On Unifying Multi-View Self-Representations for Clustering by Tensor Multi-Rank Minimization

Yuan Xie · Dacheng Tao · Wensheng Zhang · Yan Liu · Lei Zhang · Yanyun Qu

Abstract In this paper, we address the multi-view subspace clustering problem. Our method utilizes the circulant algebra for tensor, which is constructed by stacking the subspace representation matrices of different views and then rotating, to capture the low rank tensor subspace so that the refinement of the view-specific subspaces can be achieved, as well as the high order correlations underlying multi-view data can be explored. By introducing a recently proposed tensor factorization, namely tensor-Singular Value Decomposition (t-SVD) [16], we can impose a new type of low-rank tensor constraint on the rotated tensor to ensure the consensus among multiple views. Different from traditional unfolding based tensor norm, this low-rank tensor constraint has optimality properties similar to that of matrix rank derived from SVD, so the complementary information can be explored and propagated among all the views more thoroughly and effectively. The established model, called t-SVD based Multi-view Subspace Clustering (t-SVD-MSC), falls into the applicable scope of augmented Lagrangian method, and its minimization problem can be efficiently solved with theoretical convergence guarantee and relatively low computational complexity. Extensive experimental testing on eight challenging image dataset shows that the proposed method has achieved highly competent objective performance compared to several state-of-the-art multi-view clustering methods.

Y. Xie
the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
Tel.: +86-13716206758
E-mail: yuan.xie@ia.ac.cn

D. Tao
the Center for Quantum Computation & Intelligent Systems and the Faculty of Engineering & Information Technology, University of Technology, Sydney, Australia
E-mail: dacheng.tao@uts.edu.au

W. Zhang
the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
E-mail: wensheng.zhang@ia.ac.cn

L. Zhang
the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
E-mail: cslzhang@comp.polyu.edu.hk

Y. Liu
the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China
E-mail: csyliu@comp.polyu.edu.hk

Y. Qu
the School of Information Science and Technology, Xiamen University, Fujian, China
E-mail: yyqu@xmu.edu.cn

1 Introduction

Many scientific data have heterogeneous features, which are collected from diverse domains or generated from various feature extractors. For example, in real-world applications, datasets are naturally comprised of multiple views: a) webpages can be represented by using both page-text and hyperlinks pointing to them; b) images can be described by different kinds of features, such as color, edge and texture. Each type of feature is referred to as a particular view, and combining multiple views of dataset for data analysis has been a popular practice for improving performance. Commonly, the success of the multi-view learning stems from the following two **principles**: (1) Consensus principle, which aims to maximize the agreement on multiple distinct views; (2) Complementary principle, which means that each view of the data may contain some knowledge that other views do

not have; therefore, multiple views can be employed to comprehensively and accurately describe the data. For a comprehensive review of multi-view learning, please refer to [1].

In this work, we mainly focus on multi-view clustering, where the absence of a groundtruth to guide the learning process makes the underlining task much harder. **Basic assumptions of the multi-view clustering**: (1) The feature in each individual view are sufficient to discover most of the clustering information; (2) The feature in each individual view might be corrupted by noise, *i.e.,* these noise might result in a small portion of samples being assigned to wrong clusters. As different views are different representations of the same set of instances, we aim to capture the relationship among multiple views to improve the clustering results generated by the limited information from a single view.

Our work is motivated by self-representation based subspace clustering (*i.e.,* low-rank representation (LRR) [21]) and a new type of factorization for tensor and its approximation problem proposed in [31]. While having promising clustering performance, the method [21] only considers the single view feature. Then, the method [22], which is most relevant to our work, extends the LRR to the multi-view setting by imposing unfolding based low rank [23] (defined in Eqn. (19)) on tensor that stacked by the subspace representation matrices from all the views. While easy to implement, different from matrix scenarios, such a simple rank-sum tensor norm is short of a clear physical meaning for general tensors. Furthermore, it tries to model the tensor low rank in the matrix SVD-based vector space, resulting in the loss of optimality in the representation.

By contrast, the high order constraint used in our approach is based on recently proposed tensor-Singular Value Decomposition (t-SVD) and its derived tensor nuclear norm (t-TNN) [16]. t-SVD has a similar structure to the matrix SVD, and model a tensor in the matrix space through a well-defined t-product operation [31], which can be shown in the theoretical analysis in motivation subsection 4.1 (due to the need for some key notations and preliminaries, we postpone the detailed motivation until section 4.1). By applying this well-defined tensor constraint to our multi-view model, a natural physical meaning for low-rank structure underneath tensor can be achieved. More importantly, in our approach, each subspace representation matrix can be considered as a view-specific distance metric learning among different samples but with measurement noisy. The proposed method can filter out the noisy to ensure the *consensus principle* implicitly by using t-SVD based tensor multi-rank minimization (see Fig. 1 (c)). In summary, for the first time to our knowledge, we introduce a circulant algebra based low-rank tensor constraint to achieve consensus among views and explore complementary principle efficiently and thoroughly, which can be confirmed by our excellent clustering performance presented in Section 5.

In this paper, we propose a new multi-view clustering method, namely t-SVD based Multi-view Subspace Clustering (t-SVD-MSC). Fig. 1 illustrates the flowchart of our method. Given a collection of data points with multiple views $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(V)}$, t-SVD-MSC can obtain the subspace representation matrices $\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(V)}$, and then merge them to construct a 3-order tensor $\mathcal{Z}$. This tensor needs to be rotated so as to keep self-representation coefficient in Fourier domain, and the detailed merits can be found in Section 4.1. Subsequently, the rotated tensor $\tilde{\mathcal{Z}}$ is efficiently updated by t-SVD based tensor nuclear norm minimization, such that the high order information hidden among multi-view representation can be captured. After that, each $\mathbf{Z}^{(v)}$ ($v = 1, \ldots, V$) will be updated under the self-reconstruction constraint. This process runs iteratively until convergence is arrived. We need to emphasize here that our contributions are not meant as a simple replacement for the unfolding based tensor norm presented in [22]. The proposed t-SVD-MSC carefully consider the complicated structure of the subspace representation matrices from all the views, so that the subspace coefficients are transformed into Fourier domain; meanwhile the information among different samples and views can be explored by comparing every row (sample-specific) and every column (view-specific) of frontal slices over the third dimension (coefficient-specific), which is the intrinsic property of tensor low rank built upon t-SVD.

The main contributions of this paper are summarized as follows:

– We propose a new multi-view subspace clustering model, *i.e.*, t-SVD-MSC, to effectively ensure the consensus among different views by utilizing a well-founded tensor norm in a unified tensor space, so that the complementary information can be captured and propagated among all the views.

– To accommodate the circulant algebra, we design a rotated tensor structure to preserve the self-representation coefficient in Fourier domain, as well as explore the high order correlations by comparing every row (sample-specific) and every column (view-specific) of frontal tensor slices.

– We present an efficient optimization algorithm to solve the t-SVD-MSC optimization problem with relatively low computational complexity and theoretical convergence guarantee.

– We conduct the extensive evaluation of our method on several challenge datasets, where a significant improvement over state-of-the-art MSC approaches is achieved. By incorporating CNN feature as a view, the proposed model has achieved highly competent (even better) performance compared to recent proposed CNN based clustering method on some large-scale datasets.

The rest of this paper is organized as follows. Section 2 introduces related works. Section 3 gives the preliminaries on tensors and the notations that will be used throughout
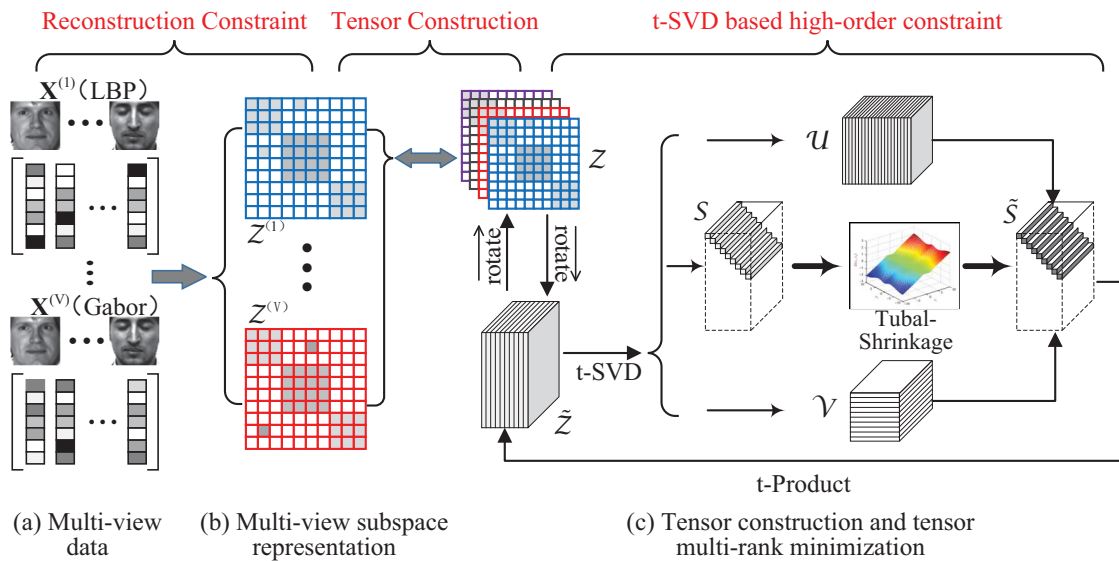
Figure 1: The Flowchart of t-SVD-MSC. Given a collection of data points with multi-view representation (a), $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(V)}$, t-SVD-MSC stacks all the subspace representations (b), $\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(V)}$, into a tensor $\mathcal{Z}$, and then rotates to $\tilde{\mathcal{Z}}$; the $\tilde{\mathcal{Z}}$ will be updated by using t-SVD based tensor multi-rank minimization (c).

the paper. In Section 4, we motivate the proposed model in detail, give an optimization algorithm to solve it, analyze its computational complexity and convergence, and provide some discussions. Experimental analysis and completion results are shown in Section 5 to verify our method. Finally, we conclude the proposed method in Section 6.

2 Related Work

Multi-view clustering methods have been extensively studied in recent years, we roughly divide them into three categories in accordance with [1]: 1) graph-based approaches, 2) co-training or co-regularized approaches, 3) subspace learning algorithms.

The first stream is the graph-based approaches [2, 3, 4, 5, 6] which exploit the relationship among different views by using multiple graph fusion strategy. [2] constructed a bipartite graph underlying the minimizing-disagreement criterion to connect the two-view feature, and then solved standard spectral clustering problem on the bipartite graph. The method [5] proposed to learn a latent graph transition probability matrix via low-rank and sparse decomposition to handle the noise from different views. Given graphs constructed separately from single view data, [6] built cross-view tensor product graphs to explore higher order information. Moreover, graph based algorithms is closely related to Multiple Kernel Learning (MKL) technique, in which views are considered as given kernel matrices. The aim is to learn the weighted combination of these kernel and the partitioning simultaneously [7].

Co-training and co-regularized style methods often construct separate learners on distinct views, then utilize the information in each learner to constrain other views. [8] provided a clustering method by interchanging the partition information among different views. [9] proposed to utilize the spectral embedding from one view to constrain the adjacent matrices in other views. By co-regularizing the clustering hypotheses across views, [10] designed novel spectral clustering objective functions that implicitly combine graphs from multiple views of the data to achieve a better result. In [47], authors extended the recent subspace clustering to multi-view domain, and utilized the Hilbert Schmidt Independence Criterion (HSIC) as a co-regularized term to explore the complementarity between views. To cluster the video face by multiple intrinsic cues, [46] considered both the video face pairwise constraints as well as the multi-view consistence, which is a co-regularization term that penalizes the disagreement among different graphs of multiple views, leading to a state-of-the-art performance on several real-world video datasets.

Subspace learning approaches are built on the assumption that all the views are generated from a latent subspace. Its goal is to capture shared latent subspace first and then conduct clustering. The representative methods in this stream are proposed in [11, 12], which applied canonical correlation analysis (CCA) and kernel CCA to project the multi-view high-dimensional data onto a low-dimensional subspace, respectively. By including robust losses to replace the squared loss used in CCA, [15] provided a convex reformulation of multi-view subspace learning that enforces conditional independence between views. Inspired by deep representation, [14] proposed a DNN-based model combining CCA

and autoencoder-based terms to exploit the deep information from two views. Since those CCA based methods are limited by capability of only handling two-view features, tensor CCA [13] generalized CCA to handle the data of an arbitrary number of views by analyzing the covariance tensor of different views.

Besides CCA, the recent proposed subspace clustering methods [54, 22] resorted to explore the relationship between samples with self-representation (*e.g.,* sparse subspace clustering (SSC) [20] and low-rank representation (LRR) [21]) in multi-view setting. Our approach is closely related to [22], which extended the LRR based subspace clustering to multi-view by employing the rank-sum of different mode unfoldings to constrain the subspace coefficient tensor. However, such a kind of tensor constraint lacks a clear physical meaning for general tensor, so that it can not thoroughly explore the complementary information among different views. On the contrary, the high order constraint within our model is built upon a new tensor decomposition scheme [16, 31], which is referred to as t-SVD and has been applied to various tasks, such as image reconstruction and tensor completion [18, 17, 19]. Therefore, the proposed model possesses good theoretical properties and clear physical meaning for handling the subspace representation tensor. The detailed motivation will be presented in Section 4.1.

3 Notations and Preliminaries

In this section, we will introduce the notations and give the basic definitions used throughout the paper. We use bold calligraphy letters for tensors, *e.g.,* $\boldsymbol{\mathcal{X}}$, bold upper case letters for matrices, *e.g.,* $\mathbf{X}$, bold lower case letters for vectors, *e.g.,* $\mathbf{x}$, and lower case letters for the entries, *e.g.,* x_{ij}. The Frobenius norm of a matrix $\mathbf{X}$ is defined as $||\mathbf{X}||_F := (\sum_{i,j} |x_{ij}|^2)^{\frac{1}{2}}$. Let $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}$ be the SVD of $\mathbf{X}$ and $\sigma_i(\mathbf{X})$ the ith largest singular value, then the matrix nuclear norm of X is $||\mathbf{X}||_* := \sum_i \sigma_i(\mathbf{X})$. The corresponding singular-value thresholding (SVT) operation with threshold τ is $\mathcal{D}_\tau(\mathbf{X}) = \mathbf{U}\boldsymbol{\Sigma}_\tau\mathbf{V}^{\mathrm{T}}$, where $\Sigma_\tau = \mathrm{diag}\left\{(\sigma_i(\mathbf{X}) - \tau)_+\right\}$ and t_+ is the positive part of t.

An N-way (or N-mode) tensor is a multi-linear structure in $\mathbb{R}^{n_1 \times n_2 \times \dots \times n_N}$. A **slice** of an tensor is a 2D section defined by fixing all but two indices, and a **fiber** is a 1D section defined by fixing all indices but one [29]. For a 3-way tensor $\boldsymbol{\mathcal{X}}$, we use the Matlab notation $\boldsymbol{\mathcal{X}}(k, :, :)$, $\boldsymbol{\mathcal{X}}(:, k, :)$ and $\boldsymbol{\mathcal{X}}(:, :, k)$ to denote the kth horizontal, lateral and frontal slices, respectively; $\boldsymbol{\mathcal{X}}(:, i, j)$, $\boldsymbol{\mathcal{X}}(i, :, j)$ and $\boldsymbol{\mathcal{X}}(i, j, :)$ to denote the mode-1, mode-2 and mode-3 fibers, and $\boldsymbol{\mathcal{X}}_f = \mathrm{fft}(\boldsymbol{\mathcal{X}}, [\,], 3)$ to denote the Fourier transform along the third dimension. In particular, $\boldsymbol{\mathcal{X}}^{(k)}$ is used to represent $\boldsymbol{\mathcal{X}}(:, :, k)$. Unfolding the tensor $\boldsymbol{\mathcal{X}}$ along the lth mode defined as $\mathrm{unfold}_l(\boldsymbol{\mathcal{X}}) = \mathbf{X}_{(l)} \in \mathbb{R}^{n_l \times \prod_{l' \neq l} n_{l'}}$, which is a matrix whose columns are

mode-l fibers [29]. The opposite operation "fold" of the unfolding is defined as $\mathrm{fold}_l(\mathbf{X}_{(l)}) = \boldsymbol{\mathcal{X}}$. The Frobenius norm of $\boldsymbol{\mathcal{X}}$ is $||\boldsymbol{\mathcal{X}}||_F := (\sum_{i,j,k} |x_{ijk}|^2)^{\frac{1}{2}}$, and the l_1 norm of $\boldsymbol{\mathcal{X}}$ is $||\boldsymbol{\mathcal{X}}||_1 := \sum_{i,j,k} |x_{ijk}|$.

Before introducing the t-SVD and its derived tensor nuclear norm, it is necessary to define five block-based operators, i.e., bcirc, bvec, bvfold, bdiag and bdfold [16]. For $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ specially, the $\boldsymbol{\mathcal{X}}^{(k)}$s can be used to form the block circulant matrix:

$$\mathrm{bcirc}(\boldsymbol{\mathcal{X}}) := \begin{bmatrix} \boldsymbol{\mathcal{X}}^{(1)} & \boldsymbol{\mathcal{X}}^{(n_3)} & \cdots & \boldsymbol{\mathcal{X}}^{(2)} \\ \boldsymbol{\mathcal{X}}^{(2)} & \boldsymbol{\mathcal{X}}^{(1)} & \cdots & \boldsymbol{\mathcal{X}}^{(3)} \\ \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{\mathcal{X}}^{(n_3)} & \boldsymbol{\mathcal{X}}^{(n_3-1)} & \cdots & \boldsymbol{\mathcal{X}}^{(1)} \end{bmatrix}, \tag{1}$$

the block vectorizing and its opposite operation

$$\mathrm{bvec}(\boldsymbol{\mathcal{X}}) := \begin{bmatrix} \boldsymbol{\mathcal{X}}^{(1)} \\ \boldsymbol{\mathcal{X}}^{(2)} \\ \vdots \\ \boldsymbol{\mathcal{X}}^{(n_3)} \end{bmatrix}, \quad \mathrm{bvfold}(\mathrm{bvec}(\boldsymbol{\mathcal{X}})) = \boldsymbol{\mathcal{X}}, \tag{2}$$

and the block diag matrix and its opposite operation

$$\mathrm{bdiag}(\boldsymbol{\mathcal{X}}) := \begin{bmatrix} \boldsymbol{\mathcal{X}}^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{X}}^{(n_3)} \end{bmatrix}, \quad \mathrm{bdfold}(\mathrm{bdiag}(\boldsymbol{\mathcal{X}})) = \boldsymbol{\mathcal{X}}. \tag{3}$$

3.1 Tensor Singular Value Decomposition (t-SVD)

To help understand the t-SVD, the following related notions, which are defined in [16], need to be introduced. The t-product between two 3-mode tensors is defined as follows:

Definition 1 (t-product) Let $\boldsymbol{\mathcal{X}}$ be $n_1 \times n_2 \times n_3$, and $\boldsymbol{\mathcal{Y}}$ be $n_2 \times n_4 \times n_3$. The t-product $\boldsymbol{\mathcal{X}} * \boldsymbol{\mathcal{Y}}$ is an $n_1 \times n_4 \times n_3$ tensor

$$\boldsymbol{\mathcal{M}} = \boldsymbol{\mathcal{X}} * \boldsymbol{\mathcal{Y}} =: \mathrm{bvfold}\{\mathrm{bcirc}(\boldsymbol{\mathcal{X}})\mathrm{bvec}(\boldsymbol{\mathcal{Y}})\}. \tag{4}$$

The t-product is analogous to the matrix multiplication except that the *circular convolution* replaces the multiplication operation between the elements, which are now mode-3 fibers [17], as follows:

$$\boldsymbol{\mathcal{M}}(i, j, :) = \sum_{k=1}^{n_2} \boldsymbol{\mathcal{X}}(i, k, :) \circ \boldsymbol{\mathcal{Y}}(k, j, :), \tag{5}$$

where $\circ$ denotes the circular convolution between two tubes. The t-product in the original domain corresponds to the matrix multiplication of the frontal slices in the Fourier domain, as follows :

$$\boldsymbol{\mathcal{M}}_f^{(k)} = \boldsymbol{\mathcal{X}}_f^{(k)} \boldsymbol{\mathcal{Y}}_f^{(k)}, \ k = 1, \dots, n_3, \tag{6}$$

Definition 2 (Tensor Transpose) Let $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the transpose tensor $\boldsymbol{\mathcal{X}}^{\mathrm{T}}$ is an $n_2 \times n_1 \times n_3$ tensor obtained by transposing each frontal slice of $\boldsymbol{\mathcal{X}}$ and then reversing the order of the transposed frontal slices 2 through n_3.

Definition 3 (Identity Tensor) The identity tensor $\boldsymbol{\mathcal{I}} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ is a tensor whose first frontal slice is the $n_1 \times n_1$ identity matrix and all other frontal slices are zero.

Definition 4 (Orthogonal Tensor) A tensor $\boldsymbol{\mathcal{Q}} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ is orthogonal if

$$\boldsymbol{\mathcal{Q}}^{\mathrm{T}} * \boldsymbol{\mathcal{Q}} = \boldsymbol{\mathcal{Q}} * \boldsymbol{\mathcal{Q}}^{\mathrm{T}} = \boldsymbol{\mathcal{I}}, \tag{7}$$

where $*$ is the t-product.

Definition 5 (f-diagonal Tensor) A tensor is called f-diagonal if each of its frontal slices is diagonal matrix. The t-production of two f-diagonal tensors with the same size $n_1 \times n_2 \times n_3$, i.e., $\boldsymbol{\mathcal{M}} = \boldsymbol{\mathcal{X}} * \boldsymbol{\mathcal{Y}}$, is also an $n_1 \times n_2 \times n_3$ f-diagonal tensor, and its diagonal tube fibers are

$$\boldsymbol{\mathcal{M}}(i,i,:) = \boldsymbol{\mathcal{X}}(i,i,:) \circ \boldsymbol{\mathcal{Y}}(i,i,:),\ i = 1,\ldots,\min(n_1,n_2). \tag{8}$$

Given the aforementioned definitions, the tensor Singular Value Decomposition (t-SVD) of $\boldsymbol{\mathcal{X}}$ is given by

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^{\mathrm{T}}, \tag{9}$$

where $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$ are orthogonal tensors of size $n_1 \times n_1 \times n_3$ and $n_2 \times n_2 \times n_3$ respectively. $\boldsymbol{\mathcal{S}}$ is an f-diagonal tensor of size $n_1 \times n_2 \times n_3$, and $*$ denotes the t-product. Fig. 2 illustrates the decomposition. As demonstrated in Eq. (6), the t-production can be computed efficiently in the Fourier domain, which leads to Algorithm 1.
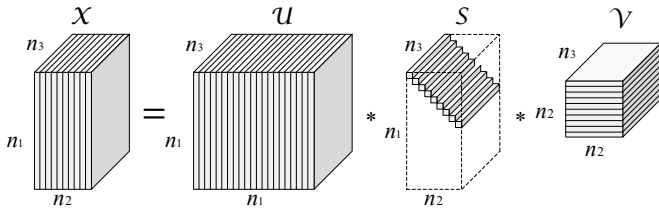

$$n_3 \quad \mathcal{X} \qquad n_3 \quad \mathcal{U} \qquad n_3 \quad S \qquad n_3 \quad \mathcal{V}$$

Figure 2: The t-SVD of an $n_1 \times n_2 \times n_3$ tensor.

3.2 Tensor Nuclear Norm via t-SVD

The t-SVD allows the tensor $\boldsymbol{\mathcal{X}}$ to be written as a finite sum of outer product of matrices [31]:

$$\boldsymbol{\mathcal{X}} = \sum_{i=1}^{\min(n_1,n_2)} \boldsymbol{\mathcal{U}}(:,i,:) * \boldsymbol{\mathcal{S}}(i,i,:) * \boldsymbol{\mathcal{V}}(:,i,:)^{\mathrm{T}}, \tag{10}$$

Algorithm 1: t-SVD [16]

Input: $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$;
Output: $\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}$;

1   $\boldsymbol{\mathcal{X}}_f = \mathrm{fft}(\boldsymbol{\mathcal{X}},[\,],3)$;
2   **for** $k = 1 : n_3$ **do**
3     $[\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}] = \mathrm{SVD}(\boldsymbol{\mathcal{X}}_f^{(k)})$;
4     $\boldsymbol{\mathcal{U}}_f^{(k)} = \mathbf{U}, \boldsymbol{\mathcal{S}}_f^{(k)} = \boldsymbol{\Sigma}, \boldsymbol{\mathcal{V}}_f^{(k)} = \mathbf{V}$;
5   **end**
6   $\boldsymbol{\mathcal{U}} = \mathrm{ifft}(\boldsymbol{\mathcal{U}}_f,[\,],3),\ \boldsymbol{\mathcal{S}} = \mathrm{ifft}(\boldsymbol{\mathcal{S}}_f,[\,],3),\ \boldsymbol{\mathcal{V}} = \mathrm{ifft}(\boldsymbol{\mathcal{V}}_f,[\,],3)$;
7   **Return** $\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{V}}$.

---

which is equivalent to the following equation in the Fourier domain [31]:

$$\begin{bmatrix} \boldsymbol{\mathcal{X}}_f^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{X}}_f^{(n_3)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{U}}_f^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{U}}_f^{(n_3)} \end{bmatrix} \cdot$$
$$\begin{bmatrix} \boldsymbol{\mathcal{S}}_f^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{S}}_f^{(n_3)} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\mathcal{V}}_f^{(1)} & & \\ & \ddots & \\ & & \boldsymbol{\mathcal{V}}_f^{(n_3)} \end{bmatrix}^{\mathrm{T}} . \tag{11}$$

where $\cdot$ denotes common matrix product, and we have n_3 blocks matrix SVD: $\boldsymbol{\mathcal{X}}_f^{(i)} = \boldsymbol{\mathcal{U}}_f^{(i)} \boldsymbol{\mathcal{S}}_f^{(i)} (\boldsymbol{\mathcal{V}}_f^{(i)})^{\mathrm{T}}, i = 1,\ldots,n_3$. Now, we can define the tensor multi-rank as follows [16, 17, 18]:

Definition 6 (Tensor multi-rank) The multi-rank of $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is a vector $\mathbf{r} \in \mathbb{R}^{n_3 \times 1}$ with the i-th element equal to the rank of the i-th frontal slice of $\boldsymbol{\mathcal{X}}_f$.

Then the t-SVD based tensor nuclear norm (t-TNN) is given as

$$||\boldsymbol{\mathcal{X}}||_{\circledast} := \sum_{i=1}^{\min(n_1,n_2)} \sum_{k=1}^{n_3} |\boldsymbol{\mathcal{S}}_f(i,i,k)|, \tag{12}$$

which is proven to be a valid norm and the tightest convex relaxation to ℓ_1 norm of the tensor multi-rank in [18, 17]. Due to the unitary invariance of matrix nuclear norm, we have

$$||\mathrm{bdiag}(\boldsymbol{\mathcal{X}}_f)||_* = ||\mathrm{bdiag}(\boldsymbol{\mathcal{S}}_f)||_* = ||\boldsymbol{\mathcal{X}}||_{\circledast}, \tag{13}$$

and since block circulant matrixes can be block diagonalized by using the Fourier transform, there is

$$||\mathrm{bdiag}(\boldsymbol{\mathcal{X}}_f)||_* = ||(\mathbf{F}_{n_3} \otimes \mathbf{I}_{n_1})\mathrm{bcirc}(\boldsymbol{\mathcal{X}})(\mathbf{F}_{n_3}^* \otimes \mathbf{I}_{n_2})||_*$$
$$= ||\mathrm{bcirc}(\boldsymbol{\mathcal{X}})||_*. \tag{14}$$

where, $\otimes$ denotes the Kronecker product, $\mathbf{F}_n$ is the $n \times n$ Discrete Fourier Transform (DFT) matrix, and $\mathbf{I}_n$ is an $n \times n$ identity matrix. Finally, we obtain

$$||\boldsymbol{\mathcal{X}}||_{\circledast} = ||\mathrm{bcirc}(\boldsymbol{\mathcal{X}})||_*. \tag{15}$$

The equivalence in Eq. (15) endows the t-TNN with interpretability in the original domain, *i.e.,* $||\mathrm{bcirc}(\boldsymbol{\mathcal{X}})||_*$ measures the rank of $\mathrm{bcirc}(\boldsymbol{\mathcal{X}})$ by comparing every row and every column of frontal slices over the third dimension, which exploits structural information of a tensor deeper than the monotonous matrix nuclear norm of certain unfolding.

4 The Proposed Approach

Subspace clustering is a technology for clustering data according to the underlying subspaces. In the paper, we consider the self-representation based subspace clustering method, specifically the LRR approach [21], which constructs affinity matrix through reconstruction coefficients, as well as explores low-dimensional subspace structures embedded in data. Suppose $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ is the matrix of data vectors, whose column is a sample vector, and d is the dimensionality of the feature space. Formally, LRR solves the following optimization problem:

$$\min_{\mathbf{Z},\mathbf{E}} \lambda ||\mathbf{E}||_{2,1} + ||\mathbf{Z}||_*, \tag{16}$$

$$\text{s.t.} \quad \mathbf{X} = \mathbf{XZ} + \mathbf{E},$$

where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N] \in \mathbb{R}^{N \times N}$ is the coefficient matrix with each $\mathbf{z}_i$ being the new representation of sample $\mathbf{x}_i$, and $|| \cdot ||_*$ is the nuclear norm, $|| \cdot ||_{2,1}$ denotes the $\ell_{2,1}$-norm of a matrix. After achieving the self-representation matrix $\mathbf{Z}$, the affinity matrix $\mathbf{A}$ is usually constructed as

$$\mathbf{A} = \frac{1}{2} \left(|\mathbf{Z}| + |\mathbf{Z}^T| \right), \tag{17}$$

where $| \cdot |$ represents the absolute operator. Then, the obtained affinity matrix $\mathbf{A}$ will be sent to a spectral clustering algorithm [24] to produce the final clustering result.

Intuitively, the above single view subspace clustering method can be extended to the multi-view setting in a simple and direct way. We use $\mathbf{X}^{(v)}$ to denote the feature matrix corresponding to the v-th view, and use $\mathbf{Z}^{(v)}$ to represent the v-th view's learned subspace representation. Hence, the objective function of the LRR based naive multi-view subspace clustering turns out to be:

$$\min_{\mathbf{Z}^{(v)}, \mathbf{E}^{(v)}} \sum_{v}^{V} \left(\lambda_v ||\mathbf{E}^{(v)}||_{2,1} + ||\mathbf{Z}^{(v)}||_* \right), \tag{18}$$

$$\text{s.t.} \quad \mathbf{X}^{(v)} = \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \mathbf{E}^{(v)}, v = 1, 2, \ldots, V,$$

where V denotes the number of all views. After obtaining $\left\{ \mathbf{Z}^{(v)} \right\}_{v=1}^{V}$, the final affinity matrix is calculated by combining all subspace representation of each view: $\mathbf{A} = \frac{1}{V}\sum_{v=1}^{V}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|)/2$. However, this formulation treats each subspace representation independently, ignoring the relationship among different views. To overcome this drawback, we propose to utilize t-SVD based tensor nuclear norm to capture the high order correlations among different views.

4.1 Motivation

4.1.1 Ensuring Consensus Principle among Views

Recall the LRR [21], it can achieve the self-representation coefficient matrix $\mathbf{Z} \in \mathbb{R}^{N \times N}$ by representing the data samples as linear combinations of the bases in a given dictionary (usually, the whole dataset itself). In other words, LRR leads to dense representation coefficients within the same subspace. When we employ multiple features to describe the data, we will have multiple self-representation coefficient matrix $\{\mathbf{Z}^{(v)}\}_{v=1}^{V}$ correspondingly. Not only should we keep the low rank constraint for each $\mathbf{Z}^{(v)}$, but also need to ensure the consensus principle by imposing low rank across all views. The proposed approach is capable of modeling those two level low rank constraints in a unified tensor space by imposing the t-TNN. Consequently, after the optimization, all the $\{\mathbf{Z}^{(v)}\}_{v=1}^{V}$ are much more close to well structure, which means that the fused $\mathbf{Z} = \frac{1}{V}\sum_{v=1}^{V}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|)/2$ can be easily segmented by common spectral clustering method.

4.1.2 Requiring a Well-Founded Low Rank Constraint in Tensor Space

To extend the self-representation based subspace clustering to multi-view setting, [22] introduced a low-rank tensor constraint [23], which directly extended the matrix nuclear norm to higher-order case:

$$||\boldsymbol{\mathcal{Z}}||_* = \sum_{m=1}^{3} \xi_m ||\mathbf{Z}_{(m)}||_*, \tag{19}$$

where the weight ξ_m needs to satisfy $\xi_m > 0$ and $\sum_{m=1}^{3} \xi_m = 1$ ($\xi_1 = \xi_2 = \xi_3$ is used in [22]), $\boldsymbol{\mathcal{Z}}$ is a 3-order tensor constructed by merging different $\mathbf{Z}^{(v)}$ along the third dimension, and $\mathbf{Z}_{(m)}$ is the unfolding matrix along the m-th mode. We refer to it as the *generalized tensor nuclear norm* (g-TNN). Albeit easy to implement, different from matrix scenarios, such a simple rank-sum term is short of a clear physical meaning for general tensors. Besides, the strategy of using the same weights to penalize all dimensionality ranks of a tensor is not always rational. However, incorporating the t-TNN in the proposed model possesses obvious and clear physical meaning. We introduce the following theorem to theoretically explain why the t-TNN is adopted in the proposed model.

Theorem 1 *[31] Let the t-SVD of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ be given by $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^{\mathrm{T}}$, and for $k < \min(n_1, n_2)$ define $\boldsymbol{\mathcal{A}}_k = \sum_{i=1}^{k} \boldsymbol{\mathcal{U}}(:, i, :) * \boldsymbol{\mathcal{S}}(i, i, :) * \boldsymbol{\mathcal{V}}(:, i, :)^{\mathrm{T}}$, then*

$$\boldsymbol{\mathcal{A}}_k = \operatorname*{argmin}_{\widetilde{\boldsymbol{\mathcal{A}}} \in \mathbb{M}} ||\boldsymbol{\mathcal{A}} - \widetilde{\boldsymbol{\mathcal{A}}}||_F \tag{20}$$

where $\mathbb{M} = \{\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{X}} * \boldsymbol{\mathcal{Y}} | \boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times k \times n_3}, \boldsymbol{\mathcal{Y}} \in \mathbb{R}^{k \times n_2 \times n_3}\}$.

Theorem 1 indicates that a truncated t-SVD representation could provide an optimal approximation in the same way as the truncated matrix SVD, which gives a best low rank approximation to the matrix in terms of the Frobenius norm under rank k constraint. Moreover, matrix nuclear norm is the tightest convex relaxation of the original rank minimization, while t-TNN also has been proven to be the tightest convex relaxation to ℓ_1 norm of the tensor multi-rank (see Section 3.2) [17]. Theoretically, the t-TNN is **more analogous** to the matrix nuclear norm than the g-TNN defined in (19).

Figure 3: The rotated coefficient tensor in our approach.

4.1.3 Constructing a Structure for Tensor Circulant Algebra

Directly utilizing the t-TNN to model the low rank constraint is still far from effectiveness, which can be evidenced by observing the performance of Ut-SVD-MSC method in experimental section. To accommodate the intrinsic circulant algebra underlying t-TNN, we choose to transform the self-represented coefficient (mode-1 fiber) into the mode-3 fiber by using *the rotation operation*[1] of the coefficient tensor, as illustrated in Fig. 3, where the marked fiber denotes a self-represented feature coefficient of a certain sample belonging to a certain view.

While relatively simple, the proposed model will benefit from the rotate operation in three aspects. First of all, through tensor rotation, the self-representation coefficient can be preserved in Fourier domain, since the Fourier transform along the third dimension. Secondly, each frontal slice in Fourier domain considers the information among different samples and different views. By measuring every row and every column of frontal slices over the third dimension, the t-TNN provides a deeper insight into multi-view feature tensor than g-TNN. Another advantage of this rotate operation is the significant reduction of computational complexity, which will be analyzed in Section 4.4 and Section

---

[1] The tensor rotation in Matlab can be achieved by using the command "shiftdim".

To sum up, the aforementioned satisfaction of principle, good theoretical properties, and well-designed tensor structure motivate us to design the proposed t-SVD-MSC model.

4.2 Problem Formulation

The objective function of the proposed method is:

$$
\min_{\mathbf{Z}^{(v)}, \mathbf{E}^{(v)}} \lambda ||\mathbf{E}||_{2,1} + ||\boldsymbol{\mathcal{Z}}||_{\circledast},
$$

$$
\begin{aligned}
\text{s.t.} \quad & \mathbf{X}^{(v)} = \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + \mathbf{E}^{(v)}, v = 1, \ldots, V, \\
& \boldsymbol{\mathcal{Z}} = \Phi(\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(V)}), \\
& \mathbf{E} = [\mathbf{E}^{(1)}; \mathbf{E}^{(2)}; \ldots, \mathbf{E}^{(V)}],
\end{aligned}
\tag{21}
$$

where the function $\Phi(\cdot)$ constructs the tensor $\boldsymbol{\mathcal{Z}}$ by merging different representation $\mathbf{Z}^{(v)}$ to a 3-mode tensor, and then rotate its dimensionality to $N \times V \times N$, as shown in Fig. 3. Also, we can easily get the following relationship:

$$
\Phi_{(v)}^{-1}(\boldsymbol{\mathcal{Z}}) = \mathbf{Z}^{(v)},
\tag{22}
$$

where $\Phi^{-1}(\cdot)$ denotes the inverse function of $\Phi(\cdot)$, and its subscript (v) means to extract the v-th frontal slice. As suggested in [21], the vertical concatenation along the column of error matrix, *i.e.,* $\mathbf{E} = [\mathbf{E}^{(1)}; \mathbf{E}^{(2)}; \ldots, \mathbf{E}^{(V)}]$, can enforce the column of $\mathbf{E}^{(v)}$ in each view to have jointly consistent magnitude values. Consequently, the objective function in Eq. (21) aims to find the optimal self-representations through capturing the informational and structural complexity of multi-view features.

The above optimization problem can be solved by using the Augmented Lagrange Multiplier (ALM) [25]. To adopt alternating direction minimizing strategy to problem (21), we need to make the objective function seperable. By introducing the auxiliary tensor variable $\boldsymbol{\mathcal{G}}$, the optimization problem can be transferred to minimize the following unconstrained problem:

$$
\begin{aligned}
& \boldsymbol{\mathcal{L}}(\mathbf{Z}^{(v)}, \ldots, \mathbf{Z}^{(V)}; \mathbf{E}^{(1)}, \ldots, \mathbf{E}^{(V)}; \boldsymbol{\mathcal{G}}) \\
& = \lambda ||\mathbf{E}||_{2,1} + ||\boldsymbol{\mathcal{G}}||_{\circledast} + \sum_{v=1}^{V} \left(\langle \mathbf{Y}_v, \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)} \rangle \right. \\
& \left. + \frac{\mu}{2} ||\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}||_F^2 \right) + \langle \boldsymbol{\mathcal{W}}, \boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{G}} \rangle \\
& + \frac{\rho}{2} ||\boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{G}}||_F^2.
\end{aligned}
\tag{23}
$$

where the matrix $\mathbf{Y}_v$ and the tensor $\boldsymbol{\mathcal{W}}$ represent two Lagrange multipliers, μ and ρ are actually the penalty parameters, which are adjusted by using adaptive updating strategy as suggested in [26]. The optimization problem (23) seems challenging to solve, not only because of the t-TNN on $\boldsymbol{\mathcal{G}}$, but also since the tensor $\boldsymbol{\mathcal{Z}}$ depends on the subspace representation of all views.

4.3 Optimization Procedure

The alternative minimization scheme is adopted for updating $\mathbf{Z}^{(v)}$, $\mathbf{E}^{(v)}$, and $\mathcal{G}$, respectively. The detailed procedure can be partitioned into three steps alternatingly.

$\mathbf{Z}^{(v)}$**-subproblem:** When $\mathbf{E}$ and $\mathcal{G}$ are fixed, since $\Phi_{(v)}^{-1}(\mathcal{W}) = \mathbf{W}^{(v)}$ and $\Phi_{(v)}^{-1}(\mathcal{G}) = \mathbf{G}^{(v)}$, we will solve the following subproblem for updating the subspace representation $\mathbf{Z}^{(v)}$:

$$\min_{\mathbf{Z}^{(v)}} \quad \langle \mathbf{Y}_v, \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)} \rangle + \frac{\mu}{2} \|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)}$$
$$- \mathbf{E}^{(v)}\|_F^2 + \langle \mathbf{W}^{(v)}, \mathbf{Z}^{(v)} - \mathbf{G}^{(v)} \rangle + \frac{\rho}{2} \|\mathbf{Z}^{(v)} - \mathbf{G}^{(v)}\|_F^2. \tag{24}$$

By setting the derivative of (24) to zero, the closed-form of $\mathbf{Z}^{(v)}$ can be obtained by

$$\mathbf{Z}^{(v)*} = (\mathbf{I} + \frac{\mu}{\rho}\mathbf{X}^{(v)^T}\mathbf{X}^{(v)})^{-1} \Big((\mathbf{X}^{(v)^T}\mathbf{Y}_v + \mu\mathbf{X}^{(v)^T}\mathbf{X}^{(v)}$$
$$- \mu\mathbf{X}^{(v)^T}\mathbf{E}^{(v)} - \mathbf{W}^{(v)})/\rho + \mathbf{G}^{(v)} \Big). \tag{25}$$

$\mathbf{E}^{(v)}$**-subproblem:**

$$\mathbf{E}^* = \operatorname*{argmin}_{\mathbf{E}} \lambda\|\mathbf{E}\|_{2,1} + \sum_{v=1}^{V} \Big(\langle \mathbf{Y}_v, \mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)} \rangle$$
$$+ \frac{\mu}{2}\|\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}\|_F^2 \Big)$$
$$= \operatorname*{argmin}_{\mathbf{E}} \frac{\lambda}{\mu}\|\mathbf{E}\|_{2,1} + \frac{1}{2}\|\mathbf{E} - \mathbf{D}\|_F^2, \tag{26}$$

where $\mathbf{D}$ is constructed by vertically concatenating the matrices $\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} + (1/\mu)\mathbf{Y}_v$ together along column. According to the Lemma 4.1 in [21], this subproblem has the following solution,

$$\mathbf{E}_{:,i}^* = \begin{cases} \dfrac{\|\mathbf{D}_{:,i}\|_2 - \frac{\lambda}{\mu}}{\|\mathbf{D}_{:,i}\|_2}\mathbf{D}_{:,i}, & \|\mathbf{D}_{:,i}\|_2 > \dfrac{\lambda}{\mu} \\ 0 & \text{otherwise.} \end{cases} \tag{27}$$

where $\mathbf{D}_{:,i}$ represents the i-th column of the matrix $\mathbf{D}$.

$\mathcal{G}$**-subproblem:** When $\mathbf{Z}^{(v)}$, $(v = 1, 2, \dots, V)$ are fixed, for updating the tensor $\mathcal{G}$, we solve the following subproblem:

$$\mathcal{G}^* = \operatorname*{argmin}_{\mathcal{G}} \|\mathcal{G}\|_{\circledast} + \frac{\rho}{2}\|\mathcal{G} - (\mathcal{Z} + \frac{1}{\rho}\mathcal{W})\|_F^2. \tag{28}$$

which is referred to as the *tensor multi-rank minimization* in this paper. The solution of the optimization problem (28) can be achieved through the following theorem[2]:

---

[2] A similar discussion about the optimization of the TNN regularized low-rank tensor completion problem can be found in [17].

Algorithm 2: t-SVD based Tensor Multi-Rank Minimization

Input: Observed tensor $\mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, scalar $\tau > 0$
Output: tensor $\mathcal{G}$

1   $\mathcal{F}_f = \text{fft}(\mathcal{F}, [\,], 3)$, $\tau' = n_3\tau$;
2   **for** $j = 1 : n_3$ **do**
3     $[\mathcal{U}_f^{(j)}, \mathcal{S}_f^{(j)}, \mathcal{V}_f^{(j)}] = \text{SVD}(\mathcal{F}_f^{(j)})$;
4     $\mathcal{J}_f^{(j)} = \text{diag}\{(1 - \frac{\tau'}{\mathcal{S}_f^{(j)}(i,i)})_+\}$, $i = 1, \dots, \min(n_1, n_2)$;
5     $\mathcal{S}_{f,\tau'}^{(j)} = \mathcal{S}_f^{(j)}\mathcal{J}_J^{(j)}$;
6     $\mathcal{G}_f^{(j)} = \mathcal{U}_f^{(j)}\mathcal{S}_{f,\tau'}^{(j)}\mathcal{V}_f^{(j)^T}$;
7   **end**
8   $\mathcal{G} = \text{ifft}(\mathcal{G}_f, [\,], 3)$;
9   **Return** tensor $\mathcal{G}$.

Theorem 2 *For $\tau > 0$ and $\mathcal{G}, \mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the globally optimal solution to the following problem*

$$\min_{\mathcal{G}} \tau\|\mathcal{G}\|_{\circledast} + \frac{1}{2}\|\mathcal{G} - \mathcal{F}\|_F^2 \tag{29}$$

is given by the tensor tubal-shrinkage operator

$$\mathcal{G} = \mathcal{C}_{n_3\tau}(\mathcal{F}) = \mathcal{U} * \mathcal{C}_{n_3\tau}(\mathcal{S}) * \mathcal{V}^T, \tag{30}$$

where $\mathcal{F} = \mathcal{U} * \mathcal{S} * \mathcal{V}^T$ and $\mathcal{C}_{n_3\tau}(\mathcal{S}) = \mathcal{S} * \mathcal{J}$, herein, $\mathcal{J}$ is an $n_1 \times n_2 \times n_3$ f-diagonal tensor whose diagonal element in the Fourier domain is $\mathcal{J}_f(i, i, j) = (1 - \frac{n_3\tau}{\mathcal{S}_f^{(j)}(i,i)})_+$.

The proof can be found in Appendix. We summarize the t-SVD based tensor multi-rank minimization in Algorithm 2. Additionally, the Lagrange multipliers $\mathbf{Y}_v$ and $\mathcal{W}$ need to be updated as follows:

$$\mathbf{Y}_v^* = \mathbf{Y}_v + \mu(\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}), \tag{31}$$
$$\mathcal{W}^* = \mathcal{W} + \rho(\mathcal{Z} - \mathcal{G}). \tag{32}$$

Finally, the optimization procedure of the proposed multiview subspace clustering method is described in Algorithm 3.

4.4 Convergence Properties and Computational Complexity

The convergence properties of the inexact ALM have been well established when the number of blocks is at most two [25]. Despite of its success in practice, its convergence properties for minimizing the objective function with N ($N \geq 3$) blocks variables linked by linear constraints, have remained unclear. Since there are several blocks (including $\{\mathbf{Z}^{(v)}\}_{v=1}^{V}$, $\mathbf{E}$, and $\mathcal{G}$) in Algorithm 3, and the objective function of (21) is not smooth, it would be not easy to prove the convergence in theory. Fortunately, as suggested in [21], two conditions are sufficient (but may not necessary) for Algorithm 3 to converge: (1) each feature matrix $\mathbf{X}^{(v)}$ is of

---

Algorithm 3: MSC via Tensor Multi-Rank Minimization

Input: Multi-view feature matrices: $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(V)}$, λ, and cluster number K

Output: Clustering results $\mathcal{C}$

1   Initialized $\mathbf{Z}^{(v)} = \mathbf{0}, \mathbf{E}^{(v)} = \mathbf{0}, \mathbf{Y}_v = \mathbf{0}, i = 1, \ldots, V$; $\mathcal{G} = \mathcal{W} = \mathbf{0}$; $\mu = 10^{-5}, \rho = 10^{-4}, \eta = 2, \mu_{\max} = \rho_{\max} = 10^{10}, \varepsilon = 10^{-7}$;

2   **while** *not converge* **do**

3   // Solving $\mathbf{Z}$

     for $v = 1 : V$ **do**

4   Update $\mathbf{Z}^{(v)}$ by using (25);

5   **end**

6   // Solving $\mathbf{E}$

     Update $\mathbf{E}$ by solving (26);

7   **for** $v = 1 : V$ **do**

8   Update $\mathbf{Y}_v$ by using (31);

9   **end**

10   Obtain $\mathcal{Z} = \Phi(\mathbf{Z}^{(1)}, \mathbf{Z}^{(2)}, \ldots, \mathbf{Z}^{(V)})$;

     // Solving $\mathcal{G}$

11   Update $\mathcal{G}$ via subproblem (28) by using Algorithm 2;

12   Update $\mathcal{W}$ by using (32);

13   Update parameters μ and ρ: $\mu = \min(\eta\mu, \mu_{\max})$, $\rho = \min(\eta\rho, \rho_{\max})$;

14     $(\mathbf{G}^{(1)}, \ldots, \mathbf{G}^{(V)}) = \Phi^{-1}(\mathcal{G})$;

15   Check the convergence conditions: $||\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}||_\infty < \varepsilon$ and $||\mathbf{Z}^{(v)} - \mathbf{G}^{(v)}||_\infty < \varepsilon$;

16   **end**

17   Obtain the affinity matrix by $\mathbf{A} = \frac{1}{V}\sum_{v=1}^{V} |\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)\mathrm{T}}|$;

18   Apply the spectral clustering method with the affinity matrix $\mathbf{A}$;

19   **Return** Clustering result $\mathcal{C}$.

---

full column rank; (2) the optimality gap produced in each iteration step is ***monotonically decreasing***. The first condition can be met by factorizing $\mathbf{Z}^{(v)}$ into $\mathbf{P}^{(v)}\hat{\mathbf{Z}}^{(v)}$, where $\mathbf{P}^{(v)}$ can be computed in advance by orthogonalizing the columns of $\mathbf{X}^{(v)\mathrm{T}}$. Moreover, due to the convexity of the Lagrangian function (23), the monotonically decreasing condition can be guaranteed to some extent according to [30]. Therefore, the proposed MSC algorithm ensures good convergence properties. Furthermore, the proposed method performs well and indeed converges fast in reality, which will be illustrated in Section 5.4.4.

Since inverse matrix can be calculated in advance in Eq. (25) for solving $\mathbf{Z}^{(v)}$, the computational bottleneck of the proposed Algorithm 3 only lies in solving the subproblems for $\mathbf{E}$ and $\mathcal{G}$. As for the $\mathbf{E}$ subproblem, it takes $\mathcal{O}(VN^2)$ in each iteration. As for the $\mathcal{G}$ subproblem, calculating the $3D$ FFT and $3D$ inverse FFT of an $N \times V \times N$ tensor and N SVDs of $N \times V$ matrices in the Fourier domain, actually dominate the main computation. Since in multi-view setting we have $N \gg V$ and $\log(N) > V$, the computation at each iteration will take $\mathcal{O}(2N^2V\log(N) + N^2V^2) \approx$

$\mathcal{O}(2N^2V\log(N))$. By considering the cost of spectral clustering (usually $\mathcal{O}(N^3)$) and the number of iterations needed to converge, the complexity of Algorithm 3 is:

$$\mathcal{O}(N^3) + \mathcal{O}(K(2N^2V\log(N))), \tag{33}$$

where K denotes the number of iterations. The iteration number K depends on the choice of η: larger η leads to a smaller K, and vice versa. In our experiments, we fix the η to 2, such that the iteration number K commonly locates within the range of $30 \sim 50$.

4.5 Discussion

Furthermore, we can analyze the contribution of each view to final clustering from the perspective of feature's characteristic, *i.e.,* discriminative power, both theoretically and experimentally (Section 5.4.1). Recent studies on sparse subspace clustering [20] have proved that a sample can be represented by its corresponding dictionary if the signals satisfy certain incoherence condition. In other words, low rank representation of a data point ideally corresponds to a combination of all the point from its own subspace, leading to a block-diagonal connectivity in affinity matrix. As it is proved in [21], for a certain feature, the closer the affinity matrix $\mathbf{A}^{(v)} = \frac{1}{2}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)\mathrm{T}}|)$ is to block-diagonal structure, the better the clustering result is. It is worth noting that the block-diagonal property does not require the data samples to have been grouped together according to their subspace memberships, because the solution produced by low rank representation is globally optimal and does not depend on the arrangements of the data samples [21].

However, in real application, different features have different capabilities of discriminative power. The feature with less discriminative power incurs more non-zero responses in the atoms belonging to different subspaces (*e.g.,* $\mathbf{Z}_{ij} \neq 0$, where samples i and j belong to different subspaces), while strongly discriminant feature will force the linear representation coefficients on different subspaces tend to zero (*e.g.,* $\mathbf{Z}_{ij} = 0$, where samples i and j belong to different subspaces). Discriminant feature will make the edges between points in different subspaces weak, such that spectral clustering can find the correct segmentation. Therefore, theoretically, discriminant feature will provide greater contribution to the final clustering results.

5 Experimental Results and Analysis

In this section, we perform experiments on several challenging image clustering datasets to present a comprehensive evaluation of the proposed method. We test our method on three applications: face clustering, scene clustering, and

generic object clustering. The statistics of all the datasets are summarized in Table 1.

Competitors: We compare the proposed method with seven representative clustering algorithms: the standard spectral clustering algorithm with the most informative view (SPC$_{best}$), LRR algorithm with the most informative view (LRR$_{best}$), robust multi-view spectral clustering via low-rank and sparse decomposition (RMSC) [5], diversity-induced multi-view subspace clustering (DiMSC) [47], low-rank tensor constrained multi-view subspace clustering (LTMSC) [22], the proposed method with unrotated coefficient tensor (Ut-SVD-MSC), learning and transferring deep ConvNet (convolutional neural network) representations with group-sparse factorization (GSNMF-CNN) [37]. The first two methods are the single view baselines. The RMSC, DiMSC, and LTMSC represent the state-of-the-art methods in multi-view clustering. Comparing with the method Ut-SVD-MSC is used to illustrate the advantage of the tensor rotation. The last approach, *i.e.*, GSNMF-CNN, does not belong to multi-view method but with the claim that it achieves state-of-the-art image clustering performance by using deep ConvNet.

Evaluation Methodology: Different experimental ***strategies*** are adopted for different applications. As for face clustering, we use relatively simple image features (*e.g.*, intensity, LBP, Gabor) to test the performance of different multi-view clustering methods. As for scene clustering, some sophisticated features (such as PHOW [43], CENTRIST [42], etc.) are considered as views to handle scene clustering. Besides traditional handcrafted features, we utilize the CNN feature trained on large-scale annotated dataset (ImageNet) to handle two challenging datasets, *i.e.*, MITIndoor-67 and Caltech-101 for scene clustering and generic object clustering, respectively. Since CNN feature is adopted in our experiments, it is necessary to compare with a state-of-the-art CNN based image clustering method, termed the GSNMF-CNN [37], which shows the transferability of the deep ConvNet trained on ImageNet to be used for enhancing image clustering. Due to the different scales, features, and challenges of different datasets, we leave the description of the detailed experimental setup to the corresponding section of each application.

Evaluation Measures. The evaluation of clustering results is a challenging problem. Two types of criteria are generally used for measuring cluster quality [32]: external and internal criteria. External criteria measures the agreement between the clustering result and an external input (usually the groundtruth of the dataset). Internal criteria, on the other hand, measures quality based on characteristic of the data and the partitioning result (*e.g.*, between-cluster and within-cluster scatter). However, good scores on an internal criterion do not necessarily translate into good effectiveness in an application [32]. Moreover, under subspace clustering setting, since data in a subspace are often distributed arbitrar-

Table 1: Statistics of different test datasets

Dataset	Images	Objective	Clusters
Yale	165	Face	15
Extended YaleB	640	Face	10
ORL	400	Face	40
Notting-Hill	4660	Face	5
Scene-15	4485	Scene	15
MITIndoor-67	5360	Scene	67
COIL-20	1440	Generic Object	20
Caltech-101	8677	Generic Object	101

ily and not around a centroid [20], standard internal criteria measurement that take advantage of the spatial proximity of the data can not be applicable. So, external criteria has been widely used for evaluating clustering performance [5, 22,47].

In our experiments, six popular external metrics are used to evaluate the performances [32,34]: Normalized Mutual Information (NMI), Accuracy (ACC), Adjusted Rank index (AR), F-score, Precision and Recall.

NMI can be information-theoretically interpreted. Suppose that C and C' represent the predicted partition and the groundtruth partition respectively, the NMI metric is calculated as:

$$NMI(C,C') = \frac{\sum_{i=1}^{K}\sum_{j=1}^{S}|C_i \cap C_j'|log\frac{N|C_i \cap C_j'|}{|C_i||C_j'|}}{\sqrt{(\sum_{i=1}^{K}|C_i|log\frac{C_i}{N})(\sum_{j=1}^{S}|C_j'|log\frac{C_j'}{N})}}, \quad (34)$$

For the definition of accuracy, suppose the clustering algorithm is tested on N samples. For a sample $\mathbf{x}_i$, the cluster label is denoted as r_i, and groundtruth is t_i. The accuracy is defined as follows:

$$ACC = \frac{\sum_{i=1}^{N}\delta(t_i,map(r_i))}{N}, \quad (35)$$

where

$$\delta(a,b) = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{otherwise,} \end{cases} \quad (36)$$

Function $map(x)$ denotes the best permutation mapping function gained by Hungarian algorithm [33], which maps cluster to the corresponding groundtruth label. So the more labels of samples are predicted correctly, the greater the accuracy is.

As for F-score, Precision, Recall, and AR, these four metrics view the clustering as a series of decisions, one for each the $N(N-1)/2$ pairs of samples on the dataset. The goal is to assign two samples to the same cluster if and only

if they are similar. For more details about their definitions, please refer to [32].

For each of the metrics, the higher it is, the better the performance is. Those metrics favor different properties in the clustering such that a comprehensive evaluation can be achieved. Note that in all dataset, the reported final results on those metrics are measured by the average and standard derivation of 20 runs. We highlight the best values in bold font in each table.

Only one parameter λ needs to be tuned, and we found its empirical value is within the range $[0.1, 2]$. More details about the parameter will be discussed in Section 5.4. The parameters in other competitors are set within ranges suggested by original papers, and we tune those parameters so as to show the best results. All experiments are implemented in Matlab on a workstation with 4.0GHz CPU, 32GB RAM, and TITANX GPU (12GB caches). To promote the culture of reproducible research, source codes and experimental results accompanying this paper can be achieved at https://www.researchgate.net/profile/Yuan_Xie4.

5.1 Experiments on Face Clustering

The *Yale*[3] face dataset contains 165 grayscale images of 15 individuals. There are 11 images per subject, one per different facial expression or configuration.

The *Extended YaleB*[4] dataset includes 38 individuals and around 64 near frontal images under different illuminations for each individual. Similarly to the works [21, 22], we use a part of images which contains the first 10 individuals, including 640 frontal face images.

The *ORL*[5] dataset consists of 40 distinct subjects, each of which contains 10 different images captured under different times, lighting, facial expressions, and facial details.

For all those datasets, similar to [22], three types of features are extracted: intensity, LBP [48] and Gabor [49]. The standard LBP features are extracted with the sampling size of 8 pixels, and the blocking number of 7×8. The Gabor feature is extracted with one scale $\lambda = 4$ at four orientations $\theta = \{0^o, 45^o, 90^o, 135^o\}$. Therefore, the dimensionalities of LBP and Gabor are 3304 and 6750, respectively.

As illustrated by the Table 2, LTMSC performs the second best in terms of all metrics on Yale dataset, while the proposed approach presents a clear advance over it, *e.g.,* 0.953 vs. 0.765 in NMI, and 0.963 vs. 0.741 in ACC. Table 4 gives the clustering results on the ORL dataset. It can be seen that quite a lot of methods achieve promising performance. However, our approach obtains a nearly perfect result in terms of all six metrics, *e.g.,* NMI 0.993, ACC 0.970,

---

[3] https://cvc.yale.edu/projects/yalefaces/yalefaces.html
[4] https://cvc.yale.edu/projects/yalefacesB/yalefacesB.html
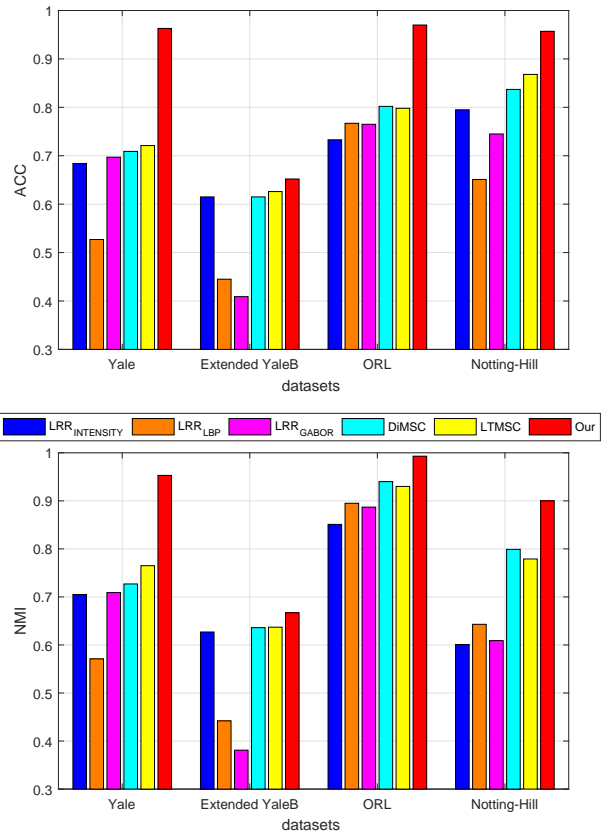[5] http://www.uk.research.att.com/facedatabase.html

Figure 4: Comparison among LRR with all the view features, DiMSC, LTMSC and the proposed t-SVD-MSC in terms of accuracy and NMI on face clustering datasets.

and Recall 0.991, which means that our method still outperforms all the alternative approaches significantly. As shown in Table 3, the improvement of t-SVD-MSC over other representative approaches (such as DiMSC and LTMSC) on Extended YaleB are not so much noticeable as on the above two datasets. We observe that, due to large variation of illumination, the LBP and Gabor features present significant lower capabilities of representation than intensity feature (see the second group bars in Fig. 4). Therefore, the basic assumption of the multi-view clustering might be violated so as to suffer the degradation of performance. This observation coincides with the corresponding conclusions obtained in [22, 47].

The dataset *Notting-Hill* [38] is constructed from the movie "Notting-Hill", where the faces of 5 main casts are collected, including 4660 faces in 76 tracks. The original dataset consists of the facial images of the size of 120×150, and we downsample each facial image to 40×50. The features utilized in this dataset are the same with the features used in other face clustering datasets. Table 5 shows the clustering result, where the proposed method also outperforms all other competitors in all metrics with clear large

Table 2: Clustering results (mean ± standard deviation) on *Yale*. We set $\lambda = 1.1$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.654 ± 0.009	0.618 ± 0.030	0.440 ± 0.011	0.475 ± 0.011	0.457 ± 0.011	0.500 ± 0.010
LRR$_{best}$	0.709 ± 0.011	0.697 ± 0.001	0.512 ± 0.005	0.547 ± 0.007	0.529 ± 0.005	0.567 ± 0.004
RMSC	0.684 ± 0.033	0.642 ± 0.036	0.485 ± 0.042	0.517 ± 0.043	0.500 ± 0.043	0.535 ± 0.044
DiMSC	0.727 ± 0.010	0.709 ± 0.003	0.535 ± 0.003	0.564 ± 0.010	0.543 ± 0.012	0.586 ± 0.009
LTMSC	0.765 ± 0.008	0.741 ± 0.002	0.570 ± 0.004	0.598 ± 0.006	0.569 ± 0.004	0.629 ± 0.005
Ut-SVD-MSC	0.756 ± 0.012	0.733 ± 0.005	0.584 ± 0.003	0.610 ± 0.006	0.591 ± 0.005	0.630 ± 0.006
t-SVD-MSC	$\mathbf{0.953 \pm 0.008}$	$\mathbf{0.963 \pm 0.006}$	$\mathbf{0.910 \pm 0.010}$	$\mathbf{0.915 \pm 0.007}$	$\mathbf{0.904 \pm 0.005}$	$\mathbf{0.927 \pm 0.007}$

Table 3: Clustering results (mean ± standard deviation) on *Extended YaleB*. We set $\lambda = 1.3$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.360 ± 0.014	0.366 ± 0.059	0.225 ± 0.018	0.308 ± 0.011	0.296 ± 0.010	0.310 ± 0.012
LRR$_{best}$	0.627 ± 0.040	0.615 ± 0.013	0.451 ± 0.002	0.508 ± 0.004	0.481 ± 0.002	0.539 ± 0.001
RMSC	0.157 ± 0.019	0.210 ± 0.013	0.060 ± 0.014	0.155 ± 0.012	0.151 ± 0.012	0.159 ± 0.013
DiMSC	0.636 ± 0.002	0.615 ± 0.003	0.453 ± 0.005	0.504 ± 0.006	0.481 ± 0.004	0.534 ± 0.004
LTMSC	0.637 ± 0.003	0.626 ± 0.010	0.459 ± 0.030	0.521 ± 0.006	0.485 ± 0.001	0.539 ± 0.002
Ut-SVD-MSC	0.479 ± 0.007	0.470 ± 0.011	0.274 ± 0.005	0.350 ± 0.007	0.327 ± 0.004	0.375 ± 0.005
t-SVD-MSC	$\mathbf{0.667 \pm 0.004}$	$\mathbf{0.652 \pm 0.000}$	$\mathbf{0.500 \pm 0.003}$	$\mathbf{0.550 \pm 0.002}$	$\mathbf{0.514 \pm 0.004}$	$\mathbf{0.590 \pm 0.004}$

Table 4: Clustering results (mean ± standard deviation) on *ORL*. We set $\lambda = 0.2$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.884 ± 0.002	0.725 ± 0.025	0.655 ± 0.005	0.664 ± 0.005	0.610 ± 0.006	0.728 ± 0.005
LRR$_{best}$	0.895 ± 0.006	0.773 ± 0.003	0.724 ± 0.020	0.731 ± 0.004	0.701 ± 0.001	0.754 ± 0.002
RMSC	0.872 ± 0.012	0.723 ± 0.007	0.645 ± 0.003	0.654 ± 0.007	0.607 ± 0.009	0.709 ± 0.004
DiMSC	0.940 ± 0.003	0.838 ± 0.001	0.802 ± 0.000	0.807 ± 0.003	0.764 ± 0.012	0.856 ± 0.004
LTMSC	0.930 ± 0.003	0.795 ± 0.007	0.750 ± 0.003	0.768 ± 0.004	0.766 ± 0.009	0.837 ± 0.005
Ut-SVD-MSC	0.874 ± 0.002	0.765 ± 0.001	0.666 ± 0.004	0.675 ± 0.005	0.643 ± 0.003	0.708 ± 0.002
t-SVD-MSC	$\mathbf{0.993 \pm 0.002}$	$\mathbf{0.970 \pm 0.003}$	$\mathbf{0.967 \pm 0.002}$	$\mathbf{0.968 \pm 0.003}$	$\mathbf{0.946 \pm 0.004}$	$\mathbf{0.991 \pm 0.003}$

Table 5: Clustering results (mean ± standard deviation) on *Notting-Hill*. We set $\lambda = 0.1$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.723 ± 0.008	0.816 ± 0.000	0.712 ± 0.020	0.775 ± 0.015	0.780 ± 0.018	0.776 ± 0.013
LRR$_{best}$	0.579 ± 0.003	0.794 ± 0.033	0.558 ± 0.007	0.653 ± 0.007	0.672 ± 0.007	0.636 ± 0.008
RMSC	0.585 ± 0.002	0.807 ± 0.013	0.496 ± 0.004	0.603 ± 0.005	0.621 ± 0.002	0.586 ± 0.011
DiMSC	0.799 ± 0.001	0.837 ± 0.021	0.787 ± 0.001	0.834 ± 0.001	0.822 ± 0.005	0.827 ± 0.009
LTMSC	0.779 ± 0.003	0.868 ± 0.000	0.777 ± 0.002	0.825 ± 0.002	0.830 ± 0.002	0.814 ± 0.004
Ut-SVD-MSC	0.837 ± 0.005	0.933 ± 0.015	0.847 ± 0.001	0.880 ± 0.005	0.900 ± 0.004	0.861 ± 0.009
t-SVD-MSC	$\mathbf{0.900 \pm 0.005}$	$\mathbf{0.957 \pm 0.010}$	$\mathbf{0.900 \pm 0.003}$	$\mathbf{0.922 \pm 0.003}$	$\mathbf{0.937 \pm 0.006}$	$\mathbf{0.907 \pm 0.005}$

margins. Our result might even be comparable with the state-of-the-art result achieved by [46] (with NMI 0.920 and ACC 0.934), where additional video-specific constraints are employed, *i.e.,* faces from the same track are likely to be from the same person, while faces do not belong to the same person if they appear together in the same video frame. Note that, the proposed method conduct video face clustering without any video-specific priori.

5.2 Experiments on Scene Clustering

Scene-15[6] dataset was gradually built by the works [39, 40, 41] with 15 categories, including office, kitchen, living room, bedroom, etc. Images are about 250×300 resolution, with 210 to 410 images per category. This dataset contains a wide range of outdoor and indoor scene environments. We extracted three kinds of handcrafted image features on this dataset: 1) Pyramid histograms of visual words (PHOW)[7] feature [43] which was extracted with 8 pixels' dense sampling step and 300 visual words, resulting in a 1800 dimensional feature. 2) Pairwise rotation invariant co-occurrence local binary pattern (PRI-CoLBP) feature, which was proven to be suitable for scene classification [44]. Different from other LBP variants, PRI-CoLBP not only captured the spatial context co-occurrence information effectively, but also possessed rotation invariance. We use gray-scale PRI-CoLBP and choose the simplest template so that the final dimensionality is $590 \times 2 = 1180$. 3) CENsus TRansform hISTogram (CENTRIST) feature [42]. It is a holistic representation which can capture structural properties such as rectangular shapes, flat surfaces and so on. By using the spatial pyramid technology, there are 1, 5, and 25 blocks for levels 0, 1, and 2, respectively. We use PCA to reduce the dimensionality of CENTRIST to 40, then a level 2 pyramid will result in a feature vector which has $40 \times (1+5+25) = 1240$ dimensions.

The clustering results are shown in Table 6, where the noticeable performance gain can be concluded by comparing with the second best LTMSC algorithm. Moreover, confusion matrices of the LTMSC and the proposed method is shown in Fig. 5, where row and column names are true and predicted labels respectively. Here, the cluster label is predicted by the best permutation mapping function used in the metric of ACC [33]. We can see that, compared with LTMSC, the proposed method wins in almost all categories in terms of clustering accuracy. The biggest confusion occurring between the indoor classes, such as bedroom and living room, coincides well with the the confusion distribution in [41].

MITIndoor-67 dataset was firstly introduced by [50], which is a challenging dataset including 15K indoor image spanning 67 different categories. It provides a training subset (5360 images) for classification task, and we perform clustering on this subset. Some samples are shown in Fig. 6. To the best of our knowledge, hardly any traditional clustering methods can achieve good performance in such a challenging dataset. To pursuit better performance, besides the features used in Scene-15, we further import the VGG-VD [51], which was pre-trained on ILSVRC12 [52], as a new view to complement handcrafted features. We use the activations of the penultimate layer for feature extraction, and resize its smaller dimension of each image to 448 for VGG19 while maintaining aspect ratio. The features are extracted from 5 scales $\{2^s, s = -1, -0.5, 0, 0.5, 1\}$, and all local features are pooling together regardless of scales and locations. The MatConvNet toolbox [53] is adopted to extract this feature.

Compared with GSNMF-CNN, our method gains significant improvement around 7.7%, 16.7%, 29.1%, 19.0%, 17.6% and 20.1% in terms of NMI, ACC, AR, F-score, Precision and Recall, respectively. Fig. 7 illustrates the comparison between SPC/LRR with different single view feature and the proposed t-SVD-MSC with multiview features. It can be observed that, the performance of SPC with raw VGG19 feature is much higher than that of SPC with traditional handcrafted features, so CNN feature is indeed an excellent representation even without any transfer. However, representing CNN feature in low-rank subspace will significantly degrade the performance, see the brown bar in LRR. The yellow bar in Fig. 7 indicates that, the proposed method could capture the complementarity between the handcrafted features and CNN feature, and boost the performance to a higher level.

5.3 Experiments on Generic Clustering

The *COIL-20*[8] dataset contains 1440 images of 20 object categories viewed from varying angels, with each category including 72 images. Similar to [22, 47], all the images are normalized to 32×32 with the same features used in Section 5.1 being extracted. As shown in Table 8, our method also outperforms three most recently published algorithms, *i.e.,* RMSC, DiMSC, and LTMSC, which further demonstrates the effectiveness of the proposed method.

The *Caltech-101* dataset [55] contains 8677 image of objects belonging to 101 categories, with about 40 to 800 images per category. Currently, image clustering method are usually evaluated under small-scale experimental configuration, *e.g.,* using relatively simple datasets (such as Yale and ORL), or cropping a small portion of categories from a large dataset (such as using 5, 7 and 20 sub-categories of

---

6 http://www-cvr.ai.uiuc.edu/ponce_grp/data/

7 This feature was extracted by using vlfeat toolbox [45]

8 http://www.cs.columbia.edu/CAVE/software/softlib/

Table 6: Clustering results (mean $\pm$ standard deviation) on *Scene-15*. We set $\lambda = 1.5$ in proposed t-SVD-MSC.

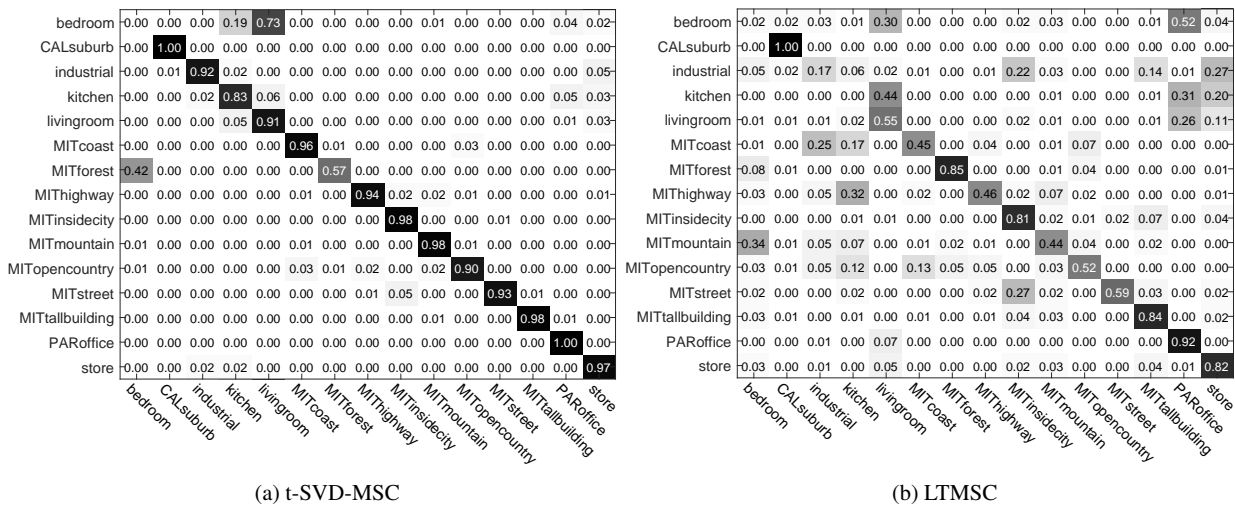| Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{\text{best}}$	0.421 ± 0.010	0.437 ± 0.015	0.270 ± 0.010	0.321 ± 0.022	0.314 ± 0.016	0.329 ± 0.020
LRR$_{\text{best}}$	0.426 ± 0.018	0.445 ± 0.013	0.272 ± 0.015	0.324 ± 0.010	0.316 ± 0.015	0.333 ± 0.015
RMSC	0.564 ± 0.023	0.507 ± 0.017	0.394 ± 0.025	0.437 ± 0.019	0.425 ± 0.021	0.450 ± 0.024
DiMSC	0.269 ± 0.009	0.300 ± 0.010	0.117 ± 0.012	0.181 ± 0.012	0.173 ± 0.016	0.190 ± 0.010
LTMSC	0.571 ± 0.011	0.574 ± 0.009	0.424 ± 0.010	0.465 ± 0.007	0.452 ± 0.003	0.479 ± 0.008
Ut-SVD-MSC	0.555 ± 0.007	0.510 ± 0.005	0.375 ± 0.003	0.422 ± 0.004	0.389 ± 0.010	0.460 ± 0.008
t-SVD-MSC	$\mathbf{0.858 \pm 0.007}$	$\mathbf{0.812 \pm 0.007}$	$\mathbf{0.771 \pm 0.003}$	$\mathbf{0.788 \pm 0.001}$	$\mathbf{0.743 \pm 0.006}$	$\mathbf{0.839 \pm 0.003}$

Table 7: Clustering results (mean $\pm$ standard deviation) on MITIndoor-67. We set $\lambda = 0.2$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{\text{best}}^{\text{CNN}}$	0.559 ± 0.009	0.443 ± 0.011	0.304 ± 0.011	0.315 ± 0.013	0.294 ± 0.010	0.340 ± 0.014
LRR$_{\text{best}}^{\text{CNN}}$	0.226 ± 0.006	0.120 ± 0.004	0.031 ± 0.007	0.045 ± 0.004	0.044 ± 0.006	0.047 ± 0.004
RMSC	0.342 ± 0.004	0.232 ± 0.009	0.110 ± 0.003	0.123 ± 0.002	0.121 ± 0.003	0.125 ± 0.003
DiMSC	0.383 ± 0.003	0.246 ± 0.000	0.128 ± 0.005	0.141 ± 0.004	0.138 ± 0.001	0.144 ± 0.002
LTMSC	0.546 ± 0.004	0.431 ± 0.002	0.280 ± 0.008	0.290 ± 0.002	0.279 ± 0.006	0.306 ± 0.005
GSNMF-CNN	0.673 ± 0.003	0.517 ± 0.003	0.264 ± 0.005	0.372 ± 0.002	0.367 ± 0.004	0.381 ± 0.001
Ut-SVD-MSC	0.518 ± 0.010	0.386 ± 0.007	0.245 ± 0.013	0.256 ± 0.007	0.249 ± 0.006	0.263 ± 0.006
t-SVD-MSC	$\mathbf{0.750 \pm 0.007}$	$\mathbf{0.684 \pm 0.005}$	$\mathbf{0.555 \pm 0.005}$	$\mathbf{0.562 \pm 0.008}$	$\mathbf{0.543 \pm 0.005}$	$\mathbf{0.582 \pm 0.004}$

the Caltech-101 dataset [36,54]). Here, we use the instances from all the categories to test whether the proposed method could handle relatively large and challenging dataset, and compare the clustering performance with the state-of-the-art unsupervised CNN-based clustering method, *i.e.,* GSNMF-CNN. We use the deep feature Inception V3 [56] in this dataset, since it achieves better results than VGG19. It is also extracted from the activations of the penultimate layer, leading to a 2048-dimensional feature vector. The same feature is used in GSNMF-CNN.

The results are shown in Table 9. By using the more powerful deep feature, the baseline algorithms (SPC and LRR) perform better than some sophisticated methods such as RMSC and DiMSC. This is probably because RMSC and DiMSC suffer from the less representation capabilities of the handcrafted features on this dataset. Surprisingly, the LTMSC is not affected by some degenerate views and even does sightly better than GSNMF-CNN. Furthermore, in the testing of all the datasets, three observations need to be worth noting: (1) The method Ut-SVD-MSC sometimes shows comparable performance to other state-of-the-art approaches, but still has a significant gap with regard to the proposed method. Because Ut-SVD-MSC does not make full use of the structure of the self-representations coefficient tensor, while the proposed method preserves those coefficients in Fourier domain, as illustrated in Fig. 3. (2) Due to noise or error in

measurement, some of the available views may be misleading in revealing the true structure of the data, so that including them in the clustering process may have negative influence. The corresponding phenomena appear several times, for example, DiMSC on Scene-15, (RMSC, DiMSC, and LTMSC) on MITIndoor-67, and (RMSC and DiMSC) on Caltech-101, where their performances are worse than directly using spectral clustering with single best feature. On the contrary, the proposed method exhibits robustness to the existence of degenerate views. (3) CNN feature is usually better than handcrafted features in terms of representation capability. But this does not mean that it is enough for clustering only by using CNN feature. The performance gains of the proposed method on all these datasets confirm the "complementary principle" in the multi-view learning, which states that each view of the data may contain some knowledge that other views do not have. The complementary information between CNN feature and handcrafted features can help to improve the clustering performance.

(a) t-SVD-MSC

(b) LTMSC

Figure 5: Comparison the confusion matrices between LTMSC and the proposed t-SVD-MSC on *Scene-15* dataset.

Table 8: Clustering results (mean $\pm$ standard deviation) on *COIL-20*. We set $\lambda = 0.25$ in proposed t-SVD-MSC.

Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}$	0.806 ± 0.008	0.672 ± 0.063	0.619 ± 0.018	0.640 ± 0.017	0.596 ± 0.021	0.692 ± 0.013
LRR$_{best}$	0.829 ± 0.006	0.761 ± 0.003	0.720 ± 0.020	0.734 ± 0.006	0.717 ± 0.003	0.751 ± 0.002
RMSC	0.800 ± 0.017	0.685 ± 0.045	0.637 ± 0.044	0.656 ± 0.042	0.620 ± 0.057	0.698 ± 0.026
DiMSC	0.846 ± 0.002	0.778 ± 0.022	0.732 ± 0.005	0.745 ± 0.005	0.739 ± 0.007	0.751 ± 0.003
LTMSC	0.860 ± 0.002	0.804 ± 0.011	0.748 ± 0.004	0.760 ± 0.007	0.741 ± 0.009	0.776 ± 0.006
Ut-SVD-MSC	0.841 ± 0.004	0.788 ± 0.005	0.732 ± 0.003	0.746 ± 0.006	0.731 ± 0.002	0.760 ± 0.002
t-SVD-MSC	$\mathbf{0.884 \pm 0.005}$	$\mathbf{0.830 \pm 0.000}$	$\mathbf{0.786 \pm 0.003}$	$\mathbf{0.800 \pm 0.004}$	$\mathbf{0.785 \pm 0.007}$	$\mathbf{0.808 \pm 0.001}$

Table 9: Clustering results (mean $\pm$ standard deviation) on *Caltech-101*. We set $\lambda = 0.5$ in proposed t-SVD-MSC.

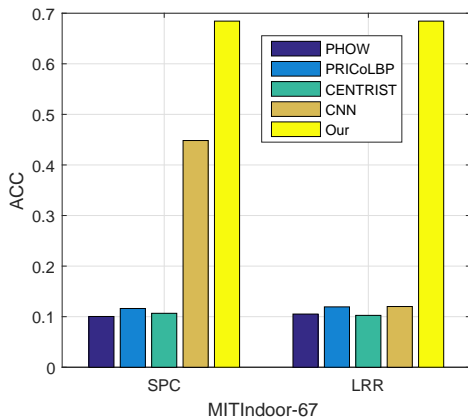| Method	NMI	ACC	AR	F-score	Precision	Recall
SPC$_{best}^{CNN}$	0.723 ± 0.032	0.484 ± 0.019	0.319 ± 0.014	0.340 ± 0.025	0.597 ± 0.018	0.235 ± 0.020
LRR$_{best}^{CNN}$	0.728 ± 0.014	0.510 ± 0.009	0.304 ± 0.017	0.339 ± 0.008	0.627 ± 0.012	0.231 ± 0.010
RMSC	0.573 ± 0.047	0.346 ± 0.036	0.246 ± 0.031	0.258 ± 0.027	0.457 ± 0.033	0.182 ± 0.031
DiMSC	0.589 ± 0.011	0.351 ± 0.008	0.226 ± 0.003	0.253 ± 0.007	0.362 ± 0.010	0.191 ± 0.007
LTMSC	0.788 ± 0.005	0.559 ± 0.012	0.393 ± 0.007	0.403 ± 0.003	0.670 ± 0.009	0.288 ± 0.012
GSNMF-CNN	0.775 ± 0.010	0.534 ± 0.012	0.246 ± 0.008	0.275 ± 0.006	0.230 ± 0.004	$\mathbf{0.347 \pm 0.006}$
Ut-SVD-MSC	0.742 ± 0.008	0.483 ± 0.003	0.334 ± 0.002	0.344 ± 0.004	0.612 ± 0.002	0.239 ± 0.002
t-SVD-MSC	$\mathbf{0.858 \pm 0.003}$	$\mathbf{0.607 \pm 0.005}$	$\mathbf{0.430 \pm 0.005}$	$\mathbf{0.440 \pm 0.010}$	$\mathbf{0.742 \pm 0.007}$	0.323 ± 0.009

5.4 Model Analysis

5.4.1 Contributions of Multi-View Features

In this subsection, we will analyze the contributions of multiple features to the final clustering result from the experimental perspective. In Fig. 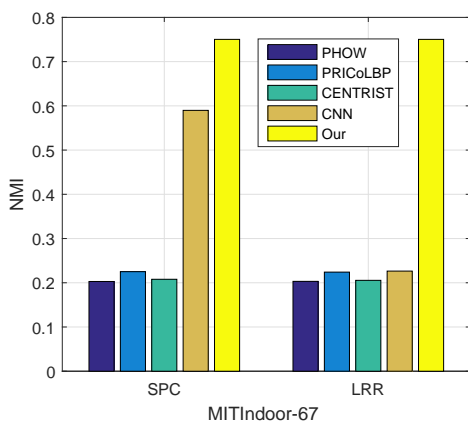8, we present the view-specific affinity matrices and the final affinity matrix for the ORL dataset. Since the LBP feature owns much more expressive capability than other two low-level feature in face description, the matrix corresponding to LBP (Fig. 8 (b)) reveals the underlying clustering structures more clearly, which further validates our conclusion that discriminant feature contributes more to final result. Similar observation can be seen

Figure 6: Samples from MITIndoor-67 dataset.

(a)

(b)

Figure 7: Comparison between SPC/LRR with the single view feature and the proposed t-SVD-MSC in terms of accuracy and NMI on MITIndoor-67 dataset.

in Fig. 9, where the block-diagonal structures are loomed and apparent for PRICoLBP and CNN-VGG19 features, respectively, while the affinity matrices for PHOW and CENTRIST features can hardly see the block-diagonal structures. This observation coincides with the conclusion that PRI-CoLBP is more suitable for scene classification than the other two handcrafted features [3], *i.e.,* PHOW and CEN-TRIST. Obviously, two discriminant features PRICoLBP and

CNN-VGG19 contribute more to final clustering, which is demonstrated in the final affinity matrix in Fig. 9 (e).

Furthermore, we analyze the changes of affinity matrix for all the views before and after the proposed optimization procedure, so that the influence of the proposed model upon each view can be explored more thoroughly. To this end, the LRR solution of each feature (denotes by $\widetilde{\mathbf{Z}}^{(v)}$) is employed to initialize the $\mathbf{Z}^{(v)}$ (see the step 1 in Algorithm 3) so as to obtain the optimized self-representation matrix (denotes by $\mathbf{Z}^{*(v)}$) in a unified tensor space. Fig. 10 shows the comparison of clustering accuracy by using $\{\mathbf{Z}^{(v)}\}_{v=1}^{V}$ before (blue bar) and after (magenta bar) optimization on the ORL and MITIndoor-67 datasets. Two key observations are listed as follows: 1) The feature type, which provides most contribution to final clustering, is kept the same before and after the optimization. 2) The performance of all the views are improved simultaneously, which is an evidence that the complementary information can be captured and propagated among all the views in high-order tensor space.

5.4.2 Parameter Tuning

We will discuss the parameter tuning in the proposed multi-view clustering model. Fortunately, the proposed model contains only one parameter λ needed to be chosen. The parameter $\lambda > 0$ is used to balance the effects of the two parts in (21). Commonly, the choice of λ depends on the prior knowledge of the error level of the data. Fig. 11 shows the evaluation results on Yale and Scene-15 datasets by using different values of λ. Although the parameter λ plays an important role on performance, most results are still better than other competitors, as can be seen from the red horizontal lines in Fig. 11 (a) and (b), which denote the second best indexes. The same indexes are not presented in Fig. 11 (c) and (d), as they are far below the minimal values of vertical ordinate. This implies the partial stability of the proposed model while λ is varying.

5.4.3 Stability

Overall, compared with all those competitors, the proposed method keeps relatively low standard deviation. Actually, the variance is mainly caused by the numerical calculation error of matrix inverse and matrix SVD, as well as the k-means algorithm in the final spectral clustering step. The matrix inverse operation is involved in optimization of $\mathbf{Z}^{(v)}$ in the step (4) in Algorithm 3, the SVD of complex matrix operation arises in updating $\mathcal{G}$ in the step (11) (details in Algorithm 2). The final step (18), *i.e.,* the spectral clustering, also can incur the variance, since it contains the real matrix SVD and k-means algorithm. As we know, k-means algorithm is sensitive to initialization. However, relatively good affinity matrix provided by the proposed t-SVD-MSC can
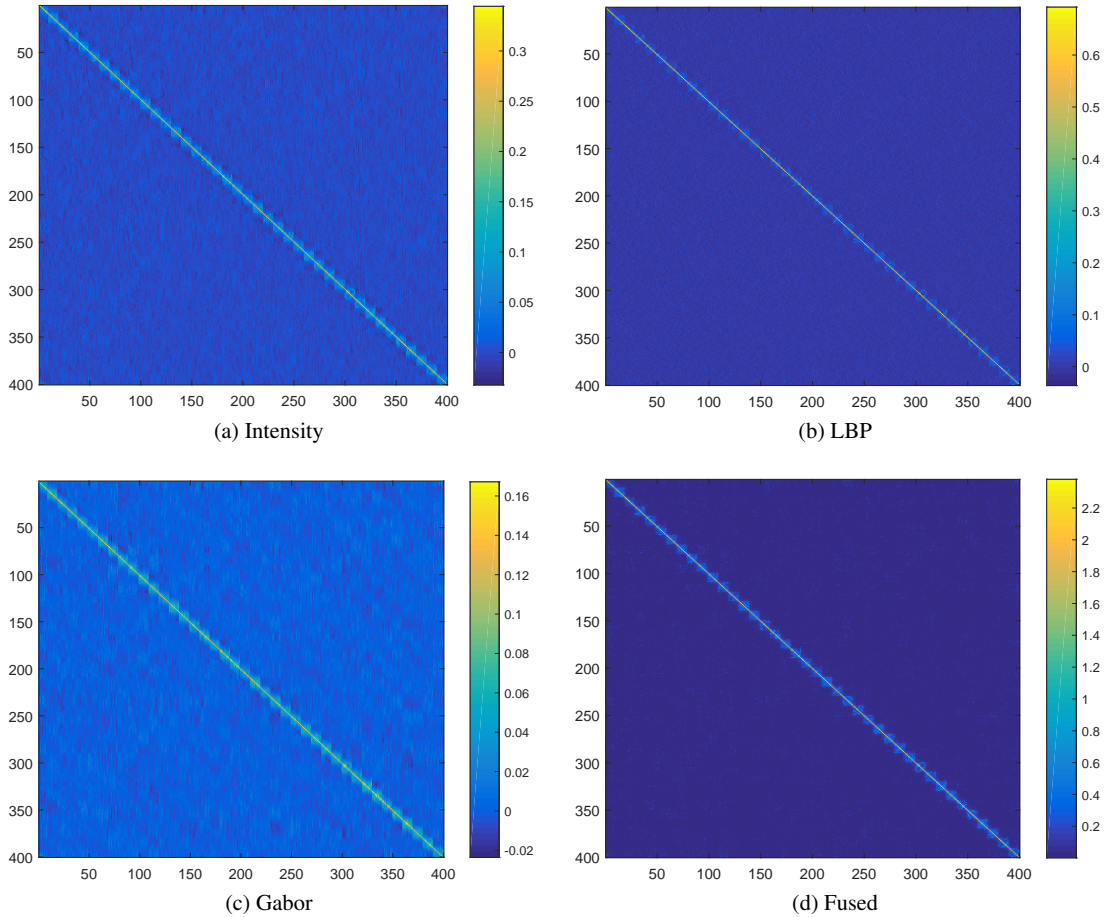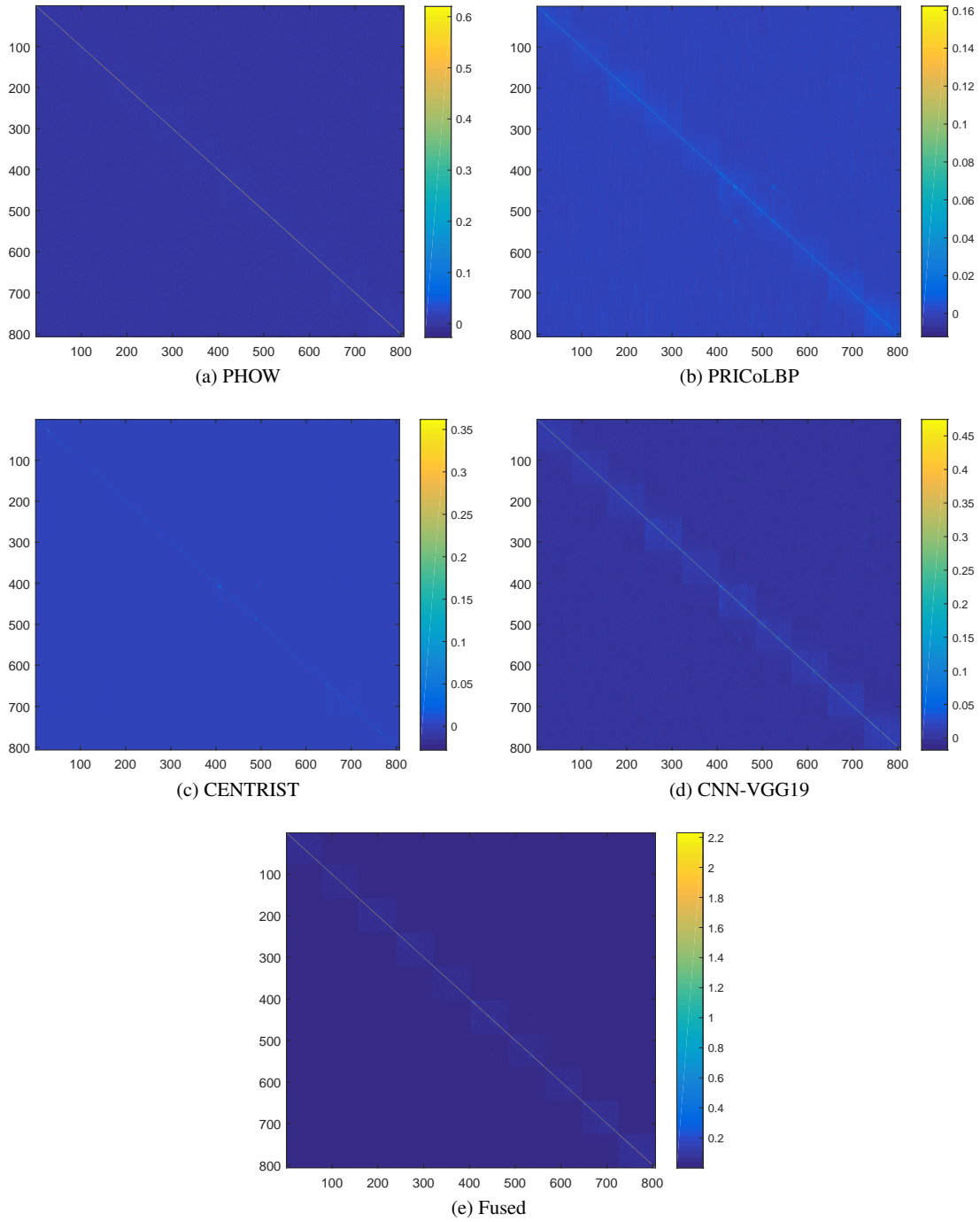
Figure 8: (a)∼(c) The illustration of affinity matrices $\mathbf{A}^{(v)} = \frac{1}{2}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|), v = 1, 2, 3$ for all the views/features in ORL datasets. (d) The final affinity matrix $\mathbf{Z} = \frac{1}{V}\sum_{v=1}^{V}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|)/2$.

reduce the variance produced by k-means to some extent. This can be evidenced by observing the standard deviation of the proposed method in all the result tables (Table 2 ∼ Table 9), which indicates that the proposed t-SVD-MSC is a stable multi-view subspace clustering method.

5.4.4 Convergence and Computational Complexity

Thanks to the rotation of the coefficient tensor (see Fig. 3), the computational complexity for SVD is reduced to $\mathcal{O}(N^2V^2)$, compared with $\mathcal{O}(N^3V)$ for the unrotated tensor. In practice, the proposed optimization method for t-SVD-MSC converges fast, which is illustrated in Fig. 12. The two curves record the reconstruction error (defined in Eq. (37)) and match error (Eq. (38)) in each iteration step. Additionally, the CPU times needed by the proposed method and its competitors are illustrated in Table 10. Compared with other state-of-the-art algorithms, the proposed model is advanced not only in clustering performance, but also in saving time, especially when handling large-scale dataset (see the third row in Table

10).

$$\text{Reconstruction Error} \doteq \frac{1}{V}\sum_{v=1}^{V}||\mathbf{X}^{(v)} - \mathbf{X}^{(v)}\mathbf{Z}^{(v)} - \mathbf{E}^{(v)}||_{\infty}$$

(37)

$$\text{Match Error} \doteq \frac{1}{V}\sum_{v=1}^{V}||\mathbf{Z}^{(v)} - \mathbf{G}^{(v)}||_{\infty} \quad (38)$$

Table 10: Comparison of CPU time of different methods, s, m and h denote second, minute and hour, respectively.

	RMSC	DiMSC	LTMSC	Our
ORL	21.07s	21.02s	42.97s	37.61s
Scene-15	187.01m	221.46m	135.68m	27.69m
Caltech-101	$\sim 24h$	$> 27h$	$\sim 5h$	$\sim 2h$

(a) PHOW

(b) PRICoLBP

(c) CENTRIST

(d) CNN-VGG19

(e) Fused

Figure 9: (a)∼(c) The illustration of affinity matrices $\mathbf{A}^{(v)} = \frac{1}{2}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|), v = 1, \ldots, 4$ for all the views/features on MITIndoor-67 dataset. (d) The final affinity matrix $\mathbf{Z} = \frac{1}{V}\sum_{v=1}^{V}(|\mathbf{Z}^{(v)}| + |\mathbf{Z}^{(v)^{\mathbf{T}}}|)/2$. To see structure clearly, only the first ten clusters are displayed.
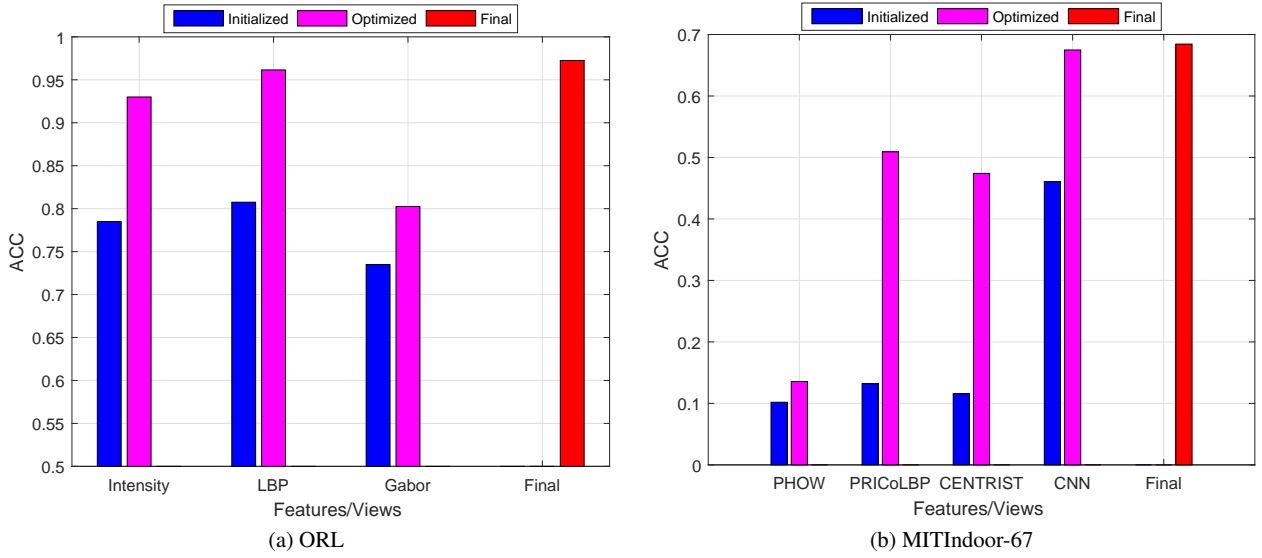
Figure 10: The comparison of clustering accuracy by using coefficient matrices $\{\mathbf{Z}^{(v)}\}_{v=1}^{V}$ before (blue bar) and after (magenta bar) optimization on ORL and MITIndoor-67 datasets.

6 Conclusions

In this paper, a t-SVD based tensor low-rank subspace model is proposed to perform data clustering from multi-view features. To capture the complementary information from different views, the proposed method constrains the rotated subspace coefficient tensor through tensor multi-rank to explore the high order correlations. Then, the multi-view clustering problem have been formulated in a unified optimization framework, and an efficient algorithm is proposed to find the optimal solution. The proposed t-SVD-MSC is then applied to three kinds of image clustering datasets: face clustering, scene clustering, and generic object clustering. Extensive evaluation of our method is conducted on several challenge datasets, where a clear advance over contemporary MSC approaches is achieved. Meanwhile, the proposed model presents strongly robust to degenerate views. By utilizing CNN feature as a new view, the results show that t-SVD-MSC is very competitive with the recent proposed CNN based clustering approach on challenge datasets.

7 Appendix

Proof of the Theorem 2:

Proof In Fourier domain, the optimization problem of Eq. (29) can be reformulated as

$$\boldsymbol{\mathcal{G}}_f = \underset{\boldsymbol{\mathcal{G}}_f}{\operatorname{argmin}}\ \tau||\mathrm{bdiag}(\boldsymbol{\mathcal{G}}_f)||_* + \frac{1}{2n_3}||\boldsymbol{\mathcal{G}}_f - \boldsymbol{\mathcal{F}}_f||_F^2 \quad (39)$$

$$= \underset{\boldsymbol{\mathcal{G}}_f}{\operatorname{argmin}}\ \sum_{j=1}^{n_3}\tau'||\boldsymbol{\mathcal{G}}_f^{(j)}||_* + \frac{1}{2}||\boldsymbol{\mathcal{G}}_f^{(j)} - \boldsymbol{\mathcal{F}}_f^{(j)}||_F^2, \quad (40)$$

where $\tau' = n_3\tau$. Then Eq. (40) can be separated into n_3 independent subproblems,

$$\boldsymbol{\mathcal{G}}_f^{(j)} = \underset{\boldsymbol{\mathcal{G}}_f^{(j)}}{\operatorname{argmin}}\ \tau'||\boldsymbol{\mathcal{G}}_f^{(j)}||_* + \frac{1}{2}||\boldsymbol{\mathcal{G}}_f^{(j)} - \boldsymbol{\mathcal{F}}_f^{(j)}||_F^2, \quad (41)$$

where $j = 1, 2, \ldots, n_3$. Note that Eq. (41) is the *F*-norm based nuclear norm low rank matrix approximation problem represented in Fourier domain. According to the result on gradients of unitarily invariant norms, Eq. (41) can also be solved by a soft-thresholding operation [27],
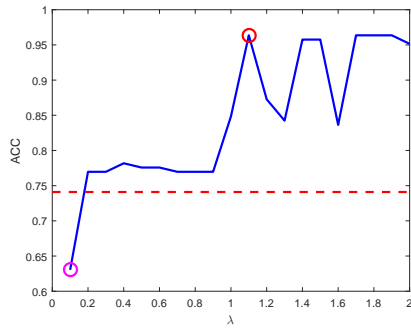
$$\boldsymbol{\mathcal{G}}_f^{(j)} = D_{\tau'}(\boldsymbol{\mathcal{F}}_f^{(j)}) = \boldsymbol{\mathcal{U}}_f^{(j)}\boldsymbol{\mathcal{S}}_{f,\tau'}^{(j)}\boldsymbol{\mathcal{V}}_f^{(j)\mathrm{T}}, \quad (42)$$

here, $\boldsymbol{\mathcal{G}}_f^{(j)} = \boldsymbol{\mathcal{U}}_f^{(j)}\boldsymbol{\mathcal{S}}_f^{(j)}\boldsymbol{\mathcal{V}}_f^{(j)\mathrm{T}}$, $\mathcal{D}_{\tau'}(\cdot)$ is the SVT operation with with threshold τ' (see Section 3), and $\mathcal{S}_{f,\tau'}^{(j)} = \mathrm{diag}\{(\mathcal{S}_f^{(j)}(i,i) - \tau')_+\}$. Then, we can get
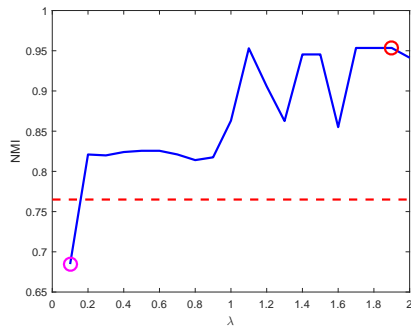
$$\boldsymbol{\mathcal{G}}_f = \mathrm{bdfold}\left\{\mathrm{bdiag}(\boldsymbol{\mathcal{U}}_f)\mathrm{bdiag}(\boldsymbol{\mathcal{S}}_{f,\tau'})\mathrm{bdiag}(\boldsymbol{\mathcal{V}}_f)^{\mathrm{T}}\right\} \quad (43)$$

and

$$\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{U}} * \tilde{\boldsymbol{\mathcal{S}}} * \boldsymbol{\mathcal{V}}^{\mathbf{T}} \quad (44)$$
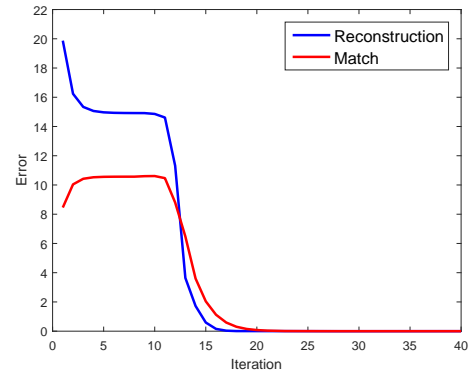
(a) ACC on Yale

Figure 12: Convergence curves on Scene-15 dataset.

where $\tilde{\mathcal{S}} = \text{ifft}(\mathcal{S}_{f,\tau'}, [\,], 3)$. Suppose that $\mathcal{J}$ is an $n_1 \times n_2 \times n_3$ f-diagonal tensor whose diagonal element in the Fourier domain is $\mathcal{J}_f(i, i, j) = (1 - \frac{\tau'}{\mathcal{S}_f^{(j)}(i,i)})_+$, then we have that $\mathcal{S}_{f,\tau'}(i, i, :) = \mathcal{S}_f(i, i, :)\mathcal{J}_f(i, i, :)$ in the Fourier domain, as well as $\tilde{\mathcal{S}}(i, i, :) = \mathcal{S}(i, i, :) \circ \mathcal{J}(i, i, :)$ in the original domain. Because both $\mathcal{S}$ and $\mathcal{J}$ are f-diagonal, $\tilde{\mathcal{S}}$ can be formulated as $\tilde{\mathcal{S}} = \mathcal{S} * \mathcal{J}$. Therefore, a convolution based tubal-shrinkage operator in the original domain is equivalent to the tensor SVT in the Fourier domain.

(b) NMI on Yale

References

1. C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv:1304.5634*, 2013.
2. V. R. de Sa, "Spectral clustering with two views," In *ICML*, 2005.
3. D. Zhou, and C. Burges, "Spectral clustering and transductive learning with multiple views," In *ICML*, 2007.
4. W. Tang, Z. Lu, and I. S. Dhillon, "Clustering with multiple graphs," In *ICDM*, 2009.
5. R. Xia, Y. Pan, L. Du, and J. Yin, "Robust multi-view spectral clustering via low-rank and sparse decomposition," In *AAAI*, 2014.
6. L. Shu, and L. J. Latecki, "Integration of single-view graphs with diffusion of tensor product graphs for multi-view spectral clustering," In *ACML*, 2015.
7. G. Tzortzis, and A. Likas, "Kernel-based weighted multi-view clustering," In *ICDM*, pp. 675-684, 2012.
8. S. Bickel, and T. Scheffer, "Multi-view clustering," In *ICDM*, pp. 19-26, 2004.
9. A. Kumar, and H. Daumé III, "A co-training approach for multi-view spectral clustering," In *ICML*, 2011.
10. A. Kumar, P. Rai, and H. Daumé III, "Co-regularized multiview spectral clustering," In *NIPS*, 2011.
11. K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," *Proc. International Conference on Machine Learning*, pp. 129-136, 2009.
12. M. B. Blaschko, and C. H. Lampert, "Correlational spectral clustering," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1-8, 2008.
13. Y. Luo, D. Tao, K. Ramamohanarao, C. Xu, and Y. Wen, "Tensor canonical correlation analysis for multi-view dimension reduction," *IEEE Trans. on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3111-3124, 2015.
14. W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," *Proc. International Conference on Machine Learning*, 2015.

(c) ACC on Scene-15

(d) NMI on Scene-15

Figure 11: Parameter (λ) tuning in terms of ACC and NMI on Yale and Scene-15 datasets.

15. M. White, X. Zhang, D. Schuurmans, and Y. I. Yu, "Convex multi-view subspace learning," In NIPS, 2012.

16. M. Kilmer, K. Braman, N. Hao, and R. Hoover, "Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 1, pp. 148-172, 2013.

17. Z. Zhang, G. Ely, S. Aeron, N. Hao, and M. Kilmer, "Novel methods for multilinear data completion and de-noising based on tensor-SVD," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2014.

18. O. Semerci, Ning Hao, M. Kilmer, and E. Miller, "Tensor-based formulation and nuclear norm regularization for multienergy computed tomography," *IEEE Trans. Image Processing*, vol. 23, no. 4, pp. 1678-1693, 2014.

19. C. Lu, J. Feng, Y. Chen, W. Liu, and Z. Lin, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," In *CVPR*, 2016.

20. E. Elhamifar, and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and application," In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765-2781, 2013.

21. G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171-184, 2013.

22. C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao. "Low-rank tensor constrained multiview subspace clustering," *Proc. International Conference on Computer Vision*, pp. 2439-2446, 2015.

23. J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 208-220, 2013.

24. A. Ng, M. Jordan, Y. Weiss. On spectral clustering: Analysis and an algorithm. *In Advances in Neural Information Processing Systems*, 2001.

25. Z. Lin, M. Chen, Y. Ma. The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Technical Report UILU-ENG-09-2215*, UIUC, 2009.

26. Z. Lin, R. Liu, Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. *In Advances in Neural Information Processing Systems*, pp. 612-620, 2011.

27. J. Cai, E. Candes, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optimization*, vol. 20, no. 4, pp. 1956-1982, 2010.

28. B. Mohar. The Laplacian spectrum of graphs. *In Graph Theory, Combinatorics, and Applications*, pp. 871-898, Wiley.

29. T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455-500, 2009.

30. J. Eckstein and D. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Math. Programming*, vol. 55, pp. 293-318, 1992.

31. M. E. Kilmer and C. D. Martin, "Factorization strategies for third-order tensors," *Linear Algebra and its Applications*, vol. 435, no. 3, pp. 641-658, 2011.

32. M. Christopher D., P. Raghavan, and H. Schtze, *Introduction to Information Retrieval*, vol. 1, Cambridge University Press, Cambridge, 2008.

33. D. Cai, X. He, and J. Han, "Document clustering using locality preserving indexing," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 12, pp. 1624-1637, 2005.

34. H. Lawrence and A. Phipps, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193-218, 1985.

35. D. Dai, and L. Van Gool, "Unsuperviesed high-level feature learning by ensemble projection for semi-supervised image classification and image clustering," Technical report, ETH Zurich, 2015.

36. C. Lu, S. Yan, and Z. Lin, "Convex sparse spectral clustering: single-view to multi-view," *IEEE Trans. on Image Processing*, vol. 25, no. 6, pp. 2833-2843, 2016.

37. L. Gui, and L. P. Morency, "Learning and transferring deep ConvNet representations with group-sparse factorization," *Proc. IEEE International Conference on Computer Vision*, 2015.

38. Y. Zhang, C. Xu, H. Lu, and Y. M. Huang, "Character identification in feature-length films using global face-name matching," *IEEE Trans. on Multimedia*, vol. 11, no. 7, pp. 1276-1288, 2009.

39. A. Oliva, and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, pp. 145-175, 2001.

40. F. -F. Li, and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 524-531, 2005.

41. S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 2169-2178, 2006.

42. J. Wu, and J. M. Rehg. "Centrist: A visual descriptor for scene categorization," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489-1501, 2011.

43. A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forest and ferns," In *Proc. IEEE International Conference on Computer Vision*, 2007.

44. X. Qi, R. Xiao, C. Li, Y. Qiao, J. Guo, and X. Tang, "Pairwise rotation invariant co-occurrence local binary pattern," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol. 36, no. 11, 2014.

45. A. Vedaldi, and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.

46. X. Cao, C. Zhang, C. Zhou, H. Fu, and H. Foroosh, "Constrained multi-view video face clustering," *IEEE Trans. on Image Processing*, vol. 24, no. 11, pp. 4381-4393, 2015.

47. X. Cao, C. Zhang, H. Fu, S. Liu, and H. Zhang, "Diversity-induced multi-view subspace clustering," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015.

48. T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution grayscale and rotation invariant texture classification with local binary patterns," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.

49. M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Trans. on Computer*, vol. 42, no. 3, pp. 300-311, 1993.

50. A. Quattoni, and A. Torralba, "Recognizing indoor scenes," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 413-420, 2009.

51. K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations*, 2014.

52. J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009.

53. A. Vedaldi, and K. Lenc, "Matconvnet - convolutional neural networks for matlab," http://www.vlfeat.org/matconvnet/.

54. H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," *Proc. IEEE International Conference on Computer Vision*, 2015.

55. F. -F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59-70, 2007.

56. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception architecture for computer vision," http://arxiv.org/abs/1512.00567v1.