



Correntropy metric-based robust low-rank subspace clustering for motion segmentation

Li Guo^{1,5} · Xiaoqian Zhang^{1,2} · Zhigui Liu¹ · Qian Wang^{3,5} · Jianping Zhou^{4,5}

Received: 19 January 2021 / Accepted: 17 October 2021 / Published online: 11 November 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract

The subspace clustering methods for motion segmentation are widely used in the field of computer vision. However, the existing methods ignore the low-rank property of motion trajectory with nonlinear structure and are sensitive to non-Gaussian noise. To this end, we seek to improve the performance of motion segmentation by effectively modeling some important characteristics of the motion trajectories, such as nonlinear structure and contained non-Gaussian noise. Specifically, we propose to use kernel function to model motion trajectory, design a variant of the correntropy-induced metric to measure noise, and integrate the block diagonal regularizer into the kernel subspace clustering to strengthen the block diagonal structure of the learned affinity matrix. More importantly, we propose a unified rank-constrained block diagonal subspace clustering method for motion segmentation, which can handle not only rigid body motion segmentation, but also non-rigid motion segmentation. And we further extend this method to deal with various noises in motion data, such as missing trajectories, corrupted trajectory and outlying trajectory. An effective algorithm HQ& AM, which is integrated by Half-quadratic theory and alternating minimization, is designed to optimize these models. Experimental results on several commonly used motion datasets indicate the effectualness and robustness of our methods.

Keywords Subspace clustering · Motion segmentation · Low rank · Block diagonal structure · Correntropy

1 Introduction

The purpose of motion segmentation [56] is to label the trajectory points of different moving objects in a dynamic scene into the corresponding groups, as shown in Fig. 1. It is the cornerstone of many machine vision applications, including

action recognition [22, 29, 30, 54], visual surveillance [6], event detection [55], motion synthesis [15], and many other applications [35, 36]. Kanatani et al. [24] proposed that the motion segmentation from tracking feature points can be converted into a subspace clustering problem, where each subspace represents an independent motion. Subspace clustering can complete many complex tasks, such as document clustering [1, 2], face clustering [5, 58], gene sequence analysis [51], image segmentation [7, 56], motion segmentation [26, 60, 61], etc. The research topic of this paper is subspace clustering for motion segmentation.

Several commonly used subspace clustering approaches were designed, such as kernel k-means (KKM) [42], efficient k-means algorithm [25] and spectral clustering (SC) [37, 50]. However, they cannot be directly used to motion segmentation because the motion trajectory feature data has the following characteristics.

- **Non-linear structure:** In motion segmentation, the trajectories of one motion are mostly located in a non-linear subspace (or submanifold) [20], because the camera always has a certain degree of perspective distortion,

✉ Xiaoqian Zhang
zhxq0528@163.com

Zhigui Liu
liuzhigui@swust.edu.cn

¹ School of Information Engineering, Southwest University of Science and Technology, Mianyang 621000, China

² School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

³ School of Optics and Electronics, Beijing Institute of Technology, Beijing 100081, China

⁴ School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China

⁵ Automatic Research Institute Co., Ltd. of China South industries Group Corporation, Mianyang 621000, China



Fig. 1 Several Image Frames from Cars5 Sequence in the Hopkins155 Motion Dataset. Different color segmentation feature points represent different moving objects (colour figure online)

which leads to the invalid assumption of the affine camera model.

- **Low-rank:** To capture smooth motion data, the capture frame rate of motion capture data is very high. Although high capture frame frequency can effectively avoid the motion jitter between adjacent frames and improve the visual effect of captured data, it also leads to high similarity between adjacent data frames and high redundancy of the entire data. If a matrix is used to represent the motion sequence, the rank of the matrix should be low.
- **Containing noise:** Real motion data obtained by a tracker is a complicated process [40], so noise is inevitable during the segmentation process. A trajectory may correspond to some random motions, which makes the assumption of the affine camera model invalid (outlying trajectory). Furthermore, some features in some frames may be lost due to complex actions and fast motions, resulting in some missing entries in the trajectory (incomplete trajectory). And if some feature points are tracked incorrectly by the tracker unintentionally, resulting in some serious errors in the track (corrupted trajectory). The incorrect trajectories mentioned above are all non-Gaussian noises embedded in the data, and their structures are much more complex [56].
- **Real-time operation:** Compared with independent static data, motion data contains a lot of successive information, which is the only clue to guide the clustering algorithm. Therefore, real-time operation is necessary for motion segmentation applications.

In view of the above special characteristics of the motion trajectory feature data, some superior subspace clustering algorithms have been proposed. In particular, Sparse Subspace Clustering (SSC) [10] can complete sparse representation of data point by selecting within-cluster data. Low-rank Representation (LRR) [31] and Least Square Regression (LSR) [34] encourage the matrix obtained to be low-rank. All these clustering algorithms, however, can only deal with linear data, but cannot handle non-linear data. A few other subspace clustering methods, such as Kernel Sparse Subspace Clustering (KSSC) [38], KSSC on SPD Riemannian

Manifold (KSSCR) [57] and Low Rank Subspace Clustering (LRSC) [45], design nonlinear subspace clustering by kernel tricks. However, these methods with the predefined kernels cannot guarantee the low rank of the data after mapping to the feature space, which hinders the improvement of segmentation performance. Ji et al. [20] proposed a Low Rank Kernel Subspace Clustering (LRKSC) method, which ensures the low-rank attribute of the matrix by learning low-rank kernel mapping. In general, LRKSC can construct an affinity matrix with block diagonal structures, resulting in correct clustering. However, the block diagonal matrix is obtained indirectly by using the low-rank structural prior, which is sensitive to noise.

Aiming at the noise in the motion data, several advanced methods for modeling noise have been put forward. In particular, [10, 20, 45] assumed that noise in the motion data is sparse and used the l_1 -norm to deal with it. A reconstruction errors measurement method based on l_2 -norm [34] was proposed, which is robust to small Gaussian noise. Liu et al. [31] proposed that the $l_{2,1}$ -norm can be used as a measure of data approximation, which is robust to outliers of a specific sample. These methods can only deal with Gaussian noise or sparse noise, but they are very sensitive to non-Gaussian noises such as severely mis-tracked features, missing entries or large outliers. In order to solve the above problems, [3, 43, 52] used the correntropy induced metric (CIM) to handle the non-Gaussian and impulsive noises, and achieved excellent performance. Unfortunately, there is no motion segmentation algorithm that can handle various noises in motion data in a unified way.

Recently, Wang et al. [49] have designed a new low-rank transfer subspace clustering method (LRTSC) for human motion segmentation by transferring existing well-labeled source data into target data. However, LRTSC has an obvious disadvantage that it takes a lot of time for data transfer operations, which is contrary to the timeliness of the motion segmentation application.

In view of the above problems of existing subspace clustering methods in motion data processing, we provide the following solutions respectively.

- To handle nonlinear data, we propose to use kernel function to model motion trajectory.
- To resist various noise contained in motion trajectory, we design a variant of CIM called the kernelized correntropy-induced metric (KCIM) to measure data error.
- To strengthen the block diagonal structure of the learned affinity matrix, we integrate BDR into the kernel subspace clustering.
- To improve the performance of motion segmentation in various scenarios, we propose a unified rank-constrained block diagonal subspace clustering method (RBDSC) and extend it to two robust versions. Furthermore, a new algorithm HQ& AM is designed to optimize these models.

The remainder of this work is structured as follows: (1) Sect. 2 summarizes the related works. (2) Section 3 elaborates our clustering methods and the corresponding optimization. (3) Section 4 carries out experiments and discussion. (4) The current work is summarized in Sect. 5.

2 Related works

We first introduce motion segmentation methods, and then review subspace clustering algorithms based on SC, including SSC [10], LRR [31] and their extensions.

2.1 Motion segmentation

Although many approaches have been proposed in the literature for motion segmentation, they haven't fully considered the important characteristics of the motion trajectory feature data presented in Sect. 1.

In early methods [8, 11, 17, 24], factorization-based approaches attempted to directly detect the motion sequence's cutting points. These methods require that the movements must be independent of each other so that it is easy to handle. However, for most dynamic scenes involving joint objects or motion cameras, the motions are at least partially interdependent.

To deal with partly dependent motions, Vidal et al. [47] put forward general principal component analysis (GPCA), which is a subspace segmentation method based on algebraic geometric. The algorithm does not have any restriction on the relative orientations of the motion subspace, such as allowing arbitrary intersections among different subspaces, so it can handle partially dependent motions. GPCA can work well if and only when the size and the number of subspaces are small, but its performance decreases with the increase of the number of subspaces. Obviously, it is only suitable for short and simple motion sequences.

Many scholars including [12, 23] describe motion segmentation as a statistical clustering problem, which can be solved by expectation–maximization (EM) or its variants. However, they may fall into suboptimal local minima because they require well-initialized iterative methods.

Recently, spectral clustering-based methods have been used for motion segmentation. They first learn an affinity matrix by utilizing local information around each trajectory, and then apply SC to obtain segmentation results [37]. The SC-based methods are the most popular. In this work, we mainly research the clustering segmentation based on spectral clustering.

2.2 Spectral clustering-based subspace clustering

Lately, some advanced SC-based subspace clustering algorithms, such as SSC [10] and LRR [31] have drawn much attention. All of them believe that data points can be expressed as a linear combination of other points from the same subspace. SSC assumes that data has self-expression and sparse-representation property [5, 10]. Specifically, for each data sample x , the coefficient matrix D satisfies

$$\min_{\mathbf{D}} \|\mathbf{D}\|_1 + \gamma \|\mathbf{E}\|_1 \quad \text{s.t. } \mathbf{X} = \mathbf{X}\mathbf{D} + \mathbf{E}, \mathbf{D}_{ii} = 0 \quad (1)$$

where $\|\cdot\|_1$ promotes sparsity of matrix, $\mathbf{E} \in \mathbf{R}^{N \times N}$ is an error matrix and γ represents an adjust parameter.

To recover the subspace structures of data, [31] proposed LRR method, which aims at resolving the following rank minimization problem:

$$\min_{\mathbf{D}, \mathbf{E}} \text{rank}(\mathbf{D}) + \gamma \|\mathbf{E}\|_q \quad \text{s.t. } \mathbf{X} = \mathbf{M}\mathbf{D} + \mathbf{E} \quad (2)$$

where \mathbf{M} denotes a dictionary matrix, the $\text{rank}(\mathbf{D})$ is named the lowest-rank of data matrix \mathbf{X} relative to dictionary \mathbf{M} , and $\|\cdot\|_q$ represents certain regularization strategy, e.g., $\|\cdot\|_1$ is used for characterizing the sparse corruptions, $\|\cdot\|_2$ for modeling small Gaussian noise, and $\|\cdot\|_{2,1}$ for dealing with sample-specific corruptions (or outliers).

To solve equation (2) easily, [31] proposed to replace the rank function with the nuclear norm. And a self-expressiveness method is proposed to represent the dictionary in Eq. (2) by the data matrix \mathbf{X} itself. Thus, Eq. (2) is rewritten as:

$$\min_{\mathbf{D}, \mathbf{E}} \|\mathbf{D}\|_* + \gamma \|\mathbf{E}\|_q \quad \text{s.t. } \mathbf{X} = \mathbf{X}\mathbf{D} + \mathbf{E} \quad (3)$$

where $\|\mathbf{D}\|_*$ represents the trace norm of matrix \mathbf{D} .

SSC and LRR have achieved excellent performance when dealing with linear data, but they cannot deal with nonlinear data. Ji et al. [20] proposed the LRKSC method, which cleverly integrates the “kernel trick” strategy, low-rank learning, and self-representation, so that nonlinear data can be handled and low rank of data can be guaranteed. It is formulated as:

Table 1 The important notations used in this work

Notation	Definition
X	The trajectory matrix
D	The coefficient matrix
B	The affinity matrix
E	The matrix of data errors
$\ X\ _*$	The nuclear norm of matrix X , $\ X\ _* = \sum_i \delta_i$, where δ_i is i th singular value of X
$\ X\ _F$	The frobenius norm of matrix X , $\ X\ _F = \sqrt{\sum_{i,j} x_{ij} ^2}$, where x_{ij} represents (i, j) th element of X
$\text{Diag}(d)$	The diagonal matrix with i th component d_i on the diagonal
$\text{Diag}(D)$	The vector of diagonal components of matrix D
$\text{Tr}(\bullet)$	The trace operator
$\Phi(X)$	The non-linear feature mapping of trajectory matrix
N	The number of data samples
I	The identity matrix
$\mathbf{1}$	The all-one column vector

$$\min_{D, M} \frac{\lambda_2}{2} \|\Phi(X) - \Phi(X)D\|_F^2 + \lambda_1 \|D\|_1 + \|M\|_* + \lambda_3 \|E\|_1 \quad (4)$$

s.t. $D \geq 0, D_{ii} = 0, K = M^T M + E$

where K represents kernel matrix, and $\Phi^T(X)\Phi(X) = M^T M$. However, the above model has two drawbacks: (1) using $\|D\|_1$ can construct an affinity matrix with block diagonal, but its diagonal structure is fragile. (2) Using $\|E\|_1$ can model arbitrary sparse corruptions, but it is a challenge for large errors in motion data.

3 The proposed methods

We propose a unified rank-constrained block diagonal subspace clustering method (RBDSC) for motion segmentation. Then, two robust RBDSC models for suppressing noise in motion data are designed:

rank-constrained block diagonal subspace clustering with modeling noise (RBDSC-MN) and rank-constrained block diagonal subspace clustering with removing noise (RBDSC-RN). Finally, the optimization procedures of the proposed models are given. Table 1 presents the important notations involved in the paper.

3.1 Rank-constrained block diagonal subspace clustering

To handle nonlinear structure of motion trajectory feature data, we adopt the low-rank kernel mapping strategy to map these data into high-dimensional Hilbert space so as to perform linear pattern analysis. And we enforce low-rank constraints on the motion trajectory in the kernel space.

Motivated by LRR [31] and LRKSC [20] methods, we use self-expressiveness-based subspace clustering frameworks to learn the affinity matrix. By comprehensively considering low-rank kernel mapping and self-expressiveness, the low-rank kernel self-expressiveness minimization problem is expressed as:

$$\min_{D, M} \frac{1}{2} \|M - MD\|_F^2 + \lambda_1 \mathfrak{R}(D) + \|M\|_* \quad (5)$$

s.t. $D \geq 0, D_{ii} = 0, K = M^T M$

where λ_1 is a nonnegative balancing parameter, and $\mathfrak{R}(D)$ is the regularization term about matrix D . According to different motivations, $\mathfrak{R}(D)$ can represent different regularization items, such as $\|D\|_*$ and $\|D\|_1$. The affinity matrixes constructed by these methods all have the common block-diagonal property, which contributes to ameliorating the performance of SC-based subspace clustering methods. However, their block diagonal structures acquired by $\|\bullet\|_*$ and $\|\bullet\|_1$ are generally frangible and will be damaged when encountering a small signal-to-noise ratio, thereby reducing the clustering results. Inspired by [33], we introduce BDR to directly pursue the block-diagonal representation of coefficient matrix D . Therefore, we propose a rank-constrained block diagonal representation method (RBDSC) and it is formulated as:

$$\min_{D, M} \frac{1}{2} \|M - MD\|_F^2 + \lambda_1 \|D\|_k + \|M\|_* \quad (6)$$

s.t. $D \geq 0, D_{ii} = 0, D = D^T, K = M^T M$

where k is the number of clusters or classes.

However, the constraint on matrix D in Eq. (6) limits its representation ability. To alleviate this issue, we add

intermediate matrix \mathbf{H} , which satisfies $\mathbf{H} = \mathbf{D}$. The problem (6) translates to

$$\min_{\mathbf{D}, \mathbf{M}, \mathbf{H}} \lambda_1 \|\mathbf{D}\|_k + \|\mathbf{M}\|_* + \frac{\|\mathbf{M} - \mathbf{MH}\|_F^2}{2} + \frac{\lambda_2 \|\mathbf{H} - \mathbf{D}\|_F^2}{2} + \frac{\lambda_3 \|\mathbf{K} - \mathbf{M}^T \mathbf{M}\|_F^2}{2}$$

$$s.t. \mathbf{D} \geq 0, \mathbf{D}_{ii} = 0, \mathbf{D} = \mathbf{D}^T \tag{7}$$

3.2 Robust rank-constrained block diagonal subspace clustering

For LRKSC in Eq. (4), the $\|\mathbf{E}\|_1$ can only model arbitrary sparse corruptions, but it cannot handle noise in motion data very well, because these noises are complex and unpredictable. Unlike L_1 -norm, correntropy-induced metric (CIM) [52] is a local metric, which is particularly robust to complex noise [39] and non-Gaussian noise [52]. It has been widely used in the field of computer vision, such as face clustering [4], object image clustering [3], feature selection [14], and semi-supervised learning [34]. The correntropy [16, 52] of arbitrary variables w and v is formulated as

$$V_\sigma(w, v) = \frac{1}{N} \sum_{i=1}^N g_\sigma(w_i - v_i) \tag{8}$$

where $g_\sigma(\bullet)$ denotes kernel function, and σ is the width of $g(w_i, v_i)$. Gaussian kernel $g_\sigma(w_i, v_i) = \exp(-\|w_i - v_i\|_2^2 / 2\sigma^2)$ is usually used as the kernel function. Generally, CIM [52] is used to evaluate the similarity of two variables w and v :

$$CIM(w, v) = \sqrt{1 - V_\sigma(w, v)}$$

$$= \sqrt{1 - \frac{1}{N} \sum_{i=1}^N g_\sigma(w_i - v_i)} \tag{9}$$

Unfortunately, we cannot directly introduce CIM into our model, because CIM is designed for linear data, while the motion tracking features are nonlinear. Fortunately, the motion data is linear in the Hilbert eigenspace [20], where correntropy can be applied to the motion data. Thus, we design a variant of CIM called the kernelized correntropy-induced metric (KCIM), which is expressed as

$$KCIM(\phi(w), \phi(v)) = \sqrt{1 - \frac{1}{N} \sum_{i=1}^N g_\sigma(\phi(e_i))} \tag{10}$$

where $\phi(e_i) = \phi(w_i) - \phi(v_i)$ for $i = 1, 2, 3 \dots N$.

It can be seen that KCIM is close to 1 when handling a large error ($\phi(e_i)$ is large), which is much smaller than the absolute error and the mean squared error used. Larger noise usually comes from data loss, data corruption and outliers [39]. Thus, we propose a rank-constrained block diagonal

subspace clustering with modeling noise (RBDSC-MN), which combines the rank-constrained block diagonal representation (RBDSC) and KCIM. And the RBDSC-MN model is formulated as:

$$\min_{\mathbf{D}, \mathbf{M}, \mathbf{H}, \mathbf{E}} \lambda_1 \|\mathbf{D}\|_k + \|\mathbf{M}\|_* + \frac{1}{2} \|\mathbf{M} - \mathbf{MH}\|_F^2 +$$

$$\frac{\lambda_2}{2} \|\mathbf{H} - \mathbf{D}\|_F^2 + \frac{\lambda_3}{2} \|\mathbf{E} - (\mathbf{K} - \mathbf{M}^T \mathbf{M})\|_F^2$$

$$+ \sum_i \sum_j (1 - g_\sigma(\mathbf{E}_{ij}))$$

$$s.t. \mathbf{D} \geq 0, \mathbf{D}_{ii} = 0, \mathbf{D} = \mathbf{D}^T$$

where \mathbf{E}_{ij} is the entry (i, j) of matrix \mathbf{E} .

For larger outliers, the prior information can be further used in the model proposed in (11) to improve the clustering accuracy. Inspired by $\|\bullet\|_{2,1}$ (group sparsity) in LRR [31] to handle large outliers [14, 31, 32], we design a new robust subspace clustering, namely RBDSC-RN, to remove outlying trajectories. And it is formulated as:

$$\min_{\mathbf{D}, \mathbf{M}, \mathbf{H}, \mathbf{E}} \lambda_1 \|\mathbf{D}\|_k + \|\mathbf{M}\|_* + \frac{1}{2} \|\mathbf{M} - \mathbf{MH}\|_F^2 +$$

$$\frac{\lambda_2}{2} \|\mathbf{H} - \mathbf{D}\|_F^2 + \frac{\lambda_3}{2} \|\mathbf{E} - (\mathbf{K} - \mathbf{M}^T \mathbf{M})\|_F^2$$

$$+ \sum_i (1 - g_\sigma(\|\mathbf{E}^i\|_2))$$

$$s.t. \mathbf{D} \geq 0, \mathbf{D}_{ii} = 0, \mathbf{D} = \mathbf{D}^T$$

where \mathbf{E}^i is the i th row of \mathbf{E} .

Therefore, to effectively deal with the motion data errors, two robust rank-constrained block diagonal subspace clustering methods are used: the RBDSC-MN model in Eq. (11) (handling incomplete trajectories and corrupted trajectories), and the RBDSC-RN model in Eq. (12) (handling outlying trajectories).

3.3 Optimization method

Since the objective functions in Eqs. (11) and (12) are non-convex, optimizing them is a challenging problem. We design a novel algorithm HQ& AM, which combines Half-quadratic theory [52] and alternating minimization [33]. Because Eqs. (7), (11) and (12) are identical except for the error term \mathbf{E} , we present the optimization process of the model RBDSC (Eq. (7)), and then illustrate the optimization schemes of model RBDSC-MN (Eq. (11)) and model RBDSC-RN (Eq. (12)).

3.3.1 Optimization model RBDSC

(1) Updating \mathbf{M} :

By fixing (\mathbf{D}, \mathbf{H}) , the optimization sub-problem corresponding to \mathbf{M}^{t+1} in Eq. (7) is

$$\min_M \|M\|_* + \frac{1}{2} \|M - MH\|_F^2 + \frac{\lambda_3}{2} \|K - M^T M\|_F^2 \tag{13}$$

where $\|M - MH\|_F^2$ can be expanded and we have

$$\|M - MH\|_F^2 = \text{tr}(I - 2H + HH^T)M^T M \tag{14}$$

Equation (13) can be rewritten as

$$\min_M \|M\|_* + \frac{\lambda_3}{2} \|M^T M - K + \frac{(I - 2H + HH^T)}{2\lambda_3}\|_F^2 \tag{15}$$

We define $K_G = K - \frac{(I - 2H + HH^T)}{2\lambda_3}$, then Eq. (15) is transformed to

$$\min_M \|M\|_* + \frac{\lambda_3}{2} \|M^T M - K_G\|_F^2 \tag{16}$$

We can obtain a closed-form solution by

$$M^{t+1} = \Sigma^* \Delta^T \tag{17}$$

where Σ^* and Δ are both related to the SVD of K_G . Let $K_G = \Gamma \Lambda \Delta^T$ represent the SVD of K_G , v_i is the i th singular value of K_G .

$$\Sigma^* = \begin{pmatrix} \zeta_1^* & & 0 \\ & \ddots & \\ 0 & & \zeta_N^* \end{pmatrix} \text{ with } \zeta_i^* = \arg \min_{\zeta_i} \frac{\lambda_3}{2} (v_i - \zeta_i)^2 + \zeta_i$$

$\zeta_i \in \{x \in \mathcal{R}_+ | x^3 - v_i x + (2\lambda_3)^{-1} = 0\} \cup \{0\}$. Δ is a matrix including the singular values of K_G .

(2) Updating H:

By fixing (D, M) , the optimization sub-problem corresponding to H^{t+1} in Eq. (7) is

$$\min_H \frac{1}{2} \|M - MH\|_F^2 + \frac{\lambda_2}{2} \|H - D\|_F^2 \tag{18}$$

We can obtain its closed-form solution given by

$$H^{t+1} = (\lambda_2 D + M^T M) / (\lambda_2 I + M^T M) \tag{19}$$

(3) Update D:

By fixing (M, H) , the optimization sub-problem corresponding to D^{t+1} in Eq. (7) is

$$\min_D \lambda_1 \|D\|_k + \frac{\lambda_2}{2} \|H - D\|_F^2 \tag{20}$$

s.t. $D \geq 0, D_{ii} = 0, D = D^T$

Define $L_D = \text{Diag}(D1) - D$ is the Laplacian matrix corresponding to matrix D . The k block-diagonal representation refers to the addition of k smallest eigenvalues:

$$\|D\|_k = \sum_{i=N+1-k}^N \lambda_i(L_D) \tag{21}$$

where $\lambda_i(L_D)$ is the i th smallest eigenvalue of L_D . According to [9], Eq. (21) can be rewritten as

$$\sum_{i=N+1-k}^N \lambda_i(L_D) = \min_Q \text{Tr}(Q L_D) \tag{22}$$

s.t. $\text{Tr}(Q) = k, 0 \leq Q \leq I$

So, Eq. (20) can be transformed to

$$\min_{D, Q} \lambda_1 \text{Tr}(Q(\text{Diag}(D1) - D)) + \frac{\lambda_2}{2} \|H - D\|_F^2 \tag{23}$$

s.t. $D \geq 0, D_{ii} = 0, D = D^T, 0 \leq Q \leq I, \text{Tr}(Q) = k$

- Fix $D = D^T$, updating Q by

$$Q^{t+1} = \arg \min_Q \lambda_1 \text{Tr}(Q(\text{Diag}(D1) - D)) \tag{24}$$

s.t. $\text{Tr}(Q) = k, 0 \leq Q \leq I$

Inspired by [9], we get the solution of Eq. (24) as

$$Q^{t+1} = V_* V_*^T \tag{25}$$

where V_* is the matrix containing the eigenvectors of k minimum eigenvalues corresponding to $\text{Diag}(D1) - D$.

- Fix $Q = Q^T$, updating D by

$$D^{t+1} = \arg \min_D \lambda_1 \text{Tr}(Q(\text{Diag}(D1) - D)) + \frac{\lambda_2}{2} \|H - D\|_F^2 \tag{26}$$

s.t. $D \geq 0, D_{ii} = 0, D = D^T$

Equation (26) can be translated into follows:

$$D^{t+1} = \arg \min_D \frac{1}{2} \left\| D - \left(H - \frac{\lambda_1}{\lambda_2} (\text{diag}(Q) \mathbf{1}^T - Q) \right) \right\|_F^2 \tag{27}$$

s.t. $C \geq 0, C_{ii} = 0, C = C^T$

Let $U \in \mathcal{R}^{N \times N}$, defining $U = H - \frac{\lambda_1}{\lambda_2} (\text{diag}(Q) \mathbf{1}^T - Q)$, $\hat{U} = U - \text{Diag}(\text{diag}(U))$. The closed-form solution of Eq.(27) is

$$D^{t+1} = \left[\frac{\hat{U} + \hat{U}^T}{2} \right]_+ \tag{28}$$

3.3.2 Optimization model RBDSC-MN

(1) Updating M:

By fixing (D, H) , the optimization sub-problem corresponding to M^{t+1} in Eq. (11) is

$$\min_M \|M\|_* + \frac{1}{2} \|M - MH\|_F^2 + \frac{\lambda_3}{2} \|K - M^T M - E\|_F^2 \tag{29}$$

Except for K_G , the problem of solving Eq. (29) is similar to that of Eq. (13).

$$K_G = K - \frac{(I - 2H + HH^T)}{2\lambda_3} - E \tag{30}$$

(2) Updating H, D :

By comparing the two Eqs. (7) and (11), the updates of H and D in Eq. (11) are the same as Eqs. (18) and (20), respectively. Thereby, the solutions of H and D are obtained by Eqs. (19) and (28), respectively.

(3) Updating E :

By fixing (M, H, D) , the optimization sub-problem corresponding to E^{t+1} in Eq. (11) is

$$\min_E \frac{\lambda_3}{2} \|E - (K - M^T M)\|_F^2 + \sum_i \sum_j (1 - g_\sigma(E_{ij})) \quad (31)$$

According to HQ analysis [52], the above problem (31) can be converted into

$$\min_{E, S} \frac{\lambda_3}{2} \|E - (K - M^T M)\|_F^2 + \|S^{\frac{1}{2}} \otimes E\|_F^2 + \sum_i \sum_j \Psi(S_{ij}) \quad (32)$$

where S is the matrix containing the auxiliary variable S_{ij} , \otimes represents the Hadamard product, and $\Psi(\bullet)$ is the dual potential function of $1 - g_\sigma(\bullet)$.

A new optimization algorithm HQ&AM is designed to solve Eq. (32). Specifically, in each iteration, S is optimized by HQ theory, and then E is optimized by alternating minimization.

- Fix E , updating S by

$$S_{ij}^{t+1} = g_\sigma(E_{ij}^t) = \exp(-(E_{ij}^t)^2 / 2\sigma^2) \quad (33)$$

- Fix S , updating E by

$$\begin{aligned} E^{t+1} &= \arg \min_E \frac{\lambda_3}{2} \|E - (K - M^T M)\|_F^2 \\ &\quad + \left\| (S^{t+1})^{\frac{1}{2}} \otimes E \right\|_F^2 \\ &= (K - M^T M) ./ \left(\mathbf{1} + \frac{2S^{t+1}}{\lambda_3} \right) \end{aligned} \quad (34)$$

where $./$ represents element-by-element division.

3.3.3 Optimization model RBDSC-RN

(1) Updating M, H, D :

By comparing the two Eqs. (11) and (12), the updates of M, H and D in Eq. (12) are the same as (29), (18) and (20), respectively. Thereby, the solutions of H and D are obtained by (19) and (28), and optimizing M is similar to solving Eq. (13), but $K_G = K - \frac{(I - 2H + HH^T)}{2\lambda_3} - E$.

(2) Updating E :

By fixing $(M, H$ and $D)$, the optimization sub-problem corresponding to E^{t+1} in Eq. (12) is

$$\min_E \frac{\lambda_3}{2} \|E - (K - M^T M)\|_F^2 + \sum_i (1 - g_\sigma(\|E^i\|_2)) \quad (35)$$

According to HQ analysis [52], the above problem (35) can be converted into

$$\min_{E, S} \frac{\lambda_3}{2} \|E - (K - M^T M)\|_F^2 + \|S^{\frac{1}{2}} \otimes E\|_F^2 + \sum_i \Psi(S_i) \quad (36)$$

where S is the matrix containing the auxiliary variable S_i .

- Fix E , updating S by

$$\begin{aligned} S_{ij}^{t+1} &= g_\sigma(\|E^t\|_2^i) \\ &= \exp(-\|E^t\|_2^i / 2\sigma^2), i = 1, 2 \dots N \end{aligned} \quad (37)$$

where $(E^t)^i$ is the i th row of matrix E^t .

- Fix S , updating E

The solutions of E can be obtained by (34).

3.4 Clustering

According to whether the motion trajectory feature data contains noise and the source of noise, we select the corresponding algorithm 1, 2 and 3 to solve (7), (11) and (12) respectively.

Algorithm 1 The RBDSC algorithm

Input: number of cluster k , data matrix \mathbf{X} , parameters $\lambda_1, \lambda_2, \lambda_3, \text{maxiter}=60$.

- 1: Initialize: $\mathbf{M}=\sqrt{\mathbf{K}}, \mathbf{D}=0, \mathbf{H}=0, \mathbf{Q}=0, \varepsilon=10^{-3}, t=0$
- 2: **while** stopping criterion is not met **do**
- 3: Calculate \mathbf{M}^{t+1} by resolving (17).
- 4: Compute \mathbf{H}^{t+1} by solving (19).
- 5: Compute \mathbf{Q}^{t+1} by resolving (25).
- 6: Compute \mathbf{D}^{t+1} by solving (28).
- 7: $t = t + 1$.
- 8: Check the stopping criterion

$$\max\{\|\mathbf{M}^{t+1} - \mathbf{M}^t\|_\infty, \|\mathbf{H}^{t+1} - \mathbf{H}^t\|_\infty, \|\mathbf{Q}^{t+1} - \mathbf{Q}^t\|_\infty, \|\mathbf{D}^{t+1} - \mathbf{D}^t\|_\infty\} \leq \varepsilon, \text{ and } t < \text{maxiter}$$

9: **end while**

10: Compute $\mathbf{B} = (|\mathbf{D}| + |\mathbf{D}|^T)/2$

Output: The value of \mathbf{B} .

Algorithm 2 The RBDSC-MN algorithm

Input: number of cluster k , data matrix \mathbf{X} , parameters $\lambda_1, \lambda_2, \lambda_3, \text{maxiter}=60$.

- 1: Initialize: $\mathbf{M}=\sqrt{\mathbf{K}}, \mathbf{D}=0, \mathbf{H}=0, \mathbf{Q}=0, \mathbf{E}=0, \varepsilon=10^{-3}, t=0$
- 2: **while** stopping criterion is not met **do**
- 3: Update $\mathbf{M}, \mathbf{H}, \mathbf{Q}, \mathbf{D}, \mathbf{S}, \mathbf{E}$ in the order of (17), (19), (25), (28), (33), (34).
- 4: $t = t + 1$.
- 5: Check the stopping criterion

$$\max\{\|\mathbf{M}^{t+1} - \mathbf{M}^t\|_\infty, \|\mathbf{H}^{t+1} - \mathbf{H}^t\|_\infty, \|\mathbf{Q}^{t+1} - \mathbf{Q}^t\|_\infty, \|\mathbf{D}^{t+1} - \mathbf{D}^t\|_\infty, \|\mathbf{S}^{t+1} - \mathbf{S}^t\|_\infty, \|\mathbf{E}^{t+1} - \mathbf{E}^t\|_\infty\} \leq \varepsilon, \text{ and } t < \text{maxiter}$$

6: **end while**

7: Compute $\mathbf{B} = (|\mathbf{D}| + |\mathbf{D}|^T)/2$

Output: The value of \mathbf{B} .

Algorithm 3 The RBDSC-RN algorithm

Input: number of cluster k , data matrix \mathbf{X} , parameters $\lambda_1, \lambda_2, \lambda_3, \text{maxiter}=60$.

- 1: Initialize: $\mathbf{M}=\sqrt{\mathbf{K}}, \mathbf{D}=0, \mathbf{H}=0, \mathbf{Q}=0, \mathbf{E}=0, \varepsilon=10^{-3}, t=0$
- 2: **while** stopping criterion is not met **do**
- 3: Update $\mathbf{M}, \mathbf{H}, \mathbf{Q}, \mathbf{D}, \mathbf{S}, \mathbf{E}$ in the order of (17), (19), (25), (28), (37), (34).
- 4: $t = t + 1$.
- 5: Check the stopping criterion

$$\max\{\|\mathbf{M}^{t+1} - \mathbf{M}^t\|_\infty, \|\mathbf{H}^{t+1} - \mathbf{H}^t\|_\infty, \|\mathbf{Q}^{t+1} - \mathbf{Q}^t\|_\infty, \|\mathbf{D}^{t+1} - \mathbf{D}^t\|_\infty, \|\mathbf{S}^{t+1} - \mathbf{S}^t\|_\infty, \|\mathbf{E}^{t+1} - \mathbf{E}^t\|_\infty\} \leq \varepsilon, \text{ and } t < \text{maxiter}$$

6: **end while**

7: Compute $\mathbf{B} = (|\mathbf{D}| + |\mathbf{D}|^T)/2$

Output: The value of \mathbf{B} .

4 Experimental evaluation

To demonstrate the superiority of our methods, we conduct extensive experiments on different video sequences. We first evaluate the performance of the proposed algorithms on rigid and non-rigid structure-from-motion: (1) rigid-body motion segmentation [33, 53] with complete trajectories: **Hopkins155** Traffic sequences [44]; (2) non-rigid motion (e.g., human motion) segmentation [53]: **Keck Gesture** [21] and **UT-Interaction** [41]. Then, we do experiments by using two frames of each sequence in the **Hopkins155** dataset [44] to compare our performance. Finally, we conduct robustness experiments: (1) missing and corrupted trajectories: **Hopkins12** motion sequences [46]; (2) outlying trajectories¹: **Cars Turning**, **Books** and **Nrbooks**.

4.1 Compared methods and settings

We compare the proposed methods with the following baselines: SSC [10], LRR [31], SSC-OMP [59], LSR [34], RSIM [19], LRKSC [20], BDR-B [33], BDR-Z [33]. Especially in the human motion segmentation experiment, in addition to comparing our models with the above models, we also compare them with specially designed human motion segmentation methods, including TSC [27], TSS [48], and LRTSC [49].

In the performance comparison test, we try our best to use the available source code and adjust the parameters to obtain excellent results. The misclassification ratio with the best mapping is widely used as a measure of clustering validity [10, 31, 34, 38, 57]. The expression is shown as follows:

¹ <http://www.vision.jhu.edu/data/hopkins155/>.

$$Err\% = \frac{\sum_{i=1}^n v(s_i, \sim \text{map}(g_i))}{n} \quad (38)$$

where $v(x, \sim y) = \begin{cases} 1, x \neq y \\ 0, x = y \end{cases}$, s_i is the ground truth of the i th element, $\text{map}(g_i)$ represents the permutation mapping function, and $\sim \text{map}(g_i)$ denotes that the cluster label g_i cannot be mapped to the corresponding ground truth. $Err\%$ is a negative metric, and a lower value means better performance. As suggested in [20], we repeat each experiment 20 times and give an average result to mitigate the impact of initialization.

4.2 Rigid-body motion segmentation with complete trajectories: Hopkins155 traffic sequences [44]

The **Hopkins155** dataset [44] is a standard motion segmentation dataset to test subspace clustering based motion segmentation algorithms. It includes the video sequences and the trajectory features tracked in all frames. It consists of 38 traffic sequences, 104 checkerboard sequences and 13 other sequences. The moving objects of Hopkins155 Traffic sequences are cars and trucks, which produce typical rigid structure-from-motions. Figure 2 shows two examples from Hopkins155 Traffic sequences.

Hopkins155 Traffic sequences can be researched under the affine assumption, because the rigid motion trajectories across multiple frames is located in the affine subspace. Considering that each video sequence in the Hopkins155 Traffic sequences has a complete feature track and outliers in the data set have been manually deleted, we apply model RBDSC (with Eq. (7) solved by Algorithm 1) to test the efficiency of our method. The parameters for model RBDSC are set for $\lambda_1 = 2.8$, $\lambda_2 = 10$, $\lambda_3 = 1.3 * 10^5$, and the polynomial kernel $k(x_1, x_2) = (0.8 + x_1^T x_2)^2$ is used to define K in our model. Table 2 reports the results of our RBDSC method and baselines. We can easily see that the RBDSC method consistently obtains better results than other advanced methods.

4.3 Non-rigid motion segmentation

Human motion is a typical non-rigid motion with great deformation, so it is very challenging to segment different moving objects. To demonstrate the effectiveness of our model for non-rigid motion, we conduct experiments on two typical human motion datasets, including **Keck Gesture Dataset (Keck)** [21] and **UT-Interaction Dataset (UT)** [41].

- **Keck** [21] contains 14 different types of gestures, which represent different military signals. Each of the three people performs these 14 gestures. In each sequence,

everyone repeats the same gesture three times. The resolution of the RGB frame is $640 * 480$. Figure 3 shows some examples of different gesture classes from the Keck dataset.

- **UT** [41] contains 20 videos, each of which contains six types of human interaction, including pushing, shaking hands, punching, kicking, pointing and hugging. The resolution of each video sequence is $780 * 480$, and the duration is about one minute. Figure 4 shows a few snapshots of human interaction videos from the UT dataset.

In the experiment, we use the low-level HoG feature extracted in [49]. Then, for the comparison methods of different theories, we carry out corresponding experimental settings. For the motion segmentation method based on transfer learning knowledge (TSC, TSS, LRTSC), we refer to [49] to segment the test video sequence based on another data (source). Specifically, TSC, TSS and LRTSC all set the **UT (Keck) Dataset** as the source and the **Keck (UT) Dataset** as the target. For other comparison algorithms (not using source information), such as LRR, LSR, SSC and LRKSC, we only input video sequences in the target dataset. We set the parameters of model RBDSC as $\lambda_1 = 2.8$, $\lambda_2 = 21$, $\lambda_3 = 140$, and the kernel function $k(x_1, x_2) = (0.8 + x_1^T x_2)^3$ is used to define K in our model. The results are recorded in Table 3. Obviously, the error rate of RBDSC is much lower than that of 11 advanced clustering methods. In particular, compared with the second-best model on UT dataset, LRTSC, our model RBDSC achieves a lower classification error rate at an average of 31.75 %.

Then, we compare the computational cost of all comparison methods, all of which were implemented with MATLAB R2019a on a Windows computer with a 3.0 GHz Intel(R) Core(TM) i5-7400 CPU and 8.00 GB memory. Figure 5 shows the running time (in seconds) of all methods on UT dataset, and these methods are ordered in ascending computational costs as SSC-OMP, LSR, RSIM, LRKSC, SSC, RBDSC, BDR-B, BDR-Z, LRR, TSS, TSC and LRTSC. We observe that RBDSC significantly reduces the computing cost compared with the second-best method (LRTSC). The results demonstrate that RBDSC is an effective and time-saving method.

4.4 Results on Two-frame of each sequence in the Hopkins155 Dataset [44]

The impact of frame counts is very important for designing real-time applications that require processing sequentially input data. To evaluate the performance of our model when inputting a small number of frames, we manually select the first and last frames [20, 28] from each video sequence in the Hopkins155 dataset to form a two-frame Hopkins155 dataset. In view of the problem that the two frames of video



Fig. 2 2 Examples from Hopkins 155 traffic sequences

sequence cannot accurately satisfy the assumption of camera affine model, [13] proposed that the subspace was derived from the rewriting limit constraint

$$x_d'^T F x_d = 0 \tag{39}$$

where $x_d^T = [x_d, y_d, 1]^T$ and $x_d'^T = [x_d', y_d', 1]^T$ were the homogeneous coordinates of the two points corresponding to the 3-D point d in the two-frame, and $F \in \mathfrak{R}^{3 \times 3}$ is the fundamental matrix. Equation (39) can be rewritten as [18]

$$f^T \text{vec}(x_d' x_d^T) = 0 \tag{40}$$

Table 2 Segmentation errors (in %) on Hopkins155 traffic sequences

Methods	LSR	SSC	LRR	RSIM	SSC-OMP	LRKSC	BDR-B	BDR-Z	RBDSC
2 motions									
Mean	1.13	0.08	0.52	0.10	19.07	0.46	1.60	0.59	0.06
Median	0.00	0.00	0.00	0.00	15.48	0.00	0.00	0.00	0.00
3 motions									
Mean	3.50	0.76	1.81	2.08	23.59	0.79	1.36	1.19	0.39
Median	0.26	0.00	0.00	0.19	28.60	0.00	0.19	0.19	0.00
ALL									
Mean	1.67	0.21	0.76	0.46	19.90	0.52	1.56	0.70	0.12
Median	0.09	0.00	0.00	0.00	17.74	0.00	0.00	0.00	0.00

Bold font represents the best results

where $f \in \mathfrak{R}^{9 \times 9}$ denotes vectorized fundamental matrix F , $\text{vec}(x_d' x_d^T) = (x_d x_d', x_d y_d', x_d y_d', y_d y_d', y_d x_d', y_d', y_d', 1)^T$. Thus, $\text{vec}(x_d' x_d^T)$ lies on the epipolar subspace of dimension 8 [18]. We assume that two perspective images with multi-motion have multi-epipolar subspaces because different motions are relative to different fundamental matrices [20].

We replicate the data 30 times for increasing the ambient dimension, and set the parameters of model RBDSC (with Eq. (7) solved by Algorithm 1) as $\lambda_1 = 3, \lambda_2 = 5, \lambda_3 = 1.3 * 10^5$. The kernel function $k(x_1, x_2) = (1.8 + x_1^T x_2)^2$ is used to define K in our model.

Table 4 reports the results of our RBDSC method and baselines. Through in-depth analysis of these results, we can get the following observations.

- We can easily see that our model RBDSC generally achieves the lowest misclassification rate compared with these advanced algorithms.
- We observe that RBDSC is superior to LRKSC overall. This is mainly because RBDSC introduces direct block diagonal constraint [33], while LRKSC uses low rank prior to indirectly approximate block diagonal structure. Figure 6 displays the comparison of the block-diagonal



Fig. 3 Examples of 7 different gesture classes in the Keck dataset [21]

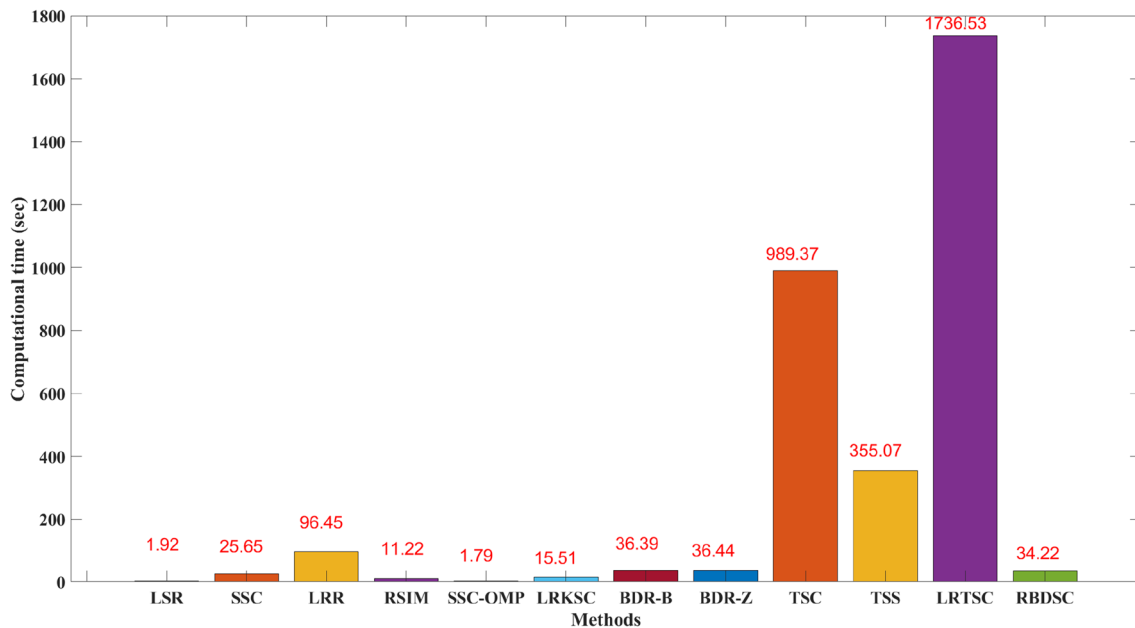


Fig. 4 Some snapshots of human interaction videos in the UT dataset [41]

Table 3 Segmentation Errors (in %) on two human motion datasets

Data	LSR	SSC	LRR	RSIM	SSC-OMP	LRKSC	BDR-B	BDR-Z	TSC	TSS	LRTSC	RBDSC
Keck	51.06	68.63	57.03	50.03	78.52	59.70	74.31	67.61	55.79	50.49	44.81	42.35
UT	48.17	56.11	58.38	38.97	57.17	52.77	72.17	69.29	47.87	46.29	38.73	6.98
Mean	49.62	62.37	57.71	44.50	67.85	56.24	73.24	68.45	51.83	48.39	41.77	24.67

Bold font represents the best results

**Fig. 5** Comparison of the running time of different algorithms on **UT Dataset**

property of the affinity matrix obtained by different clustering methods. These images can be zoomed in on the screen for best results. Obviously, RBDSC has the best block diagonal structure.

- Our method RBDSC consistently outperforms BDR-B and BDR-Z, which clearly demonstrates the benefits of combining block diagonal regularizer (BDR) [33] with kernel mapping technology and low-rank kernel learning strategy. Specifically, RBDSC has the ability to handle non-linear data by low-rank kernel mapping when the sequences cannot strictly meet the assumptions of the affine camera model.

4.5 Experimental results on robustness

4.5.1 Missing and corrupted trajectories: Hopkins12 Dataset [46]

In practice, some trajectory features in some image frames are lost due to the limitation or occlusion of the tracker. Hopkins12 dataset contains 12 motion sequences: 3 3-motion

sequences and 9 2-motion sequences. For these sequences, they have lost some of the trajectory features to some extent. And corrupted trajectories can emerge in video sequences when the tracker unknowingly loses tracking of certain feature points. Therefore, it is challenging to segment the motion subjects in video sequences.

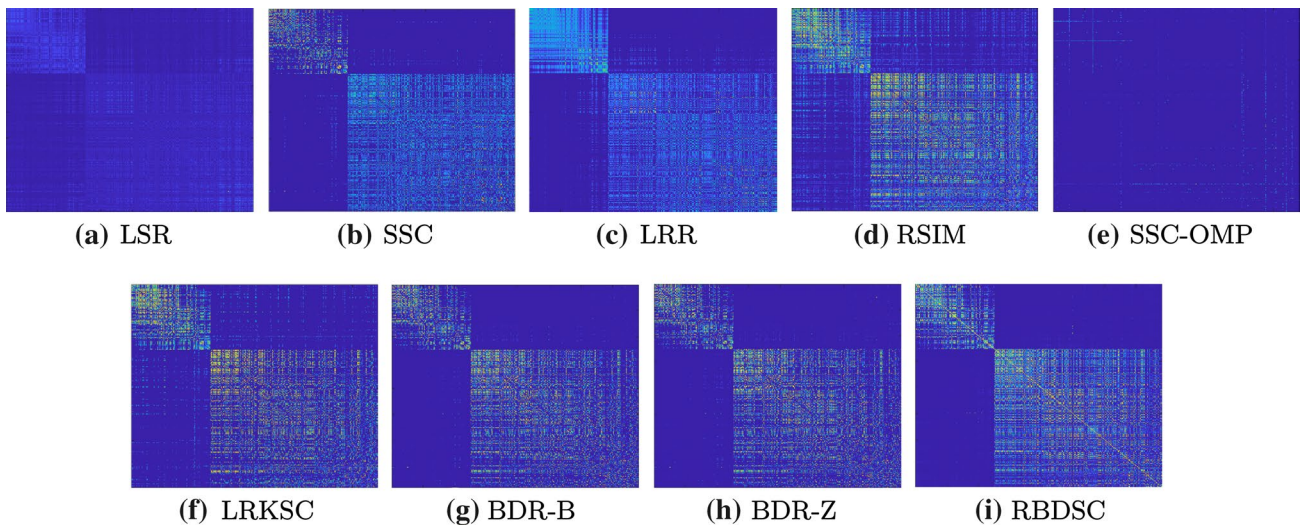
We test our model RBDSC-MN and several advanced subspace segmentation methods on the Hopkins12 dataset with incomplete or corrupted trajectories. Figure 7 shows some frames of 3-motion sequence in the Hopkins12 dataset.

We set the parameters of model RBDSC-MN (with Eq. (11) solved by Algorithm 2)) as $\lambda_1 = 2.8$, $\lambda_2 = 10$, $\lambda_3 = 1000$, and the kernel function $k(x_1, x_2) = (0.8 + x_1^T x_2)^2$ is used to define \mathbf{K} in our model. The results of the model RBDSC-MN and its comparison methods on the Hopkins12 dataset are listed in Table 5. And it distinctly indicates the benefits of model RBDSC-MN in the presence of missing trajectories. RSIM is the second-best method, but it relies on additional optimization strategy on the Grassmann manifold to acquire the estimated value of the orthogonal matrix \mathbf{V} to reduce the segmentation errors when the trajectory is incomplete. The competitiveness of

Table 4 Segmentation errors (in %) on two-frame **Hopkins155** Dataset

Methods	LSR	SSC	LRR	RSIM	SSC-OMP	LRKSC	BDR-B	BDR-Z	RBDSC
2 motions									
Mean	2.24	3.52	6.18	7.39	14.41	2.53	4.01	5.23	2.13
Median	0.10	0.10	0.10	0.00	9.60	0.36	0.00	0.10	0.00
3 motions									
Mean	4.95	10.93	14.92	14.53	37.32	5.93	11.29	12.84	4.72
Median	1.83	8.65	10.16	7.99	39.17	0.20	3.55	6.43	1.37
ALL									
Mean	2.86	5.20	8.15	9.00	19.58	3.30	5.66	6.95	2.72
Median	0.31	0.52	0.54	0.55	15.98	0.36	0.21	0.39	0.00

Bold font represents the best results

**Fig. 6** Visual comparison of the block-diagonal of the affinity matrix obtained on the **L1R2RCT_B_g23** sequence through different models

model RBDSC-MN comes from the fact that correntropy is used as the loss function for robust subspace clustering deals with missing trajectories, which are non-Gaussian noises. This demonstrates that the model RBDSC-MN can effectively handle non-Gaussian noises in motion segmentation without any preprocessing or postprocessing step.

4.5.2 Outlying trajectories: Cars Turning, Books and Nrbooks

Dynamic scenes often contain some sample outliers, which do not belong to any motion trajectory in the scene. Figure 8 displays three typical video trajectories, which contain both outlying trajectories (outliers) and missing data samples. More detailed information is shown in Table 6. We design model RBDSC-RN to remove outlier rows, and use three representative motion trajectories (Cars Turning, Books and Nrbooks) to evaluate its performance. Table 7 lists the subspace segmentation results of different methods. Obviously, the RBDSC-RN achieves the lowest mis-segmentation rates. The main reason is that the model RBDSC-RN uses

non-convex row-correntropy to eliminate outliers, and it is less affected by larger outliers.

4.6 Parameter sensitivity

Our models contain three important parameters: λ_1 , λ_2 and λ_3 , where λ_1 constrains the significance of BDR term $\|D\|_k$, λ_2 and λ_3 control the weight of $\|H - D\|_F^2$ and $\|E - (K - M^T M)\|_F^2$ (or $\|K - M^T M\|_F^2$) respectively. We use numerous values to evaluate the parameter sensitivity of the proposed model on the Hopkins12 dataset, and record the average misclassification rate of 12 sequences. Figure 9 shows the experimental results of $\lambda_1 \in [2.5 \ 2.6 \ 2.7 \ 2.8 \ 2.9 \ 3]$, $\lambda_2 \in [5 \ 10 \ 15 \ 20 \ 25 \ 30]$, and $\lambda_3 \in [0.01 \ 0.1 \ 1 \ 10 \ 100 \ 1000]$. For Hopkins12 dataset, our method can achieve lower segmentation errors when the parameters λ_1 , λ_2 and λ_3 are not less than 2.8, 10 and 0.1 respectively.

Fig. 7 Some Frames from 3-motion Sequence in **Hopkins12 Dataset** [46]

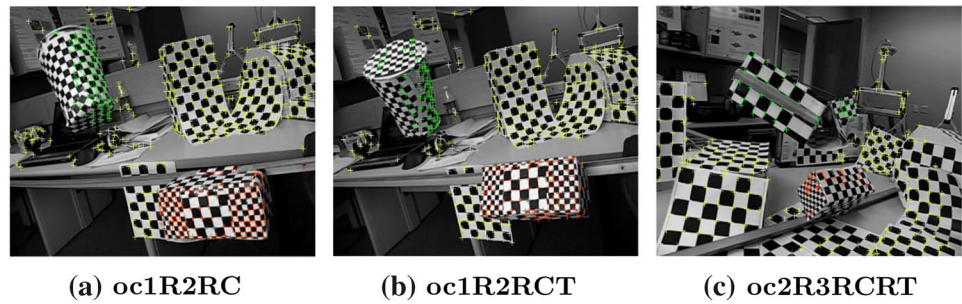
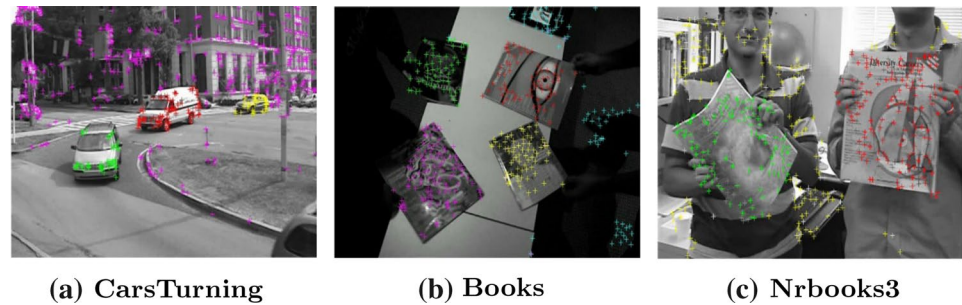


Table 5 Segmentation errors (in %) on two-frame **Hopkins12 Dataset** with missing trajectories

Methods	LSR	SSC	LRR	RSIM	SSC-OMP	LRKSC	BDR-B	BDR-Z	RBDSC-MN
2 motions									
Mean	6.06	0.73	8.16	0.55	24.31	4.56	11.28	11.76	0.22
Median	0.47	0.33	0.79	0.53	26.85	0.23	0.23	0.23	0.23
3 motions									
Mean	1.23	5.03	6.48	1.14	39.82	11.18	5.03	5.03	0.77
Median	1.07	1.09	1.27	0.91	45.18	0.91	0.36	0.36	0.86
ALL									
Mean	4.86	1.81	7.74	0.70	28.19	6.21	9.72	10.08	0.36
Median	0.99	0.60	0.98	0.70	28.26	0.48	0.30	0.30	0.26

Bold font represents the best results

Fig. 8 Example image frames from trajectories with outliers



5 Conclusion

In this work, a unified rank-constrained block diagonal subspace clustering method for motion segmentation is proposed, which can handle both rigid body and non-rigid body motion segmentation. In addition, two robust motion segmentation models (RBDSC-MN and RBDSC-RN) are put forward to deal with missing trajectories, corrupted trajectories and outlying trajectories. The experimental records based on some commonly used video sequences show that our methods can improve the accuracy of motion segmentation.

We have proved the superiority of our algorithms theoretically and experimentally, but our methods have certain limitations. First, adding kernel mapping constraints and KCIM to the block diagonal regularization representation

improves the performance of processing non-linear data (i.e. motion trajectory feature), but it brings some computational overhead. We plan to study how to speed up our methods, such as using GPUs. Second, our methods do not consider the problem of crowd movement segmentation. In crowd movement, there are many people who move independently and each person’s various small actions are included in the

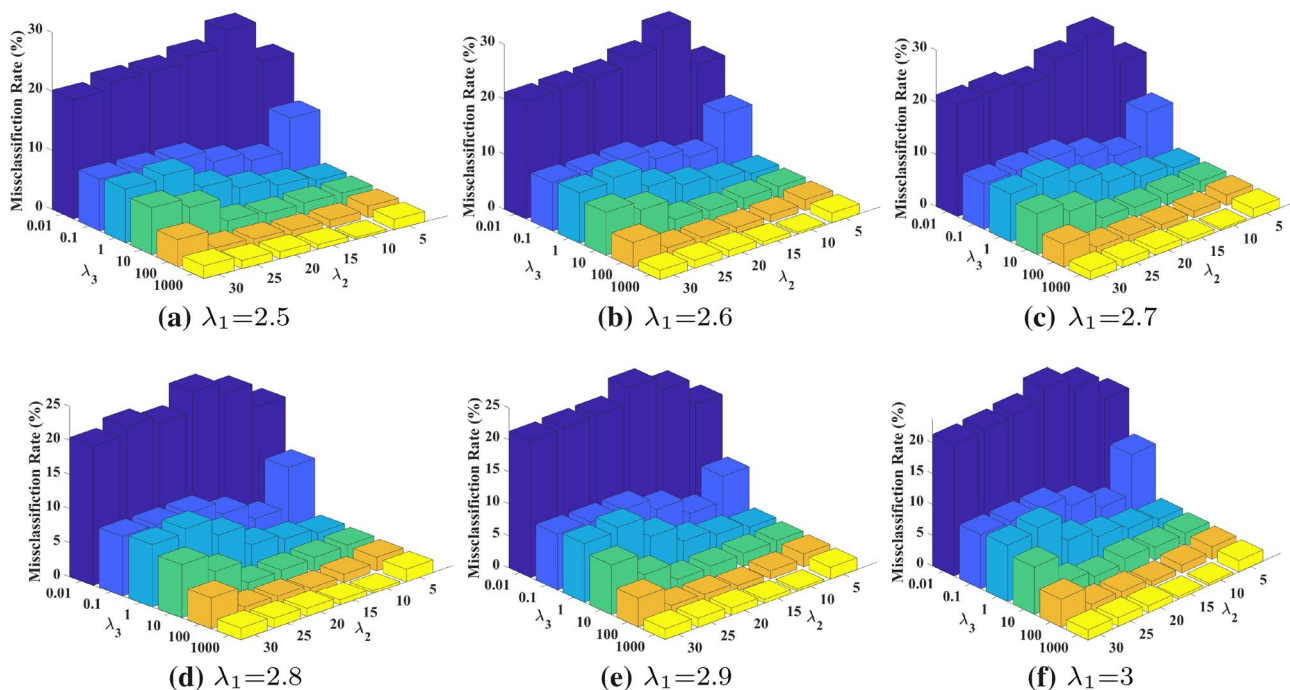
Table 6 Detailed information about the three datasets, including **Cars Turning, Books and Nrbooks3**

Dataset	Number of motions	Number of samples for each group	Number of outliers
Cars turning	4	51, 114, 52, 517	43
Books	5	45, 41, 28, 71, 30	127
Nrbooks3	3	129, 168, 91	35

Table 7 Segmentation errors (in %) on datasets with outlying trajectories

Data	LSR	SSC	LRR	RSIM	SSC-OMP	LRKSC	BDR-B	BDR-Z	RBDSC-RN
Cars turning	19.27	33.94	30.87	25.41	30.68	42.09	23.78	37.68	18.31
Books	54.83	56.65	70.86	72.95	70.49	54.46	49.00	47.83	47.72
Nrbooks3	39.58	40.11	61.49	65.47	52.99	37.72	63.48	38.11	35.19
Mean	37.89	43.57	54.41	54.61	51.39	44.76	45.42	41.21	33.74

Bold font represents the best results

**Fig. 9** Segmentation errors rates on **Hopkins12 Dataset** versus the parameters λ_1 , λ_2 and λ_3

crowd movement. In the future, we will conduct related research to make our methods suitable for crowd motion segmentation.

Acknowledgements This research work was supported by the Sichuan Science and Technology Program (Grant nos. 2020YJ0432, 2020YFS0360, 2019YJ0309), the National Natural Science Foundation of China (Grant no. 62102331), Innovative Research Group Project of the National Natural Science Foundation of China. We want to thank Canyi Lu for sharing the code for LSR, BDR-B and BDR-Z. And we thank J. Pan, Chong You and Wang L, who provide the code for LRKSC, SSC-OMP and LRTSC respectively. We also thank anonymous commenters for their substantive suggestions to improve our work.

References

- Huang S, Xu Z, Lv J (2018) Adaptive local structure learning for document co-clustering. *Knowl Based Syst* 148:74–84
- Yan W, Zhang B, Ma S, Yang Z (2017) A novel regularized concept factorization for document clustering. *Knowl Based Syst* 135:147–158
- Xue X, Zhang X, Feng X, Sun H, Chen W, Liu Z (2020) Robust subspace clustering based on non-convex low-rank approximation and adaptive kernel. *Inf Sci* 513:190–205
- Borgi MA, Nguyen TP, Labate D, Amar CB (2018) Statistical binary patterns and post-competitive representation for pattern recognition. *Int J Mach Learn Cybern* 9(6):1023–1038
- Chen L, Guo G (2019) Ordered smooth representation clustering. *Int J Mach Learn Cybern* 10(11):3301–3311
- Chen W, Zhang E, Zhang Z (2016) A Laplacian structured representation model in subspace clustering for enhanced motion segmentation. *Neurocomputing* 208(Oct 5):174–182
- Cheng B, Liu G, Wang J, Huang Z, Yan S (2011) Multi-task low-rank affinity pursuit for image segmentation. In: 2011 International conference on computer vision, IEEE, pp 2439–2446
- Costeira JP, Kanade T (1998) A multi-body factorization method for independently moving objects. *Int J Comput Vis* 29(3):159–179
- Dattorro J (2010) *Convex optimization & Euclidean distance geometry*. Lulu.com

10. Elhamifar E, Vidal R (2012) Sparse subspace clustering: algorithm, theory, and applications. *IEEE Trans Pattern Anal Mach Intell* 35(11):2765–2781
11. Gear CW (1998) Multibody grouping from motion images. *Int J Comput Vis* 29(2):133–150
12. Gruber A, Weiss Y (2004) Multibody factorization with uncertainty and missing data using the em algorithm. In: *Proceedings of the 2004 IEEE computer society conference on computer vision and pattern recognition*, 2004. CVPR 2004
13. He R, Tan T, Wang L, Zheng WS (2012) $l_{2,1}$ regularized correntropy for robust feature selection. In: *2012 IEEE conference on computer vision and pattern recognition*
14. He R, Tan T, Wang L, Zheng WS (2012) $l_{2,1}$ regularized correntropy for robust feature selection. In: *2012 IEEE conference on computer vision and pattern recognition*, IEEE, pp 2504–2511
15. Huang P, Hilton A, Starck J (2009) Human motion synthesis from 3d video. In: *2009 IEEE conference on computer vision and pattern recognition*, IEEE, pp 1478–1485
16. Huang T, Wang S, Zhu W (2020) An adaptive kernelized rank-order distance for clustering non-spherical data with high noise. *Int J Mach Learn Cybern* 11(8):1735–1747
17. Ichimura N (1999) Motion segmentation based on factorization method and discriminant criterion. *Proceedings of the seventh IEEE international conference on computer vision*, vol 1. IEEE, pp 600–605.
18. Ji P, Li H, Salzmann M, Zhong Y (2016) Robust multi-body feature tracker: a segmentation-free approach. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 3843–3851
19. Ji P, Salzmann M, Li H (2015) Shape interaction matrix revisited and robustified: efficient subspace clustering with corrupted and incomplete data. In: *Proceedings of the IEEE international conference on computer vision*, pp 4687–4695
20. Ji P, Reid I, Garg R, Li H, Salzmann M (2017) Low-rank kernel subspace clustering. *arXiv preprint arXiv:1707.04974* 1
21. Jiang Z, Lin Z, Davis L (2012) Recognizing human actions by learning and matching shape-motion prototype trees. *IEEE Trans Pattern Anal Mach Intell* 34(3):533–547
22. Zheng J, Jiang Z, Chellappa R (2016) Cross-view action recognition via transferable dictionary learning. *IEEE Trans Image Proc* 25(6):2542–2556
23. Kanatani K, Sugaya Y (2003) Multi-stage optimization for multibody motion segmentation. In: *Australia–Japan advanced workshop on computer vision*, vol 2. Citeseer, pp 7
24. Kanatani Ki (2001) Motion segmentation by subspace separation and model selection. In: *Proceedings eighth IEEE international conference on computer vision*, vol 2. ICCV 2001, IEEE, pp 586–591
25. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY (2002) An efficient k-means clustering algorithm: analysis and implementation. *IEEE Trans Pattern Anal Mach Intell* 24(7):881–892
26. Keuper M, Tang S, Andres B, Brox T, Schiele B (2019) Motion segmentation and multiple object tracking by correlation co-clustering. *IEEE Trans Pattern Anal Mach Intell* 42:140–153
27. Li S, Li K, Fu Y (2015) Temporal subspace clustering for human motion segmentation. In: *Proceedings of the IEEE international conference on computer vision*, pp 4453–4461
28. Li Z, Guo J, Cheong LF, Zhou SZ (2013) Perspective motion segmentation via collaborative clustering. In: *Proceedings of the IEEE international conference on computer vision*, pp 1369–1376
29. Lin G, Zhu H, Kang X, Fan C, Zhang E (2014) Feature structure fusion and its application. *Inf Fusion* 20:146–154
30. Lin G, Zhu H, Kang X, Miu Y, Zhang E (2015) Feature structure fusion modelling for classification. *Image Process Lett* 9(10):883–888
31. Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2013) Robust recovery of subspace structures by low-rank representation. *IEEE Trans Pattern Anal Mach Intell* 35(1):171–184
32. Lu C, Tang J, Lin M, Lin L, Yan S, Lin Z (2013) Correntropy induced l_2 graph for robust subspace clustering. In: *Proceedings of the IEEE international conference on computer vision*, pp 1801–1808
33. Lu C, Feng J, Lin Z, Mei T, Yan S (2018) Subspace clustering by block diagonal representation. *IEEE Trans Pattern Anal Mach Intell* 41(2):487–501
34. Lu CY, Min H, Zhao ZQ, Zhu L, Huang DS, Yan S (2012) Robust and efficient subspace segmentation via least squares regression. In: *European conference on computer vision*, Springer, pp 347–360
35. Lu J, Wang G, Moulin P (2014a) Human identity and gender recognition from gait sequences with arbitrary walking directions. *IEEE Trans Inf Forensics Secur* 9(1):51–61
36. Lu J, Zhou X, Tan YP, Shang Y (2014b) Neighborhood repulsed metric learning for kinship verification. *IEEE Trans Pattern Anal Mach Intell* 36(2):331–345
37. Ng AY, Jordan MI, Weiss Y (2002) On spectral clustering: analysis and an algorithm. In: *Advances in neural information processing systems*, pp 849–856
38. Patel VM, Vidal R (2014) Kernel sparse subspace clustering. In: *2014 IEEE international conference on image processing (icip)*, IEEE, pp 2849–2853
39. Ran He, Yingya Zhang, Zhenan Sun, Qiyue Yin (2015) Robust subspace clustering with complex noise. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 24(11):4001–13
40. Rao S, Tron R, Vidal R, Ma Y (2010) Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *IEEE Trans Pattern Anal Mach Intell* 32(10):1832–1845
41. Ryoo MS, Aggarwal JK (2009) Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. In: *2009 IEEE 12th international conference on computer vision*, IEEE, pp 1593–1600
42. Schölkopf B, Smola A, Müller K (2008) Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput* 10(5):1299–1319
43. Peng S, Ser W, Chen B, Sun L, Lin Z (2018) Correntropy based graph regularized concept factorization for clustering. *Neurocomput* 316:34–48
44. Tron R, Vidal R (2007) A benchmark for the comparison of 3-d motion segmentation algorithms. In: *2007 IEEE conference on computer vision and pattern recognition*, IEEE, pp 1–8
45. Vidal R, Favaro P (2014) Low rank subspace clustering (LRSC). *Pattern Recognit Lett* 43:47–61
46. Vidal R, Tron R, Hartley R (2008) Multiframe motion segmentation with missing data using powerfactorization and gpca. pp 85–105
47. Vidal R, Ma Y, Sastry S (2012) Generalized principal component analysis (GPCA). *IEEE Trans Pattern Anal Mach Intell* 27(12):1945–1959
48. Wang L, Ding Z, Fu Y (2018) Learning transferable subspace for human motion segmentation. In: *Proceedings of the AAAI conference on artificial intelligence*, vol 32
49. Wang L, Ding Z, Fu Y (2019) Low-rank transfer human motion segmentation. *IEEE Trans Image Process* 28(2):1023–1034
50. Wang L, Ding S, Wang Y, Ding L (2021) A robust spectral clustering algorithm based on gridpartition and decision-graph. *Int J Mach Learn Cybern* 12(5):1243–1254
51. Wang Q, Chen G (2017) Fuzzy soft subspace clustering method for gene co-expression network analysis. *Int J Mach Learn Cybern* 8(4):1157–1165

52. Weifeng L, Pokharel PP, Principe JC (2007) Correntropy: properties and applications in non-gaussian signal processing. *IEEE Trans Signal Process* 55(11):5286–5298
53. Yan J, Pollefeys M (2006) A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In: *European conference on computer vision*, Springer, pp 94–106
54. Yan Y, Ricci E, Liu G, Sebe N (2015) Egocentric daily activity recognition via multitask clustering. *IEEE Trans Image Process* 24(10):2984–2995
55. Yan Y, Yang Y, Deyu M, Liu G, Tong W, Hauptman A, Sebe N (2015) Event oriented dictionary learning for complex event detection. *IEEE Trans Image Process A Publ IEEE Signal Process Soc* 24(6):1867
56. Yao J, Cao X, Zhao Q, Meng D, Xu Z (2018) Robust subspace clustering via penalized mixture of gaussians. *Neurocomput* 278:4–11
57. Yin M, Guo Y, Gao J, He Z, Xie S (2016) Kernel sparse subspace clustering on symmetric positive definite manifolds. In: *proceedings of the IEEE conference on computer vision and pattern recognition*, pp 5157–5164
58. Yinfeng M, Jiye L, Fuyuan C, Yijun H (2018) A new distance with derivative information for functional k-means clustering algorithm. *Inf Sci* 463:166–185
59. You C, Robinson D, Vidal R (2016) Scalable sparse subspace clustering by orthogonal matching pursuit. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 3918–3927
60. Zhang X, Gao H, Li G, Zhao J, Huo J, Yin J, Liu Y, Zheng L (2018) Multi-view clustering based on graph-regularized non-negative matrix factorization for object recognition. *Inform Sci* 432:463–478
61. Zhu H, Member IEEE, Vial R, Lu S (2018) Youtube: searching action proposal via recurrent and static regression networks. *IEEE Trans Image Process Publ IEEE Signal Process Soc* 27(6):2609

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.