

T-Linkage: a Continuous Relaxation of J-Linkage for Multi-Model Fitting

Luca Magri

Department of Mathematics - University of Milan
Via Saldini, 50 - 20133 Milan, Italy

luca.magri@unimi.it

Andrea Fusiello

DIEGM - University of Udine
Via Delle Scienze, 208 - 33100 Udine, Italy

andrea.fusiello@uniud.it

Abstract

This paper presents an improvement of the J-linkage algorithm for fitting multiple instances of a model to noisy data corrupted by outliers. The binary preference analysis implemented by J-linkage is replaced by a continuous (soft, or fuzzy) generalization that proves to perform better than J-linkage on simulated data, and compares favorably with state of the art methods on public domain real datasets.

1. Introduction

A widespread problem in Computer Vision is estimating the parameters of mathematical models from a set of observed data that may be contaminated with noise and outliers. When multiple instances of the same structure are present in the data, the problem becomes challenging, as it is a typical example of a chicken-and-egg dilemma: in order to estimate models one needs to first segment the data, and in order to segment the data it is necessary to know the models associated with each data point. Moreover, the presence of multiple structures strains robust estimation, which have to cope with both *gross* outliers and *pseudo*-outliers (i.e. “outliers to the structure of interest but inliers to a different structure” [15]). The main challenge is therefore the simultaneous robust estimation of both segmentation and models without knowing in advance the correct number of models. This issue is ubiquitous and can be encountered in many applications, as in homography estimation, or in multi-body segmentation, just to name a few. Many solutions have been proposed, which will be discussed in Sec. 1.1. In this paper we improve the J-Linkage approach [20], that has proven successful in several applications [5, 6, 19, 16, 21]. In particular, we propose a continuous (or “soft”) relaxation of the binary, winner-take-all approach followed by J-linkage. This new algorithm, called *T-linkage*, will be shown to perform better than J-linkage on some simulated data (Sec. 3) and will be compared against state-of-the art methods on publicly available real datasets.

1.1. Related work

The existing methods aimed at fitting multiple models can be divided in parametric methods and non parametric ones. Within the first, mode finding in parameter space and Randomized Hough Transform (RHT) [26] cope naturally with multiple structures, but cannot deal with high percentage of gross outliers, especially as the number of models grows and the distribution of inliers per model is uneven.

Among the non parametric methods one of the most popular paradigms is the one based on random sampling typified by RANSAC and related multi-model estimators like Multi-RANSAC [30] and FLoSS [8]. The latter, for example, is an iterative method that constructs models similarly to RANSAC, but it considers them all simultaneously.

Somehow orthogonal to the consensus analysis (analyzing the distribution of residuals per each model) is the *preference analysis*, introduced by RHA [29], that consist in inspecting the distribution of residuals of individual data points with respect to the models. In the same spirit of RHA, J-Linkage [20] avoids a consensus oriented approach in favor of working in the preference space. J-Linkage adopts a *conceptual representation* of points: each data point is represented with the characteristic function of the set of models preferred by that point. Multiple models are revealed as clusters in the conceptual space.

Along the same line, [1] presented a method where a point is represented by the permutation that arranges the models in order of ascending residuals. A kernel is then defined, based on this representation, such that in the corresponding Reproducing Kernel Hilbert Space inliers to multiple models and outliers are well separated. This allows to remove outliers, then the clean data is over-clustered with kernel-PCA and spectral clustering and the resulting structures are merged with a sequential ad-hoc scheme that incorporates model selection criteria. In [2] this last stage has been refined with kernel optimization. Residual information is also exploited in [11], a single-model estimation technique based on random sampling, where the inlier threshold is not required. The representation of points by a permutation of models is also exploited in [28] by QP-MF a quadratic programming

aimed to maximize mutual preferences of inliers.

A model selection approach is taken in [13, 3, 9, 10, 7], where the cost function to be minimized is composed by a data term that measures goodness of fit and a penalty term which weighs model complexity (see e.g. [22]). Sophisticated and effective minimization techniques such as SA-RCM [10], ARJMC [9] and PEaRL [7] have been proposed. The latter, for example, optimize a global energy function that balances geometric errors and regularity of inlier clusters, also exploiting spatial coherence.

A recent direction of investigation focuses in particular on subspace segmentation [27, 12, 4]. Local subspace affinity (LSA) [27] is an algebraic method that uses local information around points in order to fit local subspaces and to cluster points using spectral clustering with pairwise similarities computed using angles between the local subspaces. Agglomerative Lossy Compression (ALC) [12] is a bottom up clustering algorithm that aims at segmenting the data minimizing a coding length needed to fit the points with a mixture of degenerate Gaussians up to a given distortion.

SSC [4] exploits sparse representation for segmenting data. Since high dimensional data can be expressed as linear combination of few other data points, SSC use the sparse vectors of coefficients of these linear combinations as a convenient conceptual representation of points. Then spectral clustering is performed for segmenting data in this conceptual space.

2. Method

The proposed method starts, as J-Linkage [20], with random sampling: m model hypotheses $H = \{h_j\}_{j=1\dots m}$ are generated by drawing m minimal sets of data points necessary to estimate the model, called *minimal sample sets* (MSS).

When the pool of hypotheses has been generated, the conceptual space for representing cluster of points can be defined. While in [20] each data point is represented by the characteristic function of the set of models it prefers, we propose to exploit a more general conceptual space for depicting point preferences.

2.1. Conceptual space

In J-Linkage at first the *consensus set* (CS) of each model is computed as in RANSAC. The CS of a model is the set of points such that their distance from the model is less than a threshold $\epsilon \in \mathbb{R}$. In this way each data point x can be seen as its *preference set* (PS), the set of models it has given consensus to, or equivalently it can be thought as the characteristic function on the hypothesis space H :

$$\chi(h) = \begin{cases} 1 & \text{se } d(x, h) < \theta \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Note that the PS encodes the same information of CS, but with the roles of data points and models reversed. This is the key idea of the algorithm: rather than taking models and see what points match them, J-linkage uses the models each point matches to determine which points belong to the same cluster. As a result the conceptual space adopted is $\{0, 1\}^H = \{\chi : H \rightarrow \{0, 1\}\}$.

T-Linkage generalizes this idea introducing a relaxation of the binary PS: the *preference function* of a point (PF). Each data point is described by a function ϕ taking values in the whole closed interval $[0, 1]$ and not only in $\{0, 1\}$. In this way the conceptual space is generalized from the set of characteristic functions to $[0, 1]^H = \{\phi : H \rightarrow [0, 1]\}$, and we are allowed to express more accurately the preferences of a point integrating more specific information on residuals. Please note how this parallels the difference between RANSAC and MSAC [23], if CS is considered.

In practice, the idea is to mitigate the truncating effect of threshold in (1) defining the PF of a point x as:

$$\phi(h) = \begin{cases} e^{-d(x,h)/\tau} & \text{if } d(x, h) < 5\tau \\ 0 & \text{if } d(x, h) \geq 5\tau. \end{cases} \quad (2)$$

The time constant τ plays the same role of the inliers threshold ϵ , however, compared to the discrete case, the threshold defined by (2) is less critical since it replaces the abrupt truncation in (1) with an exponential decay. Please note that setting to zero the residuals smaller than 5τ is quite natural since for $x > 5\tau$ the function $e^{-x/\tau}$ can be considered almost constant, with variations that do not exceed 0.7 %.

Observe that, since the hypothesis space H is finite, both the conceptual space described above can be embedded into vectorial spaces. As $\{0, 1\}^H \simeq \{0, 1\}^m$, PS can also be conceived as binary vector, while, since $[0, 1]^H \simeq [0, 1]^m$, PF can be figured as vector in the m -dimensional unitary cube. The rationale beyond this construction is that the i -th component $(p)_i$ of the preference function p of a point x expresses with a vote in $[0, 1]$ the preference granted by x to the hypothesis model h_i . For simplicity sake we identify these functions with their images and adopt this vectorial interpretation in the rest of the paper.

2.2. T-Linkage clustering

J-linkage [20] extracts models by agglomerative clustering of data points in the conceptual space, where each point is represented by the function of its preference. The general agglomerative clustering algorithm proceeds in a bottom-up manner: starting from all singletons, each sweep of the algorithm merges the two clusters with the higher similarity. For this purpose we need at first to define a suitable conceptual representation for clusters and then we have to endow the conceptual space with an appropriate notion of similarity. J-Linkage accomplishes these two tasks relying on the set-interpretation of $\{0, 1\}^m$. The preference set of

	T-Linkage	J-Linkage
Space	$[0, 1]^m$	$\{0, 1\}^m$
Cluster	min PF	\bigcap PS
Similarity	Tanimoto	Jaccard

Table 1: The differences between T-Linkage and J-Linkage

a cluster is defined as the intersection of preference set of its points. The similarity between two clusters is computed as the *Jaccard distance* between their PS. Given two sets A and B , the Jaccard distance is

$$d_J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}. \quad (3)$$

The Jaccard distance measures the degree of overlap of the two sets and ranges from 0 (identical sets) to 1 (disjoint sets). These notions can be extended and made apt to the continuous space $[0, 1]^m$. The preference function of a subset of data point S is defined vectorially as follows. If $S = \{x\}$ the preference function of S is simply the preference function of x , denoted by q_x , otherwise the preference function of S is a vector $p \in [0, 1]^m$ such that its i -th component is

$$(p)_i = \min_{x \in S} (q_x)_i. \quad (4)$$

The similarity measure in $[0, 1]^m$ is provided by the so called *Tanimoto distance* [18] (hence the name T-linkage):

$$d_T(p, q) = 1 - \frac{\langle p, q \rangle}{\|p\|^2 + \|q\|^2 - \langle p, q \rangle}, \quad (5)$$

here with the notation $\langle \cdot, \cdot \rangle$ we indicate the standard inner product of \mathbb{R}^m and with $\|\cdot\|$ the corresponding induced norm. The Tanimoto distance ranges in $[0, 1]$ and it is simple to show that $\forall p, q \neq 0 \in [0, 1]^m$

$$d_T(p, q) = 0 \iff p = q, \quad (6)$$

$$d_T(p, q) = 1 \iff p \perp q. \quad (7)$$

The main differences between the conceptual space adopted by T-Linkage and J-Linkage are summarized in Table 1.

Please observe that if we confine ourselves to the space $\{0, 1\}^m$ the Tanimoto distance coincides with the Jaccard one, and also the definition of PF of a cluster is identical to the one proposed for preference set.

In the continuous case, as in its discrete counterpart, the cutoff value is set to 1, which means that the algorithm will only link together elements whose preference functions are not orthogonal. As a consequence

- for each cluster there exist at least one model for which all the points have expressed a positive preference (i.e., a model that fits all the points of the cluster)

- it is not possible that two distinct clusters grant a positive preference to the same model (otherwise they would have been linked).

Each cluster of points defines (at least) one model. If more models fit all the points of a cluster they must be very similar.

Agglomerative clustering algorithms in principle fit all the data: bad models must be filtered out a-posteriori (as will be discussed in the next section). Finally, the model for each cluster of points is estimated by least squares fitting.

2.3. Outliers rejection

Following the idea of MINPRAN [14] we make the mild assumption that outliers have a uniform distribution and compute the probability that k outliers by coincidence define a model as $\alpha(k) = 1 - \mathcal{F}(k; n, p)$ where \mathcal{F} is the binomial cumulative distribution function, n is the total number of points and p is the ratio between the inlier threshold and the range of the absolute residual of outliers (estimated with Monte-Carlo simulation). Thus, based on the observation that large clusters of outliers are very unlikely, we can assume that the smallest inlier structure have a cardinality significantly larger than the largest outliers cluster. In practice, this translates into the following rejection criterion:

- sort the clusters by cardinality, add one dummy cluster with the cardinality of the MSS at the end (to cater for the outliers-free case)
- retain as inliers all those clusters that are large enough to have a negligible probability $\alpha(k)$ of being coincidences (we used $k = \alpha^{-1}(0.01)$)
- among the remaining clusters find the point where the cardinality drops, by seeking the largest difference; clusters below that point are rejected as outliers.

This strategy is far from perfect: small clusters of inliers will be inevitably filtered out. T-linkage is agnostic about the outliers rejection strategy that comes after; depending on the application, different rejection criteria can be adopted. This one, however, proven sufficiently powerful in our experiments, as will be reported in the next section.

A final remark concerns the sampling strategy. In this paper since the data we are dealing with have a geometric nature, we gain some benefits (see Fig. 1) by introducing a locality bias. In particular, we adopt a mixed sampling strategy, combining uniform sampling with local sampling for selecting neighboring points with higher probability. In this way we are able to *exploit* local information and at the same time to *explore* hypothesis space.

T-Linkage is summarized as shown in Algorithm 1:

3. Experiments with simulated data

In this section we aim at characterizing the performances of T-linkage with respect to J-Linkage on simulated data

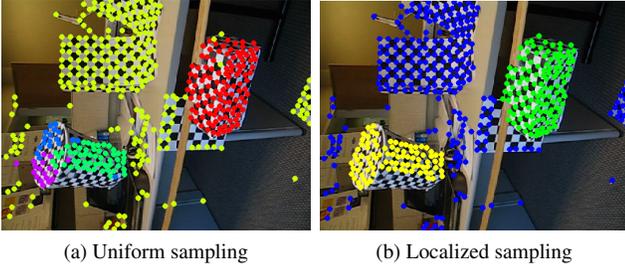


Figure 1: At equal number of hypotheses, localized sampling helps in producing more accurate results.

Algorithm 1 T-Linkage

Input: the set of data points

Output: clusters of point belonging to the same model

```

Put each point in its own cluster and compute their PF
while ( $\exists$  PF not orthogonal) do
    Find  $p, q : d_T(p, q) = \min_{r, s} d_T(r, s)$ .
    Replace these two cluster with the union of the two
    original ones and compute the PF of this new cluster
end while
Outliers rejection
Fit models to inliers clusters

```

(Fig. 2). The experimental results will confirm the benefits of working with continuous values rather than operating with hard thresholding: T-linkage takes advantage of the more expressive representation of points both in term of misclassification error and robustness to outliers.

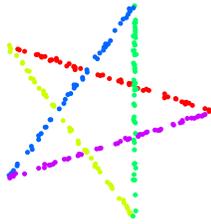


Figure 2: *Star5* data consisting of 250 points on five lines forming a star with added noise, as in [20].

We first compare the performances of J-Linkage and T-linkage on fitting lines to the *Star5* data (Fig. 2) using the *misclassification error* (ME), defined as follows:

$$ME = \frac{\# \text{ misclassified points}}{\# \text{ points}}. \quad (8)$$

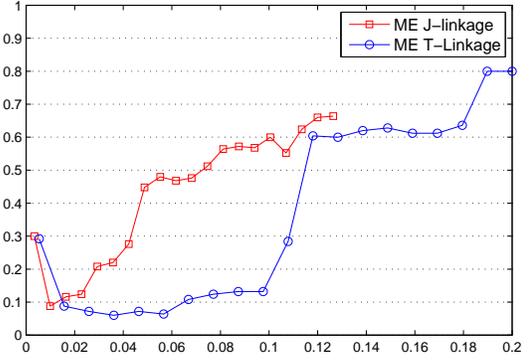


Figure 3: Misclassification errors as functions of inlier thresholds. The misclassification error committed by J-Linkage and T-Linkage on *Star5* is reported as a function of their corresponding threshold parameters. We have set $\theta = \int_0^{5\tau} e^{-\frac{x}{5\tau}} dx$ so that the area under the two voting function is the same.

where a point is *misclassified* when it is assigned to the wrong model, according to the ground-truth.

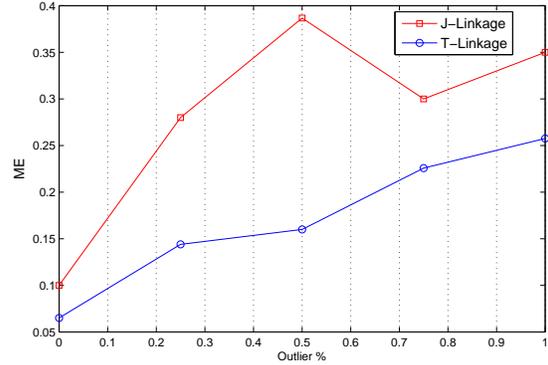


Figure 4: Misclassification error in presence of outliers in circle fitting. For each outliers level the corresponding misclassification error is reported for both J-Linkage and T-linkage.

The results can be appreciated in Fig. 3 where the corresponding ME is reported as a function of threshold parameters for both J-Linkage and T-linkage. The advantages of T-linkage over J-linkage are twofold. On the one hand T-linkage reaches a lower ME, thereby obtaining a more refined clustering. On the other hand, the soft threshold parameter τ is much less critical compared to the inlier threshold ϵ : the error function for T-linkage presents a larger plateau where for a large interval of τ T-linkage obtains values near the optimum.

The superior performance of T-linkage is due to the more

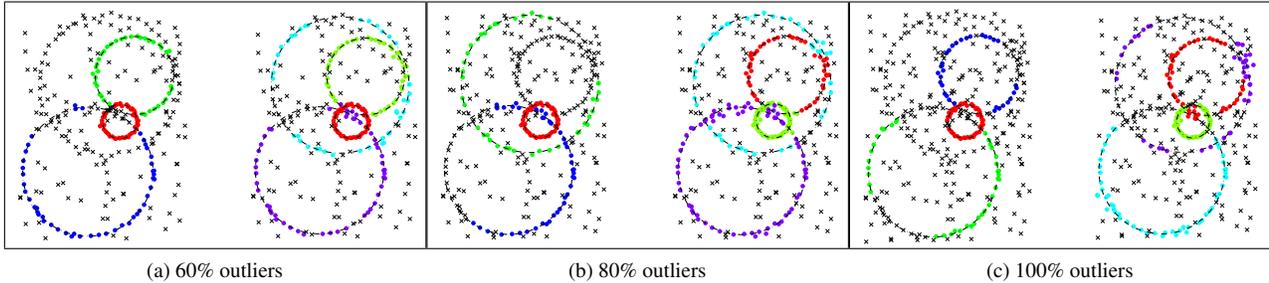


Figure 5: Circle fitting: comparison between J-Linkage (left) and T-linkage (right) for different levels of outliers' contamination. Note that J-Linkage is not able to find all the models, whereas T-linkage always estimates the correct models. This figure is best viewed in color.

expressive representation provided by the continuous conceptual space of points in proximity of models intersections, with respect to the binary classification of inliers adopted by J-Linkage.

We also compare J-Linkage and T-linkage on data contaminated by gross outliers. For this purpose we address the problem of circle fitting taking into account different levels of outliers as shown in Fig. 5. Even if J-Linkage is – in general – very robust to outliers, in some challenging cases, as the one considered here, it fails in finding the correct estimate of models. In particular it finds only three of the four circles present in the ground truth. On the contrary T-linkage can successfully handle the outliers finding all the models and giving rise to a more accurate segmentation as corroborated by the ME reported in Fig. 4.

4. Experimental results with real data

In this section we aim at benchmarking T-linkage on some real datasets that have been used in the literature. In particular, we refer to the two-views experiments reported in [25], dealing with fitting multiple homographies and multiple fundamental matrices, and to the image sequences used in [4], dealing with segmenting multiple motions in video.

4.1. Two-views segmentation

The datasets used in these experiments consist of matching points in two uncalibrated images with gross outliers. In the first case (*plane segmentation*) the (static) scene contains several planes, each giving rise to a set of point correspondences described by a specific homography. The aim is to segment different planes by fitting homographies to subsets of corresponding points.

In the second case (*motion segmentation*) the setup is similar, but the scene is not static, i.e., it contains several objects moving independently each giving rise to a set of point correspondences described by a specific fundamental matrix. The aim is to segment the different motions by fitting

fundamental matrices to subsets of corresponding points.

Using the same data as [10] we are able to compare directly to SA-RCM [10] and indirectly to ARJMC [9], PEARL [7], QP-MF [28], and FLOSS [8]. The figures regarding all methods but T-linkage are taken from [10]. All methods, including T-linkage and J-Linkage, have been tuned separately on each dataset for best performance.

	PEARL	QP-MF	FLOSS	ARJMC	SA-RCM	J-Lnkg	T-Lnkg
johnsona	4.02	18.5	4.16	6.48	5.90	5.07	4.02
johnsonb	18.18	24.65	18.18	21.49	17.95	18.33	18.33
ladysymon	5.49	18.14	5.91	5.91	7.17	9.25	5.06
neem	5.39	31.95	5.39	8.81	5.81	3.73	3.73
oldclassicswing	1.58	13.72	1.85	1.85	2.11	0.27	0.26
sene	0.80	14	0.80	0.80	0.80	0.84	0.40
Mean	5.91	20.16	6.05	7.56	6.62	6.25	5.30
Median	4.71	18.32	4.78	6.20	5.86	4.40	3.87

Table 2: Misclassification error (%) for two-view **plane** segmentation.

	PEARL	QP-MF	FLOSS	ARJMC	SA-RCM	J-Lnkg	T-Lnkg
carsturning	13.08	9.72	19.07	15.53	10.93	17.32	16.68
carsbus	9.36	11.17	18.44	12.81	8.03	19.54	17.97
biscuitbookbox	4.25	9.27	8.88	8.49	7.04	1.55	1.54
breadcartoychips	5.91	10.55	11.81	10.97	4.81	11.26	3.37
breadcubechips	4.78	9.13	10.00	7.83	7.85	3.04	0.86
breadtoycar	6.63	11.45	10.84	9.64	3.82	5.49	4.21
carchipscube	11.82	7.58	11.52	11.82	11.75	4.27	1.81
cubebreadtoychips	4.89	9.79	11.47	6.42	5.93	3.97	3.05
dinobooks	14.72	19.44	17.64	18.61	8.03	17.11	9.44
toycubecar	9.5	12.5	11.25	15.5	7.32	5.05	3.03
Mean	8.49	11.06	13.09	11.76	7.55	8.86	6.20
Median	7.99	10.17	11.49	11.39	7.58	5.27	3.21

Table 3: Misclassification error (%) for two-view **motion** segmentation.

As far as the plane segmentation is concerned, we used DLT for hypothesis generation and Sampson distance for computing the residual. The results are reported in Table 2 in which for every sequence the corresponding misclassification error obtained by each method is indicated. As can be

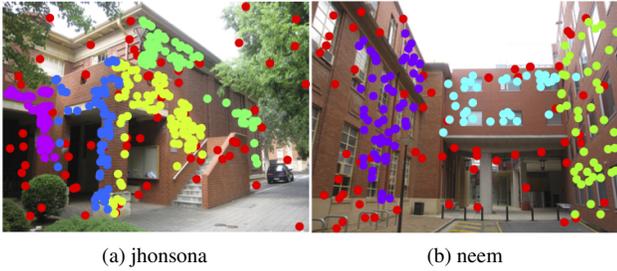


Figure 6: Sample T-linkage results in two-view **plane segmentation** (point membership is color coded, points marked in red are rejected as outliers)

appreciated, T-linkage is very accurate, achieving the best ME in 5 out of 6 cases and attaining the lowest average and median ME.

In Fig. 6 we present a two samples results obtained by T-Linkage. In particular, in Fig. 6a we can observe that the points on the stair wall are rejected as outliers even if they belong to a plane. This is due to our outliers rejection scheme that consider as outliers clusters with cardinality close to minimum sample set.

For the two-view motion segmentation problem we consider the AdelaideRMF [25] dataset and additional sequences from the Hopkins 155 [24], as in [10]. Fundamental matrices were computed with the 8-point algorithm and residuals using the Sampson distance. The results are reported in Table 3 in which for every image pair the corresponding misclassification error obtained by each method is indicated. As can be seen, T-linkage scores the best ME in 6 out of 10 cases and achieves the lowest average and median ME.

These experiments also confirm that T-Linkage perform at least as well as J-linkage, gaining some additional benefits in terms of accuracy from the soft representation of data.

In Fig. 7 we present some of the results obtained by T-Linkage. The reader might notice how in Fig. 7b some points of the background, classified as outliers according to the ground-truth, are recognized as inliers because they happen to lie near the fundamental manifold of a motion in the scene.

4.2. Motion segmentation in video sequences

In motion segmentation the input data consist in a set of features trajectories across a video taken by a moving camera, and the problem consist in recovering the different rigid-body motions contained in the dynamic scene.

Motion segmentation can be seen as a subspace segmentation problem under the modeling assumption of affine cameras. In fact under affine projection it is simple to demonstrate that all feature trajectories associated with a single moving object lie in a linear subspace of dimension at most

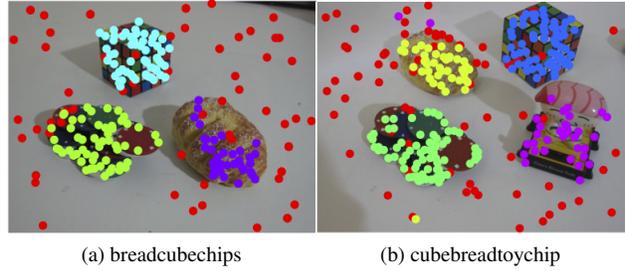


Figure 7: Sample T-linkage results in two-view **motion segmentation** (point membership is color coded, points marked in red are rejected as outliers)

4 in \mathbb{R}^{2F} (where F is the number of video frames). For this reason feature trajectories of a dynamic scene containing n rigid motion lie in the union of n low dimensional subspace of \mathbb{R}^{2F} and segmentation can be reduced to clustering data points in a union of subspaces.

We evaluate T-linkage on the Hopkins 155 motion dataset [24], which is available online at <http://www.vision.jhu.edu/data/hopkins155>. The dataset consists of 155 sequences of two and three motions, divided into three categories: checkerboard, traffic, and other (articulated/non-rigid) sequences. The trajectories are inherently corrupted by noise, but no outliers are present.

We compare T-linkage with RANSAC, J-Linkage and with algorithms tailored to subspace clustering: SSC [4] LSA [27] and ALC [12]. The average and median misclassification errors are listed in Tables 4 and 5 (the figures regarding all methods but T-linkage are taken from the site mentioned above). All methods have been tuned separately on each dataset for best performance.

In the two-motion sequences the results of T-linkage are mixed, however it always achieves a zero median error (as SSC does) and in one case (*Traffic*) also the best average error. The overall average is the second best after SSC, and fairly close to it.

On the three-motion sequences, the results of T-linkage are worse than in the other sequences, and are also somehow odd: on the traffic sequence it achieves the lowest ME, but on *Checkerboard* and *Others* it scores only third (it is second, however in the mean and median ME).

In the attempt to explain the incongruous results of T-linkage on these sequences, we plot in Fig.9, for each data point, the ratio of the distance to its “second” closest model over the distance to its ground truth model, in order to get an indication of the separation between the models. As a matter of fact, there seems to be a positive correlation between incidence of ambiguous points and increase in ME, which makes a lot of sense. Surprisingly, it turns out that in some cases the ratio is greater than one, meaning that the “second”

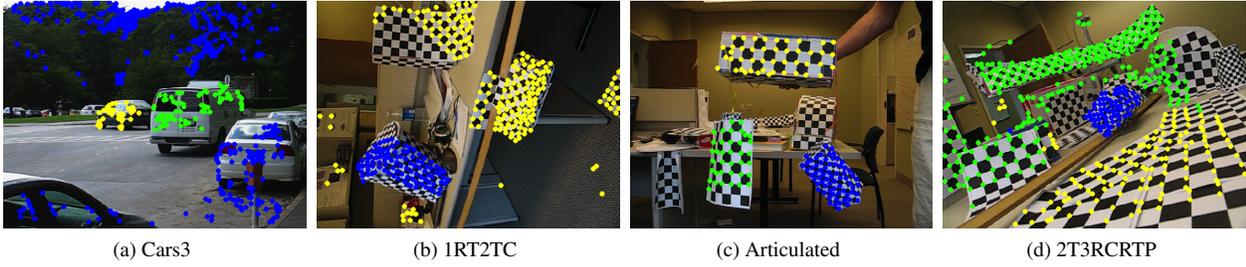


Figure 8: Sample T-linkage results for **motion segmentation** in video sequences (points membership is color coded)

		Ransac	LSA 4n	ALC 5	ALC sp	SSC	J-Lnkg	T-Lnkg
<i>Checkerboard</i>	mean	6.52	2.57	2.56	1.49	1.12	1.73	1.20
	median	1.75	0.72	0.00	0.27	0.00	0.00	0.00
<i>Traffic</i>	mean	2.55	5.43	2.83	1.75	0.02	0.70	0.02
	median	0.21	1.48	0.30	1.51	0.00	0.00	0.00
<i>Others</i>	mean	7.25	4.10	6.90	10.70	0.62	3.49	0.82
	median	2.64	1.22	0.89	0.95	0.00	0.00	0.00
<i>All</i>	mean	5.56	3.45	3.03	2.40	0.82	1.62	0.86
	median	1.18	0.59	0.00	0.43	0.00	0.00	0.00

Table 4: **Motion segmentation**: misclassification error (%) for video sequences with two motions

closest model is actually closer than the ground-truth model. The presence of such equivocal points in the dataset might be related to the problems mentioned in a very recent paper [17] which we plan to consider in the forthcoming work.

In Fig. 8 some sequences of the Hopkins 155 dataset are reported with the corresponding segmentation obtained by T-Linkage. Fig. 8a and Fig. 8c show correctly segmented scenes whereas T-Linkage fails in finding the correct number of motions in Fig. 8b (undersegmentation) and in Fig. 8d the background is split into two clusters one of which also includes the moving object in the top of the image.

Empirical results suggest that computation time is linear in the number of points as shown in Fig. 10.

5. Conclusion

We introduced a variant of J-linkage based on a continuous conceptual representation of data points, which copes better with intersecting structures and alleviates the critical dependence from the inlier threshold. Together with our outlier rejection strategy it performed better, on the average, than competing methods on two-views segmentation problems. On video sequences results are less conspicuous but still T-linkage is the second best method overall and sometimes it scores first. The MATLAB code is available on the web from the authors' web pages.

		Ransac	LSA 4n	ALC 5	ALC sp	SSC	J-Lnkg	T-Lnkg
<i>Checkerboard</i>	mean	25.78	5.80	6.78	5.00	2.97	8.55	7.05
	median	26.02	1.77	0.92	0.66	0.27	4.38	2.46
<i>Traffic</i>	mean	12.83	25.07	4.01	8.86	0.58	0.97	0.48
	median	11.45	23.79	1.35	0.51	0.00	0.00	0.00
<i>Others</i>	mean	21.38	7.25	7.25	21.08	1.42	9.04	7.97
	median	21.38	7.25	7.25	21.08	0.00	9.04	7.97
<i>All</i>	mean	22.94	9.73	6.26	6.69	2.45	7.06	5.78
	median	22.03	2.33	1.02	0.67	0.20	0.73	0.58

Table 5: **Motion segmentation**: misclassification error (%) for video sequences with three motions

References

- [1] T. Chin, H. Wang, and D. Suter. Robust fitting of multiple structures: The statistical learning approach. In *International Conference on Computer Vision*, pages 413–420, 2009. **1**
- [2] T.-J. Chin, D. Suter, and H. Wang. Multi-structure model selection via kernel optimisation. In *Conference on Computer Vision and Pattern Recognition*, pages 3586–3593, 2010. **1**
- [3] A. Delong, O. Veksler, and Y. Boykov. Fast fusion moves for multi-model estimation. In *European Conference on Computer Vision*, pages 370–384, 2012. **2**
- [4] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Conference on Computer Vision and Pattern Recognition*, 2009. **2, 5, 6**
- [5] D. F. Fouhey, D. Scharstein, and A. J. Briggs. Multiple plane detection in image pairs using j-linkage. In *International Conference on Pattern Recognition*, pages 336–339, 2010. **1**
- [6] M. Fradet, P. Robert, and P. Pérez. Clustering point trajectories with various life-spans. In *Conference for Visual Media Production*, pages 7–14, 2009. **1**
- [7] H. Isack and Y. Boykov. Energy-based geometric multi-model fitting. *International Journal of Computer Vision*, 97(2):123–147, 2012. **2, 5**
- [8] N. Lazic, I. E. Givoni, B. J. Frey, and P. Aarabi. FLoSS: Facility location for subspace segmentation. In *International Conference on Computer Vision*, pages 825–832, 2009. **1, 5**
- [9] T.-T. Pham, T.-J. Chin, J. Yu, and D. Suter. Simultaneous sampling and multi-structure fitting with adaptive reversible jump mcmc. In *Neural Information Processing Systems Foundation*, pages 540–548, 2011. **2, 5**
- [10] T.-T. Pham, T.-J. Chin, J. Yu, and D. Suter. The random cluster model for robust geometric fitting. In *Conference on*

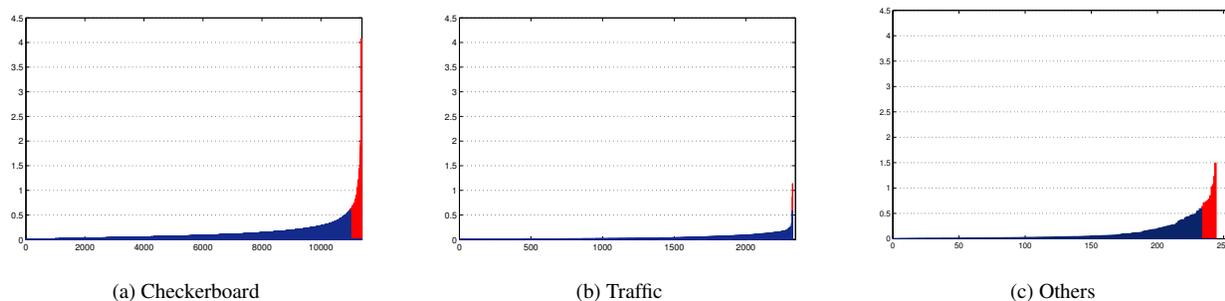


Figure 9: Ratio between distance to the second-closest and the ground-truth model of a point. The values higher than 0.6 are highlighted in red: they correspond to points whose assignment to the correct ground-truth model could be challenging. Values higher than 1.0 are definitely equivocal.

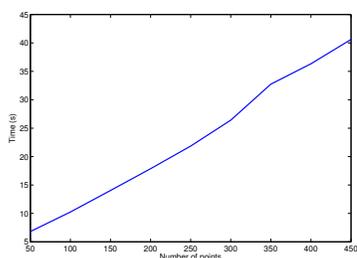


Figure 10: Computation time for the *IR2RC* sequence as a function of number of points on a Intel Core i7 2.6 GH.

- Computer Vision and Pattern Recognition*, pages 710–717, 2012. [2](#), [5](#), [6](#)
- [11] R. Raguram and J.-M. Frahm. RECON: Scale-adaptive robust estimation via residual consensus. In *International Conference on Computer Vision*, pages 1299–1306, 2011. [1](#)
- [12] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *Pattern Analysis and Machine Intelligence*, 32(10):1832–1845, 2010. [2](#), [6](#)
- [13] K. Schindler, D. Suter, and H. Wang. A model-selection framework for multibody structure-and-motion of image sequences. *International Journal of Computer Vision*, 79(2):159–177, August 2008. [2](#)
- [14] C. V. Stewart. MINPRAN: A new robust estimator for computer vision. *Pattern Analysis and Machine Intelligence*, 17(10):925–938, 1995. [3](#)
- [15] C. V. Stewart. Bias in robust estimation caused by discontinuities and multiple structures. *Pattern Analysis and Machine Intelligence*, 19(8):818–833, 1997. [1](#)
- [16] H. Su, M. Sun, L. Fei-Fei, and S. Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *International Conference on Computer Vision*, pages 213–220, 2009. [1](#)
- [17] Y. Sugaya, Y. Matsushita, and K. Kanatani. Removing mis-tracking of multibody motion video database hopkins155. In *British Machine Vision Conference*, 2013. [7](#)
- [18] T. Tanimoto. Technical report, IBM Internal Report, 17th Nov. 1957. [3](#)
- [19] J.-P. Tardif. Non-iterative approach for fast and accurate vanishing point detection. In *International Conference on Computer Vision*, pages 1250–1257, 2009. [1](#)
- [20] R. Toldo and A. Fusiello. Robust multiple structures estimation with J-linkage. In *European Conference on Computer Vision*, pages 537–547, 2008. [1](#), [2](#), [4](#)
- [21] R. Toldo and A. Fusiello. Photo-consistent planar patches from unstructured cloud of points. In *European Conference on Computer Vision*, pages 589–602, 2010. [1](#)
- [22] P. H. S. Torr. An assessment of information criteria for motion model selection. *Conference on Computer Vision and Pattern Recognition*, pages 47–53, 1997. [2](#)
- [23] P. H. S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:2000, 2000. [2](#)
- [24] R. Tron and R. Vidal. A benchmark for the comparison of 3d motion segmentation algorithms. In *Conference on Computer Vision and Pattern Recognition*, 2007. [6](#)
- [25] H. S. Wong, T.-J. Chin, J. Yu, and D. Suter. Dynamic and hierarchical multi-structure geometric model fitting. In *International Conference on Computer Vision*, 2011. [5](#), [6](#)
- [26] L. Xu, E. Oja, and P. Kultanen. A new curve detection method: randomized Hough transform (RHT). *Pattern Recognition Letters*, 11(5):331–338, 1990. [1](#)
- [27] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and nondegenerate. In *European Conference on Computer Vision*, pages 94–106, 2006. [2](#), [6](#)
- [28] J. Yu, T. Chin, and D. Suter. A global optimization approach to robust multi-model fitting. In *Conference on Computer Vision and Pattern Recognition*, 2011. [1](#), [5](#)
- [29] W. Zhang and J. Kosecká. Nonparametric estimation of multiple structures with outliers. In *European Conference on Computer Vision*, volume 4358, pages 60–74, 2006. [1](#)
- [30] M. Zuliani, C. S. Kenney, and B. S. Manjunath. The multi-RANSAC algorithm and its application to detect planar homographies. In *International Conference on Image Processing*, 2005. [1](#)