What to Select: Pursuing Consistent Motion Segmentation from Multiple Geometric Models

Yangbangyan Jiang [1,2], Qianqian Xu [3,*], Ke Ma [4], Zhiyong Yang [1,2], Xiaochun Cao [1,2,6], Qingming Huang [3,4,5,6,*]

[1] State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, China
[4] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, China
[5] Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing, China
[6] Peng Cheng Laboratory, Shenzhen, China
{jiangyangbangyan, yangzhiyong, caoxiaochun}@iie.ac.cn, xuqianqian@ict.ac.cn, {make, qmhuang}@ucas.ac.cn

Abstract

Motion segmentation aims at separating motions of different moving objects in a video sequence. Facing the complicated real-world scenes, recent studies reveal that combining multiple geometric models would be a more effective way than just employing a single one. This motivates a new wave of model-fusion based motion segmentation methods. However, the vast majority models of this kind merely seek consensus in spectral embeddings. We argue that a simple consensus might be insufficient to filter out the harmful information which is either unreliable or semantically unrelated to the segmentation task. Therefore, how to automatically select valuable patterns across multiple models should be regarded as a key challenge here. In this paper, we present a novel geometric-model-fusion framework for motion segmentation, which targets at constructing a consistent affinity matrix across all the geometric models. Specifically, it incorporates the structural information shared by affinity matrices to select those semantically consistent entries. Meanwhile, a multiplicative decomposition scheme is adopted to ensure the structural consistency among multiple affinities. To solve this problem, an alternative optimization scheme is proposed, together with a proof of its global convergence. Experiments on four real-world benchmarks show the superiority of the proposed method.

Introduction

Motion segmentation has served as a crucial upstream task in a broad spectrum of computer vision applications, such as visual SLAM (Huang et al. 2019), video object detection (Kamranian et al. 2020), video object tracking and segmentation (Zhuo et al. 2020), and visual surveillance (Sengar and Mukhopadhyay 2020). The goal of motion segmentation is to group multiple moving objects into different clusters. The objects herein are usually represented by a set of trajectories of feature points tracked in a video sequence. This problem's challenges mainly lie in the complexity of real-world scenes, which might be caused by the perspective effect, scattered feature points of moving objects, small objects.

According to how many frames involved in the segmentation at each time, studies on motion segmentation could be divided into two camps: **(a)** two-frame-based methods (Zhuo et al. 2020; Muthu et al. 2020; Ranjan et al. 2019) and **(b)** multi-frame-based methods. Compared with two-framed-based methods, the multi-frame-based methods utilize all frames in a video clip to capture the motion information, which often owns higher performance. Existing multi-frame based frameworks mainly fall into three directions. The first direction is the subspace-based methods. Methods of this kind aim at exploring the subspace structure of the trajectories, which is assumed to lie in the union of several subspaces under the affine geometric model. Typical examples include algebraic (Vidal, Tron, and Hartley 2008), information-theoretic (Rao et al. 2010) and spectral clustering-based models (Elhamifar and Vidal 2013; Liu et al. 2013; Lu et al. 2019). The second direction is targeted at exploring multi-model fitting methods that estimate the model parameters using multiple model hypotheses in the presence of data corruption and outliers. In the existing literatures, such an idea is often implemented by consensus learning (Magri and Fusiello 2016; Kluger et al. 2020) , preference fusion (Magri and Fusiello 2014; Tepper and Sapiro 2017; Magri and Fusiello 2019), hyper-graph learning (Lin et al. 2019) and energy minimization (Barath and Matas 2018; Baráth and Matas 2019). Last but not least, the third direction studies fusion-based methods that aggregate multiple geometric models (Lai et al. 2017; Xu, Cheong, and Li 2018, 2021; Jung, Ju, and Kim 2019) into a single result.

Among these approaches, (Xu, Cheong, and Li 2018) achieves a promising performance by aggregating the well-known affine, homography, and fundamental geometric models to overcome the disadvantage of each and seek out a consistent result. The affine model could hardly deal with the perspective effect in real-world scenes. For homography models, the obtained affinities between different planes of the same rigid motion are weak; thus, it is insufficient to group dispersed objects. The fundamental model could discover much richer information between trajectories, while false positive affinity might also be detected, resulting in overlapped subspace structures for the affinity matrix. To

equip the motion segmentation model with the ability to tackle various real-world scenes, it is necessary to integrate these basic models to produce a consistent segmentation result.

However, the aggregation method proposed in (Xu, Cheong, and Li 2018) is still insufficient to leverage a consistent result, which finds spectral embeddings with pairwise consensus under subset constraints, while ignoring the affinity-level consensus. As we mentioned above, the three basic models suffer from different affinity issues. Merely learning model-specific spectral embeddings with a simple fusion operation might not eliminate the influence of those incorrect affinities, *e.g.*, large fundamental affinities for unrelated trajectories, or nonzero affinities caused by outliers. Therefore, how to automatically select valuable patterns across the basic models becomes a key challenge here. Moreover, inspired by the fact that ideal affinity matrices must be block-diagonal (Lu et al. 2019), we turn to learn a consensus affinity matrix for the basic ones by exploring their structural consistency.

In this paper, we propose a new geometric-model-fusion based framework for motion segmentation, which pursues the affinity-level segmentation consistency across all the geometric models via constructing a consensus affinity matrix. The proposed framework resorts to exploit the structural information with block-diagonality shared among the basic affinity matrices to select those valuable patterns lying in affinities, *i.e.*, semantically consistent entries. More specifically, a multiplicative decomposition is utilized with a structural regularizer enforced on the shared shape mask, which ensures the structural consistency of multiple affinity matrices. We then propose an alternative optimization scheme to solve this problem and prove that it enjoys global convergence.

In summary, the contributions of this paper are three-fold:

- A novel model-fusion based motion segmentation framework is proposed to pursue the affinity-level segmentation consistency all the geometric models. In the core of this framework lies in the consensus affinity construction that captures the shared structural information with block-diagonal pursuit.
- To solve the problem, we present an algorithm to alternatively update the variables, together with a theoretical analysis for its promising global convergence property.
- Extensive experiments are conducted on four real-world benchmark datasets. The quantitative and qualitative results both validate the superiority of the proposed method.

Methodology

In this section, we first introduce how to generate affinity matrix from basic geometric models. Then the proposed method to integrate these affinities to seek consistent motion segmentation is detailed.

Affinity Construction from Geometric Models

Given the trajectories of N tracked points within F consecutive frames $\mathcal{X} = \{\boldsymbol{x}_1^f, \cdots, \boldsymbol{x}_N^f\}_{f=1}^F$, we first use them to fit plenty of hypotheses of V types of geometric models. For a geometric model, M minimal subsets of data points in each two consecutive frames are randomly sampled to estimate the model, then generate M model hypotheses. Therefore, there are $M \times F$ model hypotheses sampled for each geometric model.

With these model hypotheses, we then calculate the residual between each $\boldsymbol{x}_i^f$ and each model hypothesis by their Sampson distance (Hartley and Zisserman 2003). This results in the following residual vector for i-th point at f-th consecutive pair of frames under v-th geometric model:

$$\boldsymbol{r}_i^{f(v)} = \left[r_{i,1}^{f(v)},\ r_{i,2}^{f(v)},\ \cdots,\ r_{i,M}^{f(v)}\right]$$

After that, we adopt the Ordered Residual Kernel (ORK) (Chin, Wang, and Suter 2009) to capture the affinities between two data points and divide them by corresponding co-occurrence times throughout the frames for normalization. Then the affinity matrix is sparsified within a ϵ-neighborhood way as a customary step (Lai et al. 2017).

Consistent Segmentation with Structural Pursuit

Denote the affinity matrix set obtained by multiple geometric models as $\mathcal{A} = \{\boldsymbol{A}^{(v)}\}_{v=1}^V$, where $\boldsymbol{A}^{(v)} \in \mathbb{R}^{N\times N}$. Now we need to find the partition for k motions. Unlike (Xu, Cheong, and Li 2018) which tries to seek common spectral embeddings for all the views, we turn to construct a consensus affinity matrix by discovering reliable information lying in the shared structure of these affinity matrices, then perform the spectral clustering on the consensus matrix to obtain the final partition.

Ideally, the affinity between points belonging to different motions should be 0. Therefore, each $\boldsymbol{A}^{(v)}$ has at least k connected components, *i.e.*, each $\boldsymbol{A}^{(v)}$ is k-block diagonal. When the data points are ordered according to their motion membership, $\boldsymbol{A}^{(v)}$ could be represented as:

$$\boldsymbol{A}^{(v)} = \begin{bmatrix} \boldsymbol{A}_1^{(v)} & 0 & \cdots & 0 \\ 0 & \boldsymbol{A}_2^{(v)} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{A}_k^{(v)} \end{bmatrix},$$

where $\boldsymbol{A}_k^{(v)} \in \mathbb{R}^{n_i\times n_i}$ is not an identity matrix, n_i denotes the number of points in i-th motion, and $\sum_i n_i = N$. Note that here the block diagonality only requires the number of connected components, rather than an explicit block diagonal structure, thus it is permutation invariant – if $\boldsymbol{A}^{(v)}$ is k-block diagonal, then for any permutation matrix $\boldsymbol{P}$, $\boldsymbol{P}\boldsymbol{A}^{(v)}\boldsymbol{P}^\top$ is also k-block diagonal (Lu et al. 2019).

Apparently, $\boldsymbol{A}^{(v)}$s share the same block-diagonal structure since n_is keep unchanged in all the views. Such a structure reveals the true membership of data points, which is exactly what we pursue in this task. Borrowing the wisdom of (Yang et al. 2019), we decompose the model-specific $\boldsymbol{A}^{(v)}$ using a multiplication of a shared block-diagonal mask $\boldsymbol{S}$ and the magnitude $\boldsymbol{G}^{(v)}$, *i.e.*, $\boldsymbol{A}^{(v)} = \boldsymbol{S} \odot \boldsymbol{G}^{(v)}$. In this scheme, $\boldsymbol{S}$ is encouraged to concentrate on capturing the

consistent structure shared by all the $\boldsymbol{A}^{(v)}$s, while $\boldsymbol{G}^{(v)}$ are expected to maintain the magnitude for elements lying in this structure. Accordingly, a consensus among all the views could be reached with agreements as much as possible.

To meet our expectation, we constrain the magnitude of each element in $\boldsymbol{S}$ with a predefined upper bound S_{max} to prevent it from dominating the multiplication and force it to focus on discover the structure. Meanwhile, a nonnegative lower bound G_{min} is applied on $\boldsymbol{G}^{(v)}$ to avoid zero entries in $\boldsymbol{G}^{(v)}$, since the zero entry will cause a zero element in corresponding position for $\boldsymbol{S} \odot \boldsymbol{G}^{(v)}$ thus might break the structure. Moreover, note that the values in diagonal entries of $\boldsymbol{A}^{(v)}$s have no effect on the following spectral clustering, thus we manually set these diagonal elements to be zero before the segmentation. This also leads to a constraint for $\boldsymbol{S}$ and $\boldsymbol{G}^{(v)}$ that their diagonal entries should be zero. To sum up, we have the constraint set for $\boldsymbol{S}$ and $\boldsymbol{G}^{(v)}$s respectively,

$$\begin{aligned}\mathcal{S} = \{\boldsymbol{S} : \boldsymbol{S} \in \mathbb{R}^{N\times N}, \boldsymbol{S} = \boldsymbol{S}^\top, \\ 0 \le \boldsymbol{S}_{ij} \le S_{max}, diag(\boldsymbol{S}) = 0\}, \\ \mathcal{G} = \{\boldsymbol{G} : \boldsymbol{G} \in \mathbb{R}^{N\times N}, \boldsymbol{G}_{ij} \ge G_{min}, diag(\boldsymbol{G}) = 0\}.\end{aligned} \tag{1}$$

With the decomposition modeling, now we need to realize the block-diagonality for $\boldsymbol{S}$. From spectral graph theory, we know that the number of connected components in $\boldsymbol{S}$ equals to the multiplicity k of the eigenvalue 0 of the corresponding Laplacian matrix $\boldsymbol{L}_{\boldsymbol{S}} = diag(\boldsymbol{S}\mathbf{1}) - \boldsymbol{S}$ (Von Luxburg 2007). Therefore, a straight-forward choice to obtain a k-block-diagonal $\boldsymbol{S}$ is to apply the hard rank constraint $rank(\boldsymbol{L}_{\boldsymbol{S}}) = n - k$. However, directly solving this constrained problem is NP-hard. Fortunately, (Lu et al. 2019) proposes the following soft block diagonal regularizer based on this fact to reach this goal.

Definition 1 (k-block diagonal regularizer, (Lu et al. 2019)) *Let the eigenvalues λ_i of a $N \times N$ matrix $\boldsymbol{A}$ be arranged in a non-increasing order $\lambda_1(\boldsymbol{A}) \ge \lambda_2(\boldsymbol{A}) \ge \ldots \ge \lambda_N(\boldsymbol{A})$. For any $N \times N$ affinity matrix $\boldsymbol{S}$ that $\boldsymbol{S} \ge 0$ and $\boldsymbol{S} = \boldsymbol{S}^\top$, the k-block diagonal regularizer is defined as the sum of the k smallest eigenvalues of its Laplacian $\boldsymbol{L}_{\boldsymbol{S}}$:*

$$\|\boldsymbol{S}\|_{\boxed{k}} = \sum_{i=N-k+1}^{N} \lambda_i(\boldsymbol{L}_{\boldsymbol{S}}).$$

By adopting the Frobenius norm to regularize the magnitude $\boldsymbol{G}^{(v)}$ and implementing the above structural regularizer, our objective function could be formulated as:

$$\begin{aligned}\min_{\boldsymbol{S},\boldsymbol{G}^{(v)},\boldsymbol{U}} \frac{1}{2}\sum_v \|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 \\ + \frac{\alpha_1}{2}\sum_v \|\boldsymbol{G}^{(v)}\|_F^2 + \alpha_2\|\boldsymbol{S}\|_{\boxed{k}} \\ s.t.\ \boldsymbol{S} \in \mathcal{S},\ \boldsymbol{G}^{(v)} \in \mathcal{G}.\end{aligned} \tag{2}$$

According to Ky Fan's Theorem (Fan 1949), we could reformulate the k-block diagonal regularizer in a convex form:

$$\|\boldsymbol{S}\|_{\boxed{k}} = \min_{\boldsymbol{U}} \langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle,\ \ s.t.\ \boldsymbol{U} \in \mathcal{U} \tag{3}$$

where

$$\mathcal{U} = \{\boldsymbol{U} :\ \boldsymbol{U} \in \mathbb{R}^{N\times N}, \mathbf{0} \preceq \boldsymbol{U} \preceq \boldsymbol{I}, tr(\boldsymbol{U}) = k\}. \tag{4}$$

Using Eq. (3), we reach our final optimization problem:

$$\begin{aligned}\min_{\boldsymbol{S},\boldsymbol{G}^{(v)},\boldsymbol{U}} \frac{1}{2}\sum_v \|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 \\ + \frac{\alpha_1}{2}\sum_v \|\boldsymbol{G}^{(v)}\|_F^2 + \alpha_2\langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle \\ s.t.\ \boldsymbol{S} \in \mathcal{S},\ \boldsymbol{G}^{(v)} \in \mathcal{G},\ \boldsymbol{U} \in \mathcal{U}\end{aligned} \tag{5}$$

Optimization

Following the idea of (Yang et al. 2020), we present an optimization method to solve the following surrogate problem rather than directly solving Eq. (5):

$$\begin{aligned}\min_{\boldsymbol{S},\boldsymbol{G}^{(v)},\boldsymbol{U}} \frac{1}{2}\sum_v \|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 + \frac{\alpha_1}{2}\sum_v \|\boldsymbol{G}^{(v)}\|_F^2 \\ + \alpha_2\langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle + \frac{\alpha_3}{2}\|\boldsymbol{U}\|_F^2 \\ s.t.\ \boldsymbol{S} \in \mathcal{S},\ \boldsymbol{G}^{(v)} \in \mathcal{G},\ \boldsymbol{U} \in \mathcal{U}\end{aligned} \tag{6}$$

The introduction of the term $\frac{\alpha_3}{2}\|\boldsymbol{U}\|_F^2$ induces a promising property that the proposed algorithm could produce a parameter sequence simultaneously converging to a critical point of both Eq. (5) and Eq. (6) under certain conditions, which will be proved latter.

We propose an alternative optimization scheme to solve this non-convex and non-smooth problem (Eq. (6)). For variables $\boldsymbol{S}, \boldsymbol{U}$ and $\boldsymbol{G}^{(v)}$, we iteratively update each of them while fixing others.

Fix $\boldsymbol{S}, \boldsymbol{U}$, update $\boldsymbol{G}^{(v)}$

The subproblem for each $\boldsymbol{G}^{(v)}$ is:

$$\begin{aligned}\min_{\boldsymbol{G}^{(v)}} \frac{1}{2}\|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 + \frac{\alpha_1}{2}\|\boldsymbol{G}^{(v)}\|_F^2 \\ s.t.\ \boldsymbol{G}^{(v)} \in \mathcal{G}\end{aligned} \tag{7}$$

The solution could be easily obtained by setting the derivative to be 0. Thus we have:

$$\begin{aligned}\boldsymbol{G}^{(v)*} = \tilde{\boldsymbol{G}}^{(v)} - diag(diag(\tilde{\boldsymbol{G}}^{(v)})), \\ \tilde{\boldsymbol{G}}^{(v)} = \max\big((\boldsymbol{S} \odot \boldsymbol{A}^{(v)}) \oslash (\boldsymbol{S} \odot \boldsymbol{S} + \alpha_1), G_{min}\big).\end{aligned} \tag{8}$$

Here $\max(\cdot)$ is element-wise maximum, and $\oslash$ denotes the element-wise division, *i.e.*, $\boldsymbol{S} = \boldsymbol{A} \oslash \boldsymbol{B}$ means $S_{ij} = A_{ij}/B_{ij}$.

Fix $\boldsymbol{G}^{(v)}, \boldsymbol{U}$, update $\boldsymbol{S}$

The subproblem for $\boldsymbol{S}$ is:

$$\begin{aligned}\min_{\boldsymbol{S}} \frac{1}{2}\sum_v \|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 + \alpha_2\langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle \\ s.t.\ \boldsymbol{S} \in \mathcal{S}\end{aligned} \tag{9}$$

Obviously, when $i = j$, $\boldsymbol{S}_{ij} = 0$ is the only feasible solution. For $i \ne j$, since $\langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle = \langle \boldsymbol{S}, diag(\boldsymbol{U})\mathbf{1}^\top - \boldsymbol{U}\rangle$, denote

$\bar{\boldsymbol{U}} = diag(\boldsymbol{U})\mathbf{1}^\top - \boldsymbol{U}$, the subproblem for each pair of $\boldsymbol{S}_{ij}$ and $\boldsymbol{S}_{ji}$ could be reformulated as

$$\begin{aligned}\min_{\boldsymbol{S}_{ij}=\boldsymbol{S}_{ji}} \frac{1}{2}\sum_v \big(\boldsymbol{A}_{ij}^{(v)} - \boldsymbol{S}_{ij}\boldsymbol{G}_{ij}^{(v)}\big)^2 + \big(\boldsymbol{A}_{ji}^{(v)} - \boldsymbol{S}_{ji}\boldsymbol{G}_{ji}^{(v)}\big)^2 \\ + \alpha_2\big(\boldsymbol{S}_{ij}\bar{\boldsymbol{U}}_{ij} + \boldsymbol{S}_{ji}\bar{\boldsymbol{U}}_{ji}\big) \\ s.t.\ 0 \le \boldsymbol{S}_{ij} \le S_{max},\ 0 \le \boldsymbol{S}_{ji} \le S_{max}\end{aligned} \tag{10}$$

This problem also enjoys a closed-form solution:

$$\boldsymbol{S}_{ij}^* = \begin{cases}\min\big(S_{max}, \max(\tilde{\boldsymbol{S}}_{ij}, 0)\big), & i \neq j \\ 0, & i = j\end{cases}, \tag{11}$$

where

$$\tilde{\boldsymbol{S}}_{ij} = \frac{\sum_v \big(\boldsymbol{G}_{ij}^{(v)}\boldsymbol{A}_{ij}^{(v)} + \boldsymbol{G}_{ji}^{(v)}\boldsymbol{A}_{ji}^{(v)}\big) - \alpha_2\big(\bar{\boldsymbol{U}}_{ij} + \bar{\boldsymbol{U}}_{ji}\big)}{\sum_v \boldsymbol{G}_{ij}^{(v)2}}. \tag{12}$$

Fix $\boldsymbol{S}, \boldsymbol{G}^{(v)}$, update $\boldsymbol{U}$

The subproblem for $\boldsymbol{U}$ could be written as:

$$\min_{\boldsymbol{U}} \langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle + \frac{\alpha_3}{2\alpha_2}\|\boldsymbol{U}\|_F^2,\ \ s.t.\ \boldsymbol{U} \in \mathcal{U}. \tag{13}$$

The problem without the term $\frac{\alpha_3}{2\alpha_2}\|\boldsymbol{U}\|_F^2$ is well-studied with a solution of $\boldsymbol{U} = \boldsymbol{V}_{1:k}\boldsymbol{V}_{1:k}^\top$, where $\boldsymbol{V}$ is the eigenvector matrix with the i-th column $\boldsymbol{v}_i$ corresponding to $\lambda_i(\boldsymbol{L}_{\boldsymbol{S}})$. On the other hand, the solution when $\alpha_3 \ge 0$ could be given by Theorem 3 in (Yang et al. 2020), which is as follows.

Theorem 1 (Optimal solution of Eq. (13)) *Let $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_N]$ be the eigenvector matrix associated for $\lambda_1(\boldsymbol{L}_{\boldsymbol{S}}), \cdots, \lambda_N(\boldsymbol{L}_{\boldsymbol{S}})$. Let $\lambda_0(\boldsymbol{L}_{\boldsymbol{S}}) = -\infty, \lambda_{N+1}(\boldsymbol{L}_{\boldsymbol{S}}) = +\infty$. Furthermore, set*

$$\begin{aligned} l &= \max\{i :\ \lambda_i(\boldsymbol{L}_{\boldsymbol{S}}) < \lambda_{i+1}(\boldsymbol{L}_{\boldsymbol{S}}), 0 \le i < k\} \\ h &= \min\{i :\ \lambda_i(\boldsymbol{L}_{\boldsymbol{S}}) < \lambda_{i+1}(\boldsymbol{L}_{\boldsymbol{S}}), i \ge k\} \\ \Delta l &= \lambda_{l+1}(\boldsymbol{L}_{\boldsymbol{S}}) - \lambda_l(\boldsymbol{L}_{\boldsymbol{S}}) \\ \Delta h &= \lambda_{h+1}(\boldsymbol{L}_{\boldsymbol{S}}) - \lambda_h(\boldsymbol{L}_{\boldsymbol{S}}) \\ \breve{\delta}(\boldsymbol{L}_{\boldsymbol{S}}) &= \begin{cases}\min\{\Delta l, \Delta h\}, & l \neq 0, h \neq N, \\ \max\{\Delta l, \Delta h\}, & otherwise.\end{cases}\end{aligned}$$

Then for all $\boldsymbol{L}_{\boldsymbol{S}} \neq \mathbf{0}$ and $0 \le \frac{\alpha_3}{2\alpha_2} < \breve{\delta}(\boldsymbol{L}_{\boldsymbol{S}})$, the optimal solution of Eq. (13) is:

$$\boldsymbol{U}^\star = \boldsymbol{V}_{1:l}\boldsymbol{V}_{1:l}^\top + \tfrac{k-l}{h-l}\boldsymbol{V}_{l+1:h}\boldsymbol{V}_{l+1:h}^\top \tag{14}$$

Compared with the corresponding subproblem of the original problem (*i.e.*, Eq. (13) with $\alpha_3 = 0$), Eq. (13) with a positive α_3 is strongly convex, thus enables the global convergence property of the proposed optimization scheme. Moreover, according to Eq. (14), since the inclusion of $\boldsymbol{V}_{l+1:h}$ enables $\boldsymbol{U}^\star$ to have the whole subspace spanned by eigenvectors associated with $\lambda_k(\boldsymbol{L}_{\boldsymbol{S}})$ even if $\lambda_{k+1}(\boldsymbol{L}_{\boldsymbol{S}}) = \lambda_k(\boldsymbol{L}_{\boldsymbol{S}})$, it ensures that $\boldsymbol{U}^\star$ is well-defined and identifiable when the eigengap $\lambda_{k+1}(\boldsymbol{L}_{\boldsymbol{S}}) - \lambda_k(\boldsymbol{L}_{\boldsymbol{S}})$ vanishes.

Algorithm 1 Model-fusion-based Consistent Motion Segmentation

Input: Trajectories of N tracked points over F frames, hyper-parameter α_1, α_2, S_{max}, G_{min}
Output: Motion labels $\boldsymbol{y} = [y_1, \cdots, y_n]$

```
Construct affinity matrices A = {A^(v)}_{v=1}^V from V geometric models using the trajectories
Initialize G^(v) ← 0, S ← 1_N 1_N^⊤ − I, U ← I_N
while not converged do
    Update G^(v) with Eq. (8)
    Update S with Eq. (11)
    Update U with Eq. (14)
end while
Get consensus affinity matrix A with Eq. (15)
y ← SpectralClustering(A)
```

Clustering

After obtaining the optimal $\boldsymbol{S}$ and $\boldsymbol{G}^{(v)}$s, we could calculate the final consensus affinity matrix by:

$$\boldsymbol{A} = \sum_v^V \frac{\boldsymbol{S} \odot \boldsymbol{G}^{(v)} + (\boldsymbol{S} \odot \boldsymbol{G}^{(v)})^\top}{2} \tag{15}$$

Then spectral clustering is applied on $\boldsymbol{A}$ for final results.

We summarize our algorithm in Algorithm 1. For our model, the time complexities of updating $\boldsymbol{S}$ and $\{\boldsymbol{G}^{(v)}\}$s are both $O(VN^2)$. Updating $\boldsymbol{U}$ costs $O(N^3)$ operations to perform eigenvalue decomposition. Hence the overall computational complexity is $O(N^3 + 2VN^2)$ per iteration.

Convergence Analysis

First, we prove the global convergence property of our algorithm with respect to the surrogate problem Eq. (6).

Theorem 2 (Global Convergence of Algorithm 1 with respect to Eq. (6)) *Let $\{\boldsymbol{S}_t, \boldsymbol{G}_t^{(v)}, \boldsymbol{U}_t\}$ be the parameter sequence generated by Algorithm 1. Let $\mathcal{L}(\boldsymbol{S}, \boldsymbol{G}^{(v)}) = \frac{1}{2}\sum_v \|\boldsymbol{A}^{(v)} - \boldsymbol{S} \odot \boldsymbol{G}^{(v)}\|_F^2 + \frac{\alpha_1}{2}\sum_v \|\boldsymbol{G}^{(v)}\|_F^2 + \mathbb{1}_{\mathcal{S}}(\boldsymbol{S}) + \sum_v \mathbb{1}_{\mathcal{G}}(\boldsymbol{G}^{(v)})$, then the surrogate objective could be written as $\mathcal{F}(\boldsymbol{S}, \boldsymbol{G}^{(v)}, \boldsymbol{U}) = \mathcal{L}(\boldsymbol{S}, \boldsymbol{G}^{(v)}) + \alpha_2\langle \boldsymbol{L}_{\boldsymbol{S}}, \boldsymbol{U}\rangle + \frac{\alpha_3}{2}\|\boldsymbol{U}\|_F^2 + \mathbb{1}_{\mathcal{U}}(\boldsymbol{U})$, where $\mathbb{1}_{\mathcal{U}}(\cdot)$ is the indicator function for the set $\mathcal{U}$. Then for any $0 < \alpha_3 < 2\alpha_2 \min_t \breve{\delta}(\boldsymbol{L}_{\boldsymbol{S}_t})$, for all finite and feasible initialization, the following facts hold:*

(1) The parameter sequence $\{\boldsymbol{S}_t, \boldsymbol{G}_t^{(v)}, \boldsymbol{U}_t\}_t$ converges to a critical point $\{\boldsymbol{S}^*, \boldsymbol{G}^{(v)*}, \boldsymbol{U}^*\}$ of Eq. (6).

(2) The loss sequence $\{\mathcal{F}(\boldsymbol{S}_t, \boldsymbol{G}_t^{(v)}, \boldsymbol{U}_t)\}_t$ converges to the loss of critical point $\mathcal{F}(\boldsymbol{S}^*, \boldsymbol{G}^{(v)*}, \boldsymbol{U}^*)$ of Eq. (6).

(3) Algorithm 1 has a convergence rate of $\mathcal{O}(\frac{1}{T})$ with respect to Eq. (6).

We could then prove that the global convergence property also holds for the original problem.

Theorem 3 ***(Global Convergence of Algorithm 1 with respect to Eq. (5))*** *Under the same condition as Thm. 2, the sequence $\{\boldsymbol{S}_t, \boldsymbol{G}_t^{(v)}, \boldsymbol{U}_t\}$ generated by Algorithm 1 also satisfies (1)-(3) with respect to the original problem Eq. (5).*

Table 1: Clustering Error Rates (%) on four benchmarks. The best and second best results are highlighted in **soft red** and **soft blue**, respectively. The lower, the better. Part of results are cited from (Xu, Cheong, and Li 2018).

Type	Method	Hopkins155			Hopkins12		MTPV62			KT3DMoSeg	
		2 Motion	3 Motion	All	Mean	Median	Missing Data 12 clips	Hopkins 50 clips	All 62 clips	Mean	Median
Single-model	LSA	4.23	7.02	4.86	-	-	-	-	-	38.30	38.58
	GPCA	4.59	28.66	10.02	-	-	28.77	16.20	16.58	34.60	33.95
	ALC	2.40	6.69	3.56	0.89	0.44	0.43	18.28	14.88	24.31	19.04
	TPV	1.57	4.98	2.34	-	-	0.91	2.78	2.37	-	-
	T-Linkage	0.86	5.78	1.97	-	-	-	-	-	-	-
	SSC	1.52	4.40	2.18	-	-	17.22	2.01	5.17	33.88	33.54
	LRR	1.33	4.98	1.59	-	-	29.46	5.26	5.95	33.67	36.01
	BDR	0.95	0.85	0.93	-	-	26.63	7.81	5.09	32.88	33.01
Multi-model	MSSC	0.54	1.84	0.83	-	-	0.65	**0.65**	**0.65**	-	-
	RV	0.31	0.66	0.39	-	-	-	-	-	-	-
	KerAdd	0.27	0.66	0.36	0.11	**0.00**	1.41	0.76	0.88	8.31	1.02
	CoReg	0.37	0.75	0.46	**0.06**	**0.00**	**0.30**	0.83	0.73	**7.92**	**0.75**
	Subset	**0.23**	**0.58**	**0.31**	**0.06**	**0.00**	**0.30**	0.77	**0.65**	8.08	**0.71**
	Ours	**0.19**	**0.57**	**0.28**	**0.02**	**0.00**	**0.44**	**0.57**	**0.55**	**4.58**	1.10

Such a property ensures that our algorithm is insensitive to initialization, *i.e.*, for any initial values, both the loss and parameter sequences can converge to a stationary point and won't fluctuate around an optimum. Hence our algorithm is more stable than those without this property. Due to the space limit, the proofs of both Theorem 2 and 3 are provided in the supplementary materials[1].

Experiment

Experimental Setup

Dataset Hopkins155 dataset (Tron and Vidal 2007) consists of 155 video clips of indoor or outdoor scenes, where 120 of them are with two motions and 35 are with three motions. Hopkins12 dataset (Rao et al. 2010) is with 12 incomplete trajectories. Since Hopkins155 lacks of perspective effects and has a very imbalance number of two-motion and three-motion clips, (Lai et al. 2017) build a more complex dataset, MTPV62, by combining 50 clips in Hopkins155 and other 12 clips with object occlusions, of which 4 clips are from (Schindler, U, and Wang 2006) and 8 clips are collected by (Lai et al. 2017). The resulting dataset has 26 two-motion and 36 three-motion video clips. Moreover, (Xu, Cheong, and Li 2018) propose the KITTI 3D Motion Segmentation Benchmark (KT3DMoSeg) based on KITTI dataset (Geiger et al. 2013), which exhibits more significant camera translation, more complicated backgrounds, and interplays of multiple motions. This dataset has 22 short video clips with a maximum number of motions of 5.

Competitors We adopt the following 12 approaches as our competitors: (1) Subspace based: ALC (Rao et al. 2010), GPCA (Vidal, Tron, and Hartley 2008), SSC (Elhamifar and Vidal 2013), LRR (Liu et al. 2013), BDR (Lu et al. 2019); (2) Multimodel fitting based: T-Linkage (Magri and Fusiello 2014); (3) Fusion-based: MSSC (Lai et al. 2017), KernelAdd, CoReg and Subset (Xu, Cheong, and Li 2018), RV (Jung, Ju, and Kim 2019) (4) Two-frame based: Two-Perspective-View (TPV) (Li et al. 2013).

[1]https://github.com/jiangyangby/AAAI21-ConsistentMoSeg

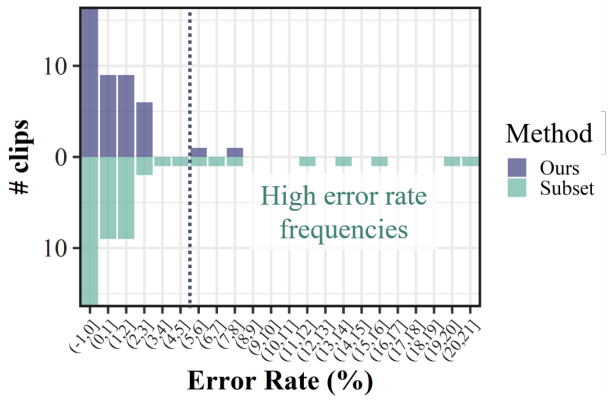

Figure 1: Histograms of error rates achieved by Ours (top half) and Subset (bottom half) on Hopkins155. For better visualization, the scale of y-axis is adjusted. The height of the first bin is 129 and 125 for Ours and Subset, respectively.

Metric Following previous studies, we use the clustering error rate as the evaluation metric for segmentation accuracy.

$$\text{ErrorRate} = \frac{\text{\# misclustered points}}{\text{\# total points}} \times 100\%$$

Implementation details We carry out our experiments on a Ubuntu 16.04 desktop with an Intel Core i7-8700K CPU and 64GB memory in MATLAB R2018a 64-bit. For those baselines which cannot deal with missing data, Chen's method (Chen 2008) is implemented to estimate the missing entries following (Xu, Cheong, and Li 2018). For the

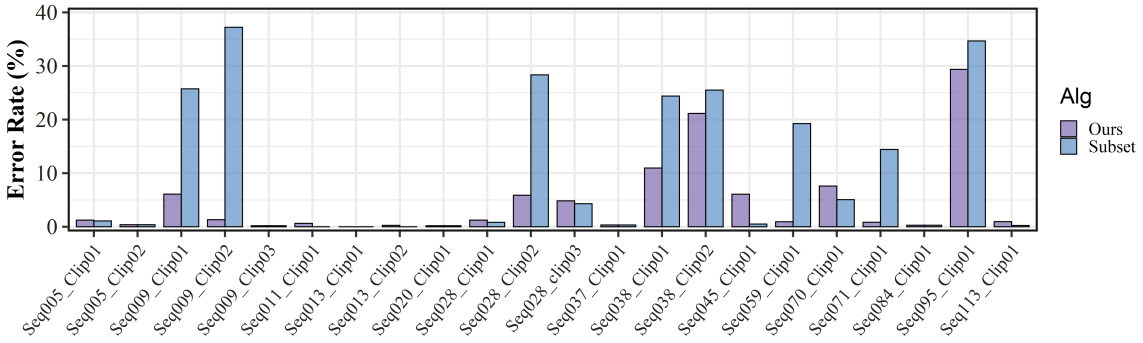

Figure 2: Error rate on each video clip of KT3DMoSeg dataset.

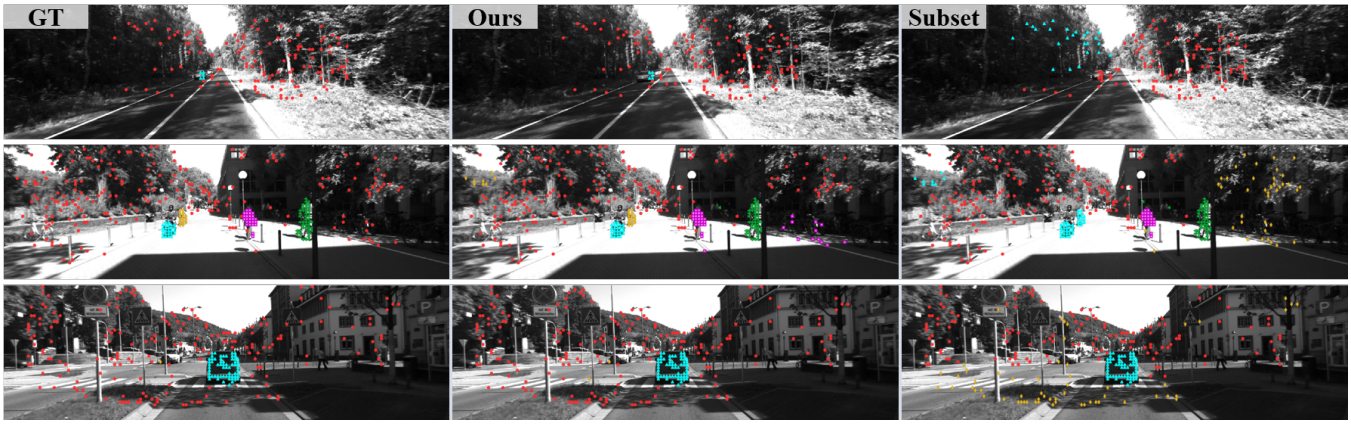

Figure 3: Examples of segmentation results on KT3DMoSeg. The left, middle, right column show the results of ground-truth (GT), Ours, and Subset, respectively. We show frames from `Seq028_Clip02`, `Seq038_Clip01`, `Seq059_Clip01` in rows.

proposed methods, the model hypotheses are fitted with linear algorithms (Hartley and Zisserman 2003). The minimal subset size for A, H, F is 8, 4 and 3. We adopt a grid search scheme for hyperparameter tuning. Specifically, α_1 is tuned within $\{0.0001, 0.0005, 0.001, 0.005, 0.01\}$, α_2 is within $\{0.001, 0.005, 0.01, 0.015, 0.03, 0.05\}$, and ϵ within $\{0, 1, 2, 3, 4, 5\}$. Meanwhile, we fix $S_{max} = 5$, $G_{min} = 0.0001$, and the tolerance as 0.001. Since we only require $\frac{\alpha_3}{\alpha_2}$ to have moderate value, α_3 is set to $0.001\alpha_2$.

Results All the results are recorded in Table 1. It shows that on all the datasets, our framework (denoted by Ours) consistently outperforms the best competitors with respect to the average error rate, where the performance is improved by 0.03%, 0.1% and 3.34%, respectively. On Hopkins155, Ours also achieves the best performance over the 2 motion and 3 motion subset. Meanwhile, on MTPV62, the error rate of Missing Data subset obtained by Ours is a little larger than Subset and CoReg. A reason might be that the performance of our proposed method largely depends on the quality of calculated affinity matrices, and the affinities for this subset might be not very good. Besides, the median error rate on KT3DMoSeg is not very promising. This may be caused by that each video clip needs individual choice of hyperparameters to achieve its best performance, while we set the same hyperparameters for all the video clips in a dataset. We will discuss the sensitivity of our model latter.

Other observations could be made that: (1) Model-fusion based methods are more competitive than single-model based ones, in particular when there is much data occlusion, which verifies the necessity of involving multiple models. (2) Among the model-fusion based approaches, integrating various types of geometric models (*i.e.*, Subset and Ours) is more effective than only utilizing homography one (*i.e.*, MSSC and RV). Namely, combining multiple geometric models brings better improvements.

Visualization In order to make a finer-grained comparison, we first plot the histograms of our proposed method and the most competitive baseline, Subset, over all the 155 video clips on Hopkins155 dataset. This is shown in Figure 1 with the bin size of 21. Though the performance of the two models is similar when the error rate ranges from 0 to 3, our proposed model obtains much less number of clips with large error rates (say those lying in (11, 21]), thus could achieve lower error rates on average. Besides, we also plot the error rate on each clip of KT3DMoSeg dataset for the two methods in Figure 2. From this figure we could also observe that the errors of our model are significantly lower than Subset's over many clips, *e.g.*, `Seq009_Clip02`,

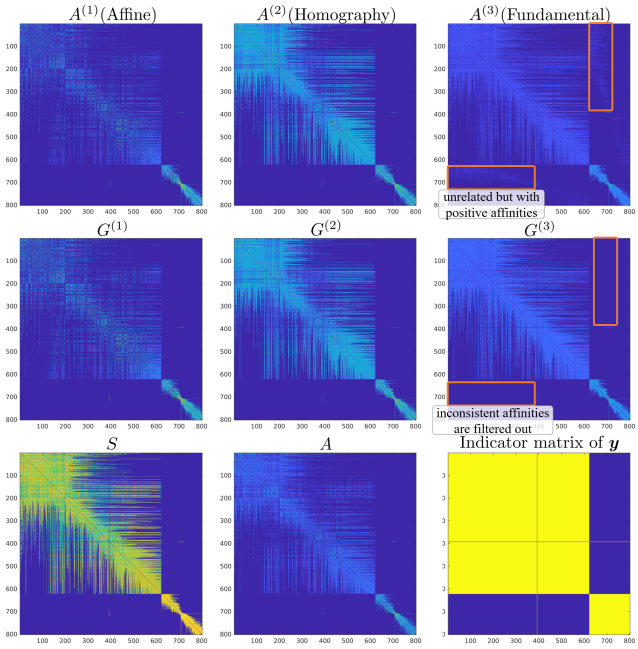

Figure 4: Visualization of $\boldsymbol{A}^{(v)}$s, obtained $\boldsymbol{S}$ and $\boldsymbol{G}^{(v)}$s, consistent affinity $\boldsymbol{A}$, and the indicator matrix for the segmentation results on the `Van` clip in MTPV62 dataset. The points are reordered according to their ground-truth labels for better illustration for the block-diagonal structure.

`Seq028_Clip02` and `Seq059_Clip01`, which is consistent with Hopkins155.

To better show that in what scenario our framework applies to, we present some segmentation results in Figure 3. It could be observed that:

- 1st row: Subset mixes up the very small object with a large portion of background points. However, our proposed model correctly discriminates the object's motion from the background motion in a higher probability.
- 2nd row: This clip contains more small objects thus is more challenging than the previous one. Compared with Ours, Subset again regards two small objects as one, and mis-clusters some points around an object. It also over-segments the entire background motion.
- 3rd row: A motion in this clip only has one tracked point (plotted in a yellow diamond), causing a significant cluster imbalance. Our method obtain most correct results, while Subset over-segments the background again.

We then move to analyze how the multiplicative decomposition work for the consensus affinity construction. Figure 4 visualize the values of basic affinity $\boldsymbol{A}^{(v)}$s, variables $\boldsymbol{S}, \boldsymbol{G}^{(v)}$, the final consensus affinity $\boldsymbol{A}$, and the indicator matrix for our segmentation results on the `Van` clip in MTPV62 dataset. First, the sparsity of $\boldsymbol{A}^{(v)}$ decreases w.r.t the increasing of v, which accords with the characteristic of geometric models. Comparing them with the indicator matrix, we see that some unrelated trajectories are assigned with positive affinities in $\boldsymbol{A}^{(3)}$, which causes totally different subspace structure with ground-truth. Nevertheless, in the obtained $\boldsymbol{G}^{(3)}$, these inconsistent affinities are filtered out. Meanwhile, the shape mask $\boldsymbol{S}$ and consensus affinity $\boldsymbol{A}$ both exhibit almost correct block-diagonal structure, resulting a nearly perfect segmentation result. Therefore, the strength of the proposed method is again verified.

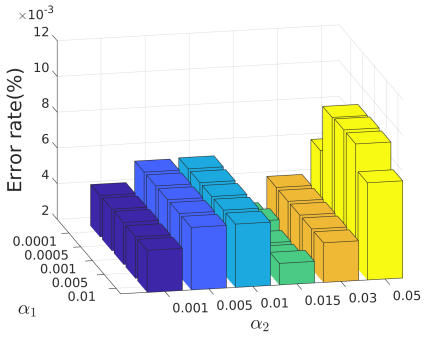

Figure 5: Sensitivity against α_1 and α_2 on Hopkins155 dataset. The error rate ranges from 0.28% to 0.89%.

Sensitivity analysis Next, we study the influence of two main hyperparameters on the proposed model, α_1 and α_2. α_1 constrains the magnitude of $\boldsymbol{G}^{(v)}$s to avoid overfitting, while α_2 helps control the block-diagonality of $\boldsymbol{S}$. To this end, a 3D-barplot is illustrated in Figure 5 based on the results of grid search on Hopkins155, where the x- and y-axis stand for the value of α_1 and α_2 respectively, and the z-axis shows the average error rate (%) on the dataset. On one hand, when α_1 is fixed, a too large α_2 (say 0.05) might rapidly increase the error rate since such values lead to structures that are either far from block-diagonal or exactly block-diagonal but with blocks involving wrong data points. On the other hand, when α_2 is not greater than 0.03, the performance change toward α_1 is moderate. Yet for $\alpha_2 = 0.05$ the performance becomes much unstable. The reason might be that a large α_2 means strict pursuit on the structure, causing the structure to be more easily broken by encouraging different magnitudes. Overall, our model is relatively sensitive toward α_2, while not much sensitive toward α_1. This is consistent with the intuition that the influence of structure should be much larger than that of magnitude.

Conclusion

In this paper, we propose a novel framework to boost the multi-model fusion for motion segmentation. Our main idea is based on seeking the consensus affinity matrix across multiple geometric models by guaranteeing their structural consistency. Specifically, the affinity matrix is decomposed into a multiplication of a model-sharing shape mask and a model-specific magnitude, with a block-diagonal structure regularization applied on the mask. Moreover, we provide a solver for this problem with a global convergence property. Finally, we evaluate our approach on four benchmarks. Experimental results show the superiority of our method.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under (Grant No. 2018AAA0102003), in part by National Natural Science Foundation of China (61620106009, U1636214, U1803264, U1636214, 61931008, 61836002, 61672514, and 61976202), in part by Key Research Program of Frontier Sciences, CAS: QYZDJ-SSW-SYS013, in part by Peng Cheng Laboratory Project of Guangdong Province PCL2018KP004, in part by Beijing Natural Science Foundation (No. 4182079), in part by Youth Innovation Promotion Association CAS, and in part by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDB28000000.

References

Barath, D.; and Matas, J. 2018. Multi-class Model Fitting by Energy Minimization and Mode-Seeking. In *European Conference on Computer Vision*, 229–245.

Baráth, D.; and Matas, J. 2019. Progressive-X: Efficient, Anytime, Multi-Model Fitting Algorithm. In *IEEE/CVF International Conference on Computer Vision*, 3779–3787.

Chen, P. 2008. Optimization Algorithms on Subspaces: Revisiting Missing Data Problem in Low-Rank Matrix. *Int. J. Comput. Vis.* 80(1): 125–142.

Chin, T.; Wang, H.; and Suter, D. 2009. The Ordered Residual Kernel for Robust Motion Subspace Clustering. In *Advances in Neural Information Processing Systems*, 333–341.

Elhamifar, E.; and Vidal, R. 2013. Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Trans. Pattern Anal. Mach. Intell.* 35(11): 2765–2781.

Fan, K. 1949. On a theorem of Weyl concerning eigenvalues of linear transformations I. *Proceedings of the National Academy of Sciences of the United States of America* 35(11): 652.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets robotics: The KITTI dataset. *I. J. Robotics Res.* 32(11): 1231–1237.

Hartley, R.; and Zisserman, A. 2003. *Multiple view geometry in computer vision*. Cambridge university press.

Huang, J.; Yang, S.; Zhao, Z.; Lai, Y.; and Hu, S. 2019. ClusterSLAM: A SLAM Backend for Simultaneous Rigid Body Clustering and Motion Estimation. In *IEEE/CVF International Conference on Computer Vision*, 5874–5883.

Jung, H.; Ju, J.; and Kim, J. 2019. Randomized Voting-Based Rigid-Body Motion Segmentation. *IEEE Trans. Circuits Syst. Video Technol.* 29(3): 698–713.

Kamranian, Z.; Naghsh-Nilchi, A. R.; Sadeghian, H.; Tombari, F.; and Navab, N. 2020. Joint motion boundary detection and CNN-based feature visualization for video object segmentation. *Neural Computing and Applications* 32(8): 4073–4091.

Kluger, F.; Brachmann, E.; Ackermann, H.; Rother, C.; Yang, M. Y.; and Rosenhahn, B. 2020. CONSAC: Robust Multi-Model Fitting by Conditional Sample Consensus. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4633–4642.

Lai, T.; Wang, H.; Yan, Y.; Chin, T.; and Zhao, W. 2017. Motion Segmentation Via a Sparsity Constraint. *IEEE Trans. Intell. Transp. Syst.* 18(4): 973–983.

Li, Z.; Guo, J.; Cheong, L.; and Zhou, S. Z. 2013. Perspective Motion Segmentation via Collaborative Clustering. In *IEEE International Conference on Computer Vision*, 1369–1376.

Lin, S.; Xiao, G.; Yan, Y.; Suter, D.; and Wang, H. 2019. Hypergraph Optimization for Multi-Structural Geometric Model Fitting. In *AAAI Conference on Artificial Intelligence*, 8730–8737.

Liu, G.; Lin, Z.; Yan, S.; Sun, J.; Yu, Y.; and Ma, Y. 2013. Robust Recovery of Subspace Structures by Low-Rank Representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 35(1): 171–184.

Lu, C.; Feng, J.; Lin, Z.; Mei, T.; and Yan, S. 2019. Subspace Clustering by Block Diagonal Representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 41(2): 487–501.

Magri, L.; and Fusiello, A. 2014. T-Linkage: A Continuous Relaxation of J-Linkage for Multi-model Fitting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3954–3961.

Magri, L.; and Fusiello, A. 2016. Multiple Models Fitting as a Set Coverage Problem. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3318–3326.

Magri, L.; and Fusiello, A. 2019. Fitting Multiple Heterogeneous Models by Multi-Class Cascaded T-Linkage. In *IEEE Conference on Computer Vision and Pattern Recognition*, 7460–7468.

Muthu, S.; Tennakoon, R. B.; Rathnayake, T.; Hoseinnezhad, R.; Suter, D.; and Bab-Hadiashar, A. 2020. Motion Segmentation of RGB-D Sequences: Combining Semantic and Motion Information Using Statistical Inference. *IEEE Trans. Image Process.* 29: 5557–5570.

Ranjan, A.; Jampani, V.; Balles, L.; Kim, K.; Sun, D.; Wulff, J.; and Black, M. J. 2019. Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 12240–12249.

Rao, S. R.; Tron, R.; Vidal, R.; and Ma, Y. 2010. Motion Segmentation in the Presence of Outlying, Incomplete, or Corrupted Trajectories. *IEEE Trans. Pattern Anal. Mach. Intell.* 32(10): 1832–1845.

Schindler, K.; U, J.; and Wang, H. 2006. Perspective *n*-View Multibody Structure-and-Motion Through Model Selection. In *European Conference on Computer Vision*, 606–619.

Sengar, S. S.; and Mukhopadhyay, S. 2020. Motion segmentation-based surveillance video compression using adaptive particle swarm optimization. *Neural Computing and Applications* 32(15): 11443–11457.

Tepper, M.; and Sapiro, G. 2017. Nonnegative Matrix Underapproximation for Robust Multiple Model Fitting. In

IEEE Conference on Computer Vision and Pattern Recognition, 655–663.

Tron, R.; and Vidal, R. 2007. A Benchmark for the Comparison of 3-D Motion Segmentation Algorithms. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Vidal, R.; Tron, R.; and Hartley, R. I. 2008. Multiframe Motion Segmentation with Missing Data Using PowerFactorization and GPCA. *Int. J. Comput. Vis.* 79(1): 85–105.

Von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and computing* 17(4): 395–416.

Xu, X.; Cheong, L. F.; and Li, Z. 2018. Motion Segmentation by Exploiting Complementary Geometric Models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2859–2867.

Xu, X.; Cheong, L. F.; and Li, Z. 2021. 3D Rigid Motion Segmentation with Mixed and Unknown Number of Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(1): 1–16.

Yang, Z.; Xu, Q.; Cao, X.; and Huang, Q. 2020. Task-Feature Collaborative Learning with Application to Personalized Attribute Prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

Yang, Z.; Xu, Q.; Zhang, W.; Cao, X.; and Huang, Q. 2019. Split Multiplicative Multi-View Subspace Clustering. *IEEE Trans. Image Process.* 28(10): 5147–5160.

Zhuo, T.; Cheng, Z.; Zhang, P.; Wong, Y.; and Kankanhalli, M. S. 2020. Unsupervised Online Video Object Segmentation With Motion Property Understanding. *IEEE Trans. Image Process.* 29: 237–249.