facial variations such as illumination changes, different expressions and facial disguises using sunglasses and scarves. Compared to Extended Yale B, this database is more challenging because of occlusions and fewer images per individual. We down sampled each image to 48×42 pixels and converted them to gray-scale. We set $p = 4$ and $s = 3$. The performance of LG-SSC with respect to the different number of clusters is compared to MG-SSC, SSC, LRR, LRSC, EDSC and S³C in Table 4. We made the following observations:

- The performance of nearly all approaches (except for LRR, MG-SSC and LG-SSC) degraded compared to that of the Extended Yale B database. This result is expected as the AR database is a more challenging database and with a higher number of clusters.
- LG-SSC shows better performance compared to that of the other approaches in all cases by a large margin. The robustness of LG-SSC is clearly evident for this database and the patch-based representations effectively guide the global self-expressive representation to a more robust clustering segmentation.
- LG-SSC consistently performs better than MG-SSC which further highlights the efficiency of LG-SSC in combining the information of local patches with calculation of robust global self-expressive representation.
- The occlusions degrade the performance of SSC, EDSC, S³C and LRSC even in the simplest case of five clusters. This shows the sensitivity of these approaches to occlusions and continuous corruptions.
- LRSC attempts to recover a clean dictionary by optimizing a nonconvex problem; however, this approach assumes that the data are contaminated by sparse error which is clearly violated in this database.
- The post-processing step of EDSC cannot improve the performance in this case because of the severely corrupted global representation that makes *correction* difficult, if not impossible.
- The third best performance is achieved by LRR. The LRR approach is considered as the extension of RPCA [48] for the union of subspaces. For data with complex noise structures, dense representations based on nuclear norm regularization appears to be more suitable compared to sparse representations.

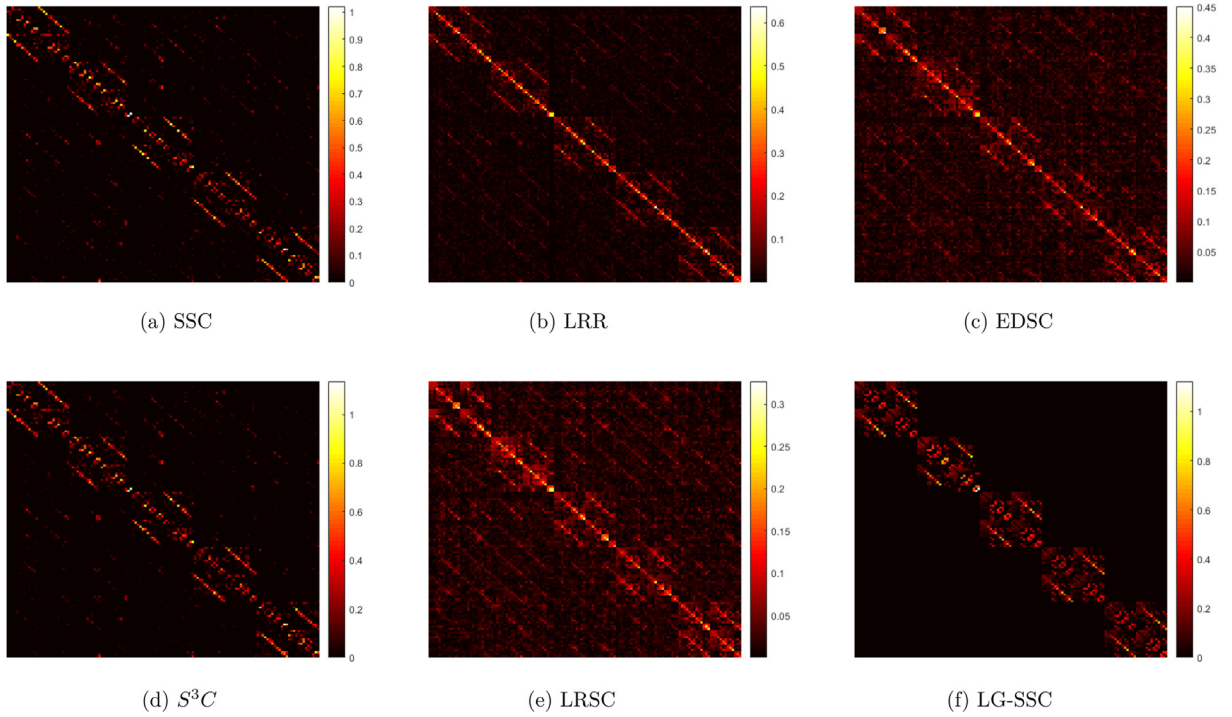


Fig. 6. Coefficient matrices of (a) SSC, (b) LRR, (c) EDSC, (d) S^3C , (e) LRSC and (f) LG-SSC for the first 5 individuals of AR database.

Table 5

Performance on the COIL-20 data set. The best accuracy is indicated in bold.

Metric	LG-SSC	MG-SSC	SSC	LRR	EDSC	S^3C	LRSC
ACC	89.58	78.26	78.68	54.86	84.51	74.86	64.09
NMI	95.34	87.92	90.39	70.03	93.52	88.28	72.29
ARI	85.53	72.30	74.89	42.19	81.54	66.88	52.22

For better visualization and comparison, the coefficient matrices corresponding to each algorithm for the first five individuals are plotted in Fig. 6. The block diagonal structure in LG-SSC's coefficient matrix is clearly evident. Two major components of a successful subspace clustering algorithm, namely, (i) subspace preserving connections and (ii) strong connectivity within each subspace, are present in LG-SSC. However, the coefficient matrices of other approaches are contaminated because of illumination effects and disguises which affect the clustering performance. Note that MG-SSC does not output a final coefficient matrix; hence, the comparison is completed against the other five approaches.

5.3. COIL-20 data set

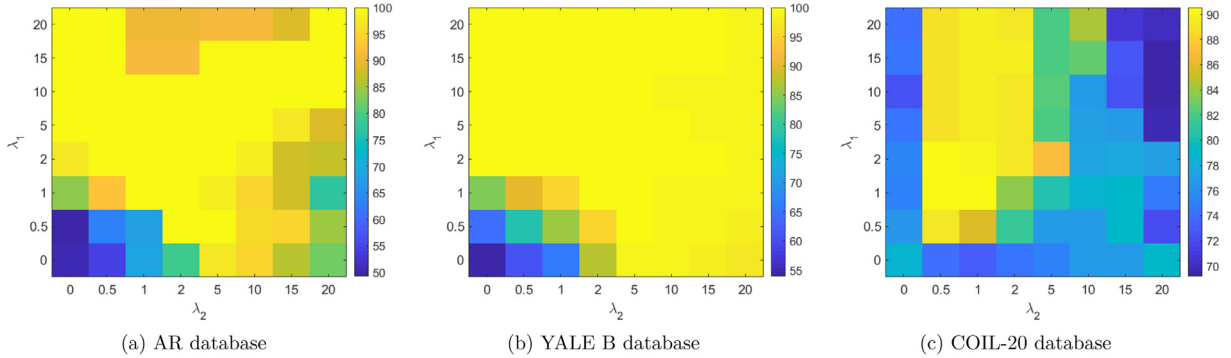
The Columbia Object Image Library(COIL-20) [44] contains 1440 gray-scale images of 20 objects in different poses. The images of the objects were taken by placing objects on a turntable against a black background. There are 72 images per object, each 128×128 pixels in size. We downsampled each image to 32×32 pixels. Although this database is clean, with no noise or occlusions, it is still interesting to observe the performance of LG-SSC in addressing data from clean subspaces. We divided each image into nine overlapping patches³ and set $s = 2$. This is done because of the fact that for object images, the patches nearer the center of the image contain more meaningful information compared to that of the other patches. The performance of our proposed LG-SSC is compared to other subspace clustering algorithms as presented in Table 5. Thus, we can make the following conclusions:

- Although MG-SSC fails to increase the accuracy in addressing clean object images, LG-SSC is able to increase the clustering accuracy by greater than 10%. This suggests that locally guided self-expressiveness might improve the quality of clustering in the challenging case of close subspaces.
- EDSC benefits from the post-processing of the coefficient matrix and shows the second best performance.

³ The patch sizes are equal to one-half of the image size; hence, we considered this a special case of $p = 4$.

Table 6Accuracy of LG-SSC with respect to different values for levels (s) and number of blocks (p).

	2×2			3×3			4×4		
	$s=2$	$s=3$	$s=4$	$s=2$	$s=3$	$s=4$	$s=2$	$s=3$	$s=4$
AR (10)	60.38	100	99.61	86.54	100	99.62	100	100	22.31
YALE B (10)	100	100	99.84	99.84	100	99.68	99.68	100	18.13
COIL-20	89.44	77.84	–	77.22	74.16	–	74.37	72.5	–

**Fig. 7.** Effect of λ_1 and λ_2 on the accuracy of LG-SSC for (a) AR database, (b) Extended Yale B database and (c) COIL-20 database.

- Sparse-based approaches (SSC and S^3C) have a higher performance compared to that of the low-rank based algorithms (LRR and LRSC). In general, sparse-based approaches have stronger theoretical guarantees compared to those of the low-rank based alternatives. Hence, for clean data sets, they are typically expected to perform better.

5.4. Parameter analysis

In LG-SSC, there are three parameters that are used in the optimization problem (7) that control the trade-off between four qualities: (i) Sparsity, (ii) ignoring cannot-links, (iii) respecting recommended-links, and (iv) self-expressive reconstruction error. Following the methodology in [6], we set the regularization parameter μ as $\frac{\alpha}{\max_{j \neq i} |x_i^T x_j|}$ where $\alpha > 1$ is tuned for each dataset. In this paper, we set this parameter to values that have been commonly used and reported in subspace clustering literature.

The behavior of LG-SSC with respect to λ_1 and λ_2 is empirically validated on all three databases (AR, Extended Yale B and COIL-20). We consider the clustering performance for the first 10 subjects of AR and Yale B and 20 objects of COIL-20. The clustering accuracy with respect to different values of these parameters is shown for each database in Fig. 7. It can be seen that for values of $\lambda_1 \in [2: 10]$ and $\lambda_2 \in [0.5: 2]$, the accuracy is quite stable in all three cases. In particular, Yale B is the least sensitive to the values of λ_1 and λ_2 and COIL-20 is the most sensitive database. For the Yale B database, as long as λ_1 and λ_2 are not too small, the accuracy is nearly 100%. Interestingly, by setting the $\lambda_1 \in [2: 20]$ and $\lambda_2 = 0$, the accuracy remains 100%. This suggests that for this database, the “cannot-links” information is more important compared to the “recommended-links” information. However, for COIL-20, the recommended-links information plays an important role in increasing the accuracy of the basic SSC (which is approximately 78.68%).

We also evaluate the effect of the number of patches (p) and number of levels (s) on the clustering accuracy. We consider $s \in \{2, 3, 4\}$ and $p \in \{4, 9, 16\}$. The performance of LG-SSC for different values of s and p for the first 10 subjects of Yale B and AR and 20 objects of COIL-20 are reported in Table 6. For the AR database, the clustering accuracy is not affected by the patch size as long as $s = 3$. Because for $s = 2$, $p = 4$ and $s = 2$, $p = 9$, the patches at the coarse level are not robust themselves and they contain occluded parts of the image, the robustness is not transferred to the fine scale. This can be confirmed by considering the case where $s = 2$ and $p = 16$. In this case, the accuracy is 100% because the patches at the second level are sufficiently small to contain robust discriminant information. By increasing the number of levels to four and number of patches to 16 ($s = 4$ and $p = 16$), the accuracy significantly decreases to 22.31%. In this case, the patches at the last level are very small and thus neither robust nor discriminant information can be fed into the upper levels. For Yale B, which is relatively less challenging compared to AR, the accuracy is nearly 100% in all cases except for the case in which $s = 4$ and $p = 16$ where the patches intuitively become very small. For this database, $s = 2$ is sufficient to increase the robustness in response to illumination variations. Interestingly, the best accuracy for the COIL-20 database is achieved only for $s = 2$ and $p = 4$. Note that in this case, each image is 32×32 pixels in size and hence we do not consider $s = 4$. For object clustering, the edges play a critical role; thus, the patches should be considered such that they contain sufficient edge information for accurate clustering.

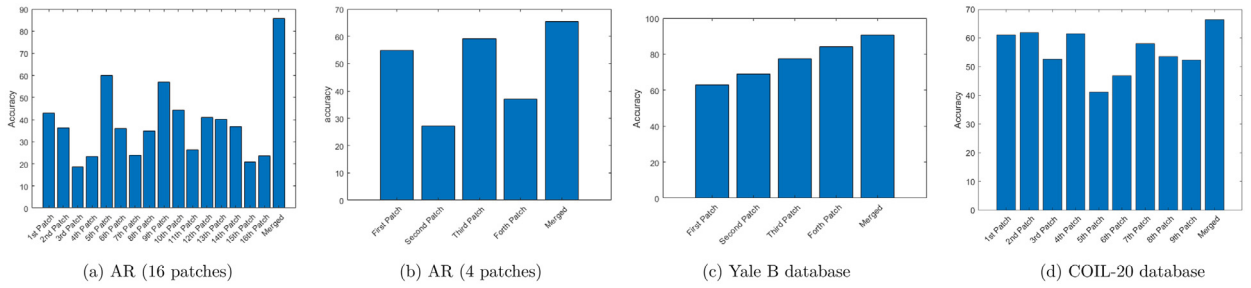


Fig. 8. Clustering accuracy of individual local patches and the corresponding merged low-dimensional embedding on (a) AR (dividing to 16 patches), (b) AR (dividing to 4 patches), (c) Extended Yale B and (d) COIL-20.

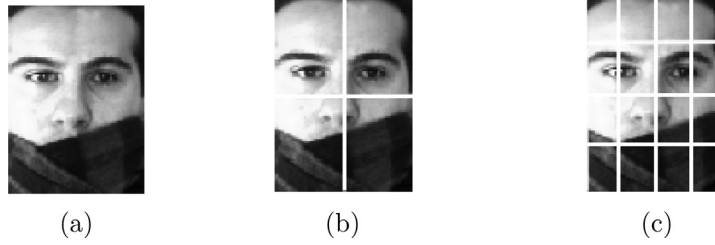


Fig. 9. The generated patches of a selected sample in different levels of the hierarchical framework. (a) selected sample in the first (fine) level, (b) 4 patches of the selected sample in the second level, (c) 16 patches of the selected sample in the third level.

5.5. Neither global nor local

In this section, we discuss the role of the multi-layer graph fusion approach and emphasize the point that nearly neither individual local patches nor global data might lead to a robust discriminant representation. However, merging the local representations using their low-dimensional embedding on a Grassmann manifold can provide a summary representation that highlights the information that the majority of local representations tend to agree on. Hence, the clustering accuracy of each local patch for the three databases (all samples for each database are considered) are plotted in Fig. 8. For the AR dataset, two cases of $p = 4$ and $p = 16$ are considered. As can be seen, in the case of $p = 16$, none of the local patches reach an accuracy greater than 65%. However, applying a k-means to the summarized low-dimensional embedding of these 16 patches leads to an accuracy near 85% (first column from right). LG-SSC further increases this robustness to near 90%. When $p = 4$, the same observation presented in Table 6 is repeated and not only do none of the local patches have an accuracy greater than 65% but also the merged information of these four local patches does not significantly increase the performance. As previously mentioned, this is because none of these patches have robust representations. In Yale B, the coefficient matrix corresponding to the 4th patch has the highest clustering accuracy and LG-SSC improves this accuracy without being affected by other patches, e.g., the 1st patch. For the COIL-20 database, not only do the local patches not lead to high clustering accuracy but also the merged low-dimensional embedding does not significantly increase the clustering accuracy either. However, LG-SSC still increases the clustering accuracy because the merged information guides the global sparse self-expressive representation and the local representations provide sufficient information to avoid miss-clustering of closely related objects.

5.6. How does LG-SSC improve clustering accuracy?

In this section, we present detailed illustrations regarding how LG-SSC can improve robustness using two examples. We first examine the role of the proposed algorithm's hierarchical structure in enhancing subspace preserving connections for occluded samples. Next, we demonstrate how robustness and separability of clusters gradually increases through the bottom-up process of LG-SSC.

5.6.1. Subspace preserving coefficients despite occlusion and disguise

We validate LG-SSC's ability to address occlusion using images of five individuals from the AR database. The chosen subset contains 130 images with 26 images per individual. Similar to previous sections, we divide the samples into a hierarchical structure with three levels ($s = 3$) and partition every patch into four patches at each level ($p = 4$). We consider a sample from the first cluster which is occluded with a scarf and is shown in Fig. 9 (a). Note that samples with indices from 1 to 26 belong to the same cluster as that of the selected image. For this image, we have four patches at the second level (shown in Fig. 9 (b)) and 16 patches at the third level (shown in Fig. 9 (c)). The selected sample's coefficient vectors (obtained from the symmetrized coefficient matrices) corresponding to the 16 patches at the third level are plotted as shown in

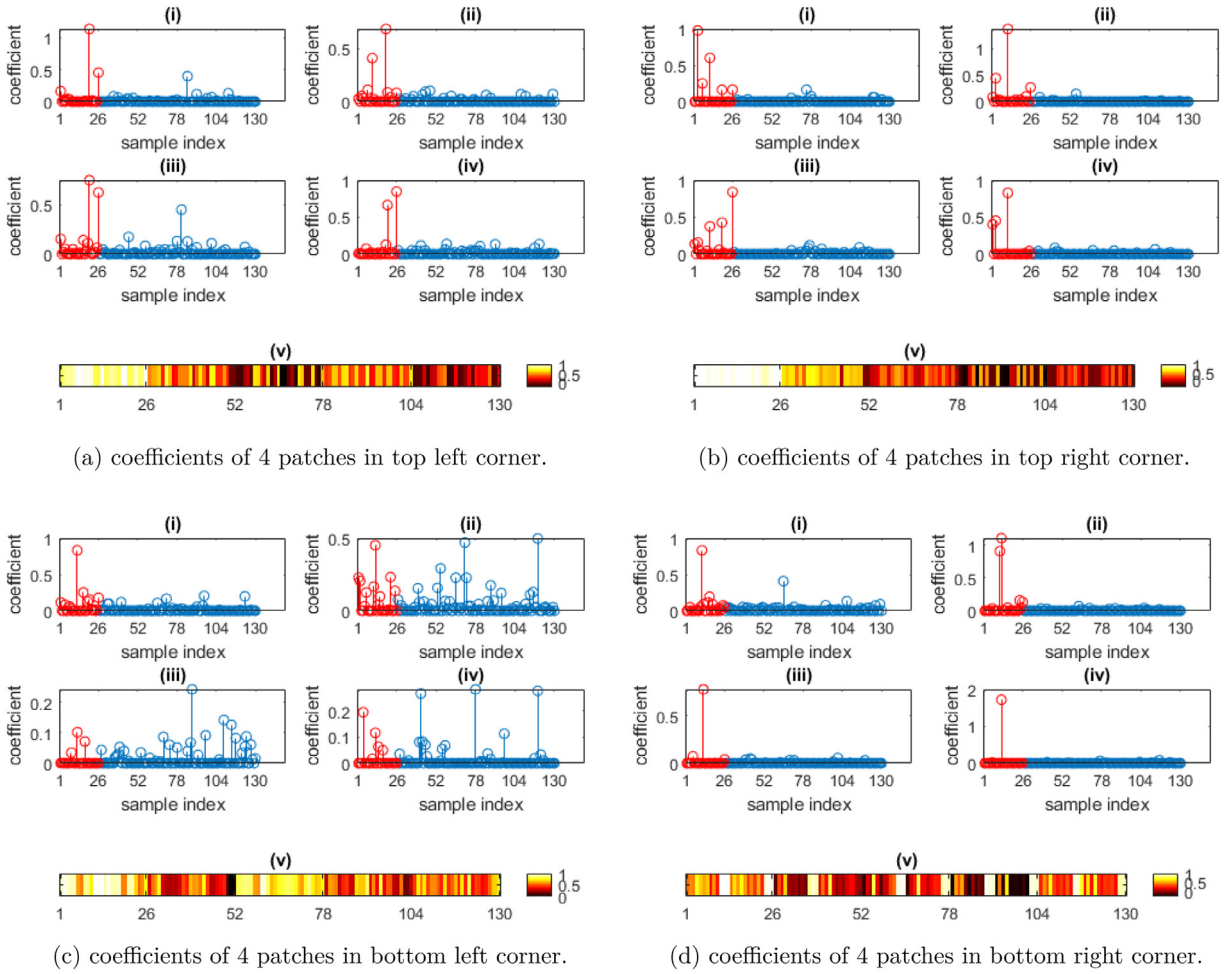


Fig. 10. Representation coefficients of 16 patches of an occluded sample in the *third* level of proposed hierarchical LG-SSC and the corresponding summarized vectors in the spectral similarity matrix.

Fig. 10 (a) - (d) such that the coefficients in (a), (b), (c) and (d) correspond to the four patches in the top left, four patches in the top right, four patches in the bottom left and four patches in the bottom right corner of the image, respectively. The red entries (coefficients) correspond to the samples of the same person (the correct cluster). For each four-patch coefficient, the corresponding summarized vector in the spectral similarity matrix (introduced in (4)) is presented at the bottom of each plot (part (v) of each subplot). The same illustration for the four patches at the second level is shown in Fig. 11.

Our observations are as follows:

- The coefficients in parts (a) and (b) of Fig. 10 correspond to *clean* partitions of the selected image (note that the chosen sample is only occluded with a scarf). These coefficients are subspace preserving (particularly in (b)) and the dominant coefficients mostly correspond to samples from the same subspace.
- The summarized spectral representation in (b) is recommending strong connections within the same subspace (samples with indices from 1 to 26 belong to the same subspace as that of the selected image). The spectral representation in (a) is fairly similar to that of (b) as well.
- The occlusion effect is significantly visible in the coefficients of the local patches shown in Fig. 10 (c). These coefficients correspond to occluded regions of the image and the dominant coefficients are clearly from the wrong clusters. As expected, the spectral vector from the summarized representation is mistakenly recommending some connections with samples from different clusters and rejecting some connections with samples from the same clusters. However, the obtained spectral vector is not completely baseless and it embeds correct information regarding some connections as well. Hence, although this spectral vector contains wrong grouping information, in general it is emphasizing the connections within the same subspace that the majority of the individual coefficient vectors correctly recommend.
- The coefficients shown in Fig. 10 (d) belong to the other corrupted region of the sample; however, interestingly for these patches, the dominant coefficients are *subspace preserving* and they mostly correspond to the samples from the

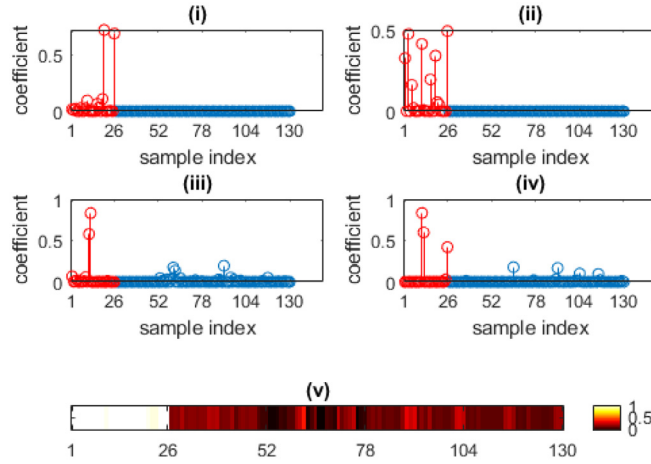


Fig. 11. Representation coefficients of 4 patches of an occluded sample in the *second* level of proposed hierarchical LG-SSC and the corresponding summarized vector in the spectral similarity matrix.

same subspace. However, these dominant entries correspond to the other *similarly* corrupted samples within the same subspace;⁴ hence, the lack of connections with the other clean samples from the same subspace leads to the wrong summarized spectral vector (shown in part (v)). Thus, despite subspace-preserving coefficients, this vector recommends many wrong connections.

- The four coefficient vectors at the second level are obtained using the summarized spectral vectors at the third level and are plotted as shown in Fig. 11. As expected, the coefficient vectors in parts (i) and (ii), which correspond to the top patches of the image, are clearly subspace preserving (with all entries from the other clusters equal to zero). The coefficients of the occluded regions in parts (iii) and (iv) are mostly subspace preserving, with few entries of samples from different clusters having small values. However, it is evident that the coefficients are in general highlighting correct connections.
- The final summarized spectral vector of the four patches at the second level is shown at the bottom of Fig. 10 (part (v)). Obviously, the vector is emphasizing correct connections from the same subspace and rejecting all the wrong connections with samples from different clusters. This confirms the efficiency of the proposed hierarchical LG-SSC in improving robustness for occluded samples.

5.6.2. Hierarchical representation for improving clustering accuracy

We reuse the selected subset of images in the previous Section (Section 5.6.1) and illustrate the gradual improvement in robustness and clustering accuracy during the bottom-up stage of the proposed hierarchical algorithm. To this end, we visualize the spectral embedding of the obtained coefficient matrices in each level of the algorithm using the well-known t-SNE method [49]. The proposed LG-SSC with parameters $p = 4$ and $s = 3$ is applied to the selected subset of images. Sixteen coefficient matrices of the gallery of 16 patches at the third level are calculated as $(\{C_{\ell_1^k}^{(3)}\}_{k=1}^4, \{C_{\ell_2^k}^{(3)}\}_{k=1}^4, \{C_{\ell_3^k}^{(3)}\}_{k=1}^4$ and $\{C_{\ell_4^k}^{(3)}\}_{k=1}^4)$.

The top five eigenvectors corresponding to their Laplacian matrices are computed as $(\{U_{\ell_1^k}^{(3)}\}_{k=1}^4, \{U_{\ell_2^k}^{(3)}\}_{k=1}^4, \{U_{\ell_3^k}^{(3)}\}_{k=1}^4$ and $\{U_{\ell_4^k}^{(3)}\}_{k=1}^4)$. The two-dimensional visualizations of these eigenvectors are plotted as shown in Fig. 12 (a)–(d) using t-SNE. The embedded samples from the same cluster are shown with the same color. We can see that nearly no coefficient matrices are efficiently separating the clusters. This indicates that the local representations have deficient discriminant information for perfect clustering.

The two-dimensional representation of the spectral embedding of the four coefficient matrices at the second level ($C_{\ell_1^1}^{(2)}$, $C_{\ell_2^1}^{(2)}$, $C_{\ell_3^1}^{(2)}$ and $C_{\ell_4^1}^{(2)}$) are shown in Fig. 12 parts (e)–(h). It can be inferred that, in general, the representations are more separable compared to those in the third layer and are less mixed and more accurate. Interestingly, the gradual flow of robustness through the hierarchy can be observed. Note that the clusters in the representation shown in Fig. 12 (f) are perfectly separated. Hence, the corresponding patch contains sufficient discriminant information for separating the clusters and in fact this representation belongs to the gallery of patches in the top right corner of the images which are occluded only by sunglasses.

The spectral embedding of the final coefficient matrix at the top of the hierarchy (first level) is also mapped to a two-dimensional space using t-SNE [49] and is plotted in Fig. 13. Here, the perfectly separated clusters are clearly visible. The

⁴ In particular, these entries correspond to other samples from the same person that are occluded with scarf but under different illumination variations.



Fig. 12. 2D visualizations of coefficient matrices corresponding to samples from 5 clusters of AR database. (a)–(d) represent embeddings of 16 local patches in the third level, (e)–(h) represent embeddings of 4 local patches in the second level.

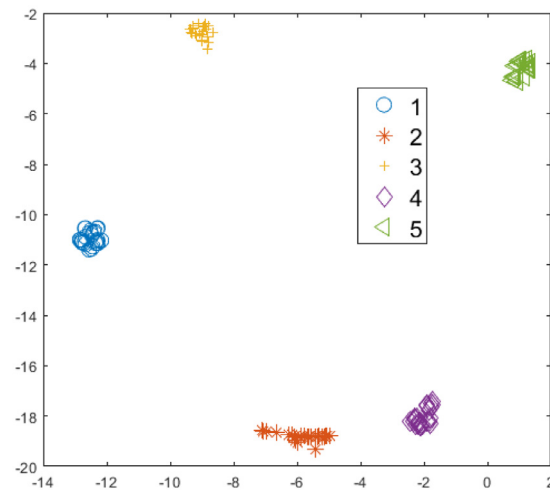


Fig. 13. 2D visualization of final (global) coefficient matrix of samples from 5 clusters of AR database.

Table 7

Accuracy of Majority Voting based merging with respect to different numbers of local blocks.

	2×2	3×3	4×4
Yale B (10)	72.34	63.28	61.72
AR (10)	53.85	61.54	74.23
Coil-20	64.24	45.90	44.65

gradual increase in the separation of the data from different clusters is noticeable from the local representations shown in Fig. 12 to the global representation shown in Fig. 13. This confirms the efficiency of the hierarchical structure of LG-SSC in balancing the robustness of local patches and discriminant nature of those that are global.

5.7. Majority voting vs LG-SSC

A simple approach for exploiting robustness of local patches is to divide the image data into blocks and independently cluster each partition (here using SSC). The obtained clustering results can be aggregated using majority voting. However, in the unsupervised setting, there is no connection between a group and a specific label. A simple practical approach is to consider the obtained clustering labels of a gallery of specific patches (for example, the top left corner patches) as the ground truth label and use the Hungarian algorithm [50] to match the labels of the remainder of the data in other blocks with the chosen ground truth. The clustering accuracy of this approach for different numbers of patches is reported in Table 7 for the three databases. There is a striking difference between the performance of the majority-voting-based algorithm and the proposed LG-SSC (for reference, the accuracy of LG-SSC is reported in Table 6). Clearly, majority voting is not able to improve robustness and clustering accuracy. This suggests that LG-SSC not only can reduce the sensitivity of clustering accuracy to patch sizes (note the consistent results of LG-SSC compared to that of majority voting) but also the proposed framework using graph fusion more efficiently merges the shared connectivity of the individual local representations.

6. Conclusion

In this paper, we demonstrated the importance of local representations in improving the robustness of self-expressive-based subspace clustering approaches. The proposed hierarchical approach bridges the gap between robust local representations and discriminant global alternatives to obtain a robust discriminant self-expressive representation for the input data. This approach consists of two major steps: 1) Efficiently summarizing local based representations using low-rank embedding on a Grassmann manifold to obtain cannot-links and recommended-links that local patches agree upon. 2) Employing this summarized information in the optimization problem for calculating robust self-expressive representation at each level using weighted group lasso regularization. The robustness of the proposed approach in response to occlusion and complex noise was confirmed by experimental results.

Although the core of the proposed algorithm is based on SSC, the suggested framework has the potential to be unified with other subspace clustering methods and extensions of SSC as well. Moreover, we believe robustness can be further improved by considering the consistency of the information in the individual local representations. In particular, the patches that correspond to representations that are not consistent with other representations (for example, based on the distance measure on the Grassmann manifold) can be abandoned to calculate the self-expressive representation. These remain interesting subjects for future work.

Declaration of competing interest

This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.

CRedit authorship contribution statement

Maryam Abdolali: Conceptualization, Methodology, Software, Writing - original draft. **Mohammad Rahmati:** Writing - review & editing, Supervision.

References

- [1] M.A. Carreira-Perpinán, A review of dimension reduction techniques, Department of Computer Science. University of Sheffield. Tech. Rep. CS-96-09 9 (1997) 1–69.
- [2] I.T. Jolliffe, Principal component analysis and factor analysis, in: Principal component analysis, Springer, 1986, pp. 115–128.
- [3] R. Vidal, Subspace clustering, IEEE Signal Process. Mag. 28 (2) (2011) 52–68.
- [4] R. Vidal, Y. Ma, S. Sastry, Generalized principal component analysis (GPCA), IEEE Trans. Pattern Anal. Mach. Intell. 27 (12) (2005) 1945–1959.
- [5] M.E. Tipping, C.M. Bishop, Mixtures of probabilistic principal component analyzers, Neural Comput. 11 (2) (1999) 443–482.
- [6] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.
- [7] 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.

- [8] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (1) (2013) 171–184.
- [9] U. Von Luxburg, A tutorial on spectral clustering, *Stat. Comput.* 17 (4) (2007) 395–416.
- [10] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, S. Yan, Robust and efficient subspace segmentation via least squares regression, in: *European conference on computer vision*, Springer, 2012, pp. 347–360.
- [11] M. Abdolali, M. Rahmati, Robust subspace clustering for image data using clean dictionary estimation and group lasso based matrix completion, *J. Visual Commun. Image Represent.* (2019).
- [12] L. Luo, L. Chen, J. Yang, J. Qian, B. Zhang, Tree-structured nuclear norm approximation with applications to robust face recognition, *IEEE Trans. Image Process.* 25 (12) (2016) 5757–5767.
- [13] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, *IEEE Trans. Pattern Anal. Mach. Intell.* 31 (2) (2008) 210–227.
- [14] J. Lai, X. Jiang, Classwise sparse and collaborative patch representation for face recognition, *IEEE Trans. Image Process.* 25 (7) (2016) 3261–3272.
- [15] P. Turaga, A. Veeraraghavan, R. Chellappa, Statistical analysis on Stiefel and Grassmann manifolds with applications in computer vision, in: *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.
- [16] J. Friedman, T. Hastie, R. Tibshirani, A note on the group lasso and a sparse group lasso (2010). arXiv: 1001.0736.
- [17] M. Abdolali, M. Rahmati, From local to global subspace clustering for image data, in: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 3787–3791.
- [18] T. Zhang, A. Szlam, G. Lerman, Median k-flats for hybrid linear modeling with many outliers, in: *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, IEEE, 2009, pp. 234–241.
- [19] M. Soltanolkotabi, E.J. Candes, et al., A geometric analysis of subspace clustering with outliers, *Ann. Stat.* 40 (4) (2012) 2195–2238.
- [20] Y.-X. Wang, H. Xu, Noisy sparse subspace clustering, *J. Mach. Learn. Res.* 17 (1) (2016) 320–360.
- [21] X. Li, Q. Lu, Y. Dong, D. Tao, Robust subspace clustering by cauchy loss function, *IEEE Trans. Neural Netw. Learn. Syst.* (2018).
- [22] J. Yao, X. Cao, Q. Zhao, D. Meng, Z. Xu, Robust subspace clustering via penalized mixture of gaussians, *Neurocomputing* 278 (2018) 4–11.
- [23] X. Guo, X. Xie, G. Liu, M. Wei, J. Wang, Robust low-rank subspace segmentation with finite mixture noise, *Pattern Recognit.* 93 (2019) 55–67.
- [24] R. He, Y. Zhang, Z. Sun, Q. Yin, Robust subspace clustering with complex noise, *IEEE Trans. Image Process.* 24 (11) (2015) 4001–4013.
- [25] X. Zhang, H. Sun, Z. Liu, Z. Ren, Q. Cui, Y. Li, Robust low-rank kernel multi-view subspace clustering based on the Schatten p-norm and coreentropy, *Information Sciences* 477 (2019) 430–447.
- [26] R. Vidal, P. Favaro, Low rank subspace clustering (Lrsc), *Pattern Recognit. Lett.* 43 (2014) 47–61.
- [27] C. Lu, J. Feng, Z. Lin, T. Mei, S. Yan, Subspace clustering by block diagonal representation, *IEEE Trans. Pattern Anal. Mach. Intelligence* 41 (2) (2019) 487–501.
- [28] C. Lu, J. Feng, Z. Lin, S. Yan, Correlation adaptive subspace segmentation by trace lasso, in: *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1345–1352.
- [29] H. Hu, Z. Lin, J. Feng, J. Zhou, Smooth representation clustering, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3834–3841.
- [30] R. Heckel, H. Bölcskei, Robust subspace clustering via thresholding, *IEEE Trans. Inf. Theory* 61 (11) (2015) 6320–6342.
- [31] Q. Qiu, G. Sapiro, Learning transformations for clustering and classification, *J. Learn. Res.* 16 (1) (2015) 187–225.
- [32] C.-G. Li, C. You, R. Vidal, Structured sparse subspace clustering: a joint affinity learning and subspace clustering framework, *IEEE Trans. Image Process.* 26 (6) (2017) 2988–3001.
- [33] P. Zhu, L. Zhang, Q. Hu, S.C. Shiu, Multi-scale patch based collaborative representation for face recognition with margin distribution optimization, in: *European Conference on Computer Vision*, Springer, 2012, pp. 822–835.
- [34] M. Abdolali, M. Rahmati, Multiscale decomposition in low-rank approximation, *IEEE Signal Process. Lett.* 24 (7) (2017) 1015–1019.
- [35] C.K. Reddy, B. Vinzamuri, A survey of partitional and hierarchical clustering algorithms, in: *Data Clustering*, Chapman and Hall/CRC, 2018, pp. 87–110.
- [36] E. Rubio, O. Castillo, F. Valdez, P. Melin, C.I. Gonzalez, G. Martinez, An extension of the fuzzy possibilistic clustering algorithm using type-2 fuzzy logic techniques, *Adv. Fuzzy Syst.* 2017 (2017).
- [37] E. Rubio, O. Castillo, Designing type-2 fuzzy systems using the interval type-2 fuzzy c-means algorithm, in: *Recent Advances on Hybrid approaches for designing Intelligent systems*, Springer, 2014, pp. 37–50.
- [38] A.M. Martinez, The ar face database, CVC Technical Report 24 (1998).
- [39] R. Basri, D.W. Jacobs, Lambertian reflectance and linear subspaces, *IEEE Trans. Pattern Anal. Mach. Intell.* (2) (2003) 218–233.
- [40] X. Dong, P. Frossard, P. Vandergheynst, N. Nefedov, Clustering on multi-layer graphs via subspace analysis on grassmann manifolds, *IEEE Trans. Signal Process.* 62 (4) (2014) 905–918.
- [41] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers, *Found. Trends Mach. Learn.* 3 (1) (2011) 1–122.
- [42] R. Chartrand, B. Wohlberg, A nonconvex admm algorithm for group sparsity with sparse groups, in: *2013 IEEE international conference on acoustics, speech and signal processing*, IEEE, 2013, pp. 6009–6013.
- [43] A.S. Georgiades, P.N. Belhumeur, D.J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. Mach. Intell.* (6) (2001) 643–660.
- [44] S.A. Nene, S.K. Nayar, H. Murase, et al., Columbia object image library (coil-20) (1996).
- [45] N.X. Vinh, J. Epps, J. Bailey, Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance, *J. Mach. Learn. Res.* 11 (Oct) (2010) 2837–2854.
- [46] L. Hubert, P. Arabie, Comparing partitions, *J. Classif.* 2 (1) (1985) 193–218.
- [47] P. Ji, M. Salzmann, H. Li, Efficient dense subspace clustering, in: *IEEE Winter Conference on Applications of Computer Vision*, IEEE, 2014, pp. 461–468.
- [48] E.J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis? *J. ACM (JACM)* 58 (3) (2011) 11.
- [49] L. Maaten, G. Hinton, Visualizing data using t-sne, *J. Mach. Learn. Res.* 9 (Nov) (2008) 2579–2605.
- [50] H.W. Kuhn, The hungarian method for the assignment problem, *Naval Res. Logist. Q.* 2 (1–2) (1955) 83–97.