Weighted Block-Sparse Low Rank Representation for Face Clustering in Videos

Shijie Xiao[1], Mingkui Tan[2], and Dong Xu[1]

[1] School of Computer Engineering, Nanyang Technological University, Singapore
[2] School of Computer Science, University of Adelaide, Australia

Abstract. In this paper, we study the problem of face clustering in videos. Specifically, given automatically extracted faces from videos and two kinds of prior knowledge (the face track that each face belongs to, and the pairs of faces that appear in the same frame), the task is to partition the faces into a given number of disjoint groups, such that each group is associated with one subject. To deal with this problem, we propose a new method called weighted block-sparse low rank representation (WBSLRR) which considers the available prior knowledge while learning a low rank data representation, and also develop a simple but effective approach to obtain the clustering result of faces. Moreover, after using several acceleration techniques, our proposed method is suitable for solving large-scale problems. The experimental results on two benchmark datasets demonstrate the effectiveness of our approach.

Keywords: low rank representation, block-sparsity, subspace clustering, face clustering.

1 Introduction

Face clustering in videos [7, 28] is an important but challenging problem in computer vision. Specifically, given the faces automatically extracted from a piece of video (*e.g.* a movie or an episode of TV series), the task is to partition these faces into a given number of clusters, such that the faces assigned to each cluster belong to the same subject. Face clustering is important for many related applications, such as video organization, video segmentation and content based video retrieval. However, the video face clustering problem is challenging because the faces are generally captured in uncontrolled environments and thus the faces are with large variations in poses, illuminations and facial expressions. Moreover, the faces may be occluded by hands, glasses or other objects.

Instead of treating each face individually, existing works [7, 28] (see Section 2 for more details) often consider the information based on face tracks (where each face track is a sequence of faces) when performing the face clustering in videos. Thus, the following two kinds of relationships among faces can be directly explored:

1. The *inner-track* relation: any two faces in the same face track should belong to the same subject.

D. Fleet et al. (Eds.): ECCV 2014, Part VI, LNCS 8694, pp. 123–138, 2014.

2. The *inter-track* relation: if any two faces appear in the same frame of the video, the corresponding two face tracks should belong to different subjects.

It is worth mentioning that, with such prior knowledge, the face clustering problem can be considered as "self-supervised" [7].

On the other hand, the subspace clustering problem [1, 10, 17] is studied in many recent works such as [20, 10, 17, 21, 27]. Specifically, given the data sampled from a union of (linear) subspaces, the goal of subspace clustering is to partition the data into several clusters, so that each cluster corresponds to one subspace. Among the subspace clustering methods, the compressed sensing based approaches [10, 17] assume that the data is self-expressible (*i.e.*, each data point in its subspace can be represented as a linear combination of the data points from the same subspace). Particularly, the low rank representation based methods [17], which encourage the data representation to be low-rank, have been successfully used in various applications. For example, the result of subspace clustering can be obtained based on the learnt data representation. Unfortunately, these unsupervised methods cannot effectively utilize the possible supervision (such as the prior knowledge) in our problem.

Motivated by the above two aspects, in this paper, we propose a low-rank representation based approach for face clustering in videos, by effectively exploiting the available prior knowledge (*i.e.* the inner-track and inter-track relations). Specifically, we design a weighted block-sparse regularizer on the data representation to incorporate both kinds of prior knowledge, so that the resultant data representation should be more discriminative. Ideally, the faces in any face track are linearly represented only by the tracks of faces from the same subject/subspace, because we encourage the sparsity of the blocks (which correspond to the face tracks) in the data representation. Moreover, if any faces from two face tracks appear in the same frame, the corresponding representation coefficients are penalized. Accordingly, we name the proposed method as weighted block-sparse low rank representation (WBSLRR). We adopt the alternating direction method (ADM) [4, 17] to solve the optimization problem and we further use several acceleration techniques to make the algorithm scalable to large-scale dataset. Moreover, we also propose an efficient method to obtain the face clustering result based on the learnt data representation.

In summary, the contributions of this work include:

- By considering both inner-track and inter-track relations of faces in videos, we develop a new method named WBSLRR to learn a more discriminative low rank representation of faces. We also propose an efficient method to obtain the face clustering result based on the learnt data representation.
- Several acceleration techniques are used to make the proposed method scalable to large-scale datasets.
- Experiments on two benchmark face datasets demonstrate the effectiveness of our WBSLRR approach for face clustering in videos.

2 Related Work

There are several existing works [7, 28] for face clustering in videos. Specifically, based on the information of face tracks, the unsupervised logistic discrinative metric learning (ULDML) method [7] learns a distance metric, so that faces in the same track are pulled closer, while faces in any face track are pushed away from the ones in another face track with the inter-track relation. More recently, based on the Hidden Markov Random Fields (HMRF) model, a probabilistic constrained clustering method called HMRF-com [28] is proposed for face clustering in videos. By exploiting the prior knowledge in the neighborhood system of HMRF, HMRF-com has shown competitive clustering performance. Besides, the problem of face clustering in videos can be treated as a constrained clustering problem, as studied in the works such as Penalized Probabilistic Clustering (PPC) [23], COP-KMeans [24] and HMRF-KMeans [2].

The subspace clustering methods [20, 10, 17, 21, 27] have been applied for face clustering. However, these methods do not exploit the valuable prior knowledge in our problem. Moreover, the face images studied in these works are usually captured under controlled environment, and may be further contaminated by artificial noises [17], while the face images in our problem are in-the-wild faces automatically detected from videos, which makes the clustering task more realistic and challenging.

The problem studied in this work is related to several other learning tasks, *e.g.*, the traditional face verification task [9], the image set based classification task [26, 6, 25, 22] and the weakly supervised learning task [12, 33, 14–16, 30]. In the traditional face verification (resp., the image set based classification) problem, each training/test example is a pair of faces (resp., a set of faces from one subject). In contrast, our problem is basically a clustering problem, where labeled data (*i.e.*, the faces with groundtruth names) are not available. In the weakly supervised learning problem, the weak supervision information usually comes from the captions of news images [12, 33] or the tags of web images [14–16, 30]. For example, in [12, 33], given a set of images, where each image contains several faces and is associated with a few names in the corresponding captions, the goal of caption-based face naming is to infer the correct name of each face. Different from such weak supervision, the prior knowledge in our task is the inner-track and inter-track relations based on the information of faces tracks.

3 Our Proposed Approach

3.1 Problem Statement

In the remainder of this paper, we use the lowercase/uppercase letter in boldface to denote a vector/matrix (e.g. , $\mathbf{a}$ denotes a vector and $\mathbf{A}$ denotes a matrix). The corresponding non-bold letter with a subscript denotes the entry in a vector/matrix (e.g. , a_i denotes the i-th entry of the vector $\mathbf{a}$, and $A_{i,j}$ denotes an entry at the i-th row and j-th column of the matrix $\mathbf{A}$). The superscript $'$

denotes the transpose of a vector or a matrix. Moreover, $\|\mathbf{A}\|_*$ denotes the nuclear norm of $\mathbf{A}$, $\mathrm{tr}(\mathbf{A})$ denotes the trace of $\mathbf{A}$ (*i.e.*, $\mathrm{tr}(\mathbf{A}) = \sum_i A_{i,i}$), $\|\mathbf{A}\|_F$ denotes the Frobenius norm of $\mathbf{A}$ (*i.e.*, $\|\mathbf{A}\|_F = (\sum_{i,j} A_{i,j}^2)^{1/2}$) and $rank(\mathbf{A})$ denotes the rank of $\mathbf{A}$. $\langle \mathbf{A}, \mathbf{B} \rangle$ denotes the inner product of two matrices (*i.e.*, $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{tr}(\mathbf{A}'\mathbf{B})$). $\mathbf{I}_n$ denotes a $n \times n$ identity matrix, and we omit the subscript when the size is obvious. $\mathbf{e}_i$ denotes the i-th column of $\mathbf{I}_n$. $diag(\mathbf{a})$ denotes a diagonal matrix where the diagonal elements are in the vector $\mathbf{a}$.

For the problem of face clustering in videos, let $\{\mathbf{X}^i\}_{i=1}^m$ denote the face tracks, where $\mathbf{X}^i \in \mathbb{R}^{d \times n_i}$ is the feature matrix corresponding to the i-th face track containing n_i faces, m is the total number of face tracks and d is the feature dimension. Besides, let $\mathbf{X} = [\mathbf{X}^1, \ldots, \mathbf{X}^m] \in \mathbb{R}^{d \times n}$ denote the feature matrix of all faces, where $n = \sum_{i=1}^m n_i$ is the total number of faces. Moreover, let us define a matrix $\mathbf{H} \in \{0, 1\}^{m \times m}$, where $H_{i,j} = 1$ if there is a face from the face track $\mathbf{X}^i$ and a face from the face track $\mathbf{X}^j$ that appear in one frame, and $H_{i,j} = 0$ otherwise, $\forall i, j = 1, \ldots, m$. The goal of our task is to cluster the n faces into l groups, where each group contains the faces from one subject.

3.2 Weighted Block-Sparse Low Rank Representation

Let us assume that the given data are drawn from a union of l independent linear subspaces, where each linear subspace corresponds to one subject [20, 17]. Following [10], we also assume that there are enough data sampled from each subspace, and data matrix is *self-expressive*, so we have $\mathbf{X} = \mathbf{XZ}$, where $\mathbf{Z} \in \mathbb{R}^{n \times n}$ is the data representation matrix. Similarly as in [10, 17], we propose to obtain the final clustering results of faces based on the data representation matrix $\mathbf{Z}$, which describes the relationship between faces. To achieve promising clustering result, we expect that the data representation $\mathbf{Z}$ has the following *ideal property* [10, 17]: $Z_{i,j} \neq 0$ (only) if the i-th face and the j-th face are from the same subject/subspace, and $Z_{i,j} = 0$ otherwise. In other words, any face from a subject should be linearly represented only by the faces from this subject.

As shown in [17], if $\mathbf{X}$ is a collection of samples *strictly* drawn from multiple independent linear subspaces (*i.e.*, $\mathbf{X}$ is *noise-free*), the optimal solution to the following problem satisfies the above mentioned ideal property:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_* \quad s.t. \quad \mathbf{X} = \mathbf{XZ}, \tag{1}$$

where $\|\mathbf{Z}\|_*$ is a convex approximation of $rank(\mathbf{Z})$ [17, 32]. The resultant data representation matrix after solving (1), which is called the shape intersection matrix (SIM) [8], has been widely used for subspace segmentation [17]. Note that the formulation in (1) essentially deals with an unsupervised learning problem, without considering the prior knowledge in our face clustering problem. Therefore, to learn a more discriminative data representation, we propose to further exploit the prior knowledge. Specifically, we additionally introduce a regularizer $\Omega(\mathbf{Z})$ to incorporate the prior knowledge, and formulate our learning problem as follows:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_* + \gamma \Omega(\mathbf{Z}), \quad s.t. \quad \mathbf{X} = \mathbf{XZ}, \tag{2}$$

where γ is a tradeoff parameter. Now, the remaining problem is how to design the regularizer $\Omega(\mathbf{Z})$ to model the prior knowledge. In this work, we propose a new regularizer that exploits both kinds of prior knowledge, which will be introduced below in details.

Recall that $\mathbf{X} = [\mathbf{X}^1, \ldots, \mathbf{X}^m]$, where each $\mathbf{X}^i$ corresponds to a face track. Accordingly, we can divide $\mathbf{Z}$ into $m \times m$ blocks as follows:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}^{(1,1)} & \ldots & \mathbf{Z}^{(1,m)} \\ \vdots & \ddots & \vdots \\ \mathbf{Z}^{(m,1)} & \ldots & \mathbf{Z}^{(m,m)} \end{bmatrix} \tag{3}$$

where each sub-matrix $\mathbf{Z}^{(i,j)} \in \mathbb{R}^{n_i \times n_j}$ contains the coefficients for representing the faces in the face track $\mathbf{X}^j$ using the ones in the face track $\mathbf{X}^i$, as shown in Figure 1.

Considering the inner-track relation, we extend the previously mentioned ideal property of the data representation $\mathbf{Z}$ to the following *block-wise ideal property*: The elements in $\mathbf{Z}^{(i,j)}$ are non-zeros (only) if the i-th face track and the j-th face track are from the same subject, otherwise the elements in $\mathbf{Z}^{(i,j)}$ should be zeros. As a result, the elements in each $\mathbf{Z}^{(i,j)}$ of such ideal representation matrix should be either large or zeros, namely the ideal $\mathbf{Z}$ should be *block-sparse*, as illustrated in Fig 1.

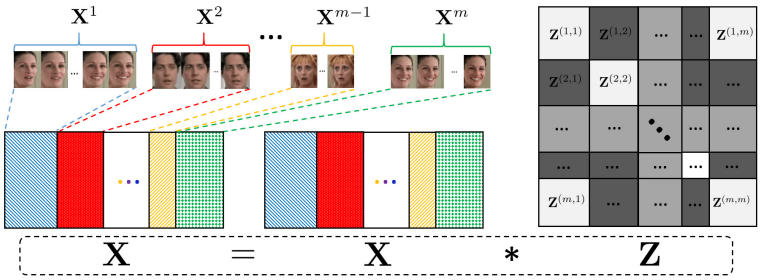

Fig. 1. Illustration of the block-sparse property of $\mathbf{Z}$, as well as the relationship between $\mathbf{X}$ and $\mathbf{Z}$ in the *noise-free* case. The different colors in $\mathbf{X}$ denote different face tracks.

Inspired by the minimization of the $\ell_{2,1}$ norm [17] which promotes the column sparsity of a matrix, we encourage the above mentioned *block-sparse* property by minimizing $\Omega_0(\mathbf{Z}) = \sum_{i=1}^{m} \sum_{j=1}^{m} \frac{1}{\sqrt{n_i n_j}} \|\mathbf{Z}^{(i,j)}\|_F$, where we use $\frac{1}{\sqrt{n_i n_j}}$ to normalize the Frobenius norms of $\{\mathbf{Z}^{(i,j)}\}_{i,j=1}^{m}$ due to different sizes of the sub-matrices.

Now, let us further consider the *inter-track* relation. Intuitively, when two faces respectively from $\mathbf{X}^i$ and $\mathbf{X}^j$ appear in the same frame, these two face tracks should be from different subjects. As a result, the elements in the corresponding two sub-matrices $\mathbf{Z}^{(i,j)}$ and $\mathbf{Z}^{(j,i)}$ are zeros in the ideal case. To this end, we

propose a new regularization term $\Omega(\mathbf{Z})$ based on $\Omega_0(\mathbf{Z})$ as follows:

$$\Omega(\mathbf{Z}) = \sum_{i=1}^{m}\sum_{j=1}^{m} Q_{i,j}\|\mathbf{Z}^{(i,j)}\|_F, \tag{4}$$

where $Q_{i,j} = \frac{1}{\sqrt{n_i n_j}} + \mu H_{i,j}$, with μ being a large scalar (which is empirically set to 1000 in our experiments). Compared with $\Omega_0(\mathbf{Z})$, if the i-th face track and the j-th face track are with the inter-track relation, the weight w.r.t. $\|\mathbf{Z}^{(i,j)}\|_F$ will be enlarged from $1/\sqrt{n_i n_j}$ to $Q_{i,j}$, so the elements in the resultant $\mathbf{Z}^{(i,j)}$ and $\mathbf{Z}^{(j,i)}$ will tend to be closer to zeros. With $\Omega(\mathbf{Z})$ defined in (4), we detail the optimization problem in (2) as follows:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_* + \gamma \sum_{i=1}^{m}\sum_{j=1}^{m} Q_{i,j}\|\mathbf{Z}^{(i,j)}\|_F \quad s.t. \quad \mathbf{X} = \mathbf{X}\mathbf{Z}. \tag{5}$$

Recall that, for the in-the-wild faces in our problem, the data $\mathbf{X}$ is often contaminated by noise, so the equality constraint in (5) may not be perfectly satisfied. Following [10], we assume that the data $\mathbf{X}$ is corrupted by the Gaussian noise, so the squared Frobenius norm [29] is used to regularize the representation error (*i.e.*, $\mathbf{X} - \mathbf{X}\mathbf{Z}$). Accordingly, we arrive at the *weighted block-sparse low rank representation* (WBSLRR) problem as follows:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_* + \gamma \sum_{i=1}^{m}\sum_{j=1}^{m} Q_{i,j}\|\mathbf{Z}^{(i,j)}\|_F + \frac{\lambda}{2}\|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2, \tag{6}$$

where λ is a tradeoff parameter. Once the optimization problem in (6) is solved, we can obtain the face clustering result based on the optimal solution $\mathbf{Z}^* \in \mathbb{R}^{n\times n}$.

Notice that, in traditional subspace clustering methods such as [17], to obtain the clustering result, spectral clustering is usually performed on $(|\mathbf{Z}^*| + |\mathbf{Z}^*|')/2$, where $|\cdot|$ denotes the element-wise absolute value operator. However, this approach does not utilize the face track information in our problem, and it may be computationally expensive when n is large. To this end, we propose to perform clustering on face tracks at first, and then propagate the labels to the faces, instead of performing clustering on the faces directly. Specifically, we convert $\mathbf{Z}^*$ into an affinity matrix $\mathbf{A} \in \mathbb{R}^{m\times m}$, where each element $A_{i,j} = \|\mathbf{Z}^{*(i,j)}\|_F/\sqrt{n_i n_j}$ describes the affinity between the corresponding pair of face tracks. Afterwards, we follow [17] to post-process this affinity matrix $\mathbf{A}$, and use the spectral clustering method [10, 20, 27] on the post-processed affinity matrix to perform clustering on the face tracks. Finally, the clustering result of faces can be directly obtained by propagating the label from each face track to the corresponding faces within this face track.

4 Optimization

There are two major challenges when solving the optimization problem in (6). Firstly, it contains a nuclear norm regularization on $\mathbf{Z}$, which is non-differentiable.

Secondly, for the face clustering problem in vedios, it is possible that a lot of faces are automatically detected, and the corresponding data matrix $\mathbf{X}$ can be very large (*e.g.*, we have 17337 faces in the BF0502 dataset, see Section 5 for more details).

To tackle the first challenge, we use the alternating direction method (ADM) [4, 17], which has been widely used in the nuclear norm related optimization problems such as [18, 17]. To address the second challenge, inspired by a recent work [18], we decompose the representation matrix $\mathbf{Z}$ as $\mathbf{Z} = \mathbf{GW}$, in which $\mathbf{G} \in \mathbb{R}^{n\times r}$ is a column-wise orthonormal matrix with r being a scalar smaller than n (r is empirically set to 1000 in our experiments), and $\mathbf{W} \in \mathbb{R}^{r\times n}$. For the convenience of optimization, we further introduce two variables $\mathbf{P} = \mathbf{I} - \mathbf{GW} \in \mathbb{R}^{n\times n}$ and $\mathbf{J} = \mathbf{P} \in \mathbb{R}^{n\times n}$, and reformulate our optimization problem as follows,

$$\begin{aligned} \min_{\mathbf{G}'\mathbf{G}=\mathbf{I},\mathbf{W},\mathbf{P},\mathbf{J}} \quad & \|\mathbf{W}\|_* + \gamma\Omega(\mathbf{I}-\mathbf{J}) + \frac{\lambda}{2}\|\mathbf{XP}\|_F^2 \\ s.t. \quad & \mathbf{I} - \mathbf{GW} = \mathbf{P},\ \mathbf{J} = \mathbf{P}. \end{aligned} \tag{7}$$

To solve the optimization problem in (7) using ADM [4, 17], we operate on the following augmented Lagrangian function:

$$\begin{aligned} \mathcal{L}(\mathbf{G},\mathbf{W},\mathbf{J},\mathbf{P},\mathbf{L},\mathbf{\Lambda},\rho) = \|\mathbf{W}\|_* + \gamma\Omega(\mathbf{I}-\mathbf{J}) + \frac{\lambda}{2}\|\mathbf{XP}\|_F^2 + \langle \mathbf{I}-\mathbf{GW}-\mathbf{P}, \mathbf{L}\rangle \\ + \langle \mathbf{J}-\mathbf{P}, \mathbf{\Lambda}\rangle + \frac{\rho}{2}\left(\|\mathbf{I}-\mathbf{GW}-\mathbf{P}\|_F^2 + \|\mathbf{J}-\mathbf{P}\|_F^2\right), \end{aligned}$$

where $\mathbf{L} \in \mathbb{R}^{n\times n}$ and $\mathbf{\Lambda} \in \mathbb{R}^{n\times n}$ are the Lagrange multipliers and ρ is the penalty parameter. The optimization problem can be solved by iteratively updating the variables $\{\mathbf{G},\mathbf{W},\mathbf{J},\mathbf{P}\}$, the Lagrange multipliers $\{\mathbf{L},\mathbf{\Lambda}\}$ and the penalty parameter ρ until convergence. We introduce the detailed updating steps at the t-th iteration as follows:

Updating G: $\mathbf{G}_{t+1}$ is calculated as $\text{argmin}_{\mathbf{G}}\mathcal{L}(\mathbf{G},\mathbf{W}_t,\mathbf{J}_t,\mathbf{P}_t,\mathbf{L}_t,\mathbf{\Lambda}_t,\rho_t)$, *i.e.*, the optimal solution to the following subproblem:

$$\min_{\mathbf{G}'\mathbf{G}=\mathbf{I}} \|(\mathbf{I}-\mathbf{P}_t+\frac{\mathbf{L}_t}{\rho_t}) - \mathbf{GW}_t\|_F^2.$$

This problem is known as the matrix procrustes problem [18]. Based on [18], the optimal solution of the above problem is given by $\mathbf{G}_{t+1} = \mathbf{U}_G\mathbf{V}_G'$, where $\mathbf{U}_G$ and $\mathbf{V}_G$ are two orthogonal matrices obtained by using singular value decomposition (SVD) of $(\mathbf{I}-\mathbf{P}_t+\frac{\mathbf{L}_t}{\rho_t})\mathbf{W}_t'$, *i.e.*, $\mathbf{U}_G\mathbf{\Sigma}_G\mathbf{V}_G' = (\mathbf{I}-\mathbf{P}_t+\frac{\mathbf{L}_t}{\rho_t})\mathbf{W}_t'$.

Updating W: $\mathbf{W}_{t+1}$ is calculated as $\text{argmin}_{\mathbf{W}}\mathcal{L}(\mathbf{G}_{t+1},\mathbf{W},\mathbf{J}_t,\mathbf{P}_t,\mathbf{L}_t,\mathbf{\Lambda}_t,\rho_t)$, *i.e.*, the optimal solution to the following subproblem:

$$\min_{\mathbf{W}} \|\mathbf{W}\|_* + \frac{\rho_t}{2}\left\|\mathbf{W} - \mathbf{G}_{t+1}'\left(\mathbf{I}-\mathbf{P}_t+\frac{\mathbf{L}_t}{\rho_t}\right)\right\|_F^2, \tag{8}$$

which can also be solved in closed form by using the Singular Value Thresholding (SVT) [5] method.

Updating J: $\mathbf{J}_{t+1}$ is calculated as $\text{argmin}_{\mathbf{J}} \mathcal{L}(\mathbf{G}_{t+1}, \mathbf{W}_{t+1}, \mathbf{J}, \mathbf{P}_t, \mathbf{L}_t, \mathbf{\Lambda}_t, \rho_t)$, *i.e.*, the optimal solution to the following subproblem:

$$\min_{\mathbf{J}} \ \Omega(\mathbf{I} - \mathbf{J}) + \frac{\rho_t}{2\gamma} \left\| \mathbf{J} - \mathbf{P}_t + \frac{\mathbf{\Lambda}_t}{\rho_t} \right\|_F^2 . \tag{9}$$

For convenience, let us define $\hat{\mathbf{J}} = \mathbf{I} - \mathbf{J}_{t+1}$ and $\mathbf{R} = \mathbf{I} - \mathbf{P}_t + \frac{\mathbf{\Lambda}_t}{\rho_t}$, the above problem can be rewritten as:

$$\min_{\hat{\mathbf{J}}} \ \Omega(\hat{\mathbf{J}}) + \frac{\rho_t}{2\gamma} \left\| \hat{\mathbf{J}} - \mathbf{R} \right\|_F^2 .$$

Similarly as in (3), we also decompose $\hat{\mathbf{J}}$ and $\mathbf{R}$ into $m \times m$ blocks. Let us denote $\hat{\mathbf{J}}^{(i,j)}$ (*resp.*, $\mathbf{R}^{(i,j)}$) as the (i,j)-th block of $\hat{\mathbf{J}}$ (*resp.*, $\mathbf{R}$), then the above problem can be equivalently rewritten as

$$\min_{\{\hat{\mathbf{J}}^{(i,j)}\}_{i,j=1}^m} \sum_{i=1}^m \sum_{j=1}^m Q_{i,j} \|\hat{\mathbf{J}}^{(i,j)}\|_F + \frac{\rho_t}{2\gamma} \sum_{i=1}^m \sum_{j=1}^m \left\| \hat{\mathbf{J}}^{(i,j)} - \mathbf{R}^{(i,j)} \right\|_F^2 ,$$

which can be divided into the following m^2 subproblems:

$$\min_{\hat{\mathbf{J}}^{(i,j)}} \ \tau_{i,j} \|\hat{\mathbf{J}}^{(i,j)}\|_F + \frac{1}{2} \left\| \hat{\mathbf{J}}^{(i,j)} - \mathbf{R}^{(i,j)} \right\|_F^2 , \quad i,j = 1, \ldots, m, \tag{10}$$

where $\tau_{i,j} = \gamma Q_{i,j}/\rho_t$. Based on Lemma 3.3 in [31], the closed form solution of the problem in (10) can be obtained as $\hat{\mathbf{J}}^{*(i,j)} = \max\left(1 - \frac{\tau_{i,j}}{\|\mathbf{R}^{(i,j)}\|_F}, 0\right) \mathbf{R}^{(i,j)}$. With $\{\hat{\mathbf{J}}^{*(i,j)}\}_{i,j=1}^m$ obtained after solving the m^2 subproblems in (10), $\mathbf{J}_{t+1}$ can be recovered by $\mathbf{J}_{t+1} = \mathbf{I} - \hat{\mathbf{J}}^*$. In this way, the optimization problem in (9) can be solved.

Updating P: $\mathbf{P}_{t+1}$ is calculated as $\text{argmin}_{\mathbf{P}} \mathcal{L}(\mathbf{G}_{t+1}, \mathbf{W}_{t+1}, \mathbf{J}_{t+1}, \mathbf{P}, \mathbf{L}_t, \mathbf{\Lambda}_t, \rho_t)$, *i.e.*, the optimal solution to the following subproblem:

$$\min_{\mathbf{P}} \ \frac{\lambda}{2} \|\mathbf{X}\mathbf{P}\|_F^2 + \rho_t \left\| \mathbf{P} - \mathbf{C}_{t+1} \right\|_F^2 . \tag{11}$$

where $\mathbf{C}_{t+1} = \frac{1}{2}(\mathbf{I} - \mathbf{G}_{t+1}\mathbf{W}_{t+1} + \mathbf{J}_{t+1} + \frac{1}{\rho_t}\mathbf{L}_t + \frac{1}{\rho_t}\mathbf{\Lambda}_t)$. Note that, the gradient of the above objective function *w.r.t.* $\mathbf{P}$ is $\lambda \mathbf{X}'\mathbf{X}\mathbf{P} + 2\rho_t(\mathbf{P} - \mathbf{C}_{t+1})$. By setting the gradient to zeros, we obtain the optimal solution for (11) as

$$\mathbf{P}_{t+1} = \left(\frac{\lambda}{2\rho_t} \mathbf{X}'\mathbf{X} + \mathbf{I} \right)^{-1} \mathbf{C}_{t+1}. \tag{12}$$

We now discuss how to calculate $\mathbf{P}_{t+1}$ in (12) more efficiently without using matrix inversion. Let $\mathbf{V}_X diag([\sigma_1, \ldots, \sigma_{r_X}, 0, \ldots, 0]')\mathbf{V}_X' = \mathbf{X}'\mathbf{X}$ denote the SVD of $\mathbf{X}'\mathbf{X}$, where $\mathbf{V}_X \in \mathbb{R}^{n \times n}$ is an orthogonal matrix and $\{\sigma_i\}_{i=1}^{r_X}$ are *positive*

Algorithm 1. The algorithm for solving WBSLRR.

Input: $\mathbf{X} = [\mathbf{X}^1, \ldots, \mathbf{X}^m] \in \mathbb{R}^{d \times n}$, $\mathbf{H} \in \{0,1\}^{m \times m}$, λ and γ.
Initialize $\mathbf{P}_0$, $\mathbf{G}_0$, $\mathbf{W}_0$, $\mathbf{J}_0$, $\mathbf{L}_0$, $\mathbf{\Lambda}_0$ as zero matrices and set $t = 0$.
while not converge **do**
1. Calculate $\mathbf{G}_{t+1}$ by using $\mathbf{G}_{t+1} = \mathbf{U}_G \mathbf{V}_G'$, where $\mathbf{U}_G \mathbf{\Sigma}_G \mathbf{V}_G' = (\mathbf{I} - \mathbf{P}_t + \frac{\mathbf{L}_t}{\rho_t})\mathbf{W}_t'$.
2. Calculate $\mathbf{W}_{t+1}$ by solving (8) using the SVT method [5].
3. Calculate $\mathbf{J}_{t+1}$ by using $\mathbf{J}_{t+1} = \mathbf{I} - \hat{\mathbf{J}}^*$, with $\hat{\mathbf{J}}^*$ obtained by solving (10) for $\{\hat{\mathbf{J}}^{*(i,j)}\}_{i,j=1}^m$.
4. Calculate $\mathbf{P}_{t+1}$ as in (13).
5. Calculate $\mathbf{L}_{t+1}$ as $\mathbf{L}_{t+1} = \mathbf{L}_t + \rho_t(\mathbf{I} - \mathbf{G}_{t+1}\mathbf{W}_{t+1} - \mathbf{P}_{t+1})$, and compute $\mathbf{\Lambda}_{t+1}$ as $\mathbf{\Lambda}_{t+1} = \mathbf{\Lambda}_t + \rho_t(\mathbf{J}_{t+1} - \mathbf{P}_{t+1})$.
6. Calculate ρ_{t+1} as $\rho_{t+1} = \min(\rho_t(1 + \Delta\rho), \rho_{max})$.
7. Check the following convergence conditions: $\|\mathbf{I} - \mathbf{G}_{t+1}\mathbf{W}_{t+1} - \mathbf{P}_{t+1}\|_\infty \leq \epsilon$ and $\|\mathbf{J}_{t+1} - \mathbf{P}_{t+1}\|_\infty \leq \epsilon$.
8. $t \leftarrow t + 1$.
end while
Output: the data representation $\mathbf{Z}^* = \mathbf{G}_t\mathbf{W}_t$.

singular values sorted in descending order, with $r_X = rank(\mathbf{X}) = rank(\mathbf{X}'\mathbf{X})$. As a result, we have $\frac{\lambda}{2\rho_t}\mathbf{X}'\mathbf{X} + \mathbf{I} = \mathbf{V}_X diag([1+\omega\sigma_1, \ldots, 1+\omega\sigma_{r_X}, 1, \ldots, 1]')\mathbf{V}_X'$ and $(\frac{\lambda}{2\rho_t}\mathbf{X}'\mathbf{X} + \mathbf{I})^{-1} = \mathbf{V}_X diag([\frac{1}{1+\omega\sigma_1}, \ldots, \frac{1}{1+\omega\sigma_{r_X}}, 1, \ldots, 1]')\mathbf{V}_X'$, where $\omega = \frac{\lambda}{2\rho_t}$. For convenience, we define $\mathbf{V}_{r_X} \in \mathbb{R}^{n \times r_X}$ which contains the first r_X columns of $\mathbf{V}_X$, and we define $\mathbf{\Lambda} \in \mathbb{R}^{r_X \times r_X}$ as $\mathbf{\Lambda} = diag([\frac{\omega\sigma_1}{1+\omega\sigma_1}, \ldots, \frac{\omega\sigma_{r_X}}{1+\omega\sigma_{r_X}}]')$. Accordingly, we obtain $(\frac{\lambda}{2\rho_t}\mathbf{X}'\mathbf{X} + \mathbf{I})^{-1} = \mathbf{I} - \mathbf{V}_{r_X}\mathbf{\Lambda}\mathbf{V}_{r_X}'$, so $\mathbf{P}_{t+1}$ in (12) can be equivalently calculated as

$$\mathbf{P}_{t+1} = \mathbf{C}_{t+1} - \mathbf{V}_{r_X}\mathbf{\Lambda}\mathbf{V}_{r_X}'\mathbf{C}_{t+1}, \tag{13}$$

for which the computational cost is $O(r_X n^2)$.

Other details, including updating the Lagrange multipliers and the penalty parameter, as well as the details of the convergence conditions, are summarized in Algorithm 1, where $\{\rho_0, \Delta\rho, \rho_{max}, \epsilon\}$ is set similarly as in [17].

Time Complexity and Convergence Analysis. The computational complexities of the main steps for updating the variables $\{\mathbf{G}, \mathbf{W}, \mathbf{J}, \mathbf{P}\}$ in each iteration are $O(rn^2)$, $O(rn^2)$, $O(n^2)$ and $O(r_X n^2)$, respectively. Therefore, the overall computational complexity for each iteration in Algorithm 1 is $O((r + r_X)n^2)$, and the step 3 for updating $\mathbf{J}$ can be efficiently performed in parallel.

The theoretical convergence of ADM with more than two blocks is still an open issue [17]. However, it has been widely used in many applications because it empirically converges well in general [17]. In the following experiments, we also show the empirical convergence of Algorithm 1 (see Section 5 for more details). Alternatively, the optimization problem in (6) can be addressed by using a recently proposed algorithm LADMPSAP [19] with convergence guarantees in theory, which will be studied in our future work.

5 Experiments

5.1 Datasets

We evaluate the performances of the proposed method and the baseline methods on two benchmark face datasets (*i.e.*, the Notting-Hill dataset [28] and the BF0502 dataset [11]) used in [28]. On both datasets, we strictly follow the experimental setting in [28]. The Notting-Hilldataset contains 4460 faces in 76 tracks detected from the movie "Notting Hill", and the faces are corresponding to 5 main casts. Following [28], we use the pixel intensities as the feature, so that each face is represented as a 18000 dimensional feature vector. The BF0502 dataset contains faces detected from the TV series "Buffy the Vampire Slayer". In our experiments, we use the 17337 faces in 229 tracks corresponding to 6 main casts. To represent each face, we use the 1937-dimensional descriptor [11] extracted from 13 facial points (*e.g.* the left and right corners of each eye).

We define the distinguishability value to as the criterion to evaluate how difficult the clustering problem is on each dataset. Specifically, based on the groundtruth labels, let us call a pair of face tracks as "same-subject pair" if these two tracks are from the same subject, otherwise we call it aa "different-subject pair". For each pair of face tracks, we calculate the mean of the squared Euclidean distance between the faces in one track and the faces in the other track. Then, we define the distinguishability value as the ratio between the average distance corresponding over same-subject pairs and the average distance over different-subject pairs. Generally speaking, a larger distinguishability value indicates that the corresponding clustering problem on this dataset is easier.

A brief summary of the information of the two datasets can be found in Table 1. According to this table, the BF0502 dataset, with more faces and face tracks and a smaller distinguishability value, should be more challenging than the Notting-Hill dataset.

5.2 Baselines and Evaluation Criterion

We compare our proposed method with the most recent work HMRF-com [28], as well as the baselines mentioned in [28]. Specifically, the following methods are used as the baselines:

- *the traditional clustering method*: Kmeans [3] is used in two ways as the baselines in [28]. Specifically, "Kmeans-1" is directly performed on the whole dataset after using PCA, and "Kmeans-2" denotes the Algorithm 2 in [28], in which Kmeans is used in Stage 2. Note that neither of these two approaches utilize the prior knowledge.
- *the constrained clustering method*: The Penalized Probabilistic Clustering (PPC) method [23] used the Gaussian mixture models, with the prior knowledge in our problem considered as pairwise constraints.
- *the metric learning based method*: The unsupervised logistic discriminant metric learning (ULDML) method [7] proposed for the face track clustering in

videos. In [28], two methods (called "ULDML-cl" and "ULDML-km") are proposed based on the learnt metric. Specifically, for ULDML-cl, a complete-link hierarchical clustering method is employed on the corresponding distance matrix between the face tracks [7]. For ULDML-km, Kmeans is performed based on the learnt metric.

– *the hidden markov random fields based method*: HMRF-com [28] is a probablistic constrained clustering approach based on HMRF. Note that the prior knowlege is considered in the combined neighborhood system.

To further study our proposed method, we also compare our WBSLRR with the following five compressed sensing based subspace clustering methods on both datasets: Least Squares Regression (LSR) [20] , Sparse Subspace Clustering (SSC) [10], Low Rank Representation (LRR) [17], Correlation Adaptive Subspace Segmentation (CASS) [21] and Low Rank Sparse Subspace Clustering (LRSSC) [27]. For fair comparison, we apply our acceleration techniques in our implementation of LRSSC [27], and we use the LRR method with the squared Frobenius norm regularization on the representation error. Basically, both WBSLRR and these subspace clustering methods seek for a desired data representation $\mathbf{Z}$, based on which clustering can be performed. The major difference between these approaches is the regularizations on $\mathbf{Z}$ in their objective functions, which are briefly summarized in Table 3. Based on Table 3, we can observe that LRR is a special case of our WBSLRR if we drop $\Omega(\mathbf{Z})$ in (6), while LRSSC can be treated as a special case of our WBSLRR by replacing $\Omega(\mathbf{Z})$ with $\|\mathbf{Z}\|_1$ (*i.e.*, LRSSC encourages the general sparsity without considering of the information of the face tracks). For fair comparison, for all these subspace clustering methods (*i.e.*, LSR, SSC, LRR, CASS and LRSSC), the clustering is also performed based on the affinity matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$ (as we introduced in Section 3.2), so that the information of face tracks is also utilized.

Following [28], we use *accuracy* (based on the confusion matrix) for performance evaluation, which is defined as the number of correctly clustered faces over the total number of faces. The confusion matrix is derived from the best 1-to-1 match between the partition of all faces and the groundtruth labels, which is obtained by using the Hungarian method [13]. As suggested in [28], each algorithm is repeated for 30 times, and the mean accuracy and standard deviation are reported. Due to the page limitation, we omit the parameter settings of these baselines, which can be found in [28]. The results of the state-of-the-art baselines listed in Table 2 are from the tables in [28]. For fair comparision, we manually tune the parameters and report the best results of our method and the subspace clustering methods as suggested in [28].

5.3 Experimental Results

In this section, we verify the effectiveness of our proposed method with two experiments. In the first experiment, we compare the results of our proposed method with the state-of-the-art results in [28] on the two datasets. In the second experiment, we compare our method with several subspace clustering approaches.

Table 1. A brief summary of information about the two datasets. "#" means "the number of", and "dim" stands for the feature dimension.

Dataset	m(#tracks)	n(#faces)	d(dim)	l(#subjects)	distinguishability value
BF0502	229	17337	1937	5	1.09
Notting-Hill	76	4660	18000	6	1.46

Table 2. The clustering accuracies (mean±standard deviation%) of the state-of-the-art methods and our proposed method on two datasets under two settings. The results of the baseline methods are from [28]. The best accuracies are highlighted in boldface.

Methods	BF0502		Notting-Hill	
	Setting 1	Setting 2	Setting 1	Setting 2
Kmeans-1	39.31 ± 4.51	39.31 ± 4.51	69.16 ± 3.22	69.16 ± 3.22
Kmeans-2	42.05 ± 5.45	42.05 ± 5.45	73.43 ± 8.12	73.43 ± 8.12
PPC	43.64 ± 4.61	42.54 ± 3.98	79.71 ± 2.14	78.88 ± 5.15
ULDML-km	29.05 ± 2.84	41.62 ± 0.00	72.66 ± 12.78	73.18 ± 8.66
ULDML-cl	39.01 ± 0.00	49.29 ± 0.00	51.72 ± 0.00	36.87 ± 0.00
HMRF-com	47.77 ± 3.31	50.30 ± 2.73	81.33 ± 0.43	84.39 ± 1.47
WBSLRR (ours)	**59.55** $\pm$ 0.51	**62.76**$\pm$1.10	**95.24** $\pm$0.00	**96.29**$\pm$0.00

Comparison with the State-of-the-Art Methods. In this experiment, we evaluate all the methods under the following two settings:

- Setting 1: we only utilize the inner-track relation in all the methods,
- Setting 2: both inter-track and inter-track relations are available to all the methods.

For our WBSLRR method, we solve the optimization problem in (6) with the second term, namely $\Omega(\mathbf{Z})$, replaced with $\Omega_0(\mathbf{Z})$, in order to exclude the consideration of inter-track relation under setting 1. and directly solve the optimization problem implement (6) under setting 2. For each method, the clustering accuracies on two datasets are shown in Table 2. According to Table 2, we have the following observations:

Firstly, our proposed method WBSLRR outperforms all the baseline methods (on both datasets) under both settings. Comparing WBSLRR with the second best method (*i.e.* the HMRF-com method) on the two datasets, the relative improvement is about 20% (resp., 15%) on the BF0502 dataset (resp., the Notting-Hill dataset). The results clearly demonstrate that WBSLRR can make better use of the prior knowledge (*i.e.* the inner-track and inter-track relation) for the face clustering problems in videos .

On both datasets, the performances of WBSLRR under setting 2 are better when compared with those under setting 1, which demonstrates that it is beneficial to additionally consider the inter-track relation in (6). For HMRF-com, we have similar observations, *i.e.*, the results under setting 2 are better than those under setting 1 on both datasets.

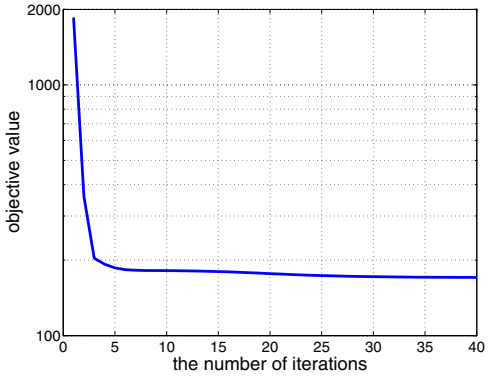

Fig. 2. The objective values of our optimization problem in (6) with respect to the number of iterations, on the Notting-Hill dataset

For almost all the methods, the performances on the Notting-Hill dataset are generally better than those on the BF0502 dataset. One possible explanation is that, the BF0502 dataset contains more faces and face tracks and it is also with smaller distinguishability value, which indicates that the face clustering task on the BF0502 dataset is more challenging.

Last but not least, we take the Notting-Hill dataset as an example to show the objective values of the optimization problem in (6) with respect to different iteration numbers in Figure 2. We can observe that our optimization algorithm empirically converges well.

Comparison with the Subspace Clustering Methods. The mean accuracies and the standard deviations as well as the running times of five existing subspace clustering methods and our WBSLRR on both datasets are reported in Table 3. All the algorithms are executed on a desktop with Intel Xeon CPU (3.2Ghz) and 16GB memory).

From Table 3, we observe that our WBSLRR achieves the best accuracies on both datasets and it is also reasonably fast compared with other methods. Note that the difference between our WBSLRR and LRR is that we additionally use the regularizer $\Omega(\mathbf{Z})$ to encourage the block-sparsity of the representation matrix $\mathbf{Z}$. WBSLRR outperforms LRR, which clearly demonstrates the effectiveness of the proposed regularizer $\Omega(\mathbf{Z})$ for exploiting the available prior knowledge in our task.

Moreover, we observe that both WBSLRR and LRSSC outperform LRR in terms of the clustering accuracy, which indicates that more robust results can be achieved by further encouraging the sparsity of $\mathbf{Z}$. Our WBSLRR achieves better results than LRSSC, which demonstrates it is more beneficial to encourage the weighted block-sparsity on the data representation according to the prior knowledge in our WBSLRR, rather than promoting the general sparsity by using the ℓ_1 norm regularization as in LRSSC.

Table 3. The regularizations on $\mathbf{Z}$ in different subspace clustering methods and our WBSLRR, and their clustering accuracies (mean±standard deviation%) as well as running times (in seconds) on two datasets. The results of CASS on the BF0502 dataset are not available because CASS cannot be used on large datasets like the BF0502 dataset. The best accuracies are highlighted in boldface.

Methods	Regualrization on $\mathbf{Z}$	BF0502		Notting-Hill	
		Accuarcy(%)	Time(s)	Accuarcy(%)	Time(s)
LSR	$\|\mathbf{Z}\|_F^2$	50.19 ± 1.93	131.53	89.89 ± 0.00	7.36
SSC	$\|\mathbf{Z}\|_1$	36.52 ± 0.91	24554.59	75.50 ± 7.90	2931.00
LRR	$\|\mathbf{Z}\|_*$	51.17 ± 2.94	1208.53	93.11 ± 0.00	31.92
CASS	$\sum_{i=1}^{n} \|\mathbf{X}\ diag(\mathbf{Z}\mathbf{e}_i)\|_*$	N/A	N/A	93.18 ± 0.00	29610.15
LRSSC	$\|\mathbf{Z}\|_* + \gamma\|\mathbf{Z}\|_1$	58.08 ± 5.37	8211.20	94.03 ± 0.00	545.93
WBSLRR (ours)	$\|\mathbf{Z}\|_* + \gamma\Omega(\mathbf{Z})$	**62.76**±1.10	693.43	**96.29**±0.00	194.14

LRR outperforms LSR and SSC on both datasets. One possible explanation is that by using the nuclear norm regularizer on $\mathbf{Z}$, LRR can better grasp the global structure [17] of the given data. While CASS achieves relatively better results than LRR, LSR and SSC on the Notting-Hill dataset, it is slow and thus cannot be applied on the large dataset BF0502.

Finally, the five subspace clustering methods generally achieve better results than the baseline methods in Table 2. One possible explanation is that, the self-expressiveness assumption of the data is generally satisfied and the prior knowledge is also considered in the post-processing procedures of these subspace clustering methods.

6 Conclusions

To effectively solve the face clustering problem in videos, in this paper, we have proposed the WBSLRR method, which exploits the two kinds of prior knowledge (*i.e.* the inner-track and inter-track relations) while learning a low rank representation of the given data. We also propose a post-processing approach to efficiently obtain the clustering result of faces based on the resultant data representation. After using several acceleration techniques in our algorithm, the proposed method is scalable for solving large scale problems. The experimental results have demonstrated the effectiveness of our approach when compared with several state-of-the-art baselines and subspace clustering methods.

Acknowledgement. This work is supported by the Singapore MoE Tier 2 Grant (ARC42/13).

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic subspace clustering of high dimensional data for data mining applications. In: SIGMOD (1998)

2. Basu, S., Bilenko, M., Banerjee, A., Mooney, R.J.: Probabilistic semi-supervised clustering with constraints. In: Semi-Supervised Learning. MIT Press (2006)
3. Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning 3(1), 1–122 (2011)
5. Cai, J., Emmanuel, C., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM Journal on Optimization 20(4), 1956–1982 (2010)
6. Cevikalp, H., Triggs, B.: Face recognition based on image sets. In: CVPR, pp. 2567–2573 (2010)
7. Cinbis, R.G., Verbeek, J.J., Schmid, C.: Unsupervised metric learning for face identification in TV video. In: ICCV, pp. 1559–1566 (2011)
8. Costeira, J.P., Kanade, T.: A multibody factorization method for independently moving objects. International Journal of Computer Vision 29(3), 159–179 (1998)
9. Cui, Z., Li, W., Xu, D., Shan, S., Chen, X.: Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In: CVPR, pp. 3554–3561 (2013)
10. Elhamifar, E., Vidal, R.: Sparse subspace clustering. In: CVPR, pp. 2790–2797 (2009)
11. Everingham, M., Sivic, J., Zisserman, A.: Hello! my name is... Buffy – automatic naming of characters in TV video. In: BMVC, pp. 899–908 (2006)
12. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Automatic face naming with caption-based supervision. In: CVPR, pp. 1–8 (2008)
13. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Research Logistics Quarterly 2(1-2), 83–97 (1955)
14. Li, W., Duan, L., Tsang, I.W., Xu, D.: Batch mode adaptive multiple instance learning for computer vision tasks. In: CVPR, pp. 2368–2375 (2012)
15. Li, W., Duan, L., Tsang, I.W., Xu, D.: Co-labeling: A new multi-view learning approach for ambiguous problems. In: ICDM, pp. 419–428 (2012)
16. Li, W., Duan, L., Xu, D., Tsang, I.W.: Text-based image retrieval using progressive multi-instance learning. In: ICCV, pp. 2049–2055 (2011)
17. Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., Ma, Y.: Robust recovery of subspace structures by low-rank representation. TPAMI 35(1), 171–184 (2013)
18. Liu, G., Yan, S.: Active subspace: Toward scalable low-rank learning. Neural Computation 24(12), 3371–3394 (2012)
19. Liu, R., Lin, Z., Su, Z.: Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. In: ACML, pp. 116–132 (2013)
20. Lu, C.Y., Min, H., Zhao, Z.Q., Zhu, L., Huang, D.S., Yan, S.: Robust and efficient subspace segmentation via least squares regression. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VII. LNCS, vol. 7578, pp. 347–360. Springer, Heidelberg (2012)
21. Lu, C., Feng, J., Lin, Z., Yan, S.: Correlation adaptive subspace segmentation by trace Lasso. In: ICCV, pp. 1345–1352 (2013)
22. Lu, J., Wang, G., Moulin, P.: Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning. In: ICCV, pp. 329–336 (2013)
23. Lu, Z., Leen, T.K.: Penalized probabilistic clustering. Neural Computation 19(6), 1528–1567 (2007)
24. Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S., et al.: Constrained k-means clustering with background knowledge. In: ICML, pp. 577–584 (2001)

25. Wang, R., Guo, H., Davis, L.S., Dai, Q.: Covariance discriminative learning: A natural and efficient approach to image set classification. In: CVPR, pp. 2496–2503 (2012)
26. Wang, R., Shan, S., Chen, X., Gao, W.: Manifold-manifold distance with application to face recognition based on image set. In: CVPR, pp. 1–8 (2008)
27. Wang, Y.X., Xu, H., Leng, C.: Provable subspace clustering: When LRR meets SSC. In: NIPS, pp. 64–72 (2013)
28. Wu, B., Zhang, Y., Hu, B.G., Ji, Q.: Constrained clustering and its application to face clustering in videos. In: CVPR, pp. 3507–3514. IEEE (2013)
29. Xu, D., Huang, Y., Zeng, Z., Xu, X.: Human gait recognition using patch distribution feature and locality-constrained group sparse representation. IEEE Transactions on Image Processing 21(1), 316–326 (2012)
30. Xu, X., Tsang, I.W., Xu, D.: Handling ambiguity via input-output kernel learning. In: ICDM, pp. 725–734 (2012)
31. Yang, J., Yin, W., Zhang, Y., Wang, Y.: A fast algorithm for edge-preserving variational multichannel image restoration. SIAM Journal on Imaging Sciences 2(2), 569–592 (2009)
32. Zeng, Z., Chan, T.H., Jia, K., Xu, D.: Finding correspondence from multiple images via sparse and low-rank decomposition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 325–339. Springer, Heidelberg (2012)
33. Zeng, Z., Xiao, S., Jia, K., Chan, T.H., Gao, S., Xu, D., Ma, Y.: Learning by associating ambiguously labeled images. In: CVPR, pp. 708–715 (2013)