Robust Motion Segmentation From Pairwise Matches

Federica Arrigoni and Tomas Pajdla
CIIRC – Czech Technical University in Prague
Federica.Arrigoni@cvut.cz, pajdla@cvut.cz

Abstract

In this paper we consider the problem of motion segmentation, where only pairwise correspondences are assumed as input without prior knowledge about tracks. The problem is formulated as a two-step process. First, motion segmentation is performed on image pairs independently. Secondly, we combine independent pairwise segmentation results in a robust way into the final globally consistent segmentation. Our approach is inspired by the success of averaging methods. We demonstrate in simulated as well as in real experiments that our method is very effective in reducing the errors in the pairwise motion segmentation and can cope with large number of mismatches.

1. Introduction

Motion segmentation is an essential task in many applications in Computer Vision and Robotics, such as surveillance [18], action recognition [56] and scene understanding [12]. The classic way to state the problem is the following: given a set of feature points that are *tracked* through a sequence of images, the goal is to cluster those trajectories according to the different motions they belong to. It is assumed that the scene contains multiple objects that are moving rigidly and independently in 3-space. There is a plenty of available techniques to accomplish such task, as detailed in Sec. 1.1. Among them, the methods developed in [17, 21, 58] achieve a very low misclassification error on the Hopkins155 benchmark [47], which is a well established dataset to test the performance of motion segmentation. However, the tracks available in the dataset are not realistic at all since they were filtered with the aid of manual operations. In this paper we take motion segmentation one step further by considering more difficult/realistic assumptions, namely we assume that pairwise matches (e.g. those computed from SIFT keypoints [26]) are available only, and we address the task of classifying image points (instead of tracks), as shown in Fig. 1. This problem has not been considered before but it has great practical relevance since it does not require to compute tracks in advance, which is a challenging task in the presence of multiple moving objects.

Figure 1: Segmentation results are reported on four images for the technique in [58] combined with [30] (top) and our method (middle). Image points are drawn in different colours: green (correctly classified); red (misclassified); blue (unknown). Ground-truth segmentation is also reported (bottom) where different colors encode the membership to different motions.

More precisely, we formulate motion segmentation as a two-step process:

1. segmentation of corresponding points is performed on each image pair in isolation;
2. segmentation of image points is computed without relying on the feature locations, using only the classification of matching points derived in Step 1.

Our new formulation is detailed in Sec. 2. Regarding Step 2, we develop a simple scheme that classifies each point based on the frequencies of labels of that point in different image pairs, which is reported in Sec. 3. The resulting method is a general framework that can be combined with any algorithm able to perform motion segmentation in two images.

The idea of combining results from individual image pairs was also present in [24], where all the pairs were considered, and in [21, 58], where only pairs of consecutive frames were used. These techniques, however, are different from our approach since they do not completely perform segmentation of image pairs but they rely on *partial* results only (i.e. correlation of corresponding points). Such results are used to build an affinity matrix that encodes the similarity between different tracks, to which spectral clustering

[54] (or its multi-view variations [6, 20, 55]) is applied. As a consequence, they perform segmentation of tracks, whereas our method classifies image points. A related approach [16] considers the scenario where correspondences are unknown and uses the Alternating Direction Method of Multipliers (ADMM) [3] to jointly perform motion segmentation and tracking, where sequences with at most 200 trajectories are analyzed only due to algorithmic complexity.

Experiments on both synthetic and real data were performed to validate our approach. Robust Preference Analysis (RPA) [28] was used in Step 1. A new dataset was also created, consisting of five sequences of moving objects in an indoor environment, where SIFT keypoints [26] were extracted and manually labelled to get ground-truth segmentation. Results are reported in Sec. 4, where it is shown that: our method is comparable or better than most traditional (track-based) solutions on Hopkins155 [47]; it outperforms the methods developed in [17, 58] on synthetic/real datasets with mismatches; it is very effective in reducing the errors in the pairwise segmentations; it can be profitably used to segment SIFT keypoints in a collection of images.

Our two-step formulation of motion segmentation is inspired by the success gained by *synchronization* or *averaging* methods (e.g. [42, 46, 14, 1, 4]) that formulate other computer vision problems (e.g. structure from motion and 3D registration) in an analogous manner. For instance, 3D registration – where the task is to bring multiple scans into alignment – can be addressed by first computing rigid transformations between each pair of scans in isolation, and then globally optimizing these transformations without considering points. In particular, our method – which computes the segmentation of one image at a time (as explained in Sec. 3) – presents similarities with [46, 14], which estimate the transformation of one camera/scan at a time.

1.1. Related Work

Motion segmentation lies at the intersection of several computer vision problems, including subspace separation, multiple model fitting and multibody structure from motion.

The goal of *subspace separation* is to cluster high-dimensional data drawn from multiple low-dimensional subspaces. Existing solutions include Generalized Principal Component Analysis (GPCA) [50], Local Subspace Affinity (LSA) [59], Power Factorization (PF) [52], Agglomerative Lossy Compression (ALC) [36], Low-Rank Representation (LRR) [25], Sparse Subspace Clustering (SSC) [11], Structured Sparse Subspace Clustering (S^{3}C) [22], and Robust Shape Interaction Matrix (RSIM) [17]. Motion segmentation can be cast as subspace separation since – under the affine camera model – the point trajectories lie in the union of d subspaces in $\mathbb{R}^{2n}$ of dimension at most 4, where d denotes the number of motions and n denotes the number of images. Subspace separation techniques can also be used to solve motion segmentation in two images under the perspective camera model, since corresponding points undergoing the same motion – after a proper rearrangement of coordinates – belong to a subspace of $\mathbb{R}^9$ of dimension at most 8, as observed in [24].

The goal of *multiple model fitting* is to estimate multiple models (e.g. geometric primitives) that fit data corrupted by outliers and noise, without knowing which model each point belongs to. Some methods follow a consensus-based approach, namely they focus on the estimation part of the problem, with the aim of finding models that describe as many points as possible. The Hough transform [57], Sequential RANSAC [53], Multi-RANSAC [62] and Random Sample Coverage (RansaCov) [29] belong to this category. Other techniques follow a preference-based approach, namely they concentrate on the segmentation side of the problem, from which model estimation follows. Solutions of this type include Residual Histogram Analysis (RHA) [61], J-Linkage [44], Kernel Optimization [5], T-linkage [27], Random Cluster Model (RCM) [34] and Robust Preference Analysis (RPA) [28]. The problem of fitting multiple models can also be expressed in terms of energy minimization [8, 9], as done by PEARL (Propose Expand and Re-estimate Labels) [15] and Multi-X [2]. Model fitting techniques can be exploited to solve motion segmentation under the affine camera model, by fitting multiple subspaces to feature trajectories in an image sequence, similarly to subspace separation methods. They can also be used to solve motion segmentation in two images under the perspective camera model, by fitting multiple fundamental matrices to corresponding points in an image pair.

The goal of *multibody structure from motion* is to simultaneously estimate the motion between each object and the camera as well as the 3D structure of each object, given a set of images of a dynamic scene. This problem can be seen as the generalization of structure from motion [32] to the dynamic case, where motion segmentation has to be solved in addition to 3D reconstruction. Geometric solutions are available for two images [51] and three images [49]. Other techniques follow a statistical approach [45, 35, 41, 43, 31, 38], whereas in [13, 7, 23, 60] motion segmentation and structure from motion are combined. More details can be found in survey [40].

Ad-hoc solutions to motion segmentation are also present in the literature [24, 21, 58], which are not explicitly related to the aforementioned problems. The authors of [24] formulate a joint optimization problem which builds upon the SSC algorithm, where it is required that all image pairs share a common sparsity profile. In [21] an accumulated correlation matrix is built by sampling homographies over consecutive image pairs, and spectral clustering [54] is applied to get the sought segmentation. Such approach is generalized in [58] where multiple models (affine,

fundamental and homography) are combined to get an improved segmentation. Different approaches are analyzed to reach such task, namely Kernel Addition (KerAdd) [6], Co-Regularization (Coreg) [20] and Subset Constrained Clustering (Subset) [55]. Motion segmentation is also addressed in [39, 37], where existing algorithms are customized for specific scenarios and acquisition platforms.

2. Problem Formulation

Let n denote the number of images and let d denote the number of motions. Suppose that a number p_i of points are found in image i using a feature extraction algorithm, so that the total amount of points over all the images is given by $p = \sum_{i=1}^{n} p_i$. Let $\mathbf{s}_i \in \{0, 1, \ldots, d\}^{p_i}$ denote the labels of points in image i, which identify the membership to a specific motion. The meaning of the zero label, which essentially represents outliers, will be clarified in Sec. 3.3. The vector $\mathbf{s}_i$ is referred to as the *total segmentation* of image i, since it represents labels of points considering a *global* numbering of motions. The goal here is to estimate $\mathbf{s}_i$ for $i = 1, \ldots, n$, as shown in Fig. 2. In other words, we aim at classifying image points as opposed to existing methods which segment tracks. In order to reach such a task, we assume that points have been matched in image pairs and that segmentation between pairs of images is available. Note that the knowledge of matches, which involve two images at a time, is a weaker assumption than the presence of tracks, which involve all the images simultaneously.

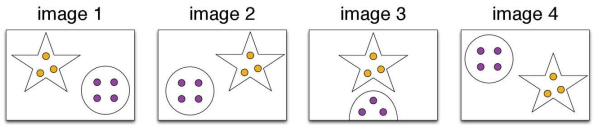


Figure 2: A set of points is detected in multiple images and the goal is to assign them a label (purple or yellow) based on the moving object (star or circle) they belong to.

Let $\mathbf{s}_{ij} \in \{0, 1, \ldots, d\}^{m_{ij}}$ denote the labels of corresponding points in images i and j, where the zero label corresponds to outliers and let $m_{ij} \leq \min\{p_i, p_j\}$ denote the number of matches of the pair (i, j). Vector $\mathbf{s}_{ij}$ is referred to as the *partial segmentation* of the pair (i, j), since it represents labels of corresponding points considering a *local* numbering of motions, as shown in Fig. 3. Observe that each $\mathbf{s}_{ij}$ may contain some errors, which can be caused either by mismatches or by failure of the algorithm used for pairwise segmentation, and some image points may not have a label assigned in some pairs due to missing correspondences.

Thus we have to face the problem of how to assign a unique/global label to all image points such that the constraints coming from pairwise segmentation are best satisfied. In other words, the segmentation task can be reduced to the problem of estimating the total segmentations $\mathbf{s}_1, \ldots, \mathbf{s}_n$ starting from the knowledge of partial segmentations $\mathbf{s}_{ij}$ with $i, j = 1, \ldots, n$. It is worth noting that in this way the actual coordinates of image points are not used anymore after pairwise segmentation, since only labels matter for the final segmentation. Observe also that this general formulation does not assume any particular camera model or scene geometry.

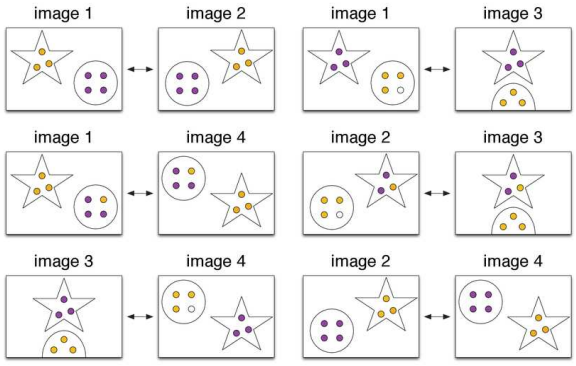


Figure 3: Motion segmentation is performed on image pairs (with possible errors). The same motion (star or circle) may be given a different label (purple or yellow) in different pairs.

3. Proposed Method

Our method (sketched in Fig. 7) takes as input the results from pairwise segmentation. It first computes the total segmentation of each image individually and then updates all these estimates in order to have a single/global numbering of motions.

3.1. Segmenting a single image

The key observation is that each partial segmentation $\mathbf{s}_{ij} \in \{0, 1, \ldots, d\}^{m_{ij}}$ gives rise to two vectors

$$\widehat{\mathbf{s}}_i^j \in \{\text{NaN}, 0, 1, \ldots, d\}^{p_i} \quad (1)$$

$$\widehat{\mathbf{s}}_j^i \in \{\text{NaN}, 0, 1, \ldots, d\}^{p_j} \quad (2)$$

which contain labels of matching points in images i and j, respectively, where NaN accounts for missing correspondences. This implies that, if we fix one image (e.g. image i), then several estimates are available for its total segmentation, which define a set $\mathcal{B}_i$

$$\mathcal{B}_i = \{\widehat{\mathbf{s}}_i^k \text{ s.t. } k = 1, \ldots, n,\ k \neq i\}. \quad (3)$$

However, these estimates are not absolute since they may differ by a permutation of the labels associated with each motion, as shown in Fig. 4.

In order to resolve such ambiguity, we consider a graph where each node is an element in $\mathcal{B}_i$ (i.e. a partial segmentation involving image i) and the edge between nodes h and

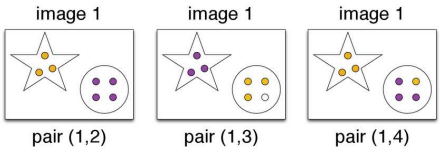


Figure 4: A possible solution for the total segmentation of image 1 is given by each partial segmentation where image 1 is involved. The same motion (star or circle) may be given a different label (purple or yellow) in different pairs.

k is associated with a permutation P_{hk} of labels that best maps $\widehat{\mathbf{s}}_i^k$ (i.e. labels of image i in the pair (i,k)) into $\widehat{\mathbf{s}}_i^h$ (i.e. labels of image i in the pair (i,h)). Computing such permutation is a *linear assignment problem*, which can be solved using the Hungarian algorithm [19]. The task here is to compute a permutation P_k for each node that reveals the true numbering of motions. It can be seen that this can be expressed as a *permutation synchronization*, that is the problem of estimating P_k for $k = 1, \dots, n$ $(k \neq i)$ such that $P_{hk} = P_h P_k^{-1}$, which can be solved via eigenvalue decomposition [33].

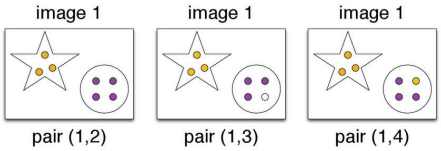


Figure 5: After solving a permutation synchronization problem, several estimates for the total segmentation of image 1 are available, where the same motion (star or circle) has the same label (purple or yellow) in different pairs.

After this step, the set in Eq. (3) contains several estimates of $\mathbf{s}_i$ with respect to a single numbering of motions, as shown in Fig. 5. Thus a scheme that assigns a unique label to each point in image i is required, which can be regarded as the best over the set $\mathcal{B}_i$. A reasonable approach consists in labelling each point with the most frequent label (i.e. the *mode*) among all the available measures. In other words, the label of point r is given by

$$\mathbf{s}_i(r) = \text{mode}\,\{\widehat{\mathbf{s}}_i^k(r) \text{ s.t. } \widehat{\mathbf{s}}_i^k \in \mathcal{B}_i,\ \widehat{\mathbf{s}}_i^k(r) \neq \text{NaN}\} \quad (4)$$

where only labels of actual correspondences are considered, with $r = 1, \dots, p_i$. As long as the algorithm used for pairwise segmentation correctly classifies all the points in most pairs, this procedure works well, as confirmed by experiments in Sec. 4.

3.2. Segmenting multiple images

The above procedure is applied to all the images in order to estimate the sought total segmentations $\mathbf{s}_1, \mathbf{s}_2 \dots, \mathbf{s}_n$. Such

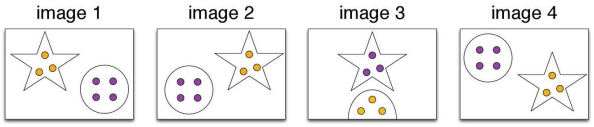


Figure 6: Motion segmentation is performed on each image individually. The same motion (star or circle) may be given a different label (purple or yellow) in different images.

estimates, however, are not absolute since each image has been treated independently from the others, and hence results may differ by a permutation of the labels associated with each motion, as shown in Fig. 6.

In order to address this issue, we consider a graph where each node corresponds to an image and the edge between images i and j is associated with a permutation P_{ij} that best maps $\mathbf{s}_j$ into $\mathbf{s}_i$. In order to compute such permutation, we ground on pairwise segmentation, since labels of the same points are required: in order to map $\mathbf{s}_j$ (labels of image j) into $\mathbf{s}_i$ (labels of image i), we first map $\widehat{\mathbf{s}}_i^j$ (labels of image i in the pair (i,j)) into $\mathbf{s}_i$, and then we map $\mathbf{s}_j$ into $\widehat{\mathbf{s}}_j^i$ (labels of image j in the pair (i,j)). These are linear assignment problems [19]. Thus the task is to compute a permutation P_i for each image that reveals the true numbering of motions such that $P_{ij} = P_i P_j^{-1}$, which can be viewed as a permutation synchronization [33]. Hence all the total segmentations are expressed with respect to the same numbering of motions, as in Fig. 2.

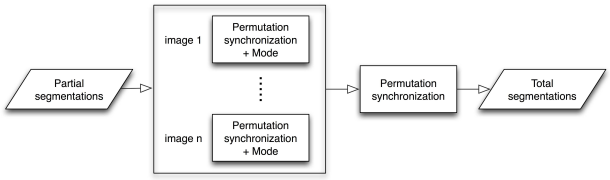


Figure 7: Outline of the proposed approach.

3.3. Dealing with outliers

When doing pairwise segmentation, it is expected that mismatched points are classified as outlier (zero label). When dealing with total segmentation, instead, the situation is different: in principle, there exists no outlier since each image point actually belongs to a motion. However, in the presence of high corruption in the input matches, one may not be able to assign a valid label to all image points. Indeed, it may happen that a point is mismatched (and hence assigned the zero label) in all the pairs, so that there is no valid information to classify it. Such points are expected to have zero label in the absolute segmentation. However, since they are not actual outliers, we will refer to them as "unclassified" or "unknown" in the experiments.

Table 1: Average misclassification error [%] for several methods on the Hopkins155 benchmark [47]. Results are copied from [58].

	LSA [59]	GPCA [50]	ALC [36]	SSC [11]	TPV [24]	LRR [25]	T-Linkage [27]	S^3C [22]	RSIM [17]	MSSC [21]	KerAdd [58]	Coreg [58]	Subset [58]	Baseline	MODE
2 Motions	4.23	4.59	2.40	1.52	1.57	1.33	0.86	1.94	0.78	0.54	0.27	0.37	**0.23**	2.26	1.00
3 Motions	7.02	28.66	6.69	4.40	4.98	4.98	5.78	4.92	1.77	1.84	0.66	0.75	**0.58**	9.04	2.67
All	4.86	10.02	3.56	2.18	2.34	1.59	1.97	2.61	1.01	0.83	0.36	0.46	**0.31**	3.79	1.37

Table 2: Average and median misclassification error [%] for several methods on the Hopkins12 benchmark [52]. Results for different variants of ALC and SSC are taken from [17] whereas results for the remaining methods are copied from the respective papers.

	PF [52]	PF+ALC [36]	RPCA+ALC [36]	ℓ_1+ALC [36]	SSC-R [11]	SSC-O [11]	RSIM [17]	KerAdd [58]	Coreg [58]	Subset [58]	Baseline	MODE
Mean	14.94	10.81	13.78	1.28	3.82	8.78	0.61	0.11	**0.06**	**0.06**	7.45	4.33
Median	9.31	7.85	8.27	1.07	0.31	4.80	0.61	**0.00**	**0.00**	**0.00**	2.16	0.38

In order to deal with those points, a reasonable approach is to ignore the labels which are set to zero by pairwise segmentation and compute the mode over the remaining measures, i.e. substitute them with NaN before using Eq. (4). In this way all the image points are assigned a valid label (except those which are deemed as outlier in all the pairs), meaning that this approach tends to classify a high amount of points even in the presence of mismatches.

4. Experiments

In order to evaluate the performance of our approach – named MODE[1] – we ran experiments on both synthetic data and real images, in addition to the real data Hopkins155 [47] and Hopkins12 [52]. For pairwise segmentations – which constitute the input to our method – we fitted multiple fundamental matrices to correspondences in each image pair using RPA [28] (code available online[2]). Default values specified in the original paper were used for the algorithmic parameters in all the experiments.

Note that there are no direct competitors to our method, since the task of segmentation from pairwise matches has not been addressed so far. For this reason, we focus on the comparison with a trivial solution (named the "baseline") which takes the *same* input as our approach (i.e. the results from pairwise segmentation) and it is constructed as follows: first, a maximum-weight spanning tree is computed, where each node in the graph is an image and edges are weighted with the number of inliers; then, the results from pairwise segmentation are used to segment each image along the tree, where the global numbering of motions is fixed at the root and propagated to the leaves.

In order to enrich the evaluation, we also considered traditional methods requiring tracks as input. In the case of the Hopkins datasets such tracks are available (Sec. 4.1), whereas in the remaining scenarios (Sec. 4.2 and 4.3) they were recovered from pairwise matches with two different approaches (i.e. [30, 48]). Similarly to most works in motion segmentation literature, we assumed that the number of motions was known in advance and gave this value as input to all the analysed techniques.

4.1. Hopkins Datasets

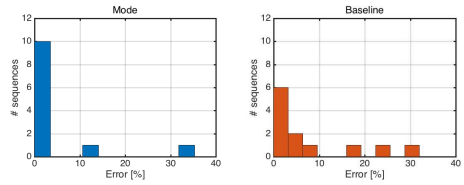
The Hopkins155 benchmark [47] contains 155 sequences of indoor and outdoor scenes with two or three motions, which are categorized into checkerboard, traffic and articulated/nonrigid sequences, and the Hopkins12 dataset [52] provides 12 additional sequences with missing data. We emphasize that these datasets provide (cleaned) tracks over multiple images, so they are not suitable for the task addressed in this paper, which is segmentation from raw pairwise matches. However, we report results on these sequences since they are widely used in the literature.

In order to make a meaningful comparison with the state of the art, a scheme that assigns a unique label to each track is required, starting from labels of image points. To accomplish such a task, we use the same criterion as the one developed in Sec. 3 to label each image point given multiple measures derived from pairwise segmentation. We assign to each track the mode of the labels of points belonging to the track, and the same procedure is applied to the baseline. Performance is measured in terms of *misclassification error*, that is the percentage of misclassified tracks, as it is customary in motion segmentation literature. Tracks labelled as zero (if any) were counted as errors, since we know that outliers are not present in these datasets.

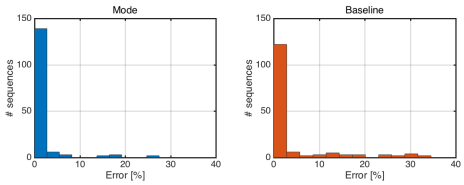
Results are reported in Tab. 1 and Tab. 2 where MODE is compared to several motion segmentation algorithms. Our approach clearly outperforms the baseline and it performs comparably or better than most of the state-of-the-art techniques, with a mean error of 1.37% over all the sequences in Hopkins155 and a median error of 0.38% over all the sequences in Hopkins12. In particular, it is noticeable that our method achieves (nearly) zero error in 139 out of 155 sequences in Hopkins155 and in 10 out of 12 sequences in Hopkins12, as shown in Fig. 8. After inspecting the solution, it was found that the remaining sequences correspond to situations where the algorithm used for pairwise segmentation (RPA) performed bad in most image pairs.

[1] https://github.com/federica-arrigoni/ICCV_19

[2] http://www.diegm.uniud.it/fusiello/demo/rpa/



(a) Hopkins12 [52]



(b) Hopkins155 [47]

Figure 8: Histograms of misclassification errors achieved by MODE and the baseline on the Hopkins155 and Hopkins12 datasets. The horizontal axis corresponds to a possible misclassification error in an individual sequence, and the vertical axis corresponds to the number of sequences where a given error is reached.

The fact that our method is not the best is not surprising since we are making much weaker assumptions (matches between image pairs instead of tracks over multiple images), i.e., we are addressing a more difficult task. Nevertheless, our method achieves good performances. In general, there is no reason to use our approach when tracks are available and one out of the best traditional methods (e.g.[17, 21, 58]) can be used. *Our method is designed for the scenario where pairwise matches are available only*. The next sections demonstrate the benefits of our approach for this specific task.

4.2. Simulated Data

We considered the *cars1* dataset from the traffic sequences in Hopkins155, where $d = 2$, $n = 20$ and $p = 6140$. Noise-free pairwise matches were obtained from the available tracks and synthetic errors were added to these correspondences in order to produce mismatches. More precisely, in each image pair a fraction of the correspondences – which ranged from 0 to 0.8 in our experiments – was randomly switched. This scenario resembles unordered image collections (e.g. in multibody structure from motion) where errors are ubiquitous among pairwise matches. For each configuration the test was repeated 10 times and average results were computed.

We compared MODE with the baseline, which – as our method – takes as input the results from pairwise segmentation. We also included in the comparison two traditional methods which require tracks over multiple images as input, namely RSIM[3] [17] and Subset[4] [58], whose implementations are available online. The former provides a robust solution to subspace separation, whereas the latter can be regarded as the current state of the art in motion segmentation with mean error of 0.31% on the Hopkins155 benchmark (see Tab. 1). We used two different techniques for computing tracks from pairwise matches, namely StableSfM[5] [30] and QuichMatch[6] [48].

Performance was measured in terms of misclassification errors, which is defined here as the percentage of misclassified points over the total amount of classified image points. In other words, unlike in Sec. 4.1, segmentation results were evaluated considering only points with a nonzero label (i.e. points with zero label did not contribute to the error). Indeed, due to the presence of mismatches, one may not expect to give a valid label to all the image points, as observed in Sec. 3.3. We also computed the percentage of points classified by each method.

Results are reported in Fig. 9, which clearly shows the robustness to mismatches gained by our approach: it is remarkable that the error remains constant (around 0%) with up to 60% of mismatches. MODE is significantly better than the baseline both in terms of misclassification error and percentage of classified points. The former exploits redundant measures in order to produce the final segmentation, whereas the latter uses results from a tree only.

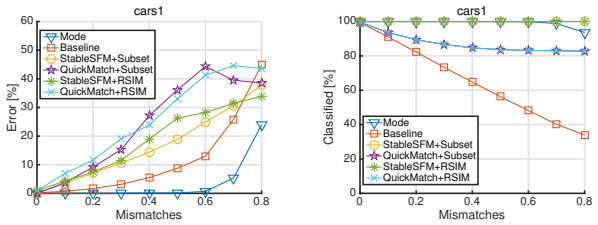


Figure 9: Misclassification error [%] and classified points [%] versus fraction of mismatches for several methods on the *cars1* sequence from Hopkins155 [47].

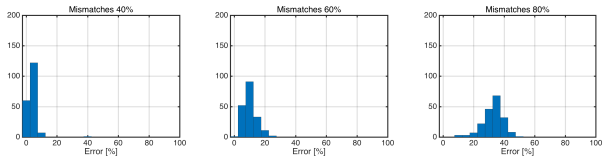


Figure 10: Histograms of misclassification error achieved by RPA [28] on *cars1* [47] for a single trial. The horizontal axis corresponds to the misclassification error in an individual image pair. The vertical axis corresponds to the number of pairs where a given error is obtained. The error and the percentage of points classified by MODE are, respectively, 0.2% and 100% for 40% of mismatches (left), 0.7% and 99.8% for 60% of mismatches (middle), 24.8% and 94.3% for 80% of mismatches (right).

[3] https://github.com/panji1990/Robust-shape-interaction-matrix

[4] https://alex-xun-xu.github.io/ProjectPage/CVPR_18/

[5] http://www.maths.lth.se/matematiklth/personal/calle/sys_paper/sys_paper.html

[6] https://bitbucket.org/tronroberto/quickshiftmatching

Concerning traditional methods, it was found by inspecting the solution that Subset and RSIM actually segment all the tracks, and unclassified data correspond to image points that were not included in any track by the algorithm used for computing tracks. Such techniques achieve a low misclassification error only when mismatches are below 10% and performances degrade with increasing ratio of mismatches. Indeed, wrong correspondences propagate into the tracks making traditional motion segmentation really hard to solve. Notice that a track can even contain points of different motions, in which case errors in the output segmentation appear by assigning a unique label to the entire track. This clearly motivates the need of our method for segmentation from raw pairwise matches.

In order to give a full picture on the performance of our approach, we report in Fig. 10 the histograms of misclassification error achieved by RPA over all the image pairs, which gives an idea about how hard it is to solve the motion segmentation *given* results of pairwise segmentation. Indeed, RPA may fail to detect errors in the input matches and it may not correctly segment some points since it lacks theoretical guarantees, thus producing errors in the individual pairwise segmentations. As expected, the histograms shift to the right as the percentage of input mismatches increases. Let us consider the central histogram, which corresponds to 60% of mismatches: it is worth noting that, despite individual pairwise segmentations are noisy, our method achieves nearly zero error. In other words, MODE is able to successfully solve motion segmentation while reducing errors in the pairwise segmentations, thanks to the fact that it exploits redundant measures in a principled manner. Further analysis can be found in the supplementary material.

4.3. Real Data

In order to evaluate the performance of our approach on real data, we considered both indoor and outdoor images. SIFT keypoints [26] were extracted in all the images and correspondences between image pairs were established using the nearest neighbor and ratio test as in [26], using the VLFeat library[7]. For each image pair (i, j), we kept only those correspondences that were found both when matching image i with j and when matching image j with i, and isolated features (i.e. points that are not matched in any image) were removed. No further filtering was applied.

4.3.1 Indoor scenes

Since there are no standard datasets for segmentation from pairwise matches, we created a small benchmark consisting of five image sequences. We considered indoor scenes containing two or three motions where one object is fixed (i.e. it is a part of the background), and we acquired from 6 to 10 images of size 2922×2000 with a moving camera. Fig. 11 shows a sample image from each sequence and more details about the dataset[8] can be found in the supplementary material. SIFT correspondences on such images are very noisy, as shown in Fig. 12, making motion segmentation a challenging task. In the case of the *Penguin* sequence there is no motion between some frames, so pairwise segmentation was not performed. In the remaining sequences, RPA was applied to all the image pairs.

(a) Flowers (b) Pencils (c) Bag (d) Bears (e) Penguin

Figure 11: Sample images from our dataset.

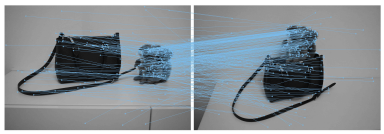


Figure 12: SIFT matches on an image pair from the *Bag* sequence.

As in Sec. 4.2, we compared MODE with the baseline, which takes as input the results from pairwise segmentation, and we also considered two traditional methods, namely RSIM [17] and Subset [58], where StableSfM [30] and QuichMatch [48] were used to compute tracks over multiple images. In order to evaluate results quantitatively, we manually labelled points in each sequence, thus producing a ground-truth segmentation of each image, that was used to compute the misclassification error. Results are shown in Tab. 3, which also reports the percentage of points classified by each method. See also Fig. 1 and the supplementary material for qualitative evaluations.

While there are no significant differences between MODE and the baseline in terms of misclassification error, the former is superior in terms of the percentage of classified points since it exploits *redundant* two-frame segmentations. Both our method and the baseline – with a misclassification error lower than 5% in all the sequences – are significantly better than Subset and RSIM. Traditional methods exhibit poor performances on our dataset since they do not deal with mismatches, confirming the outcome of the experiments on synthetic data.

We also tested the method developed in [16], which does not require pairwise matches but feature locations and descriptors only. We ran the available Matlab implementation of [16] on *Pencils*. It did not return any solution after several hours of computation due to "out of memory" error. We conclude that it is not yet a practical approach to motion segmentation on the scenarios considered in our paper.

[7] http://www.vlfeat.org/

[8] https://github.com/federica-arrigoni/ICCV_19

Table 3: Misclassification error [%] and classified points [%] for several methods on our dataset. The number of motions d, the number of images n, and the total number of image points p are also reported for each sequence.

				MODE		Baseline		StableSfM + Subset [58]		QuichMatch + Subset [58]		StableSfM + RSIM [17]		QuichMatch + RSIM [17]	
Dataset	d	n	p	Error	Classified	Error	Classified	Error	Classified	Error	Classified	Error	Classified	Error	Classified
Penguin	2	6	5865	**0.76**	69.17	0.95	33.95	32.27	99.59	41.05	70.11	41.50	99.59	41.54	70.11
Flowers	2	6	7743	**1.23**	73.65	2.84	32.70	8.55	99.50	8.59	72.59	16.65	99,50	14.20	72.59
Pencils	2	6	2982	3.80	65.33	**2.30**	30.65	41.46	99.56	40.88	66.36	23.07	99.56	23.45	66.36
Bag	2	7	6114	**1.52**	57.95	1.54	26.56	14.22	99.69	15.67	65.85	34.55	99.69	39.92	65.85
Bears	3	10	15888	4.82	73.65	**2.72**	29.80	38.13	99.58	35.21	63.12	49.48	99.58	53.80	63.12

(a) *helicopter* [10] (b) *boat* [24] (c) *cars7* [47] (d) *cars8* [47]

Figure 13: Segmentation results are reported on sample images for MODE (top), the baseline (middle) and StableSfM + Subset [58] (bottom). Different colours encode the membership to different motions. For better visualization, unclassified points are not drawn.

4.3.2 Outdoor scenes

To study a more realistic scenario, we considered four outdoor scenes, namely *helicopter* [10], *boat* [24], *cars7* [47] and *cars8* [47]. A subset of the images was chosen for each sequence in order to ensure enough motion between consecutive frames. The properties of each dataset are presented in Tab. 4, which also reports the percentage of points classified by MODE, the baseline and Subset [58] combined with StableSfM [30]. The latter provided the best results among all possible combinations of traditional segmentation methods and tracking algorithms. In the case of the *helicopter* sequence, a subset of the images has ground-truth pixel-wise annotation, which was used to compute the misclassification error (see Tab. 4). For the remaining sequences no ground-truth is available, so only qualitative evaluation can be provided, which is reported in Fig. 13 and in the supplementary material.

Results show that our solution is of good quality in all the images, outperforming the baseline in terms of amount of classified data. This is particularly evident in the right column of Fig. 13a where the baseline is not able to classify any point in the moving object. The poor performance of the baseline on some images gives an idea about how noisy the individual pairwise segmentations are. Our method is able to reduce such errors thanks to the fact that it exploits redundant measures. There are no significant differences between Subset and MODE in the *boat* sequence, which, however, is a simple scene for matching due to slow motion. In the *helicopter*, *cars7* and *cars8* sequences, Subset produces useless results.

Table 4: Misclassification error [%] and classified points [%] for several methods on outdoor scenes. The number of motions d, the number of images n, and the total number of image points p are also reported for each sequence.

				MODE		Baseline		StableSfM + Subset [58]	
Dataset	d	n	p	Error	Classified	Error	Classified	Error	Classified
helicopter [10]	2	10	17139	2.01	80.82	**0.78**	45.93	16.81	99.52
boat [24]	2	10	21183	–	87.34	–	56.31	–	99.62
cars7 [47]	2	21	16602	–	92.27	–	57.38	–	99.66
cars8 [47]	2	19	13438	–	93.12	–	50.53	–	99.61

5. Conclusion

We presented a new solution to the motion segmentation where the problem is split in two steps. First, a segmentation is performed independently on pairs of images. Then, the partial/local results are combined to segment points in all the images. This general framework – combined with a robust solution to two-frame segmentation (e.g. RPA [28]) – handles realistic situations such as the presence of mismatches that have been overlooked so far in previous work. Our approach does not require tracks as input but only pairwise correspondences. Thus it could be exploited to build tracks that are aware of segmentation, which constitute the foundation of a multibody structure from motion pipeline. Future research will explore this direction.

Acknowledgements. The authors would like to thank Luca Magri and Stanislav Steidl for their help with the experiments. This work was supported by the European Regional Development Fund under the project IMPACT (reg. no CZ.02.1.01/0.0/0.0/15_003/0000468).

References

[1] Federica Arrigoni, Beatrice Rossi, and Andrea Fusiello. Spectral synchronization of multiple views in SE(3). *SIAM Journal on Imaging Sciences*, 9(4):1963 – 1990, 2016.

[2] Daniel Barath and Jiri Matas. Multi-class model fitting by energy minimization and mode-seeking. In *Proceedings of the European Conference on Computer Vision*, pages 229–245. Springer International Publishing, 2018.

[3] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, Jan. 2011.

[4] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[5] Tat-Jun Chin, David Suter, and Hanzi Wang. Multi-structure model selection via kernel optimisation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3586–3593, 2010.

[6] Corinna Cortes, Mehryar Mohri, and Afshin Rostamizadeh. Learning non-linear combinations of kernels. In *Neural Information Processing Systems*, pages 396–404. 2009.

[7] João Paulo Costeira and Takeo Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159–179, 1998.

[8] Andrew Delong, Lena Gorelick, Olga Veksler, and Yuri Boykov. Minimizing energies with hierarchical costs. *International Journal of Computer Vision*, 100(1):38–58, 2012.

[9] Andrew Delong, Anton Osokin, Hossam N. Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 96(1):1–27, 2012.

[10] Ralf Dragon, Jörn Ostermann, and Luc Van Gool. Robust realtime motion-split-and-merge for motion segmentation. In *German Conference on Pattern Recognition*, pages 425–434. Springer Berlin Heidelberg, 2013.

[11] Ehsan Elhamifar and René Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.

[12] Andreas Ess, Tobias Mueller, Helmut Grabner, and Luc Van Gool. Segmentation-based urban traffic scene understanding. In *British Machine Vision Conference*, 2009.

[13] Charles William Gear. Multibody grouping from motion images. *International Journal of Computer Vision*, 29(2):133–150, 1998.

[14] Richard Hartley, Khurrum Aftab, and Jochen Trumpf. L1 rotation averaging using the Weiszfeld algorithm. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3041–3048, 2011.

[15] Hossam Isack and Yuri Boykov. Energy-based geometric multi-model fitting. *International Journal of Computer Vision*, 97(2):123–147, 2012.

[16] Pan Ji, Hongdong Li, Mathieu Salzmann, and Yuchao Dai. Robust motion segmentation with unknown correspondences. In *Proceedings of the European Conference on Computer Vision*, pages 204–219. Springer International Publishing, 2014.

[17] Pan Ji, Mathieu Salzmann, and Hongdong Li. Shape interaction matrix revisited and robustified: Efficient subspace clustering with corrupted and incomplete data. In *Proceedings of the International Conference on Computer Vision*, pages 4687–4695, 2015.

[18] Jong Bae Kim and Hang Joon Kim. Efficient region-based motion segmentation for a video monitoring system. *Pattern Recognition Letters*, 24(1):113 – 128, 2003.

[19] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly 2*, 2:83 – 97, 1955.

[20] Abhishek Kumar, Piyush Rai, and Hal Daume. Co-regularized multi-view spectral clustering. In *Neural Information Processing Systems*, pages 1413–1421. 2011.

[21] Taotao Lai, Hanzi Wang, Yan Yan, Tat-Jun Chin, and Wan-Lei Zhao. Motion segmentation via a sparsity constraint. *IEEE Transactions on Intelligent Transportation Systems*, 18(4):973–983, 2017.

[22] Chun-Guang Li and R. Vidal. Structured sparse subspace clustering: A unified optimization framework. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 277–286, 2015.

[23] Ting Li, Vinutha Kallem, Dheeraj Singaraju, and René Vidal. Projective factorization of multiple rigid-body motions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6, 2007.

[24] Zhuwen Li, Jiaming Guo, Loong-Fah Cheong, and Steven Zhiying Zhou. Perspective motion segmentation via collaborative clustering. In *Proceedings of the International Conference on Computer Vision*, pages 1369–1376, 2013.

[25] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 171–184, 2013.

[26] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[27] Luca Magri and Andrea Fusiello. T-Linkage: A continuous relaxation of J-Linkage for multi-model fitting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3954–3961, June 2014.

[28] Luca Magri and Andrea Fusiello. Robust multiple model fitting with preference analysis and low-rank approximation. In *Proceedings of the British Machine Vision Conference*, pages 20.1–20.12. BMVA Press, September 2015.

[29] Luca Magri and Andrea Fusiello. Multiple models fitting as a set coverage problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3318–3326, June 2016.

[30] Carl Olsson and Olof Enqvist. Stable structure from motion for unordered image collections. In *Proceedings of the 17th Scandinavian conference on Image analysis (SCIA'11)*, pages 524–535. Springer-Verlag, 2011.

[31] Kemal Egemen Ozden, Konrad Schindler, and Luc Van Gool. Multibody structure-from-motion in practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1134–1141, 2010.

[32] Onur Ozyesil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion. *Acta Numerica*, 26:305 – 364, 2017.

[33] Deepti Pachauri, Risi Kondor, and Vikas Singh. Solving the multi-way matching problem by permutation synchronization. In *Advances in Neural Information Processing Systems 26*, pages 1860–1868. Curran Associates, Inc., 2013.

[34] Trung-Thanh Pham, Tat-Jun Chin, Jin Yu, and David Suter. The random cluster model for robust geometric fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1658–1671, 2014.

[35] Gang Qian, R. Chellappa, and Qinfen Zheng. Bayesian algorithms for simultaneous structure from motion estimation of multiple independently moving objects. *IEEE Transactions on Image Processing*, 14(1):94–109, 2005.

[36] Shankar Rao, Roberto Tron, Rene Vidal, and Yi Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *Pattern Analysis and Machine Intelligence*, 32(10):1832–1845, 2010.

[37] Cosimo Rubino, Alessio Del Bue, and Tat-Jun Chin. Practical motion segmentation for urban street view scenes. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2018.

[38] Reza Sabzevari and Davide Scaramuzza. Monocular simultaneous multi-body motion segmentation and reconstruction from perspective views. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 23–30, 2014.

[39] Reza Sabzevari and Davide Scaramuzza. Multi-body motion estimation from monocular vehicle-mounted cameras. *IEEE Transactions on Robotics*, 32(3):638–651, 2016.

[40] Muhamad Risqi U. Saputra, Andrew Markham, and Niki Trigoni. Visual SLAM and structure from motion in dynamic environments: A survey. *ACM Computing Surveys*, 51(2):37:1–37:36, 2018.

[41] Konrad Schindler, David Suter, and Hanzi Wang. A model-selection framework for multibody structure-and-motion of image sequences. *International Journal of Computer Vision*, 79(2):159–177, 2008.

[42] Amit Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and Computational Harmonic Analysis*, 30(1):20 – 36, 2011.

[43] Ninad Thakoor, Jean Gao, and Venkat Devarajan. Multibody structure-and-motion segmentation by branch-and-bound model selection. *IEEE Transactions on Image Processing*, 19(6):1393–1402, 2010.

[44] Roberto Toldo and Andrea Fusiello. Robust multiple structures estimation with J-Linkage. In *Proceedings of the European Conference on Computer Vision*, pages 537–547, 2008.

[45] Philip H. S. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 356(1740):1321–1340, 1998.

[46] Andrea Torsello, Emanuele Rodolà, and Andrea Albarelli. Multiview registration via graph diffusion of dual quaternions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2441 – 2448, 2011.

[47] Roberto Tron and René Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[48] Roberto Tron, Xiaowei Zhou, Carlos Esteves, and Kostas Daniilidis. Fast multi-image matching via density-based clustering. In *Proceedings of the International Conference on Computer Vision*, pages 4077–4086, 2017.

[49] René Vidal and Richard Hartley. Three-view multibody structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):214–227, 2008.

[50] René Vidal, Yi Ma, and S. Shankar Sastry. Generalized principal component analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959, 2005.

[51] René Vidal, Yi Ma, Stefano Soatto, and Shankar Sastry. Two-view multibody structure from motion. *International Journal of Computer Vision*, 68(1):7–25, 2006.

[52] René Vidal, Roberto Tron, and Richard Hartley. Multiframe motion segmentation with missing data using powerfactorization and GPCA. *International Journal of Computer Vision*, 79(1):85–105, 2008.

[53] Esther Vincent and Robert Laganiere. Detecting planar homographies in an image pair. In *International Symposium on Image and Signal Processing and Analysis*, pages 182–187, 2001.

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

[55] Xiang Wang, Buyue Qian, and Ian Davidson. On constrained spectral clustering and its applications. *Data Mining and Knowledge Discovery*, 28(1):1–30, 2014.

[56] Daniel Weinland, Remi Ronfard, and Edmond Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224 – 241, 2011.

[57] Lei Xu, Erkki Oja, and Pekka Kultanen. A new curve detection method: randomized Hough transform (RHT). *Pattern Recognition Letters*, 11(5):331–338, 1990.

[58] Xun Xu, Loong-Fah Cheong, and Zhuwen Li. Motion segmentation by exploiting complementary geometric models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[59] Jingyu Yan and Marc Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and nondegenerate. In *Proceedings of the European Conference on Computer Vision*, pages 94–106, 2006.

[60] Luca Zappella, Alessio Del Bue, Xavier Lladó, and Joaquim Salvi. Joint estimation of segmentation and structure from motion. *Computer Vision and Image Understanding*, 117(2):113 – 129, 2013.

[61] Wei Zhang and Jana Kosecká. Nonparametric estimation of multiple structures with outliers. In *Workshop on Dynamic*

Vision, European Conference on Computer Vision 2006, volume 4358 of *Lecture Notes in Computer Science*, pages 60–74. Springer, 2006.

[62] Marco Zuliani, Charles S. Kenney, and B. S. Manjunath. The multiRANSAC algorithm and its application to detect planar homographies. In *Proceedings of the IEEE International Conference on Image Processing*, pages III–153–6, 2005.