J Math Imaging Vis (2011) 39: 140–160
DOI 10.1007/s10851-010-0229-z

Rank Estimation in Missing Data Matrix Problems

Carme Julià · Angel D. Sappa · Felipe Lumbreras · Joan Serrat · Antonio López

Published online: 23 October 2010

Abstract A novel technique for missing data matrix rank estimation is presented. It is focused on matrices of trajectories, where every element of the matrix corresponds to an image coordinate from a feature point of a rigid moving object at a given frame; missing data are represented as empty entries. The objective of the proposed approach is to estimate the rank of a missing data matrix in order to fill in empty entries with some matrix completion method, without using or assuming neither the number of objects contained in the scene nor the kind of their motion. The key point of the proposed technique consists in studying the frequency behaviour of the individual trajectories, which are seen as 1D signals. The main assumption is that due to the rigidity of the moving objects, the frequency content of the trajectories will be similar after filling in their missing entries. The proposed rank estimation approach can be used in different computer vision problems, where the rank of a missing data matrix needs to be estimated. Experimental results with synthetic and real data are provided in order to empirically show the good performance of the proposed approach.

Keywords Rank estimation · Missing data · Matrix completion techniques

C. Julià (✉)
Intelligent Robotics and Computer Vision Group,
Dept. of Computer Science and Mathematics, ETSE-URV,
Av. Països Catalans 26, 43007, Tarragona, Spain
e-mail: carme.julia@urv.cat

A.D. Sappa
Computer Vision Center, UAB, Edifici O Campus UAB,
08193 Bellaterra, Barcelona, Spain
e-mail: asappa@cvc.uab.es

F. Lumbreras · J. Serrat · A. López
Computer Vision Center, Computer Science Department, UAB,
Edifici O Campus UAB, 08193 Bellaterra, Barcelona, Spain

F. Lumbreras
e-mail: felipe@cvc.uab.es

J. Serrat
e-mail: joans@cvc.uab.es

A. López
e-mail: antonio@cvc.uab.es

1 Introduction

Several problems can be reduced to find a low-rank matrix approximation to the given data matrix W (e.g., structure from motion [29], optical flow estimation [17], photometric stereo [16], structure from sound [28], data mining [31]). In general, the given matrix is assumed full and considered as the basic representation of the inputs. Then, according to the problem, different processing algorithms could be applied: singular value decomposition of W for obtaining the shape and motion in the SFM problems [26]; clustering of trajectories—i.e., columns of W—in the multiple object segmentation problem [9]. Unfortunately, in real-life situations, missing or incomplete data are unavoidable, for instance: variable out of range, noisy data, feature mismatch, occluded data, are the common reasons for such missing inputs and neglected empty entries. Therefore, most of the aforementioned algorithms cannot be directly applied to W.

The easiest way to deal with missing data is to restrict the process to the completely observed inputs. Another simple option is to assume a fixed user defined value for those missed entries in the matrix—for instance, row or column average [27, 31]. A more elaborated alternative is to use matrix completion methods, which fill in missing entries by suitable estimates (e.g., [8, 18, 19]). In low-dimensional problems, it is better to use matrix completion methods that take into account the fact that the data matrix has a reduced rank. For instance, there exist several problems in the computer vision framework where data are stacked into a low-rank matrix of trajectories $W_{2f\times n}$, where f and n are the number of frames and feature points respectively (from now on it will be referred as W). The factorization technique [29] can be used to decompose this matrix W into the product of two matrices $A_{2f\times r}$ and $B_{r\times n}$, where r is the rank of W. The product AB gives the best rank-r approximation to W. The *Alternation* technique [35] is a factorization technique commonly used when there are missing data in W (e.g., [6, 12, 15, 25]). This iterative algorithm starts with an initial A_0 or B_0 random factor and, at each iteration k, computes alternatively each of the factors A_k and B_k, until the product $A_k B_k$ converges to the known values in W. This product is used to fill in missing entries in W. The main drawback of the Alternation technique is that it depends on the initialization and it converges to a local minimum when there are missing data in the matrix W.

More recently, a spectrally optimal factorization method has been proposed in [3]. It is based on [1] and gives a global solution for a particular pattern of missing data known as Young diagram. Unfortunately, in the structure from motion problem, the pattern of the missing data consists in two or more Young diagrams. Hence, in this case the solution proposed by [3] is sub-optimal. Candes et al. [7] present a matrix completion method that uses convex optimization. They propose to minimize the *nuclear norm* of the matrix W, which is the sum of their singular values, instead of minimizing the rank of W. Haldar et al. [13] point out that, although nuclear norm minimization (NNM) is effective, it can be computationally demanding. They propose an incremented-rank version of the PowerFactorization [15] (IRPF) to fill in missing entries in W. Reported results in [13] empirically show that the performance of the IRPF at filling in low-rank matrices is better than the NNM.

The matrix completion techniques, although appropriate to generate a full matrix by filling in missing data, need some additional prior knowledge of the problem to avoid wrong results. Concretely, a proper matrix rank assumption would help to find the right solution. Focussing on this problem, in the current paper a novel approach for rank estimation is presented, which could be used later on for filling in missing entries by means of a matrix completion method. A preliminary version of this rank estimation technique was introduced in [20] for the motion segmentation problem. It is expected that other applications based on low-rank data sets analysis, where missing data are often found, could benefit from the proposed approach (e.g., [5, 24]).

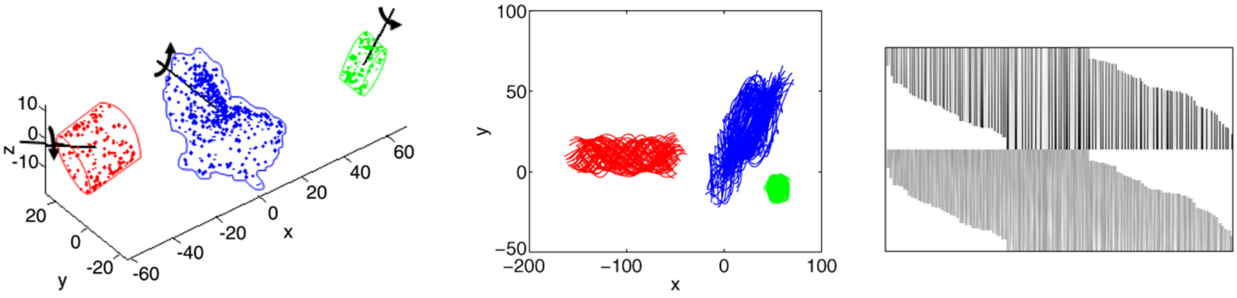
The motivation to define the rank estimation technique [20] was that, although several techniques have been proposed for the multiple object motion segmentation problem (e.g., [14, 22, 33]), most of them assume a full matrix of trajectories W. In those cases, the rank of the matrix W can be estimated by examining the ratio between its singular values (e.g., [37]). However, trajectories are often incomplete or split due to objects occlusions, errors on the tracking or simply because they are not more in the camera field of view. Unfortunately, when the matrix of trajectories contains missing data, the rank cannot be directly computed. Recall that the rank of the matrix W depends on both the number of objects and their motion. In [32], and more recently in [34], Vidal et al. propose an approach able to deal with missing data in the multiple object case. Their approach consists in fixing the rank of W to five and then applying a factorization technique to project the point trajectories from $\mathbb{R}^{2f}$ to $\mathbb{R}^5$, where f is the number of frames. That projection is proposed by using the fact that the maximum dimension of each motion subspace is four, in the case of a rigid object. Then, projecting onto a generic subspace of dimension five will preserve the number and dimension of motion subspaces. This assumption could give poorly recovered missing entries, when the number of different objects and motions increases. Experimental results provided in [20] show that the error obtained in the missing entries recovering can affect the motion segmentation results.

In the current paper a novel approach to estimate the rank is proposed, instead of defining it beforehand. The proposed approach estimates the rank without using neither the number of rigid objects nor the kind of motion. It is based on the fact that the frequency spectra of the motion, treated as a signal and depicted as columns in the input matrix W, should be preserved after recovering missing entries. Behind this idea there is the assumption that, due to the rigidity of the moving objects, the behavior of missing and known data, studied as trajectories (columns of W), is the same; therefore, both generate a similar spectral representation. Once the rank has been estimated and used for obtaining a full matrix through factorization, applications such as motion segmentation could be performed using classical approaches [36].

The remainder of this paper is organized as follows. Section 2 introduces the proposed technique. Experimental results, using scenes containing different numbers of objects, are presented in Sect. 3 in order to empirically show the performance of the proposed rank estimation technique. Finally, concluding remarks are summarized in Sect. 4.

2 Missing Data Matrix Rank Estimation

As mentioned in Sect. 1, matrix completion methods can be used to fill in missing entries in the particular context of computer vision. Next, the Alternation technique [35], which has been widely used in the computer vision framework (e.g., [6, 12, 15]), is briefly presented since it is used as a matrix completion method in the current approach. The proposed rank estimation technique is also valid with other matrix completion methods, as it is shown in Sect. 3.3.



(a) (b) (c)

Fig. 1 (**a**) Scene defined by three synthetic objects (two cylinders and a sculptured surface). (**b**) Feature point trajectories plotted in the image plane. (**c**) Trajectory matrix with 30% of missing data ($W_{30\%missing}$)

2.1 Alternation Technique

Given a matrix $W_{m\times n}$ with rank $r < \min\{m, n\}$, the aim is to find $A_{m\times r}$ and $B_{r\times n}$ such that minimize the following expression:

$$\|W - AB\|_F^2 \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius matrix norm [11]. Working with missing data, the expression to minimize is:

$$\|W - AB\|_F^2 = \sum_{i,j} |W_{ij} - (AB)_{ij}|^2 \tag{2}$$

where i and j correspond to the index pairs where W_{ij} is defined.

The Alternation technique uses the fact that if one of the factors (A or B) is known, the other factor can be computed by solving the least squares problem (2). Therefore, the two factors are computed alternately, until the product AB converges to W. Hence, by multiplying the resulting matrices A and B, a full matrix, which is the best rank-r approximation to W, is obtained.

In the case of a single rigid moving object, and under affine camera model, the rank of the matrix of trajectories W, is at most 4 (see [29], for details). However, when this idea is extended to the multiple objects case, the rank of matrix W is not bounded when the number of objects in the scene and the kind of motion are unknown. Furthermore, it is not easy to estimate due to the missing entries. Therefore, matrix rank estimation becomes a chicken-egg problem, since the rank value is needed for recovering missing entries and computing an estimation of the full matrix. Moreover, it should be noticed that although several ranks could be tested for filling in missing entries, there is not a direct way to measure the *goodness* of recovered data in order to define which is the most appropriated rank value. The next section presents a brief study of results that would be obtained if only known entries in W were used for computing its rank.

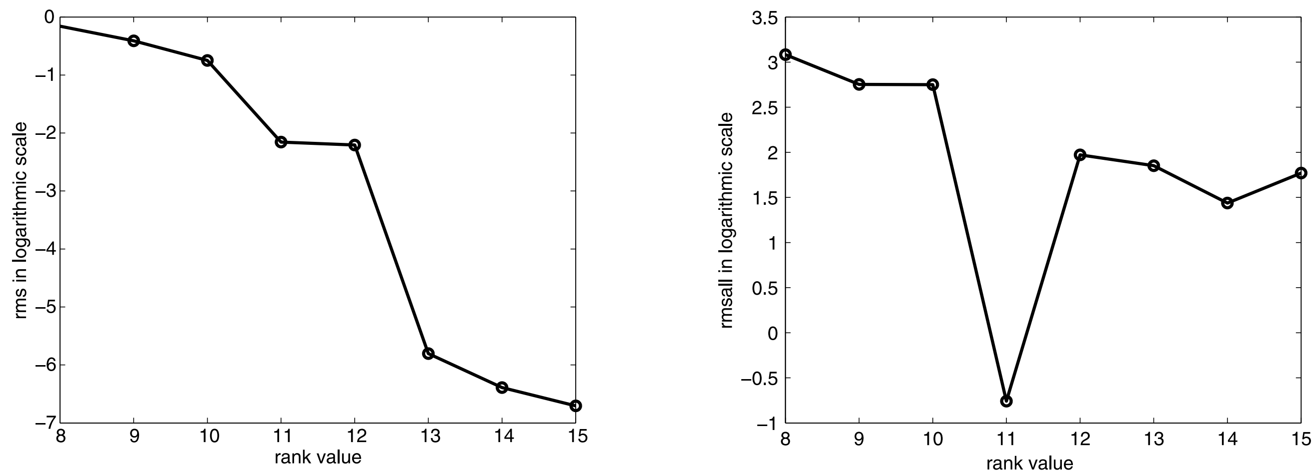
2.2 Partial Information versus Full Information

This section shows the importance of using some kind of information about missing entries to estimate the correct rank of W. Several tests were considered obtaining similar conclusions (i.e., different percentages of missing data, trajectory matrices defined by the motion of different number of objects, different amount of noise, etc.).

The study presented in this section is performed by using a particular data set corresponding to a synthetic scene, defined by three rigid objects moving independently. Feature points are distributed over the surface of the objects and tracked through several frames. A full matrix W_{full} is generated and then a trajectory matrix with a 30% of missing data ($W_{30\%missing}$) is obtained by removing information, simulating the behavior of tracked feature points (more information about experiments set up is given in Sect. 3). Figure 1(a) shows the synthetic scene used through this section; feature point trajectories, in the image plane, are presented in Fig. 1(b). Finally, a trajectory matrix with a 30% of missing data is presented in Fig. 1(c). The half-top of this matrix corresponds to x coordinates of feature points; while half-bottom to y coordinates. White elements correspond to empty entries in the matrix; on the contrary black (or grey) elements in the matrix represent known entries in the matrix.

The first stage is to fill in empty entries by means of a matrix completion method assuming different rank values (in our case the Alternation technique is used). For every assumed rank value a different solution is obtained. This section aims at showing that a criterion based only on known entries will give a wrong rank value estimation. The used criterion consists in selecting the rank value of the matrix with a minimum *root mean square error* (*rms*). That is, in the current example, the aim is to select the rank for which the matrix obtained with the product $A_{2f\times r}B_{r\times p}$ minimizes the following expression:

$$rms = \|W_{30\%missing} - A_{2f\times r}B_{r\times p}\|_F/\sqrt{q} \tag{3}$$



(a) (b)

Fig. 2 (**a**) The *rms* as a function of the rank values for the missing data matrix presented in Fig. 1(c), when only known entries in $W_{30\%missing}$ are considered. Hence, the *rms* measures how good the known entries are approximated. (**b**) The rms_{full} as a function of the rank when all data points in W_{full} are considered. The rms_{full} measures how good the initially known data are approximated as well as how good missing data are recovered

where $2f$ and p are the number of rows and columns in W and q is the number of known entries in $W_{30\%missing}$. On the other hand, since the W_{full} matrix is known, entries filled in during the matrix completion are also compared with the original values by means of:

$$rms_{full} = \|W_{full} - A_{2f\times r} B_{r\times p}\|_F / \sqrt{q} \tag{4}$$

where q is the number of all entries in W_{full}. Notice that rms_{full} can be computed, since in our experiments all the entries in W_{full} are initially known. Hence, two different *rms* values are computed; the first by only taking into account known entries in $W_{30\%missing}$, while the second by considering all the entries in W_{full}.

The result of this comparison is presented in Fig. 2. It can be seen that the trend of these plots is different. Furthermore, the minimum values are obtained at different rank values. In the first case, when only known entries in $W_{30\%missing}$ are used, the best rank estimation corresponds to 15 (Fig. 2(a)); while in the second case, when all entries in W_{full} are used, the best rank estimation is 11 (Fig. 2(b)). Note that the correct rank value is 12, which is quite similar to the value estimated with the rms_{full}; the correct rank value of W_{full} is directly obtained by computing its singular values [11] (Fig. 3 shows the 12th first singular values of W_{full}, in logarithmic scale). In this particular example missing entries are filled in better considering $r = 11$ instead of $r = 12$. This can be due to the random initialization of Alternation.

The previous result can be understood by studying the way missing data in a single trajectory are recovered, after assuming different rank values. Figure 4 presents the trajectory of a given feature point (i.e., a single column in W_{full}),

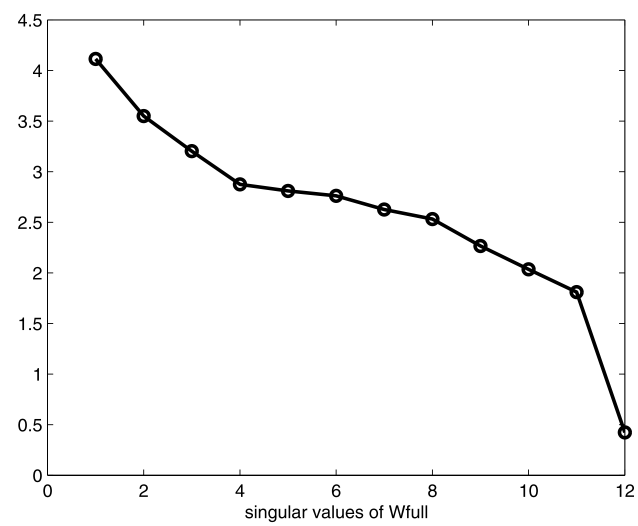


Fig. 3 Singular values of the initially full matrix W_{full} (only the first 12 ones are plotted), in logarithmic scale; notice that the ratio between the 11th and 12th singular values is larger than the rest of all previous ratios

and also the filled in data considering different rank values (concretely, from $r = 9$ up to $r = 13$); a thicker line corresponds to known values in $W_{30\%missing}$, a thinner line shows original values that were removed to generate the missing data matrix and finally, a dashed line corresponds to the filled in entries. An enlargement is presented in the cases of $r = 9, 10, 12$ and 13 in order to have a better visualization due to the high values that take the filled in entries in the column (note that in these cases dashed lines go out the range of plot). As an illustration, the whole filled in trajec-

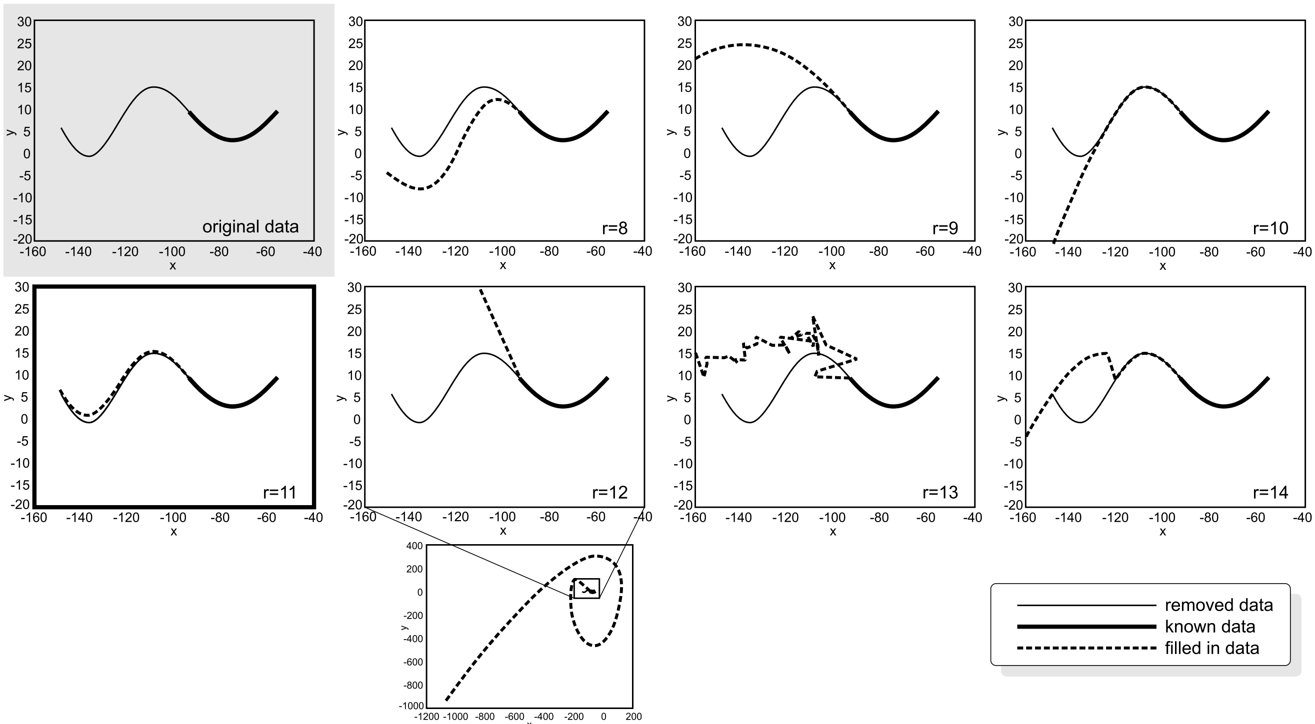


Fig. 4 (*top*)–(*left*) Trajectory of a given feature point (column in W_{full}). (From top to bottom, from left to right) Filled in data (*dashed line*) assuming rank values from 9 up to 13

tory obtained in the case of $r = 12$ is also plotted. The objective of presenting Fig. 4 is to show the way missing entries are filled in by assuming different rank values. It can be appreciated that in all the examples known data are preserved. However, missing data are almost correctly recovered when the assumed rank value is 11, as presented in Fig. 2(b).

2.3 Proposed Approach

Having in mind that the goodness of recovered data cannot be measured, since in general missing entries are not known, an approach based on the study of changes on the input matrix after recovering missing data is proposed. A preliminary version has been presented in [20]. Since the rank of W cannot be computed, different rank values (r) are tested, obtaining a full matrix for each case. Then, by using both the initially known and recovered missing entries in W a novel measure of goodness is introduced. The underlying idea is that, since feature points trajectories belong to surfaces of rigid objects, the motion generated by recovered missing entries should be similar to the one of the initial known entries. This motion similarity is identified with the fact that the frequency spectra of the input matrix W should be preserved after recovering missing data. It is based on an energy and frequency content preservation. In order to study the frequency content of the matrices, the Fast Fourier Transform (*FFT*) is applied to each of the columns of the full matrices (obtained by using different rank values as presented below) and also to those of the input matrix W (adding zeros to its missing entries).[1] Analogous results are obtained with the Discrete Cosine Transform (*DCT*). Since the idea is to study the trajectories along the frames, only columns of the matrices are considered, instead of using rows or two dimensions at the same time. Concretely, the x and y coordinates of the trajectories, which are stacked separately in W, as mentioned in Sect. 2.2, are considered as the real and imaginary part of each column when the *FFT* is applied. In the preliminary approach [20], the x and y coordinates of each trajectory are considered as a 1D signal. The goal of splitting trajectories up into real and imaginary parts is to avoid the artificial step between the x and y coordinates that introduces a ripple in the frequency domain.

In a different approach, Akhter et al. [4] describe the time varying 3D trajectories of a non-rigid object as a linear combination of trajectory bases for which they choose the Discrete Cosine Transform (*DCT*). Concretely, they describe the 3D trajectories by using k vectors of the *DCT* basis. The idea of the approach proposed in the current paper is different: as mentioned above, the 2D trajectories (columns of W) are interpreted as 1D signals and the modulus of the *FFT* of these signals is used to study the goodness of the filled in missing data in these 2D trajectories.

[1] Similar results are obtained by adding the mean of the corresponding column to the missing entries of W.

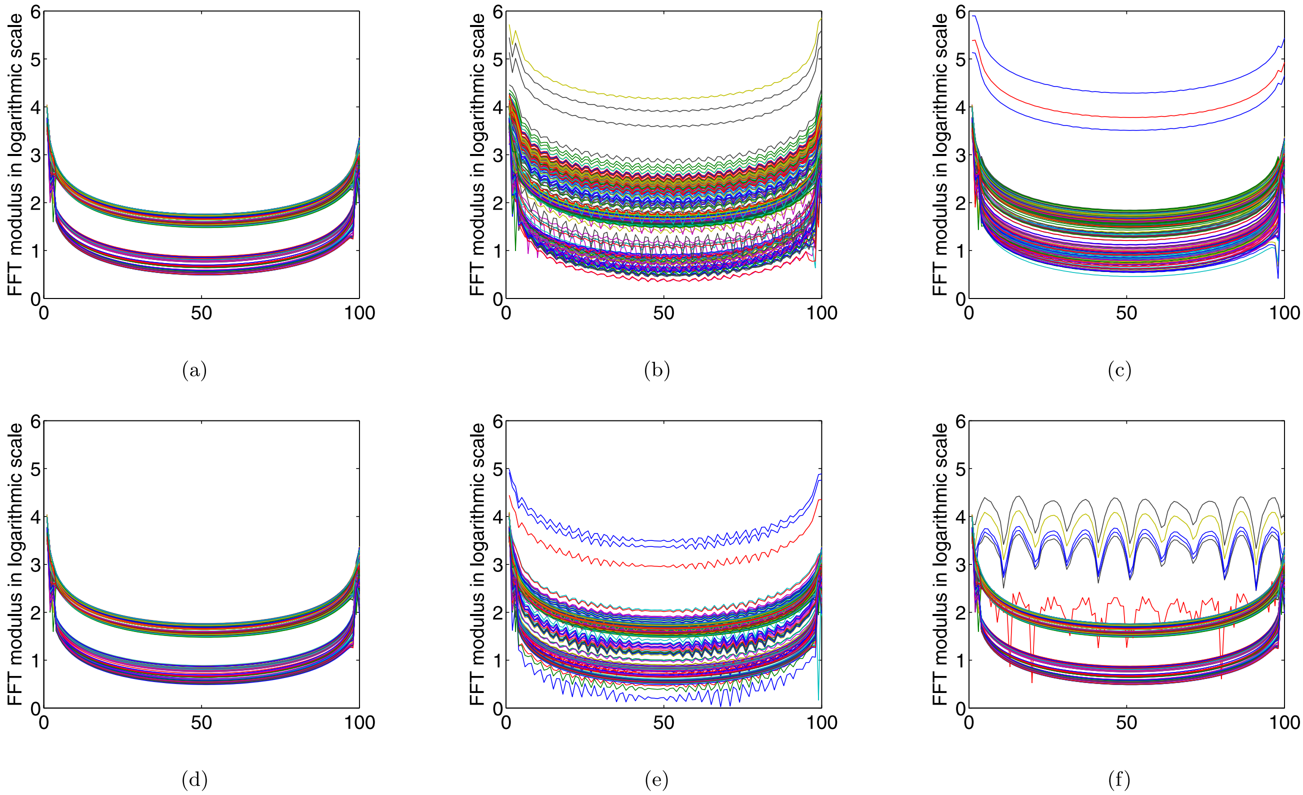


Fig. 5 Modulus of the *FFT* (logarithmic scale) when trajectories are split up into real and imaginary parts: (**a**) of the original matrix $W_{30\%missing}$; (**b**)–(**f**) of the filled in matrices considering rank values from 9 up to 13

A methodology similar the one proposed in [10] is presented, although for a completely different problem. The proposed approach works as follows:

Algorithm Set $r = r_0$, compute the modulus of the *FFT* (denoted as $|\cdot|$ from now on) of the original input matrix W ($F = |FFT(W)|$) and repeat the following steps until $r = r_{max}$[2]:

1. By using the current rank r, apply a matrix completion technique to W to obtain a full matrix of trajectories, W_r (in our experiments the result of Alternation technique is used to fill in missing data in W).
2. Apply the *FFT* to W_r and compute its modulus: $F_r = |FFT(W_r)|$.
3. Compute the following difference:

$$e(r) = \|F - F_r\|_F. \tag{5}$$

4. If $r < r_{max}$, increase $r = r + 1$ and go to step 1.

Solution The W_r that gives the minimum value of e is the best full matrix and the corresponding r is the estimation of the rank of W we were looking for. In the preliminary version [20], the estimated r was the one that gave the first local minimum of e.

In the current experiments, an initial rank $r_0 = 2$ is assumed. In [20], r_0 was set to 5, which could give wrong rank estimations in the case of dependent motions (the rank of W can be 3 or 4 in those cases).

The three object scene presented in the previous section is now used to illustrate the performance of the proposed measure of goodness. The same trajectory matrix, with a 30% of missing data, is used ($W_{30\%missing}$). The modulus of the *FFT*, of the obtained filled matrices considering rank values from 9 up to 13 are plotted in Fig. 5. Additionally, the modulus of the original input matrix ($W_{30\%missing}$) is also plotted (Fig. 5(a)). It can be appreciated that, the most similar *FFT* modulus to the one of the initial matrix (Fig. 5(a)) is obtained in the case of $r = 11$ (Fig. 5(d)).

Figure 6 shows the difference between the modulus of $FFT(W_{30\%missing})$ and each one of the FFT_r, obtained considering different rank values. Concretely, the value obtained with (5) is plotted. It can be seen that the minimum of e is found at $r = 11$, which is the same that the one obtained in the previous section when rms_{all} is considered as a measure of goodness (see Fig. 2). Finally, it should be noticed

[2]This upper-bound could be automatically computed by using some criteria for stopping the iteration. This will reduce CPU time, but it is out of scope of the current manuscript.

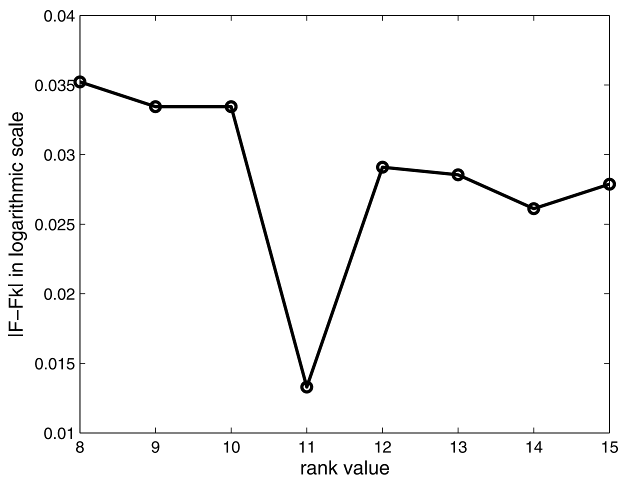


Fig. 6 Difference between F and each one of the F_r (see (5)); the minimum is found at $r = 11$

that the shape of the plot presented in Fig. 6 is quite similar to the one obtained when all data points in W_{full} were considered for plotting the rms_{full} as a function of the rank, compare Fig. 6 with Fig. 2(b).

Hence, this measure gives similar results than the one that uses all the entries in the matrix of trajectories. The advantage is that in this case, the initially missing data are not used.

3 Experimental Results

A study of the performance of the proposed rank estimation approach is presented in this section. Sequences with different numbers of objects and percentages of missing data are considered. Synthetic and real data, introduced in Sect. 3.1, are used to validate results.

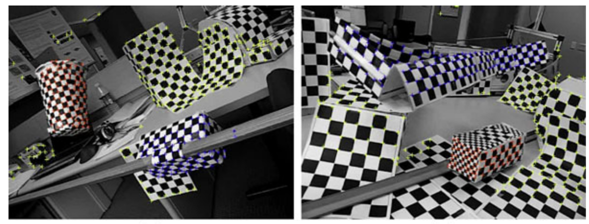
Experimental results are focused on the use of the Alternation technique as a matrix completion method (see Sect. 3.2). Results obtained by considering two different global matrix completion techniques are reported in Sect. 3.3 in order to show the validity of the proposed approach with other matrix completion techniques. In particular, results obtained with the Singular Value Decomposition (SVD) [11] and the SPectrally Optimal Completion (SPOC) [3] are included.

3.1 Data

Synthetic data sets are generated by randomly distributing 3D feature points over the surfaces of a cylinder and a triangular mesh (nodes) representing a sculptured surface. Taking these objects, different sequences are obtained by rotating and translating both of them. At the same time, the camera also rotates and translates. Different numbers of cylinders and sculptured surfaces are considered in order to generate sequences with multiple objects.

(a) (b)

Fig. 7 Objects used in the real data experiments



(a) (b)

Fig. 8 (**a**) First frame of the three objects sequence. (**b**) First frame of the two objects sequence

The same procedure applied to the synthetic data is also used to process real data. Two different objects are considered (see Fig. 7). From each object, a real video sequence with a resolution of 640×480 pixels is generated. A single rotation around a vertical axis is performed to each of the objects. Feature points are selected by means of a corner detector algorithm and only points distributed over the squared-surfaces (box and cylinders) visible in all the frames are considered. More details about the corner detection and tracking algorithm can be found in [23]. Full trajectory matrices corresponding to sequences of multiple objects are generated by merging different matrices of single object trajectories, after swapping x and y coordinates. Overlapping between objects is avoided for the sake of presentation simplicity by applying translations. Furthermore, sequences from the benchmark presented in [30] are also tested. Concretely, in the current paper, results obtained with two checkerboard sequences that contain two and three objects respectively are reported. The first frame of each of these two sequences is shown in Fig. 8.

It should be noticed that since real images usually contain noisy data all singular values are nonzero. Therefore, the smallest ones must be truncated in order to estimate the correct rank of W. Since it is difficult to set the appropriate threshold, [21] proposes the *model selection* for rank detection. Based on that, the following expression is used to estimate the rank of a full matrix in presence of noise:

$$r_m = \arg\min_r \frac{\lambda_{r+1}^2}{\sum_{j=1}^r \lambda_j^2} + \mu r, \qquad (6)$$

where λ_i corresponds to the i-th singular value of the matrix, and μ is a parameter that depends on the amount of noise. The r that minimizes this expression is considered as the rank of W. The higher the noise level is, the larger μ should be [30]—in our sequences with real data, this parameter has been empirically set to $\mu = 10^{-7}$, which gives the most coherent rank value, taking into account the motion and number of objects in each studied real sequence.

3.2 Rank Estimation by Considering a Local Matrix Completion Method: The Alternation Technique

As mentioned in Sect. 1, the Alternation technique converges to a local solution when there are missing data in the matrix of trajectories W. In order to avoid a wrong solution associated with the random nature of the Alternation initialization, 25 attempts are performed for each sequence and each percentage of missing data. Recall that the Alternation starts with an initial A_0 or B_0 random factor and proceeds until the product $A_k B_k$ converges to the known values in W. This motivates the use of a quartile-based representation of the results as detailed below.

From a given full matrix, missing data are automatically generated by removing parts of random columns, simulating the behavior of tracked features. The removing process randomly selects a cell in the given column, splitting it up into two parts. One of these parts is randomly removed, simulating features missed by the tracker or new features detected after the first frame, respectively. Since the full matrix is initially known, its rank can be directly computed by means of its singular values [11] and compared with the estimated rank obtained with the proposed approach.

The experimental results are focused on matrices of trajectories. However, a preliminary study, which takes into account different kind of low-rank matrices is carried out in the next section.

3.2.1 General Matrix Completion and Rank Estimation

This section aims at showing that the proposed rank estimation technique can be applied to different low-rank data matrices. Concretely, the following cases are studied: (i) a random matrix; (ii) a matrix of trajectories with randomly permuted rows (the idea is to simulate non-smooth camera motion); (iii) the same matrix of trajectories, with the originally given rows. Given a low-rank missing data matrix, unknown entries are filled in with the Alternation technique by considering different rank values. Then, three different measures of goodness are computed for each rank value, in order to know which rank gives the best filled in matrix. In particular, the *rms*, the rms_{all} and the difference between F and F_k (5) are computed. The three studied matrices in this experiment have the same size (180×145) and the same rank value ($r = 8$).

In a first experiment, a random rank-8 matrix is considered. Figure 9 shows the results obtained in this case. Actually, the mean of the obtained error at 25 attempts is plotted, for each rank value. The smallest error value is highlighted with a dashed circle in the plots. It can be seen that the smallest error is obtained for $r = 8$ (independently of the measure of goodness chosen), when there is no noise in the data (see Fig. 9(a), (b) and (c)). However, when working with noisy data, the *rms* decreases as the rank value increases (Fig. 9(d), (e) and (f)). Therefore, another measure of goodness is needed to be used. Notice that the trend of the plots obtained with the rms_{all} (Fig. 9(b) and (e)) and with the proposed measure of goodness (Fig. 9(c) and (f)) are very similar.

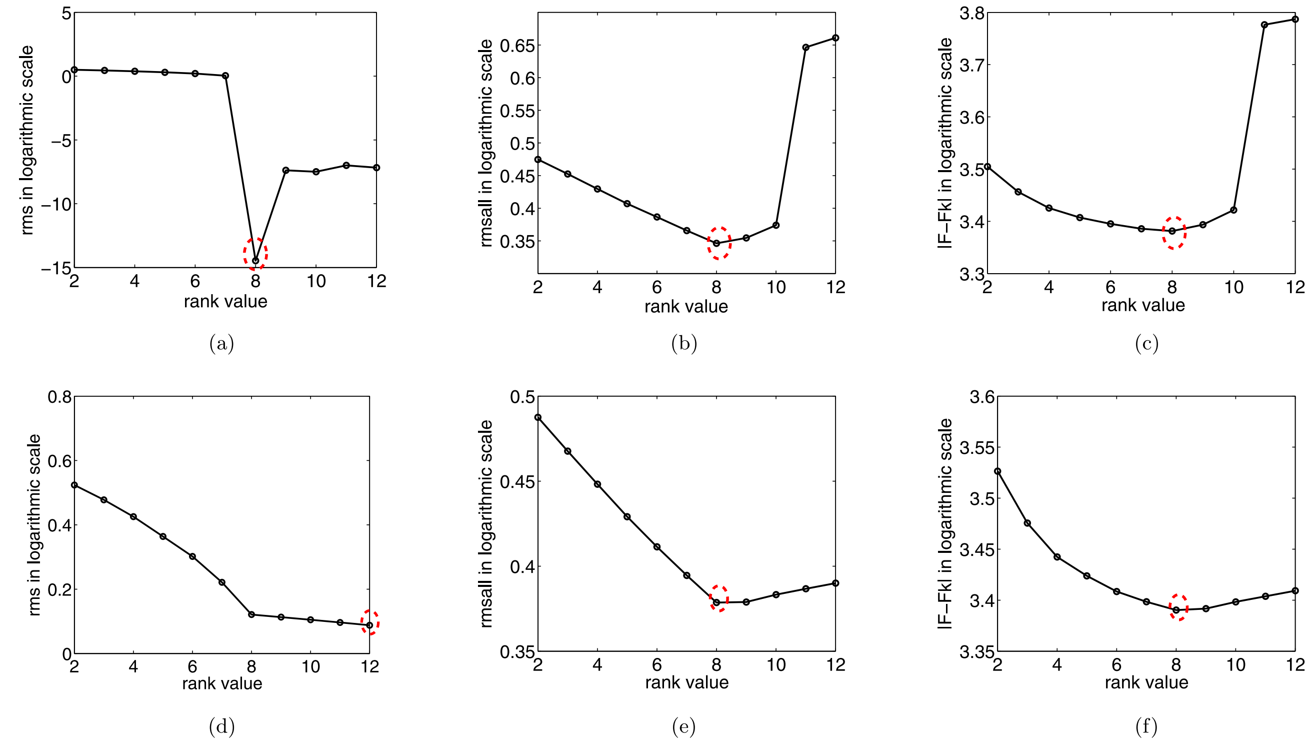
In a second experiment, the studied sequence contains two cylinders, which are generated as explained in Sect. 3.1. Feature point trajectories in the image plane are plotted in Fig. 10(a), while the obtained matrix of trajectories W_{full} is shown in Fig. 10(b). The rank of W_{full} is 8. First, the rows of the matrix are randomly permuted, in order to simulate a non-smooth movement of the camera. Results are plotted in Fig. 11. In this case the rank is not as properly estimated as in the previous experiment, for any measure of goodness. It can be concluded that the proposed rank estimation technique is not as suitable for sequences with a high degree of non-smooth camera motion. Notice that the trend of the plots obtained with rms_{all} and with the proposed measure of goodness are similar (see Fig. 11(b) and (c)). Again, it is shown that the *rms* cannot be used as a measure of goodness with noisy data: the estimated rank is 12 in this case, as it can be seen in Fig. 11(d).

Finally, the Alternation technique is applied to the matrix plotted in Fig. 10(b). Figure 12 shows the results obtained in this case. Results are similar to the ones obtained in the random data matrix: the smallest error is obtained when $r = 8$, except when the *rms* is used in the noisy data case (see Fig. 12(d)).

Experimental results presented in Sects. 3.2.2 and 3.2.3 are focused on matrices of trajectories.

3.2.2 Experiments with Synthetic Data

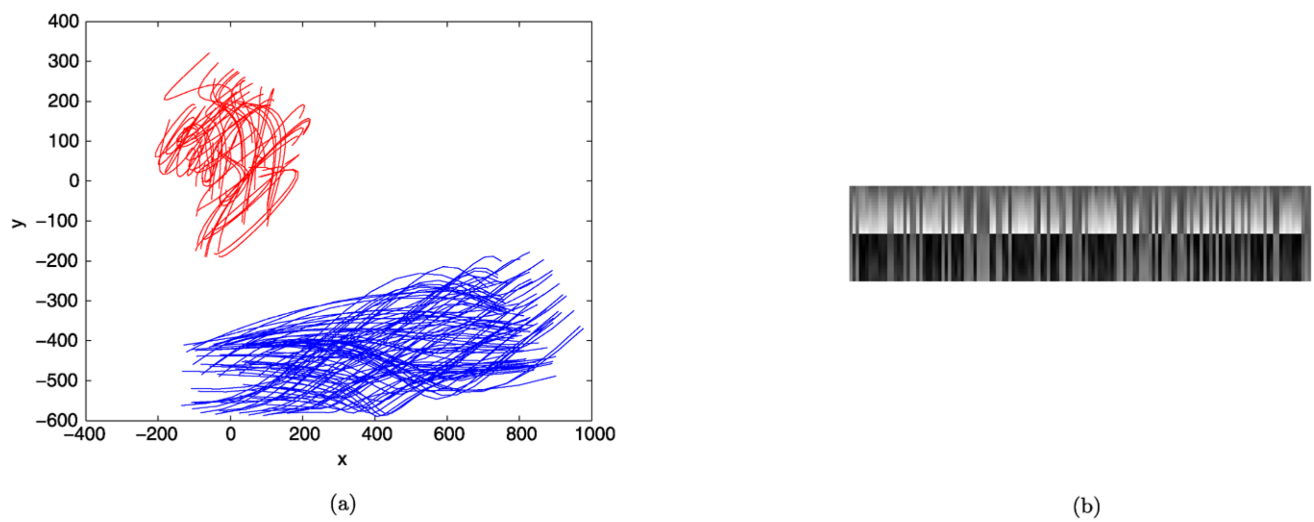
Figure 1(a) shows an illustration of the first studied sequence with synthetic data. It is defined by 100 frames containing 340 feature points (114 and 119 from two cylinders and 107 from a sculptured surface). Feature point trajectories are depicted in Fig. 1(b) and a trajectory matrix, for the case of 30% of missing data, is presented in Fig. 1(c). Figure 13(a) shows the estimated rank values considering the sequence of Fig. 1 and different percentages of missing data (the rank of the full matrix is 12). The *rms* values, which study how the initially known values are recovered with the Alternation technique, are plotted in Fig. 13(b).



(a) (b) (c)

(d) (e) (f)

Fig. 9 Random matrix W_{full}. (**a**) The *rms* as a function of the rank values, no noisy data case. (**b**) The rms_{all} as a function of the rank values, no noisy data case. (**c**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**d**) The *rms* as a function of the rank values, noise is added to W_{full}. (**e**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**f**) Difference between F and each one of the F_k, noisy data case



(a) (b)

Fig. 10 (**a**) Feature point trajectories plotted in the image plane. (**b**) Full data trajectory matrix (W_{full}), with $r = 8$

Analogously, the obtained results considering sequences of 5, 7 and 9 objects are plotted in Figs. 14–16. In these sequences, the rank of each input full matrix is 16, 20 and 25, respectively. In general, the rank is quite well estimated even with a percentage of missing data of about 40%, which is a very remarkable performance considering for instance the large number of objects contained in the last scene—nine objects. Notice that no prior knowledge about the objects contained in the scene nor about their motion is given. Figures 14(c), 15(c) and 16(c) show the *rms* obtained when the Alternation is applied to the missing data matrix, considering the corresponding rank value. Figures 13(a) and 14(b) show that in most of the cases the estimated rank is very close to the correct value. In fact, the obtained error is not

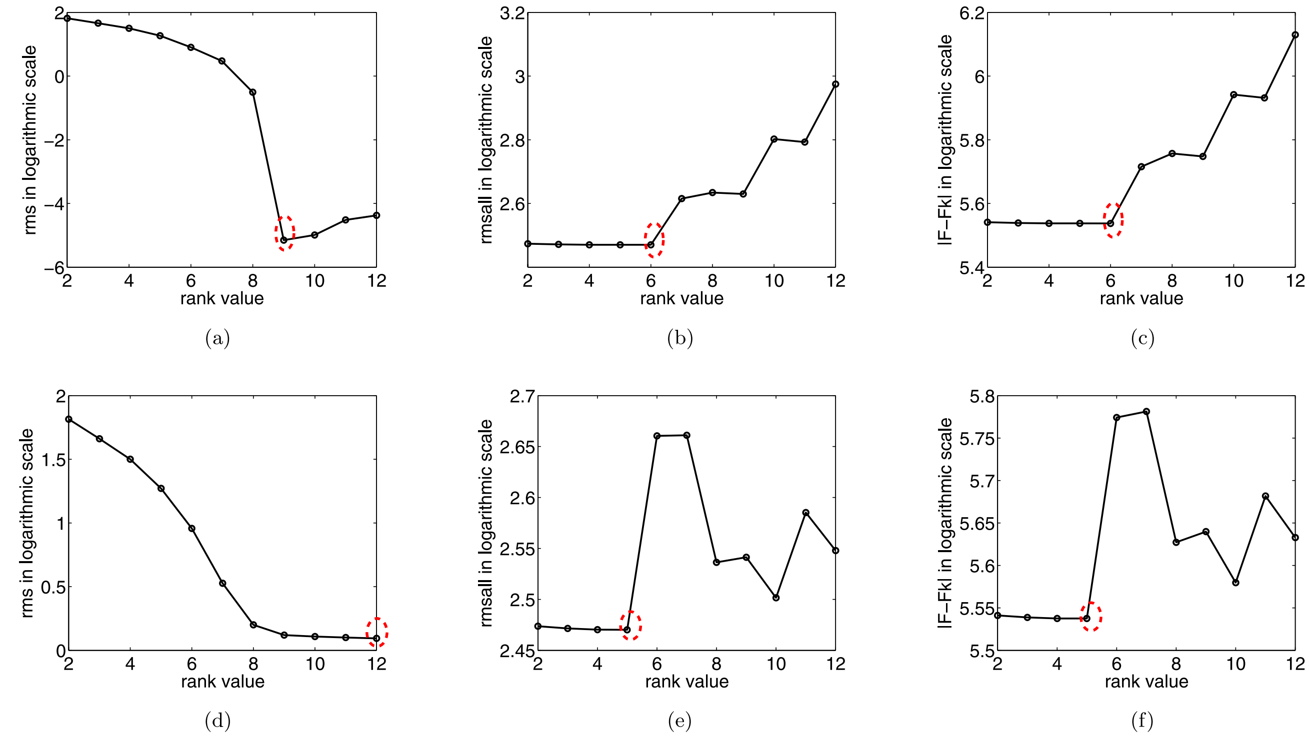


Fig. 11 Matrix of trajectories W_{full} obtained by randomly permuting the rows of the originally matrix shown in Fig. 10(b). (**a**) The *rms* as a function of the rank values, no noisy data case. (**b**) The rms_{all} as a function of the rank values, no noisy data case. (**c**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**d**) The *rms* as a function of the rank values, noise is added to W_{full}. (**e**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**f**) Difference between F and each one of the F_k, noisy data case

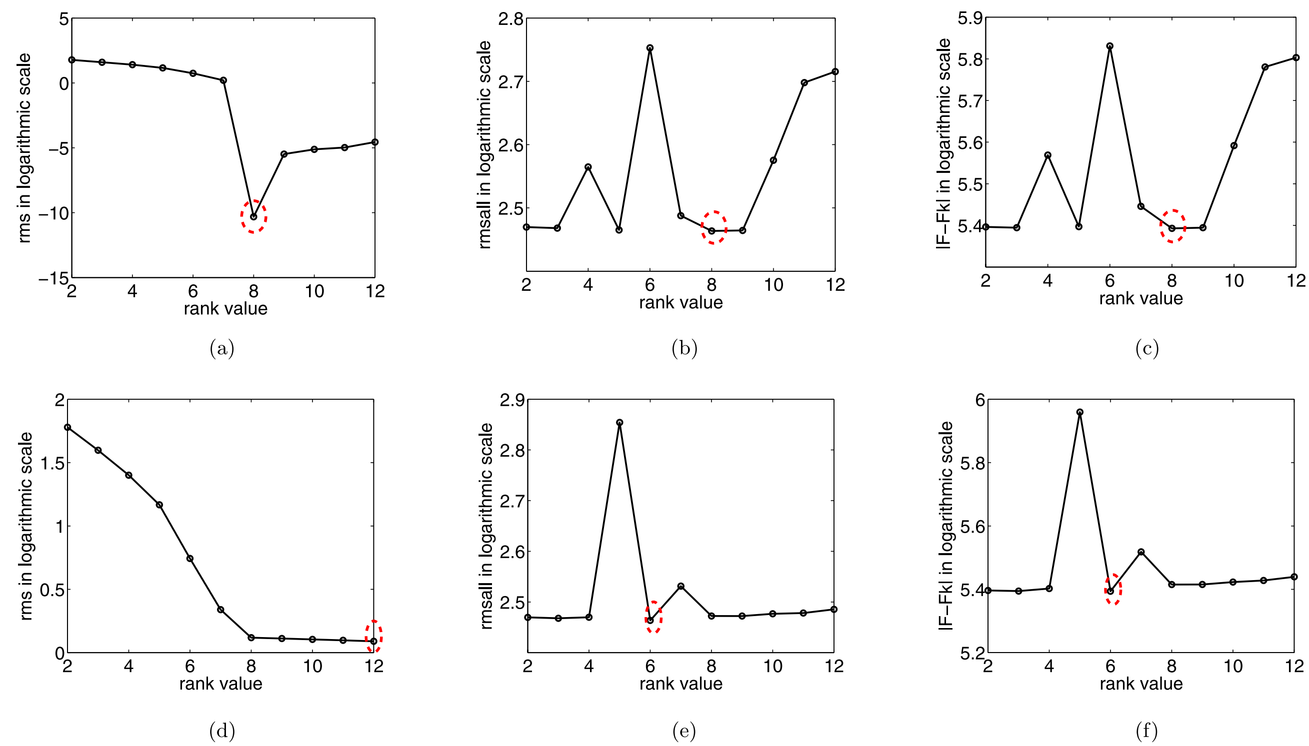


Fig. 12 Matrix of trajectories W_{full} of Fig. 10(b). (**a**) The *rms* as a function of the rank values, no noisy data case. (**b**) The rms_{all} as a function of the rank values, no noisy data case. (**c**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**d**) The *rms* as a function of the rank values, noise is added to W_{full}. (**e**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**f**) Difference between F and each one of the F_k, noisy data case

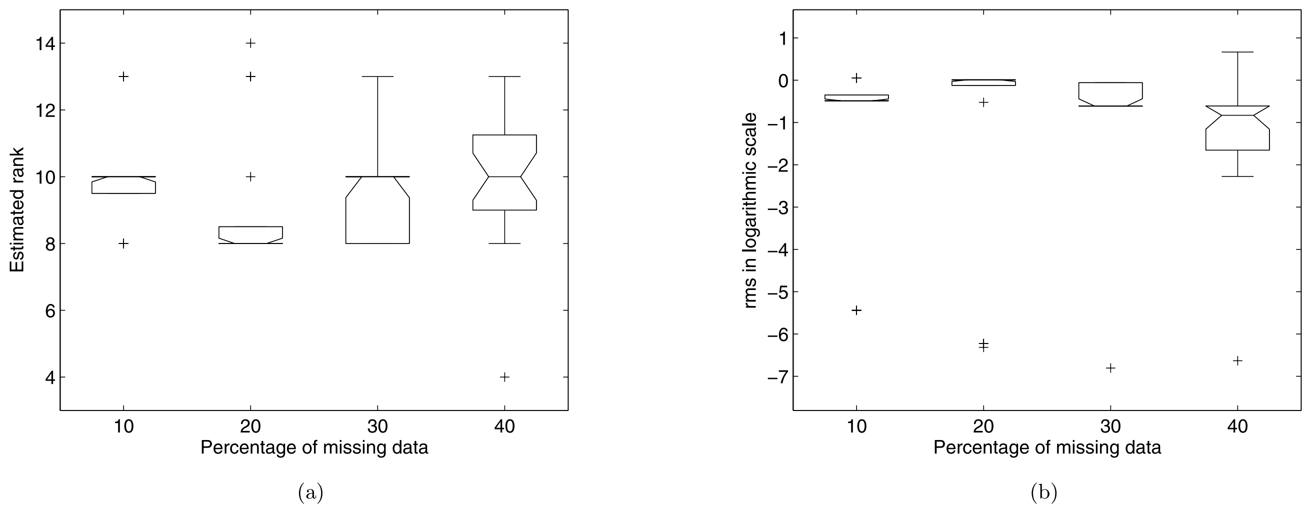


Fig. 13 Three objects, rank of full matrix 12: (**a**) estimated rank values for different percentages of missing data; (**b**) *rms* obtained with Alternation in logarithmic scale. Concretely, boxes enclose data in between lower and upper quartiles (medians are represented by *horizontal lines* in *thinner regions*). *Vertical lines*, outside these boxes, correspond to the rest of data

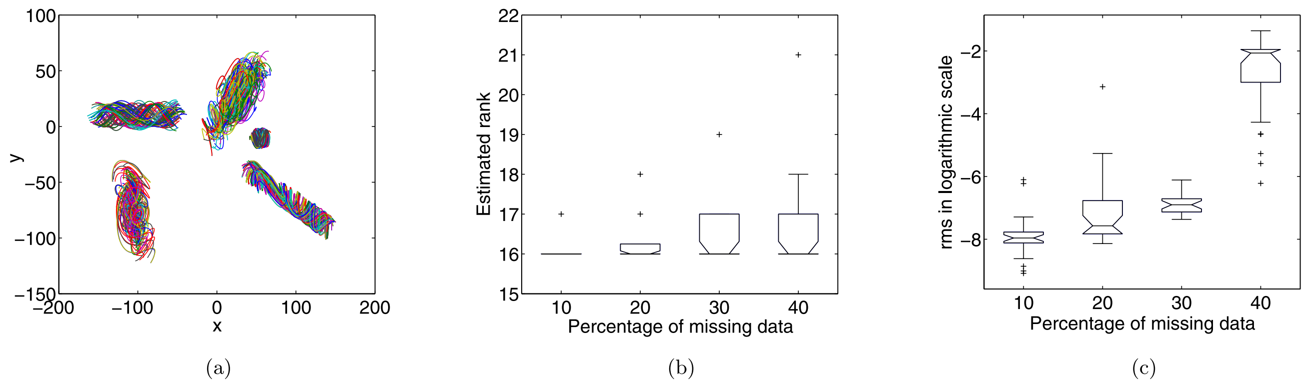


Fig. 14 Five objects, rank of full matrix 16: (**a**) feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

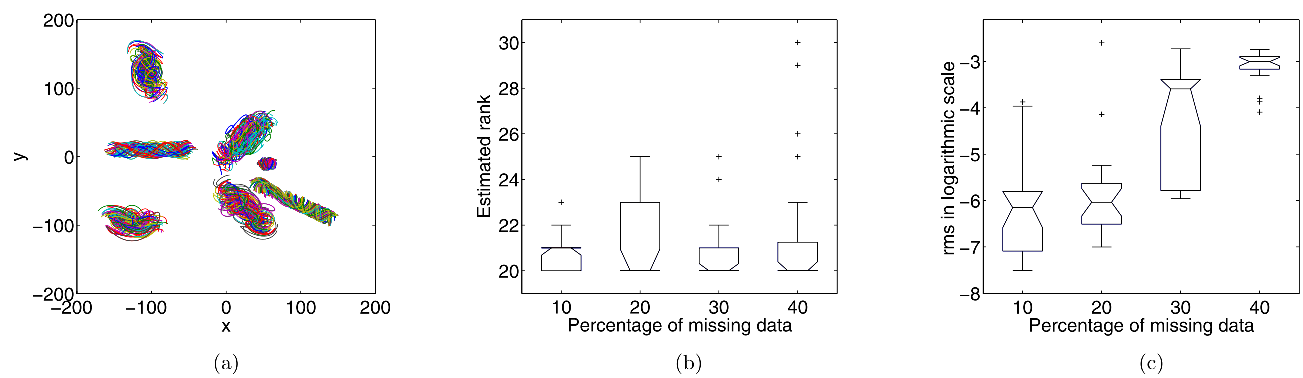
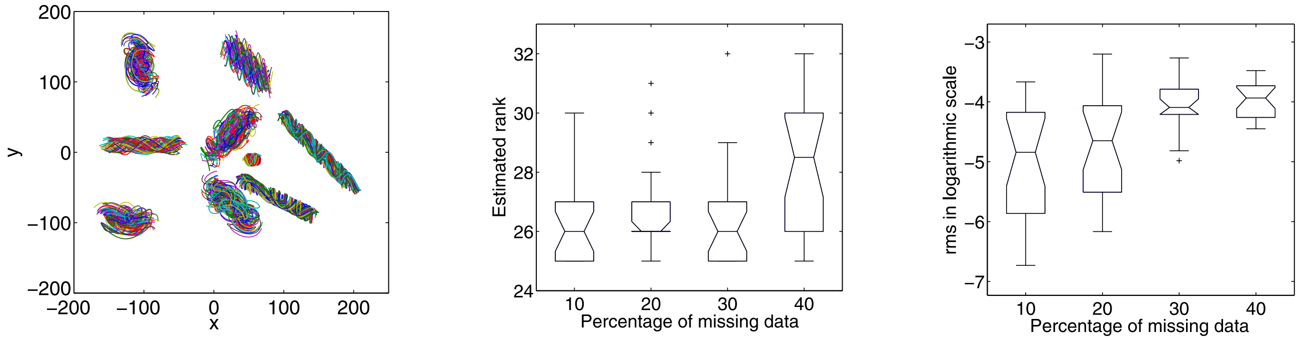


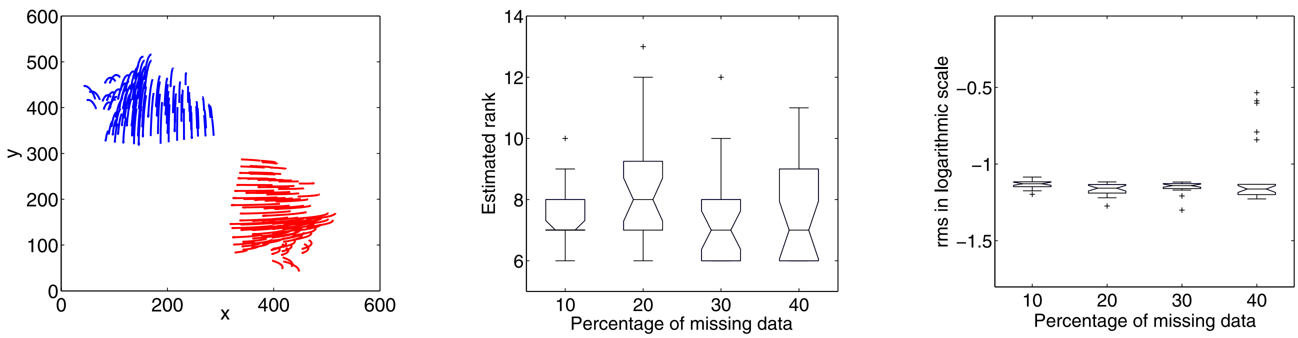
Fig. 15 Seven objects, rank of full matrix 20: (**a**) feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

significant in applications such as motion segmentation. For the case of the scene with seven objects (Fig. 15(b)) the maximum error is reached in the case of 20% of missing data; it is about 20% the correct value. Note that in the rest of cases that error is less than 15% of the correct rank value. Some outliers appear in the case of 40% of missing data. Finally,



(a) (b) (c)

Fig. 16 Nine objects, rank of full matrix 25: (**a**) feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale



(a) (b) (c)

Fig. 17 First sequence, rank of full matrix 6: (**a**) full feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

for the scene with nine objects (Fig. 16(b)), the maximum error appears when 40% of missing data are considered. That estimated rank value is 28% higher than the correct one. In the rest of cases the rank is estimated with an error smaller than 20% of the correct value.

Once the rank is estimated and a full matrix of trajectories is obtained, further post processing techniques (e.g., motion segmentation, SFM) that require a full input matrix of trajectories can be performed. It should be highlighted that the estimated rank is always considerable closer to the correct value than when a predefined fixed value was used (e.g., five, as in [32]). It has been shown in [20] that results of further processing (e.g., motion segmentation) depend on the accuracy of the estimated rank.

3.2.3 Experiments with Real Data

The first studied sequence with real data is generated by using the first object twice (Fig. 7(a)). The obtained full feature point trajectories are plotted in the image plane in Fig. 17(a); it contains 174 feature points tracked through 101 frames. A second sequence is generated considering both objects together (Fig. 7(a) and (b)); the corresponding full trajectories in this second case are plotted in Fig. 18(a). It is defined by 61 frames and 275 feature points (87 from the first object and 188 from the second one). Finally a three object sequence is generated by considering twice the first object (Fig. 7(a)) together with the second object (Fig. 7(b)). This third sequence contains 362 feature points (87, 87 and 188 from the first and second object respectively), which are tracked through 61 frames. The obtained trajectories are depicted in Fig 19(a).

Figure 17(b) shows the estimated rank values for the first sequence. The rank of the full matrix of trajectories is 6; it is obtained by using (6) and by setting $\mu = 10^{-7}$. In this particular case, the objects contained in the scene define a degenerate motion. That is, each one of them does not generate a full rank motion matrix (rank 4). In this real data experiment the estimated rank takes a wider range of values than in the synthetic data case. Similar results are obtained with the second sequence, in which the full matrix has also rank 6. Figure 18(b) shows the estimated rank values. Finally, results for the three object sequence are presented in Fig. 19(b). In this case, the rank of the full matrix is 7. In the

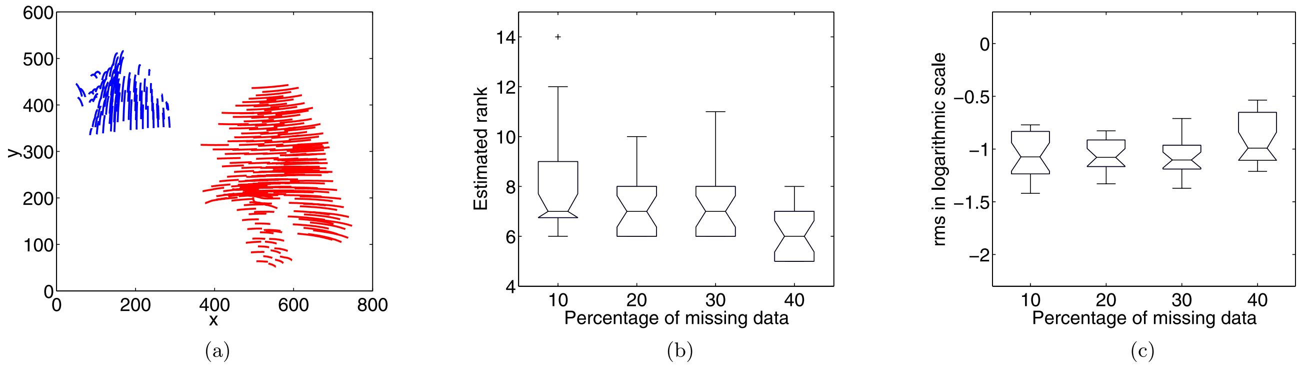


Fig. 18 Second sequence, rank of full matrix 6: (**a**) full feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

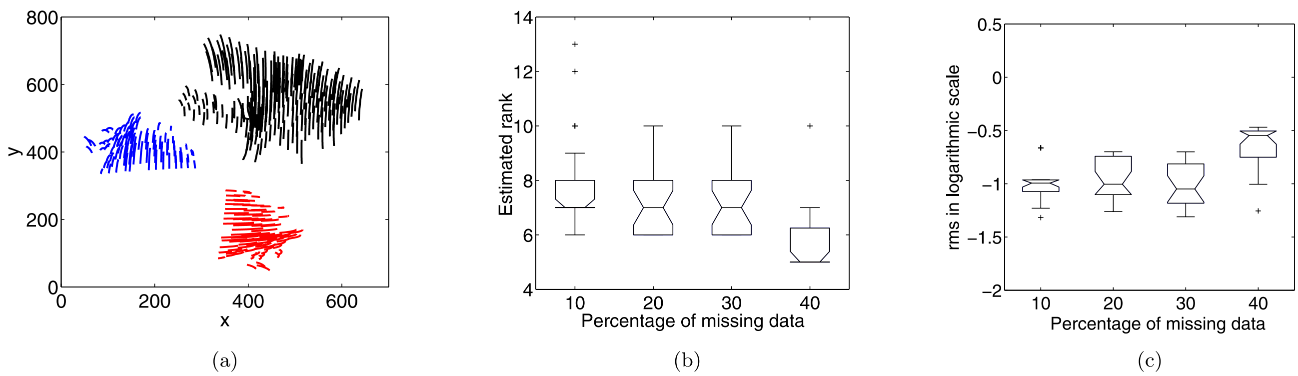


Fig. 19 Third sequence, rank of full matrix 7: (**a**) full feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

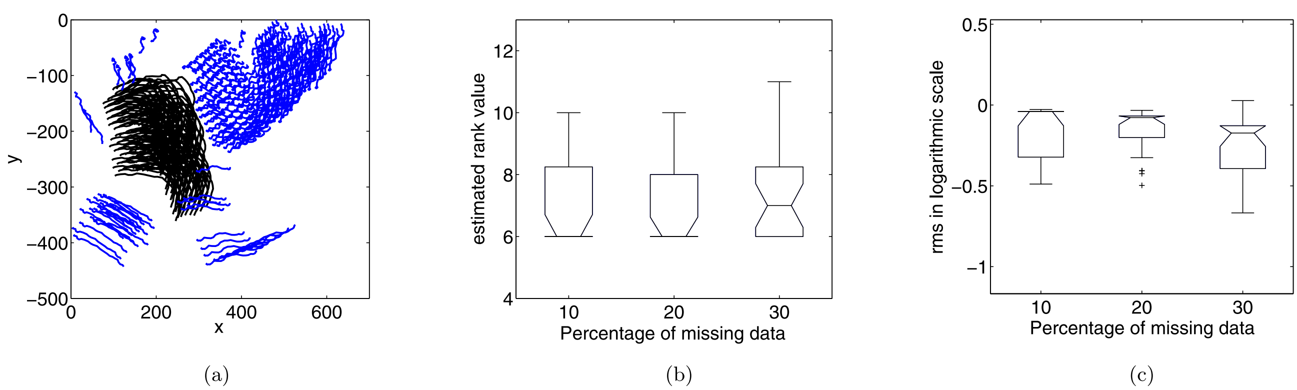


Fig. 20 (Color online) Checkerboard sequence containing 2 objects, rank of full matrix 6: (**a**) full feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

three examples the *rms* values, computed with the values initially known in each case, are quite large (see Figs. 17(c), 18(c), 19(c)), in comparison with the scenes with synthetic data, since the real images contain noisy data.

Finally, sequences from the benchmark presented in [30] are considered. Results obtained with the first sequence (Fig. 8(a)), taking 2 of the objects contained in the scene, are shown in Fig. 20. The rank of the full data matrix, obtained by assuming $\mu = 10^{-7}$, is 6. The numbers of frames and feature points are 24 and 341, respectively. With such a few number of frames, a 40% of missing data would result in a matrix without enough information. Therefore, the percentages of missing data considered with these sequences are from 10% up to 30%. The first object, whose trajecto-

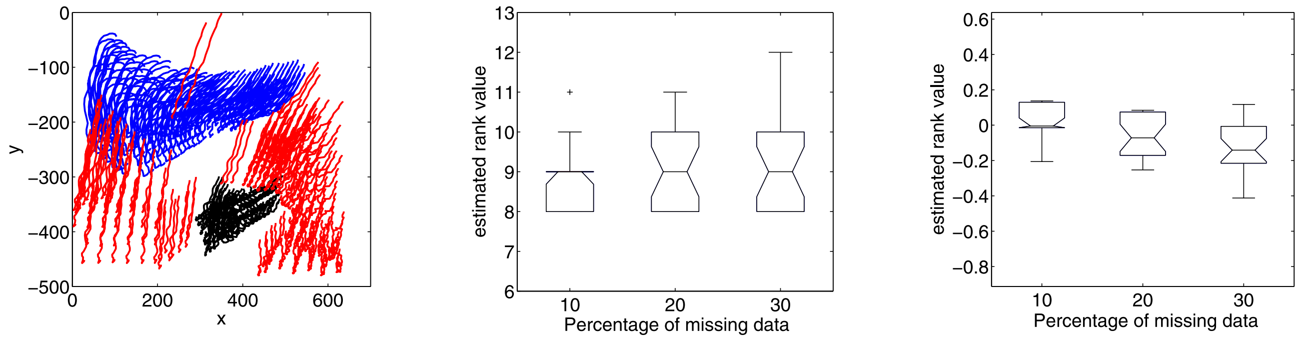


Fig. 21 (Color online) Checkerboard sequence containing 3 objects, rank of full matrix 8: (**a**) full feature point trajectories plotted in the image plane; (**b**) estimated rank values for different percentages of missing data; (**c**) *rms* obtained with Alternation in logarithmic scale

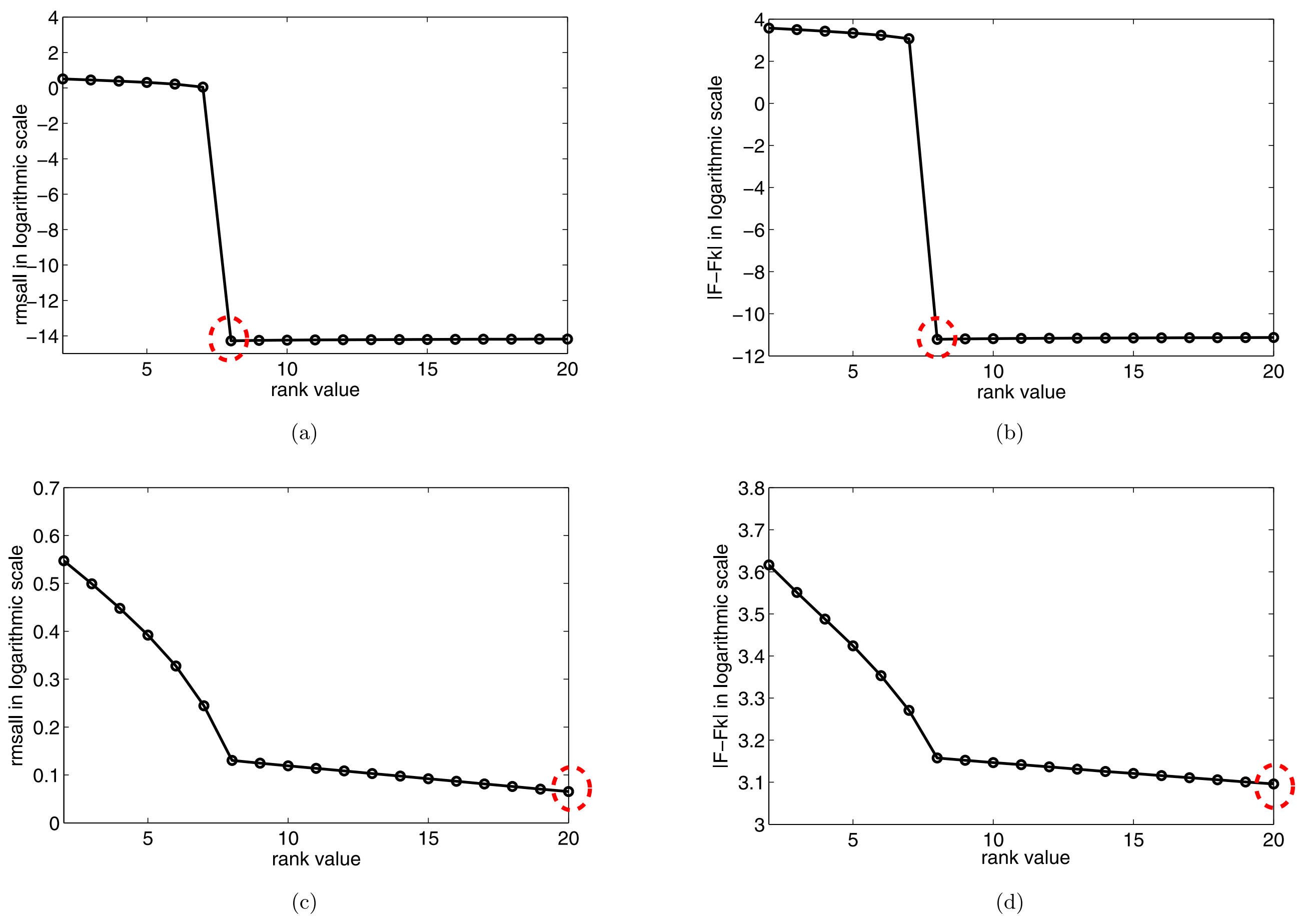


Fig. 22 Random matrix W_{full}. (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

ries are marked in black in Fig. 20(a), rotates on one axis and translates on a plane orthogonal to this axis; the camera rotates around one axis (static points are marked in blue in Fig. 20(a)). The rank value is quite well estimated for every percentage of missing data (notice that the median of the values is 6 for a percentage of missing data below 30% and 7 in the 30% case). The maximum error is obtained in the case of 30% of missing data.

Figure 21 shows results obtained with the second sequence (Fig. 8(b)), which contains 3 objects. The numbers of frames and feature points are 30 and 411, respectively. The rank of the full data matrix, obtained by assuming $\mu = 10^{-7}$,

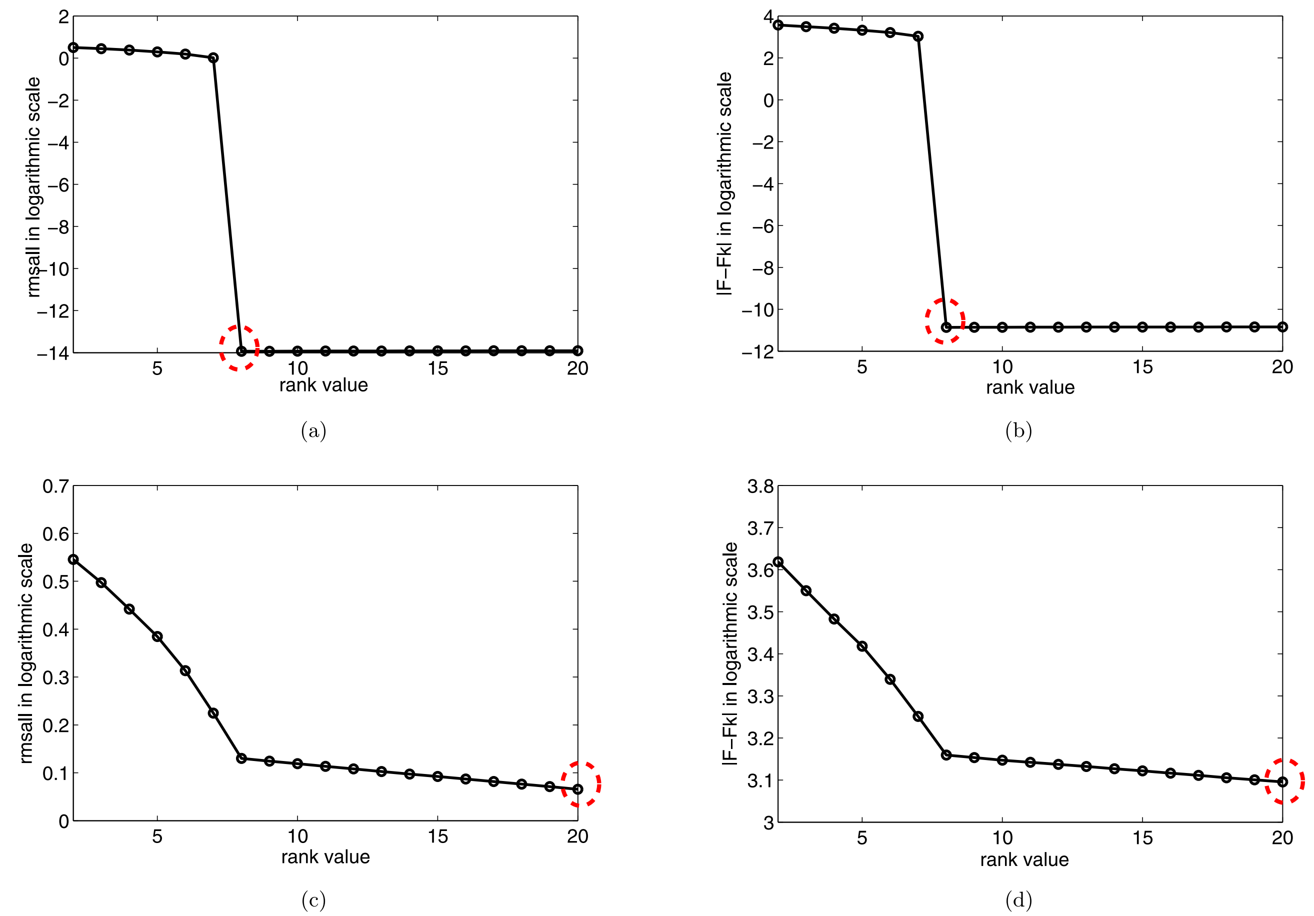


Fig. 23 Matrix of trajectories obtained by randomly permuting the rows of the originally given matrix of trajectories W_{full}. (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

is 8. Trajectories plotted in the image plane are depicted in Fig. 21(a). The first object (blue trajectories) rotates, the second one (black trajectories) rotates and translates and the camera rotates and translates (static points are marked in red). The rank is quite well estimated for any percentage of missing data (notice that the median of the estimated values is 9 in all the cases). Again, the maximum error is obtained in the case of 30% of missing data.

Although in the experiments with real data the estimated rank takes, in general, a wider range value than in the experiments with synthetic data, the median of all the rank values is equal or quite similar to the correct one (±10% in most cases).

3.3 Rank Estimation Technique by Considering Global Completion Techniques

The main objective of including this section is to show that the proposed rank estimation technique is also valid with other matrix completion methods. Therefore, only experiments with sequences that contain synthetic data are provided. Concretely, the experimental results are analogous to the ones presented in Sect. 3.2.1 for the case of the Alternation technique. The main advantage of using a global technique is that only one attempt is needed, since results do not depend on the initialization.

Actually, this section aims at showing that, independently of the matrix completion method used, the smallest *rms* does not always correspond to the correct rank value of W, as it was shown in Sect. 2.2 and Sect. 3.2.1.

3.3.1 Singular Value Decomposition (SVD)

Given a matrix of trajectories $W_{m\times n}$, the Singular Value Decomposition gives a global optimal solution when approximates W by a low-rank matrix. The main drawback of the SVD is that it cannot be used with missing data in W.

In a first experiment, a random rank-8 matrix is considered, as in Sect. 3.2.1. Figure 22(a) shows the obtained rms_{all} when W_{full} is free of noise. It can be seen that $rms_{all} = 0$ for $r \geq 8$ (recall that the rank of W_{full} is 8). Figure 22(b) shows the error obtained by comparing the modu-

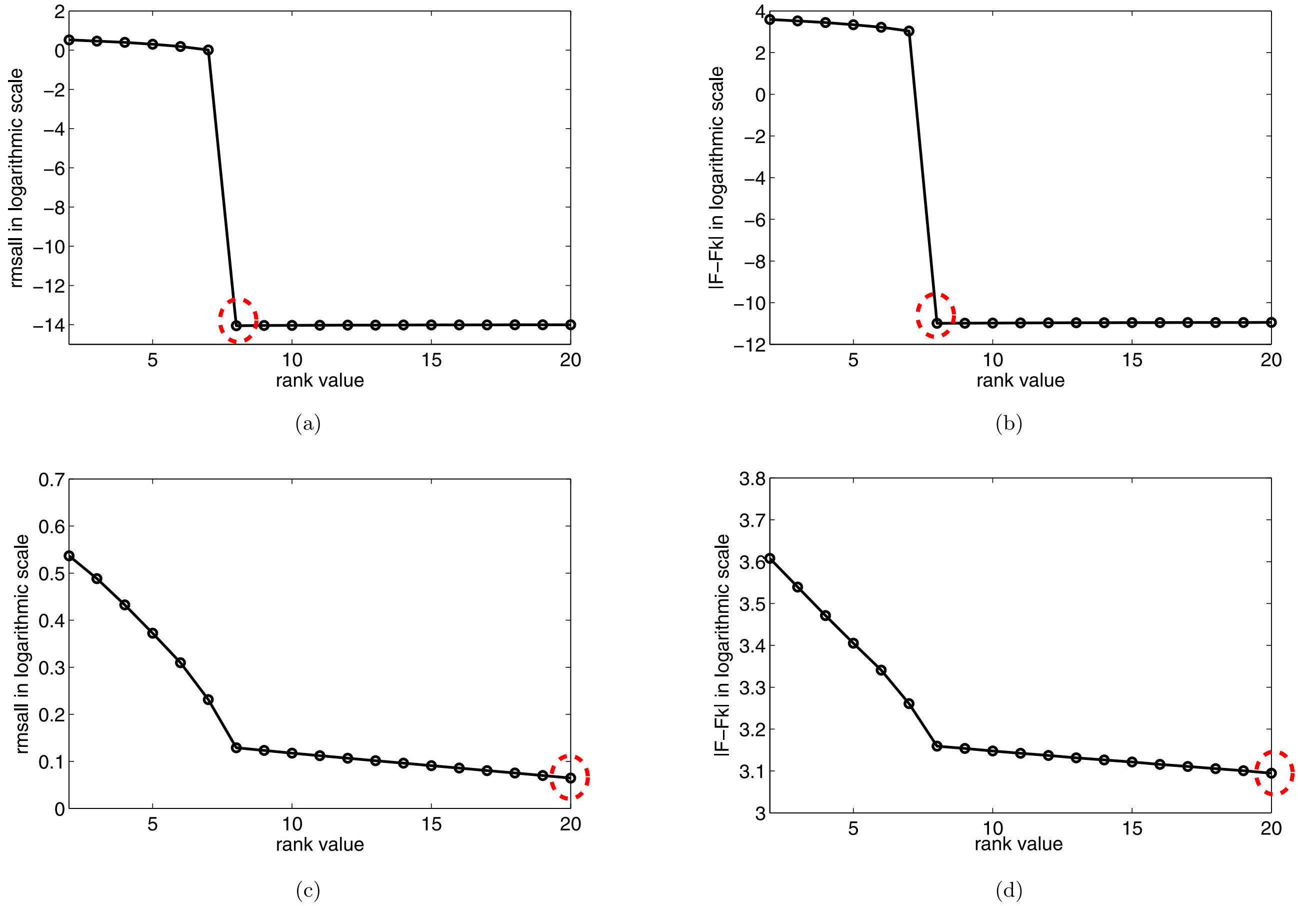


Fig. 24 Matrix of trajectories W_{full} of Fig. 10(b). (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

lus of the *FFT* (see (5)). Notice that the plot is very similar to the previous one. Again, the minimum error is obtained when $r = 8$, which is the correct one. However, in real situations matrices of trajectories use to have noise. In order to simulate these situations, a Gaussian noise with standard deviation $\sigma = 1$ and zero mean is added to the 2D feature point trajectories. In these cases, the rms_{all} and the proposed measure of goodness decrease as the rank value increases, as it can be seen in Fig. 22(c) and (d). Both measures of goodness decrease as the rank value increases.

The second experiment, consists in taking the matrix of trajectories shown in Fig. 10(b). The SVD is applied to the full matrix W_{full} (Fig. 10(b)), but their rows have been randomly permuted. Thus, in order to simulate a non-smooth camera motion. Figure 23 shows that similar results to the previous ones are obtained.

Finally, in a third experiment, the SVD is applied directly to the matrix W_{full} (Fig. 10(b)). Results are plotted in Fig. 24. Again, the smallest error (both rms_{all} and the difference between F and F_k) is obtained when $r = 8$, only in the case of no noise in the data.

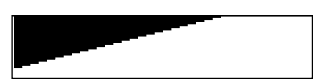


Fig. 25 Mask used to enforce a Young diagram in the missing data matrix. *Black entries* in the mask correspond to missing data, while *white entries* correspond to known data in the matrix of trajectories. A percentage of missing data of about 30% is generated

3.3.2 SPectrally Optimal Completion (SPOC)

Aguiar et al. [3] present the SPectrally Optimal Completion (SPOC) algorithm, which is a matrix completion method that proposes a global optimal solution, for particular patterns of missing entries. Concretely, this method gives a global solution when missing entries produce a Young diagram. That is, when missing entries are arranged in the first n_1 entries of the 1st row, the first n_2 entries of the 2nd row, ..., the first n_k entries of the kth row, such that $n_1 \geq n_2 \geq \cdots \geq n_k$. Figure 25 shows an example of a matrix where missing entries, which correspond to the black entries in the matrix, produce a Young diagram. The SPOC

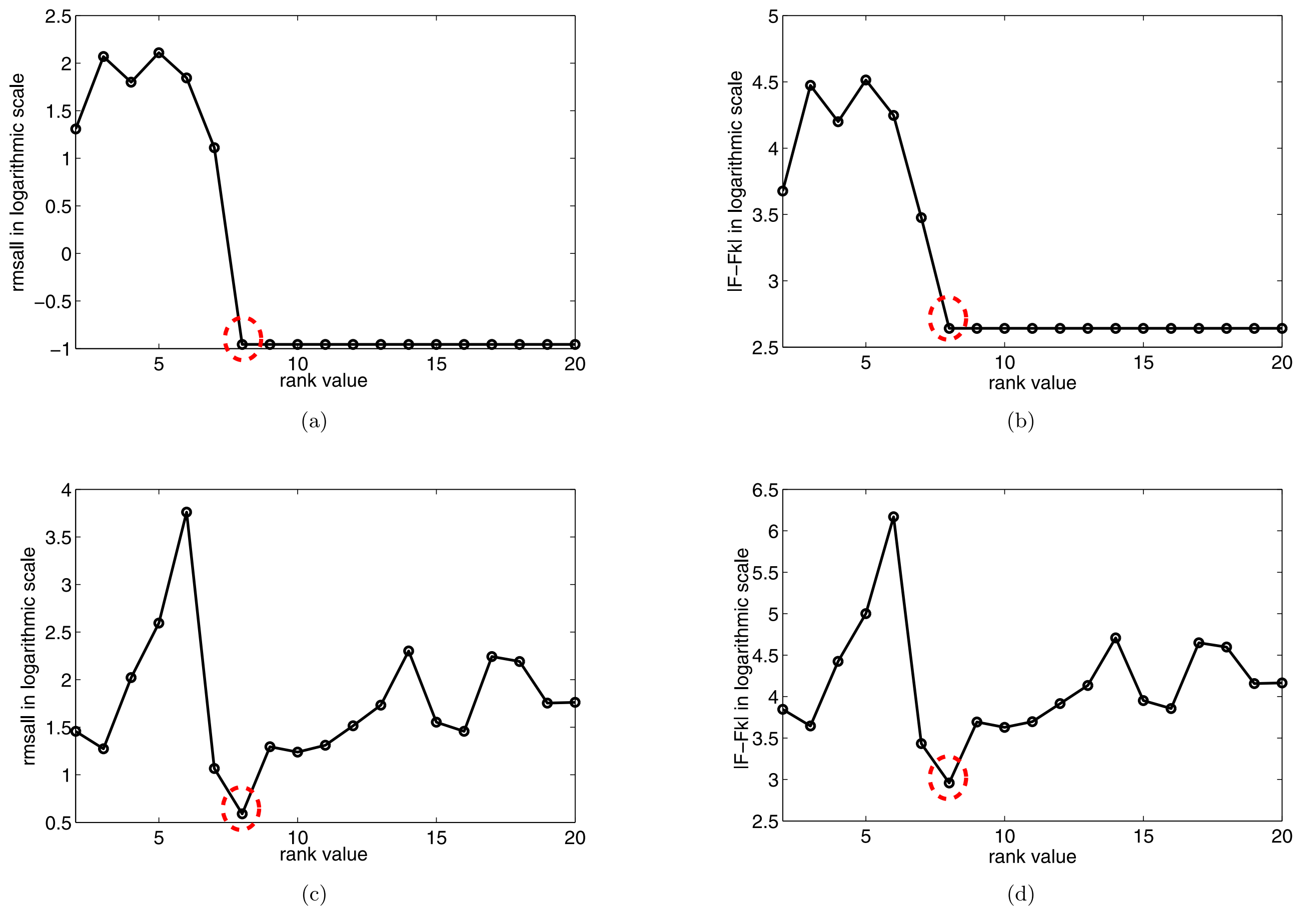


Fig. 26 Random matrix W that contains 30% of missing data. (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

algorithm is based on inequalities presented in [1] that relate the singular values of a matrix with those of its submatrices. The source code of the SPOC, publicly available at http://users.isr.ist.utl.pt/~aguiar/spoc.m, has been used in this experiment. See [3] for more details on the method.

Unfortunately, working with a matrix of trajectories, the pattern of missing data consists of two or more Young diagrams, as pointed out by [2]. In order to tackle that case, a Young-wise optimal iterative algorithm is proposed in [3]. This iterative algorithm gives a sub-optimal solution. Since the goal at this section is to test global methods, it is enforced that missing data of the matrix of trajectories W produce a Young diagram in these experiments. Figure 25 shows the mask $M_{30\times 145}$ applied to the studied full matrices W_{full} in this section to enforce that missing entries produce a Young diagram. Concretely, this mask generates a percentage of missing data of about 30% in the matrix of trajectories.

Experimental results are carried out as in Sect. 3.2.1 and as in the SVD case (Sect. 3.3.1). The first experiment consists in considering a random rank-8 matrix W_{full} with the same size as the mask shown in Fig. 25, that is 30×145. A matrix W with 30% of missing data is obtained by multiplying W_{full} by the mask. Figure 26 shows the errors obtained by considering rank values from 2 up to 20. The *rms* obtained by taking only the initially known entries is 0 for any rank value, due to the fact that the initially known data are not modified by the SPOC algorithm. Hence, another measure is needed to be defined in order to know which rank gives the best filled in matrix. As all data are initially known in W_{full}, the rms_{all} can be computed. As it can be seen in Fig. 26(a), the smallest rms_{all} is obtained for $r = 8$, which is the rank of W_{full}. Figure 26(b) shows the error obtained by comparing the modulus of the *FFT* (see (5)). It can be seen that the trend of the plot is quite similar to the previous one. The smallest error is also obtained when $r = 8$. Analogous conclusions can be derived from the results obtained when noise is added to the matrix W (see Fig. 26(c) and (d); Fig. 26(e) and (f)). Again, the smallest errors are obtained when $r = 8$, in both cases.

In a second experiment, the W_{full} matrix plotted in Fig. 10(b) is studied. First, this matrix is multiplied with

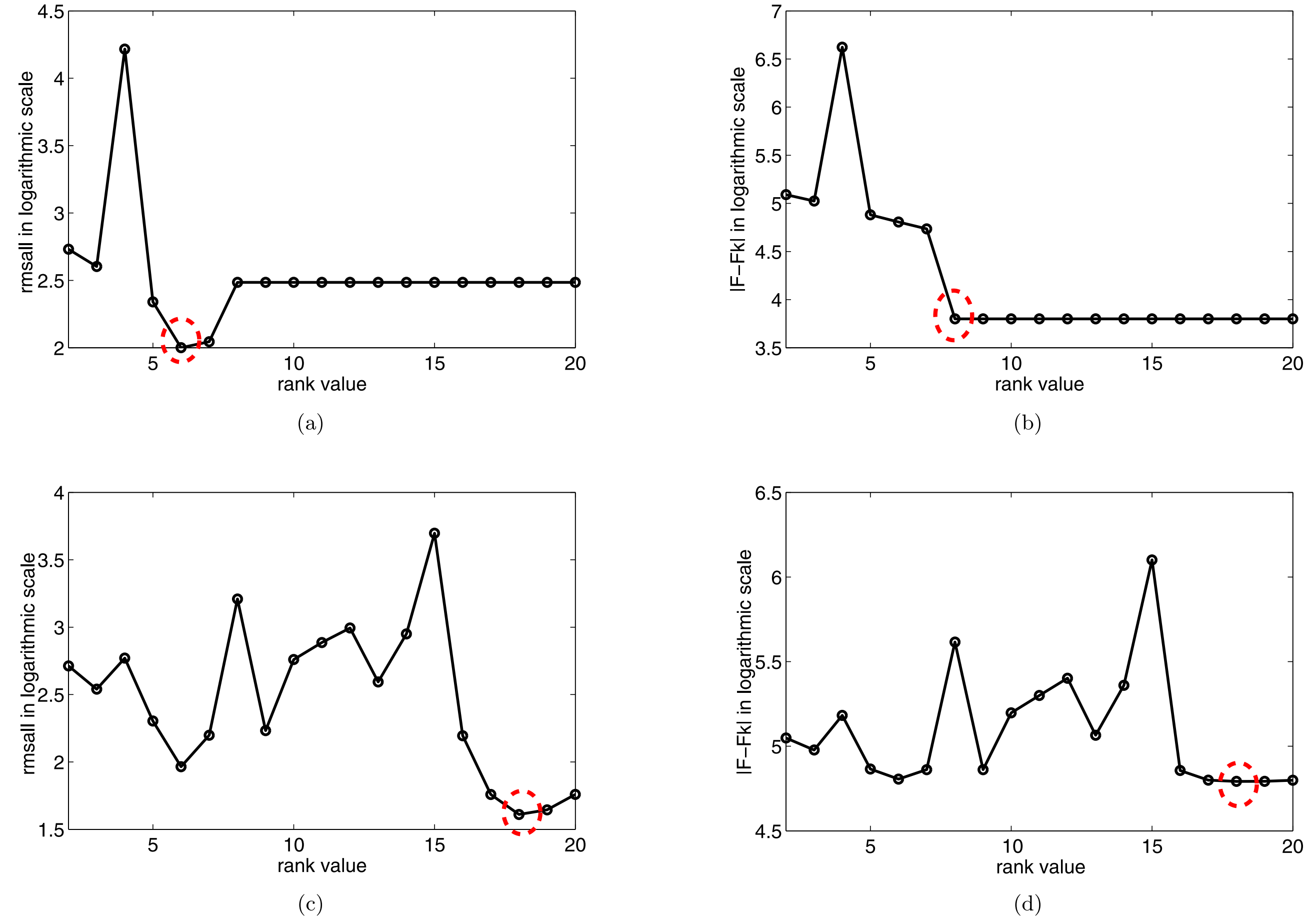


Fig. 27 Matrix of trajectories obtained by randomly permuting the rows of W (matrix of the trajectories plotted in Fig. 10(b)). W contains 30% of missing data generated with the mask of Fig. 25. (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

the mask, given a matrix of 30% of missing data. The rows of this matrix are randomly permuted before applying the SPOC algorithm, in order to simulate a non-smooth motion of the camera. As in the case of the Alternation (see Sect. 3.2.1), the estimated rank is not correct neither considering rms_{all} nor the proposed measure of goodness. Furthermore, the plots are not similar when there is no noise in the data.

Finally, results obtained by applying SPOC to the matrix of trajectories W are shown in Fig. 28. The smallest error is obtained when $r = 6$ in the case of no noise in the data and for both measures of goodness (see Fig. 28(a) and (b)). When working with noisy data, the smallest error is obtained for $r = 7$, for both the rms_{all} and the difference between F and F_k, as can be seen in Fig. 28(c) and (d).

The main drawback of the SPOC algorithm is that it gives a global optimal solution only for a particular structure of missing data. Although, in some cases, this missing data distribution could be enforced, it is not always possible. Additionally, it seems that the SPOC algorithm does not give as good results when the rows of W are randomly permuted and when there is noise in the data. A more extensive study should be done before using this method in the proposed rank estimation technique.

3.3.3 Summary

It is important to remark that the main goal of the paper is to estimate the rank of the matrix, without using the number of objects in the scene nor the kind of their motion. Independently of the matrix completion used to fill in the missing entries in the matrix, it is needed to define a measure of goodness in order to study the recovered missing data and select the rank that gives the best filled in matrix.

Although the proposed rank estimation technique uses the Alternation technique as a matrix completion method, the previous two Sections show it is also valid with other methods. The SVD is not suitable for our proposed rank estimation technique, since it cannot deal with missing data. However, it has been shown that the proposed measure of

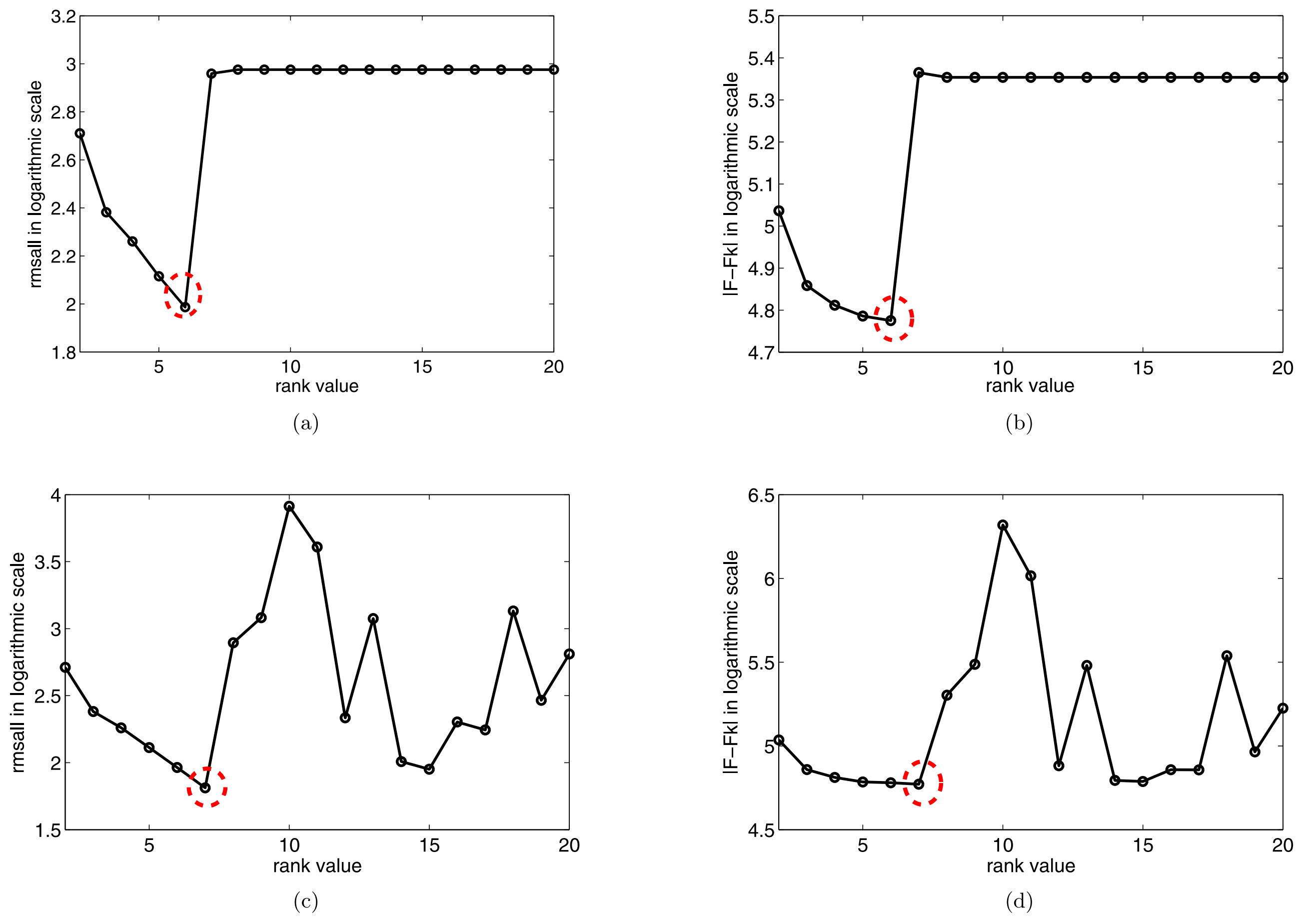


Fig. 28 Matrix of trajectories W, which contains 30% of missing data. (**a**) The rms_{all} as a function of the rank values, no noisy data case. (**b**) Difference between F and each one of the F_k (see (5)), no noisy data case. (**c**) The rms_{all} as a function of the rank values, noise is added to W_{full}. (**d**) Difference between F and each one of the F_k, noisy data case

goodness is also valid with the SVD, while there is no noise in the data. Otherwise, the obtained error decreases as the rank value increases, for both tested measures of goodness.

It has been also shown that it would be interesting to use a globally optimal method, such as the SPOC [3], in order to obtain a global solution when completing the matrix. However, the main drawback of using this method is that it is not always easy to work with matrices whose missing entries produce a Young diagram. Furthermore, the computational cost of this method is very high when the matrix is large.

4 Conclusions

A novel technique to estimate the rank of a given missing data matrix is proposed. It is based on the study of the frequency spectra of the input matrix W. The motivation is that, since feature points belong to surfaces of rigid objects, the frequencies of the signal generated by the movement of these points, summarized at each column of W, should be preserved after filling in missed entries. In other words, the recovered full matrix studied as columns should contain a frequency spectra similar to the one of the input matrix.

The obtained results empirically show the good performance of the proposed rank estimation technique when scenes containing different numbers of objects and percentages of missing data are considered. From the synthetic scene experiments (no noisy data) it can be concluded that the rank of the input matrix is well estimated, even with percentages of missing data of about 40%. Real scene experiments show that the proposed technique is also able to deal with real noisy images. The *rms* obtained with Alternation, which is used as matrix completion method, is also analyzed in order to study the error added to the data during the filling in process. It has been shown that the *rms* value grows as the percentage of missing data is increased. Hence, in cases with a high percentage of missing data, filled in matrices contain a high amount of noise and the estimated rank value could be less accurate.

In addition to the Alternation technique, global matrix completion methods have also been studied. Concretely, results obtained with the Singular Value Decomposition and with the SPectrally Optimal Completion show that the proposed rank estimation technique is also valid with other matrix completion techniques. However, the SVD will not be used, since it cannot deal with missing data.

It should be highlighted that the proposed rank estimation technique does not require the prior knowledge of the objects contained in the scene, nor any assumption about their motion. Although the viability of the proposed rank estimation technique has not been theoretically demonstrated, its good performance has been empirically shown for both synthetic and real data sequences.

Acknowledgements This work has been partially supported by the Spanish Government under projects TRA2007-62526/AUT and DPI2007-66556-C03-03; research programme Consolider-Ingenio 2010: MIPRCV (CSD2007-00018); and Catalan Government under project CTP 2008ITT 00001.

References

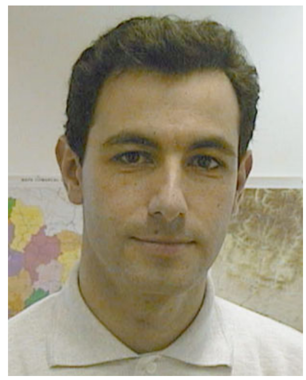
1. Aguiar, P.M.Q., Stosic, M., Xavier, J.: On singular values of partially prescribed matrices. Linear Algebra Appl. **429**, 2136–2145 (2008)
2. Aguiar, P.M.Q., Xavier, J., Stosic, M.: Globally optimal solution to exploit rigidity when recovering structure from motion under occlusions. In: IEEE International Conference on Image Processing (ICIP), pp. 197–200 (2008)
3. Aguiar, P.M.Q., Xavier, J., Stosic, M.: Spectrally optimal factorization of incomplete matrices. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008
4. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Nonrigid structure from motion in trajectory space. In: Neural Information Processing Systems, 2008
5. Broersen, P., Waele, S., Bos, R.: Application of autoregressive spectral analysis to missing data problems. IEEE Trans. Instrum. Meas. **53**, 981–986 (2004)
6. Buchanan, A., Fitzgibbon, A.W.: Damped Newton algorithms for matrix factorization with missing data. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 316–322 (2005)
7. Candes, E.J., Recht, B.: Exact low-rank matrix completion via convex optimization. In: Forty-Sixth Annual Allerton Conference, pp. 806–812 (2008)
8. Chen, P., Suter, D.: Recovering the missing components in a large noisy low-rank matrix: application to SFM. IEEE Trans. Pattern Anal. Mach. Intell. **26**, 1051–1063 (2004)
9. Costeira, J.P., Kanade, T.: A multibody factorization method for independently moving objects. Int. J. Comput. Vis. (IJCV) (3):159–179, 1998
10. Farid, H., Kosecká, J.: Estimating planar surface orientation using bispectral analysis. IEEE Trans. Image Process. **16**, 2154–2160 (2007)
11. Golub, G.H., Van Loan, C.F. (eds.): Matrix Computations. The Johns Hopkins University Press, Baltimore (1989)
12. Guerreiro, R.F.C., Aguiar, P.M.Q.: Estimation of rank deficient matrices from partial observations: two-step iterative algorithms. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR), pp. 450–466 (2003)
13. Haldar, J.P., Hernando, D.: Rank-constrained solutions to linear matrix equations using powerfactorization. IEEE Signal Process. Lett. **16**, 584–587 (2009)
14. Han, M., Kanade, T.: Reconstruction of a scene with multiple linearly moving objects. Int. J. Comput. Vis. (IJCV) **53**, 285–300 (2000)
15. Hartley, R., Schaffalitzky, F.: Powerfactorization: 3D reconstruction with missing or uncertain data. In: Australian-Japan Advanced Workshop on Computer Vision, 2003
16. Hayakawa, H.: Photometric stereo under a light source with arbitrary motion. Opt. Soc. Am. **11**, 3079–3089 (1994)
17. Irani, M.: Multi-frame optical flow estimation using subspace constraints. In: ICCV, vol. 1, pp. 626–633 (1999)
18. Jacobs, D.W.: Linear fitting with missing data for structure-from-motion. Comput. Vis. Image Underst. (CVIU) **1**, 57–81 (2001)
19. Jia, H., Fortuna, J., Martinez, A.: Perturbation estimation of the subspaces for structure from motion with noisy and missing data. In: Third International Symposium on 3D Data Processing, Visualization, and Transmission, pp. 1101–1007 (2006)
20. Julià, C., Sappa, A.D., Lumbreras, F., Serrat, J., López, A.: Rank estimation in 3d multibody motion segmentation. Electron. Lett. **44**, 279–280 (2008)
21. Kanatani, K.: Statistical optimization and geometric inference in computer vision. Philos. Trans., Math. Phys. Eng. Sci. **356**, 1303–1320 (1998)
22. Kanatani, K.: Motion segmentation by subspace separation and model selection. In: IEEE Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 586–591 (2001)
23. Ma, Y., Soatto, J., Kosecká, J., Sastry, S.: An Invitation to 3D Vision: From Images to Geometric Models. Springer, New York (2004)
24. Myrtveit, I., Stensrud, E., Olsson, U.: Analyzing data sets with missing data: an empirical evaluation of imputation methods and likelihood-based methods. IEEE Trans. Softw. Eng. **27**, 999–1013 (2001)
25. Okatani, T., Deguchi, K.: On the Wiberg algorithm for matrix factorization in the presence of missing components. Int. J. Comput. Vis. **72**, 329–337 (2007)
26. Poelman, C.J., Kanade, T.: A paraperspective factorization method for shape and motion recovery. IEEE Trans. Pattern Anal. Mach. Intell. **19**, 206–218 (1997)
27. Sarwar, M.S., Karypis, G., Konstan, J.A., Riedl, J.T.: Application of dimensionality reduction in recommender system—a case study. In: E-Commerce Workshop, pp. 309–324 (2000)
28. Thrun, S.: Affine structure from sound. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Advances in Neural Information Processing Systems, vol. 18, pp. 1353–1360. MIT Press, Cambridge (2006)
29. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. Int. J. Comput. Vis. **9**(2), 137–154 (1992)
30. Tron, R., Vidal, R.: A benchmark for the comparison of 3D motion segmentation algorithms. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2007
31. Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B.: Missing value estimation methods for DNA microarrays. Bioinformatics **17**, 1–6 (2001)
32. Vidal, R., Hartley, R.: Motion segmentation with missing data using powerfactorization and GPCA. In: IEEE Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 310–316 (2004)
33. Vidal, R., Ma, Y.: A unified algebraic approach to 2-D and 3-D motion segmentation and estimation. J. Math. Imaging Vis. **25**, 403–421 (2006)
34. Vidal, R., Tron, R., Hartley, R.: Multiframe motion segmentation with missing data using powerfactorization and GPCA. Int. J. Comput. Vis. **78**, 85–105 (2008)
35. Wiberg, T.: Computation of principal components when data is missing. In: Second Symposium of Computational Statistics, pp. 229–326 (1976)

36. Yan, J., Pollefeys, M.: A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In: European Conference on Computer Vision (ECCV), pp. 94–106 (2006)
37. Zelnik-Manor, L., Irani, M.: Degeneracies, dependencies and their implications in multi-body and multi-sequence factorization. In: IEEE Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 287–293 (2003)

Carme Julià received the BSc degree in Mathematics in 2002 from the Polytechnic University of Catalonia, Barcelona, Spain. She started her PhD studies in Computer Science in the Computer Vision Center, at the Universitat Autònoma de Barcelona in October 2002. In September 2004 she got the MSc degree and in 2008 the PhD degree with the thesis entitled *Missing Data Matrix Factorization Addressing the Structure from Motion Problem*. Since September 2008 she is a member of the Intelligent Robotics and Computer Vision Group, in the Universitat Rovira i Virgili, Tarragona, Spain. Her research interest is focused on the structure from motion problem through factorization techniques. She is currently working on non-rigid structure from motion with missing data and on image segmentation through photometric stereo.

Angel D. Sappa received the electromechanical engineering degree in 1995 from National University of La Pampa, Argentina, and the PhD degree in industrial engineering in 1999, from the Polytechnic University of Catalonia, Barcelona, Spain. In 2003, after holding research positions in France, the UK, and Greece, he joined the Computer Vision Center, Barcelona, Spain, where he is currently a senior researcher. He is a member of the Advanced Driver Assistance Systems Group. His research interests span a broad spectrum within the 2D and 3D image processing.

Felipe Lumbreras received the BSc degree in physics from the Universitat de Barcelona in 1991 and the PhD degree in computer science from the Universitat Autònoma de Barcelona in 2001. He is currently an associated professor in the Computer Science Department and member of the Centre de Visió per Computador (CVC). His research interest includes wavelet and texture analysis, 3D reconstruction, and computer vision for automotive applications.



Joan Serrat is associate lecturer at the computer science department of the Universitat Autònoma de Barcelona, and member of the Computer Vision Center. His areas of interest are structure from motion, 3D reconstruction and computer vision applied to driving assistance systems. He has also been head of several machine vision projects for local industries. He is member of the board of the IAPR spanish chapter.

Antonio López received the BSc degree in computer science from the Polytechnic University of Catalonia, Barcelona, Spain, in 1992 and the MSc degree in image processing and artificial intelligence and the PhD degree from the Universitat Autònoma de Barcelona (UAB) in 1994 and 2000, respectively. Since 1992, he has been giving lectures at the Computer Science Department, UAB, where he is currently an Associate Professor. In 1996, he participated in the foundation of the Computer Vision Center, UAB, where he has held different institutional responsibilities and is presently responsible for the research group on advanced driver-assistance systems by computer vision. He has been responsible for public and private projects. He has coauthored more than 50 papers, all in the field of computer vision.