Multiple Structure Recovery via Probabilistic Biclustering

M. Denitto[1(✉)], L. Magri[1], A. Farinelli[1], A. Fusiello[2], and M. Bicego[1]

[1] Department of Computer Science, University of Verona, Verona, Italy
matteo.denitto@univr.it
[2] DPIA, University of Udine, Udine, Italy

Abstract. Multiple Structure Recovery (MSR) represents an important and challenging problem in the field of Computer Vision and Pattern Recognition. Recent approaches to MSR advocate the use of clustering techniques. In this paper we propose an alternative method which investigates the usage of biclustering in MSR scenario. The main idea behind the use of biclustering approaches to MSR is to isolate subsets of points that behave "coherently" in a subset of models/structures. Specifically, we adopt a recent generative biclustering algorithm and we test the approach on a widely accepted MSR benchmark. The results show that biclustering techniques favorably compares with state-of-the-art clustering methods.

1 Introduction

The extraction of multiple models from noisy or outlier-contaminated data – a.k.a. Multiple Structure Recovery (MSR) – is an important and challenging problem that emerges in many Computer Vision applications [7,10,31]. With respect to single-model estimation in presence of noise and outliers, MRS aims at facing the so called *pseudo*-outliers (i.e. "outliers to the structure of interest but inliers to a different structure" [27]), which push robust estimation to its limit. If, in addition, the number of structures is not known in advance, MSR turns into a thorny model-selection problem, as one have to pick, among all the possible interpretations of the data, the most appropriate one.

In the literature, the problem of MSR has been successfully tackled by leveraging on clustering techniques [13,18,19,25]. Generally, the data matrix to analyze reports the points to cluster on one dimension and the features/descriptors on the other dimension [1]. Clustering approaches group the rows (or the columns) of a given data matrix on the basis of a similarity criterion. For example in these recent approaches *J-linkage*[30], *T-linkage* [18] and *RPA* [19]. The feature vector used to represent data is derived form the preferences expressed by the data points for a pool of tentative structures obtained by random sampling. Hence cluster analysis is performed via either agglomerative or partitional methods where distances measure the (dis)agreement between preferences.

Although it has been shown that clustering provides good solution to the MSR problem, there are situations where the performances of clustering can

A. Robles-Kelly et al. (Eds.): S+SSPR 2016, LNCS 10029, pp. 274–284, 2016.
DOI: 10.1007/978-3-319-49055-7_25

be highly compromised by data matrix structure (e.g. noisy data matrices; or rows behaving similarly only in a small portion of the data matrix). Retrieving information in scenarios where clustering struggles can be done through a recent class of approaches called biclustering.

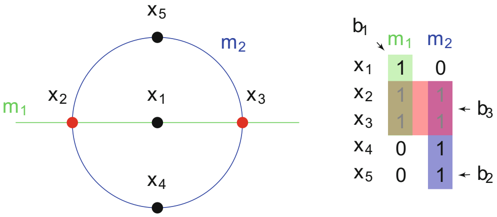


Fig. 1. Example of overlapping biclusters on Multiple Structure Recovery. In this case the possible biclusters are the: b_1) the points lying on m_1, b_2) the points lying on m_2 and b_3) the points lying on the intersections.

With the term *bi-clustering* we refer to a specific category of algorithms performing clustering on both rows and columns of a given data matrix [17]. The goal of biclustering is to isolate sub-matrices where the rows present a "coherent behavior" in a restricted columns subset, and vice-versa. Compared with clustering, biclustering exploits *local* information to retrieve structures that cannot be found performing analysis on whole rows or columns. The problem of biclustering (also known as co-clustering, and strongly related to subspace clustering) is gaining increasing attention in the Pattern Recognition community, with many papers being published in recent years (e.g. [4,8,11,22]). Even if it was originally proposed in biological scenarios (i.e. analysis of gene expression microarray datasets [2,17,20]), biclustering has been widely adopted in many other contexts ranging from market segmentation to data mining [6,14,21].

This paper is positioned exactly in this context, investigating the performance of biclustering on MSR. The main advantage of using biclustering on MSR is that biclustering can isolate portions of the data matrix where a subset of points share similar attitudes in a subset of models. Compared to clustering, biclustering retrieves additional information on which models better describes a particular subset of points. Moreover, biclustering provides better understanding of the data since different biclusters can overlap (on rows, columns or both). This means that a certain point can belong to different biclusters and hence it can be characterized by distinct subsets of models (see Fig. 1 for an example). Thus biclustering approaches can easily deal with intersections, which is a known problematic situation when using clustering algorithms [28].

To the best of our knowledge there exists only a preliminar work applying biclustering techniques to MSR [28]. In [28] authors show that the application of biclustering techniques to MSR is promising and provides superior solution when compared with clustering. While this provides a significant contribution to the state of the art, there is large room for improvements since the method adopted by the authors has some limitations (i.e. it works with sparse binary matrices and it needs some pre-processing/post-processing operations to retrieve the final solutions).

In this paper we investigate the use of a recent probabilistic biclustering approach, namely Factor Analysis for BIcluster Acquisition (FABIA) [12] for MSR. We choose this algorithm because it has been shown to perform better than the state of the art in the Microarray Gene Expression analysis, a widely exploited scenario to test biclustering algorithms [12,17].

We evaluate the performances of probabilistic biclustering on a real benchmark dataset (Adelaide dataset) and we compare against the recent clustering algorithms adopted for MSR. Results confirm that biclustering favorably compares with the state-of-the-art.

The reminder of the paper is organized as follows: Sect. 2 provides a brief review of MSR and the clustering methods we compare to. Section 3 formalizes the biclustering problem and the algorithm we adopt. Section 4 presents and discusses the experimental evaluation. Finally, some concluding remarks are presented in Sect. 5

2 Background and Related Work

This section provides background knowledge about the proposed framework. It formalizes the MSR problem and the clustering approaches currently adopted.

2.1 Multiple Structure Recovery

MSR aims at retrieving parametric models from unstructured data in order to organize and aggregate visual content in significant higher-level geometric structures. This task is commonly found in many Computer Vision applications, a typical example being 3D reconstruction, where MSR is employed either to estimate multiple rigid moving objects (to initialize multibody Structure from Motion [7,23]), or to produce intermediate geometric interpretations of reconstructed 3D point cloud [10,31]. Other instances include face clustering, body-pose estimation and video motion segmentation. In all these scenarios the information of interest can be extracted from the observed data and aggregated in suitable structures by estimating some underlying geometric models, e.g. planar patches, homographies, fundamental matrices or linear subspaces.

More formally, to set a general context, let μ be a model e.g. lines, subspace, homography, fundamental matrices or other geometric primitives and $X = \{x_1, \dots, x_n\}$ be a finite set of n points, possibly corrupted by noise and outlier. The problem of MSR consists in extracting k instances of μ termed structures from the data, defining, at the same time, subsets $C_i \subset X, i = 1, \dots,$ such that all points described by the i-th structures are aggregated in C_i. Often the models considered are parametric, i.e. the structures can be represented as vectors in a proper parameter space.

2.2 Clustering

The extensive landscape of approaches aimed at MSR can be broadly categorized along two mutually orthogonal strategies, namely *consensus analysis* and *preference analysis*. Focusing on preference analysis, these methods reverse the role of data and models: rather than considering models and examining which points match them, the residuals of individual data points are taken into account [3,29,32]. This information is exploited to shift the MSR problem from the ambient space where data lives to a *conceptual* [24] one where it is addressed via cluster analysis techniques.

T-Linkage [18] and RPA [19] can be ascribed to these clustering-based methods as they share the same *first-represent-then-clusterize* approach. Both the algorithms represent data points in a m-dimensional unitary cube as vectors whose components collect the preferences granted to a set of m hypotheses structures instantiated by drawing at random m minimal sample sets – the minimum-sized set of data points necessary to estimate a structure. Preferences are expressed with a soft vote in $[0, 1]$ according to the continuum of residuals in two different fashions. As regards T-linkage, a voting function characterized by an hard cutoff is employed. RPA, instead, exploits the Cauchy weighting function (of the type employed in M-estimators) that has an infinite rejection point mitigating the sensitivity of the inlier threshold.

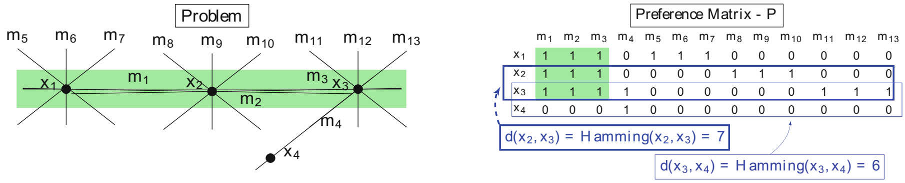
The rationale beyond both these representations is that the agreement between the preferences of two points in this conceptual space reveals the multiple structures hidden in the data: points sharing the same preferences are likely to belong to the same structures.

T-Linkage captures this notion through the Tanimoto distance, which in turn is used to segment the data via a tailored version of average linkage that succeeds in detecting automatically the number of models. If rogue points contaminate the data, outlier structures need to be pruned via ad hoc-post processing techniques.

RPA, on the contrary, requires the number of desired structures as input but inherently caters for gross contamination. At first a kernelized version of the Tanimoto distances is feed to Robust Principal Analysis to remove outlying preferences. Then Symmetric Non Negative Factorization [15] is performed on the low rank part of the kernel to segment the data. Hence, the attained partition is refined in a MSAC framework. More precisely, the consensus of the sampled hypotheses are scrutinized and the structures that, within each segment, support more points are retained as solutions.

While T-linkage can be considered as a pure preference method, RPA attempts to combine also the consensus-side of the MSR problem. However it does not fully reap the benefit of working with both the dimensions of the problem, as biclustering does, for preference and consensus are considered only sequentially.

All these methods can be regarded as processing the *Preference Matrix*, where each entry (i, j) represents the vote granted by the i-th point to the j-th tentative structures. Rows of that matrix provides the representation of points that are used to derive the affinity matrices for clustering. Column of that matrix are the consensus set of a model hypothesis.



	m1	m2	m3	m4	m5	m6	m7	m8	m9	m10	m11	m12	m13
x1	1	1	1	0	1	1	1	0	0	0	0	0	0
x2	1	1	1	0	0	0	0	1	1	1	0	0	0
x3	1	1	1	1	0	0	0	0	0	0	1	1	1
x4	0	0	0	1	0	0	0	0	0	0	0	0	0

$d(x_2, x_3) = \text{Hamming}(x_2, x_3) = 7$

$d(x_3, x_4) = \text{Hamming}(x_3, x_4) = 6$

Fig. 2. Shortfalls of clustering on MRS.

An example where clustering struggles is provided in Fig. 2, which describes a simple MSR problem: to group similar points on the basis of their behavior with respect to the proposed models we should perform clustering on the Preference Matrix P which describes the relationship between the points $\{x_1, x_2, x_3, x_4\}$ and the models $\{m_1, \cdots, m_{13}\}$. Assume we perform clustering adopting the Hamming distance (i.e. number of different bits): since the distance between the x_3 and the x_4 is smaller than the distance between x_3 and x_2, clustering would assign the third and the fourth point to the same group. However looking at the problem diagram it is clear that points x_1, x_2 and x_3 should belong to the same cluster. This information can be retrieved performing a simultaneous clustering of both rows and columns of the Preference Matrix, isolating a subset of models (m_1, m_2 and m_3) where the points x_1, x_2, x_3 share a similar behavior (shaded area in Fig. 2). This is exactly what biclustering techniques do.

Next section provides a more formal definition of the biclustering problem, focusing on the approach adopted to analyze the Preference Matrix (FABIA [12]).

3 Biclustering

As mentioned in Sects. 1 and 2.2 the goal of biclustering applied to MSR is the simultaneous clustering of points and structures/models of a given Preference Matrix, merging the well known concepts of consensus analysis and preference analysis. Due to the similarity of RPA (where the kernelized matrix is factorized in order to obtain point clusters) we present biclustering from a *sparse low-rank matrix factorization* perspective, also the most suitable to understand the insights behind the FABIA algorithm [12].

We denote as $D \in \mathbb{R}^{n \times m}$ the given data matrix, and let $R = \{1, \ldots, n\}$ and $C = \{1, \ldots, m\}$ be the set of row and column indices. We adopt D_{TK}, where $T \subseteq R$ and $K \subseteq C$, to represent the submatrix with the subset of rows in T and the subset of columns in K. Given this notation, we can define a *bicluster* as a submatrix D_{TK}, such that the subset of rows of D with indices in T exhibits a "coherent behavior" (in some sense) across the set of columns with indices in K, and vice versa. The choice of coherence criterion defines the type of biclusters to be retrieved (for a comprehensive survey of biclustering criteria, see [8,17,22]).

A possible coherence criterion for a bicluster (sub-matrix) is for the corresponding entries to have a similar value, significantly different from the other entries of the matrix. For example, a data matrix containing one bicluster with rows $T = \{1, 2, 3, 4\}$ and columns $K = \{1, 2\}$ may look like

$$D = \begin{bmatrix} \mathbf{10} & \mathbf{10} & 0 & 0 \\ \mathbf{10} & \mathbf{10} & 0 & 0 \\ \mathbf{10} & \mathbf{10} & 0 & 0 \\ \mathbf{10} & \mathbf{10} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad v = \begin{bmatrix} \mathbf{5} \\ \mathbf{5} \\ \mathbf{5} \\ \mathbf{5} \\ 0 \\ 0 \end{bmatrix}, \quad z = \begin{bmatrix} \mathbf{2} \\ \mathbf{2} \\ 0 \\ 0 \end{bmatrix}.$$

From an algebraic point of view, this matrix can be represented by the outer product $D = vz^T$ of the sparse vectors v and z. We call these vectors *prototypes* (for v) and *factors* (for z). Generalizing to k biclusters, we can formulate the biclustering problem as the decomposition of the given data matrix D as the sum of k outer products,

$$D = \sum_{i=1}^{k} v_i z_i^T = VZ, \tag{1}$$

where $V = [v_1, \ldots, v_k] \in \mathbb{R}^{n \times k}$ and $Z = [z_1, \ldots, z_k]^T \in \mathbb{R}^{k \times m}$.

The connection between biclustering and sparse low-rank matrix factorization can be highlighted by observing that the factorization of the original data matrix shows that it has rank no larger than the number of biclusters (usually much lower than the number of rows or columns). Moreover, if the size of the matrix D is much bigger than the bicluster size (as it is typically the case in many applications), the resulting prototype and factor vectors should be composed mostly by zeros (*i.e.*, the prototypes and factors should be sparse).

3.1 FABIA

In the biclustering literature, there are several proposals of biclustering methods through matrix factorization (*e.g.*, [16,33]); however, to the best of our knowledge, the only probabilistic approach is FABIA.

FABIA is a generative model for biclustering based on factor analysis [12]. The model proposes to decompose the data matrix by adding noise to the strict low rank decomposition in (1),

$$D = \sum_{i=1}^{k} v_i z_i^T + Y = VZ + Y, \tag{2}$$

where matrix $Y \in \mathbb{R}^{n \times m}$ accounts for random noise or perturbations, assumed to be zero-mean Gaussian with a diagonal covariance matrix. As explained above (Sect. 3) the prototypes in V and factors in Z should be sparse. To induce sparsity, FABIA uses two types of priors: (i) an independent Laplacian prior, and (ii) a prior distribution that is non-zero only in region where prototypes are sparse (for further details, see [12]). The model parameters are estimated using a variational EM algorithm [9,12], for all the details about FABIA derivation and implementation[1], please refer to [12].

4 Experimental Evaluation

This section provides the performances comparison between some clustering methods recently applied to MSR [18,19,25] and the probabilistic biclustering approach presented in Sect. 3.1. The comparison with [28] was not possible since the code is not available. The workflow of the overall procedure can be sketched as follows: starting from an image (i) we generate the hypothesis and compute the Preference Matrix following the guidelines in [19]; (ii) then the probabilistic biclustering technique have been applied.

To assess the quality of the approaches we used the widely adopted *Adelaide* real benchmark dataset[2]. Moreover we conduct a reproducibility analysis, since it is known that RPA algorithm can produce very different solutions due to the random initialization required by the Symmetric NNMF step.

4.1 Adelaide Dataset

We explored the performances of probabilistic biclustering on two type of experiments, namely motion and plane estimation. In motion segmentation experiments, we were provided with two different images of the same scene composed by several objects moving independently; the aim was to recover fundamental matrices to subsets of point matches that undergo the same motion. With respect to the plane segmentation scenario, given two uncalibrated views of a scene, the goal was to retrieve the multi-planar structures by fitting homographies to point correspondences. The AdelaideRMF dataset is composed of 38 image pairs (19 for motion segmentation and 19 for plane segmentation) with matching points contaminated by gross outliers. The ground-truth segmentations are also available. In order to assess the quality of the results, we adopted the misclassification errors, that counts the number of wrong point assignment according to the map between ground-truth labels and estimated ones that minimize the overall number of misclassified points (as in [26]). For fair comparison, the Preference Matrix fed to FABIA was generated relying on the guided sampling scheme presented in [19].

FABIA parameters which regulate the factors/prototypes sparsity and the threshold to retrieve biclusters memberships have been varied in the range suggested by the authors in [12]. The best results on the whole Adelaide dataset (motion and plane estimation) are reported in Table 1. The performances of other methods are taken from [19]; results show that FABIA provides higher quality solutions on the motion segmentation dataset, and on average it performs better on the planar segmentation. Focusing on the motion segmentation dataset, there

[1] Code available from http://www.bioinf.jku.at/software/fabia/fabia.html.

[2] The dataset can be downloaded from https://cs.adelaide.edu.au/~hwong/doku.php?id=data.

Table 1. Misclassification error (ME %) for motion segmentation (left) and planar segmentation (right). k is the number of models and % out is the percentage of outliers.

	k	%out	T-lnkg	RCMSA	RPA	FABIA
biscuitbookbox	3	37.21	**3.10**	16.92	3.88	3.86
breadcartoychips	4	35.20	14.29	25.69	7.50	**4.22**
breadcubechips	3	35.22	3.48	8.12	5.07	**0.87**
breadtoycar	3	34.15	9.15	18.29	7.52	**0.60**
carchipscube	3	36.59	4.27	18.90	6.50	**1.52**
cubebreadtoychips	4	28.03	9.24	13.27	4.99	**1.07**
dinobooks	3	44.54	20.94	23.50	15.14	**9.72**
toycubecar	3	36.36	15.66	13.81	9.43	**9.50**
biscuit	1	57.68	16.93	14.00	1.15	**0**
biscuitbook	2	47.51	3.23	8.41	3.23	**1.32**
boardgame	1	42.48	21.43	19.80	11.65	**8.96**
book	1	44.32	3.24	4.32	2.88	**0**
breadcube	2	32.19	19.31	9.87	**4.58**	19.42
breadtoy	2	37.41	5.40	3.96	**2.76**	19.62
cube	1	69.49	7.80	8.14	3.28	**1.66**
cubetoy	2	41.42	3.77	5.86	4.04	**2.21**
game	1	73.48	1.30	5.07	3.62	**0**
gamebiscuit	2	51.54	9.26	9.37	2.57	**2.44**
cubechips	2	51.62	6.14	7.70	4.57	**0.53**
mean			9.36	12.37	5.49	**4.61**
median			7.80	9.87	4.57	**1.66**

	k	%out	T-lnkg	RCMSA	RPA	FABIA
unionhouse	5	18.78	48.99	2.64	**10.87**	21.54
bonython	1	75.13	11.92	17.79	15.89	**6.82**
physics	1	46.60	29.13	48.87	**0.00**	**0.00**
elderhalla	2	60.75	10.75	29.28	**0.93**	3.04
ladysymon	2	33.48	24.67	39.50	24.67	**11.81**
library	2	56.13	24.53	40.72	31.29	**20.47**
nese	2	30.29	7.05	46.34	**0.83**	4.92
sene	2	44.49	7.63	20.20	**0.42**	2.20
napiera	2	64.73	28.08	31.16	**9.25**	21.85
hartley	2	62.22	21.90	37.78	**17.78**	23.59
oldclassicswing	2	32.23	20.66	21.30	25.25	**7.92**
barrsmith	2	69.79	49.79	**20.14**	36.31	29.88
neem	3	37.83	25.65	41.45	19.86	**11.20**
elderhallb	3	49.80	31.02	35.78	**17.82**	18.63
napierb	3	37.13	**13.50**	29.40	31.22	36.68
johnsona	4	21.25	34.28	36.73	**10.76**	17.96
johnsonb	7	12.02	24.04	16.46	26.76	**24.50**
unihouse	5	18.78	33.13	**2.56**	5.21	15.76
bonhall	6	6.43	21.84	19.69	41.67	24.02
mean			24.66	28.30	17.20	**15.94**
median			23.38	29.40	**17.53**	17.96

are only three situations where FABIA works worse than clustering approaches. A possible explanation on why FABIA struggles could be because general biclustering approaches are tested in scenarios where the number of biclusters is much higher than in MSR (i.e. ∼100 in Gene Expression analysis versus 3–7 in this dataset). To overcome this behavior we run FABIA increasing the number of biclusters to retrieve and aggregating the results on the basis of column overlap as done in [5], this leads to an improvement of the solution quality; results are reported in Table 2.

Table 2. Increasing the number of biclusters improve the results obtained by FABIA on the motion segmentation dataset.

biscuitbookbox	k = 3	3.86
	k = 4	**1.35**
breadcube	k = 2	19.42
	k = 4	11.36
breadtoy	k = 2	19.62
	k = 4	**1.22**

4.2 Reproducibility

In this section we assess the reproducibility of the two methods that better perform on the Adelaide dataset: RPA and FABIA. The goal is to demonstrate that probabilistic approaches can overcome the problem of reproducibility present in RPA. For a fair comparison we adopted the *2R3RTCRT* video sequence from the Hopkins dataset[3]: a dataset where both the approaches retrieve good and similar solutions.

Hopkins dataset is a motion segmentation benchmark where the input data consists 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 clustering problem under the modeling assumption of affine cameras. In fact, under the assumption of 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 4 in $\mathbb{R}^{2F}$ (where F is the number of video frames). Feature trajectories of a dynamic scene containing k rigid motion lie in the union of k low dimensional subspace of $\mathbb{R}^{2F}$ and segmentation can be reduced to clustering data points in a union of subspaces.

To test the reproducibility of RPA and FABIA we run the algorithms on the same Preference Matrix hundred times, and for each trial we assess the misclassification error; the results are reported in Fig. 3. The figure shows the result obtained by the approaches in each iteration and its distance from the average results. Results clearly show that while the two approaches are compatible on average, FABIA retrieves the same solution in each iteration while RPA is much less stable.

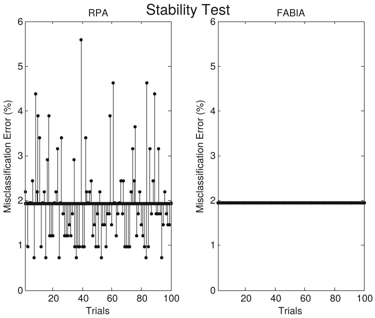


Fig. 3. Reproducibility. The methods have been run 100 times on the same Preference Matrix. Plots show the misclassification error in of each trials along with the distance from the mean (RPA mean = 1.93 %, FABIA mean = 1.95 %).

5 Conclusion and Discussion

In this paper we present an alternative to clustering for the problem of Multiple Structure Recovery (MSR), namely biclustering. In general, biclustering techniques allow to retrieve superior and more accurate information than clustering

[3] The dataset can be downloaded at http://www.vision.jhu.edu/data/hopkins155/.

approaches, characterizing each cluster of points with the subset of features that better describes them. The goal of biclustering approaches applied to MSR is isolate submatrices inside the Preference Matrix where a subset of points behave "coherently" in a subset of models/structures. We tested the recent probabilistic biclustering approach FABIA on the Adelaide benchmark dataset, proving that it favorably compares with the state of the art. Moreover we tested the reproducibility of the analyzed methods showing that FABIA is much more stable than the second competitor RPA.

References

1. Bishop, C.M.: Pattern recognition and Machine Learning. Springer, Heidelberg (2006)
2. Cheng, Y., Church, G.: Biclustering of expression data. In: Proceedings of 8th International Conference on Intelligent Systems for Molecular Biology (ISMB 2000), pp. 93–103 (2000)
3. Chin, T., Wang, H., Suter, D.: Robust fitting of multiple structures: the statistical learning approach. In: International Conference on Computer Vision, pp. 413–420 (2009)
4. Denitto, M., Farinelli, A., Bicego, M.: Biclustering gene expressions using factor graphs and the max-sum algorithm. In: Proceedings of 24th International Conference on Artificial Intelligence, pp. 925–931. AAAI Press (2015)
5. Denitto, M., Farinelli, A., Franco, G., Bicego, M.: A binary factor graph model for biclustering. In: Fränti, P., Brown, G., Loog, M., Escolano, F., Pelillo, M. (eds.) S+SSPR 2014. LNCS, vol. 8621, pp. 394–403. Springer, Heidelberg (2014). doi:10.1007/978-3-662-44415-3_40
6. Dolnicar, S., Kaiser, S., Lazarevski, K., Leisch, F.: Biclustering overcoming data dimensionality problems in market segmentation. J. Travel Res. **51**(1), 41–49 (2012)
7. Fitzgibbon, A.W., Zisserman, A.: Multibody structure and motion: 3-D reconstruction of independently moving objects. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1842, pp. 891–906. Springer, Heidelberg (2000). doi:10.1007/3-540-45054-8_58
8. Flores, J.L., Inza, I., Larrañaga, P., Calvo, B.: A new measure for gene expression biclustering based on non-parametric correlation. Comput. Methods Prog. Biomed. **112**(3), 367–397 (2013)
9. Girolami, M.: A variational method for learning sparse and overcomplete representations. Neural Comput. **13**(11), 2517–2532 (2001)
10. Häne, C., Zach, C., Zeisl, B., Pollefeys, M.: A patch prior for dense 3D reconstruction in man-made environments. In: 2012 2nd International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), pp. 563–570. IEEE (2012)
11. Henriques, R., Antunes, C., Madeira, S.C.: A structured view on pattern mining-based biclustering. Pattern Recogn. **48**(12), 3941–3958 (2015)
12. Hochreiter, S., Bodenhofer, U., Heusel, M., Mayr, A., Mitterecker, A., Kasim, A., Khamiakova, T., Van Sanden, S., Lin, D., Talloen, W., et al.: Fabia: factor analysis for bicluster acquisition. Bioinformatics **26**(12), 1520–1527 (2010)
13. Jain, S., Govindu, V.M.: Efficient higher-order clustering on the Grassmann manifold (2013)

14. Kaytoue, M., Codocedo, V., Buzmakov, A., Baixeries, J., Kuznetsov, S.O., Napoli, A.: Pattern structures and concept lattices for data mining and knowledge processing. In: Bifet, A., May, M., Zadrozny, B., Gavalda, R., Pedreschi, D., Bonchi, F., Cardoso, J., Spiliopoulou, M. (eds.) ECML PKDD 2015. LNCS (LNAI), vol. 9286, pp. 227–231. Springer, Heidelberg (2015). doi:10.1007/978-3-319-23461-8_19
15. Kuang, D., Yun, S., Park, H.: SymNMF: nonnegative low-rank approximation of a similarity matrix for graph clustering. J. Glob. Optim. 1–30 (2014)
16. Lee, M., Shen, H., Huang, J.Z., Marron, J.: Biclustering via sparse singular value decomposition. Biometrics **66**(4), 1087–1095 (2010)
17. Madeira, S., Oliveira, A.: Biclustering algorithms for biological data analysis: a survey. IEEE Trans. Comput. Biol. Bioinform. **1**, 24–44 (2004)
18. Magri, L., Fusiello, A.: T-linkage: a continuous relaxation of J-linkage for multi-model fitting. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 3954–3961 (2014)
19. Magri, L., Fusiello, A.: Robust multiple model fitting with preference analysis and low-rank approximation. In: Xie, X., Tam, Jones, M.W., Tqam, G.K.L. (eds.) Proceedings of British Machine Vision Conference (BMVC), pp. 20.1–20.12. BMVA Press, September 2015
20. Mitra, S., Banka, H., Pal, S.K.: A MOE framework for biclustering of microarray data. In: 18th International Conference on Pattern Recognition, 2006, ICPR 2006, vol. 1, pp. 1154–1157. IEEE (2006)
21. Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., Coello, C.A.C.: Survey of multiobjective evolutionary algorithms for data mining: Part II. IEEE Trans. Evol. Comput. **18**(1), 20–35 (2014)
22. Oghabian, A., Kilpinen, S., Hautaniemi, S., Czeizler, E.: Biclustering methods: biological relevance and application in gene expression analysis. PloS ONE **9**(3), e90801 (2014)
23. Ozden, K.E., Schindler, K., Van Gool, L.: Multibody structure-from-motion in practice. IEEE Trans. Pattern Anal. Mach. Intell. **32**(6), 1134–1141 (2010)
24. Pekalska, E., Duin, R.P.: The Dissimilarity Representation for Pattern Recognition: Foundations And Applications (Machine Perception and Artificial Intelligence). World Scientific Publishing, Singapore (2005)
25. Pham, T.T., Chin, T.J., Yu, J., Suter, D.: The random cluster model for robust geometric fitting. Pattern Anal. Mach. Intell. **36**(8), 1658–1671 (2014)
26. Soltanolkotabi, M., Elhamifar, E., Candès, E.J.: Robust subspace clustering. Ann. Stat. **42**(2), 669–699 (2014)
27. Stewart, C.V.: Bias in robust estimation caused by discontinuities and multiple structures. IEEE Trans. Pattern Anal. Mach. Intell. **19**(8), 818–833 (1997)
28. Tepper, M., Sapiro, G.: A biclustering framework for consensus problems. SIAM J. Imaging Sci. **7**(4), 2488–2525 (2014)
29. Toldo, R., Fusiello, A.: Robust multiple structures estimation with J-linkage. In: European Conference on Computer Vision (2008)
30. Toldo, R., Fusiello, A.: Robust multiple structures estimation with J-linkage. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008. LNCS, vol. 5302, pp. 537–547. Springer, Heidelberg (2008). doi:10.1007/978-3-540-88682-2_41
31. Toldo, R., Fusiello, A.: Image-consistent patches from unstructured points with J-linkage. Image Vis. Comput. **31**(10), 756–770 (2013)
32. Zhang, W., Kosecká, J.: Nonparametric estimation of multiple structures with outliers. In: Vidal, R., Heyden, A., Ma, Y. (eds.) WDV 2005-2006. LNCS, vol. 4358, pp. 60–74. Springer, Heidelberg (2007). doi:10.1007/978-3-540-70932-9_5
33. Zhang, Z.Y., Li, T., Ding, C., Ren, X.W., Zhang, X.S.: Binary matrix factorization for analyzing gene expression data. Data Min. Knowl. Disc. **20**(1), 28–52 (2010)