Low-Rank Matrix Fitting Based on Subspace Perturbation Analysis with Applications to Structure from Motion

Hongjun Jia, *Student Member, IEEE,* and Aleix M. Martínez, *Member, IEEE*

Abstract

The task of finding a low-rank (r) matrix that best fits an original data matrix of higher rank is a recurring problem in science and engineering. The problem becomes especially difficult when the original data matrix has some missing entries and contains an unknown additive noise term in the remaining elements. The former problem can be solved by concatenating a set of r-column matrices which share a common, single r-dimensional solution space. Unfortunately, the number of possible submatrices is generally very large and, hence, the results obtained with one set of r-column matrices will generally be different from that captured by a different set. Ideally, we would like to find that solution which is least affected by noise. This requires that we determine which of the r-column matrices (i.e., which of the original feature points) are less influenced by the unknown noise term. This paper presents a criterion to successfully carry out such a selection. Our key result is to formally prove that the more distinct the r vectors of the r-column matrices are, the less they are swayed by noise. This key result is then combined with the use of a noise model to derive an upper-bound for the effect that noise and occlusions have on each of the r-column matrices. It is shown how this criterion can be effectively used to recover the noise-free matrix of rank r. Finally, we derive affine and projective structure from motion (SFM) algorithms using the proposed criterion. Extensive validation on synthetic and real data sets shows the superiority of the proposed approach over the state of the art.

Index Terms

Low-rank matrix, noise, missing data, random matrix, matrix perturbation, subspace analysis, structure from motion, computer vision, pattern recognition.

I. Introduction

MANY computer vision problems, as well as several others in computer graphics, pattern recognition and bioinformatics reduce to finding an appropriate low-rank matrix that successfully approximates the original data matrix [37], [28]. This problem becomes especially challenging when the original matrix contains noise and has several missing elements. One classical example is in the estimation of optical flow from video sequences, where several of the tracked fiducials can become occluded or be imprecisely detected (i.e., noisy measurements) [1], [14]. Other classical applications are in the recognition of faces using the so-called appearance-based approach [24], and in the classification of patients using microarray technology in medicine and bioinformatics [35], [9].

In this paper, we will focus on yet another classical problem – that of structure from motion (SFM). SFM is one of the fundamental problems in computer vision, with a large variety of applications [20], [12]. The SFM problem requires that we compute the 3D structure of an arbitrary scene from a set of 2D image point correspondences. These point correspondences are drawn from a set of images obtained by (usually) uncalibrated cameras. As above, the SFM problem becomes especially difficult when the feature points used to solve the correspondence problem cannot be precisely detected or become occluded for the duration of some frames. The former of these two problems is known as *data noise*, while the latter is usually referred to as *missing data*.

Manuscript received —; revised — —.

Hongjun Jia and Aleix M. Martínez are with the Department of Electrical and Computer Engineering at The Ohio State University, Columbus, OH 43210.

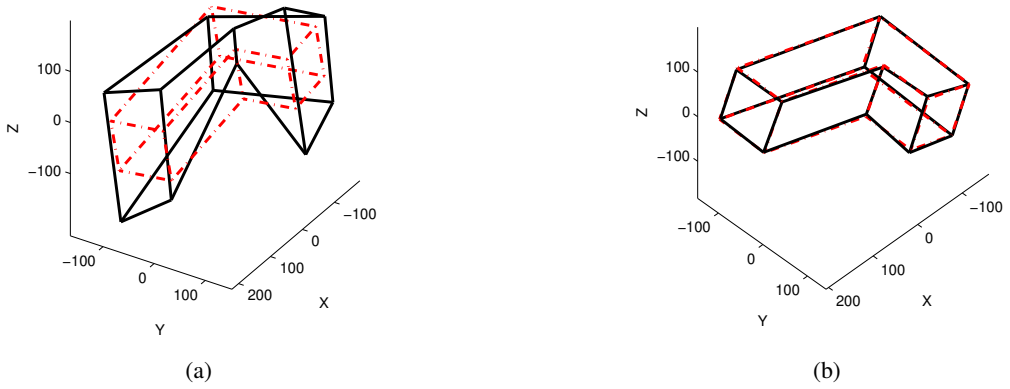


Fig. 1. The effects of noise on SVD. (a) The original data matrix is obtained by adding uniformally distributed Gaussian noise (with a standard deviation of 5). An additional error in one of the rows (with an extra 50% noise term) is then included. The dashed shape specifies the ground-truth; the solid shape the recovered result. (b) The recovered result obtained when the row containing most of the noise is eliminated before applying SVD.

The data noise problem can be stated as follows. Let the measurement matrix $\widehat{\mathbf{W}} \in \mathbb{R}^{m \times n}$ be the noisy version of an unknown matrix $\mathbf{W}$ of rank $r \ll \min(m, n)$. Here, m is a multiple of the number of frames through which each of the image feature points is being tracked (e.g., in affine SFM $m = 2q$, q the number of frames, whereas in projective SFM $m = 3q$), and n is the number of feature points. The noisy data matrix can be formally defined as

$$\widehat{\mathbf{W}} = \mathbf{W} + \mathbf{E}, \tag{1}$$

where $\mathbf{E}$ is the matrix containing the unknown noise terms. In general, the addition of the noise matrix $\mathbf{E}$ will enforce $\widehat{\mathbf{W}}$ to be of full-rank, while the original data matrix is known to be of a lower rank r. Hence, our objective is to find that rank-r matrix $\mathbf{W}_r$ that best approximates the noise-free data matrix $\mathbf{W}$, i.e, we wish to minimize the difference between $\mathbf{W}$ and $\mathbf{W}_r$. Unfortunately, $\mathbf{W}$ is not known and, hence, one is generally left to estimate $\mathbf{W}_r$ from $\widehat{\mathbf{W}}$ by means of an appropriate metric.

A typical solution to this problem is to employ the classical Singular Value Decomposition (SVD) [10]. The popularity of SVD is due to it providing the least squares error solution. Among other problems, this has been applied to affine SFM [34], [26], appearance-based recognition of faces [30] and objects [24], optical flow [1], [14], and microarray analysis [35], [9].

However, SVD does not generally work under the conditions of large noise. This is because the least-squares solution for factorization favors those feature vectors associated to the largest variances (i.e., the outliers). In many practical cases, the points carrying most of the variance are actually those associated with noise, since it is the noise which makes the variance increase. If one fails to eliminate the subset of column vectors of the measurement matrix carrying a large noise term, the SVD result (which is otherwise optimal in the least-squares sense) cannot guarantee a precise recovery of the 3D structure. Fig. 1 shows an example with one of the rows of $\widehat{\mathbf{W}}$ containing a large amount of noise. In this example the ground-truth (a 3D shape in the form of a letter L) is shown in (a) using dashed lines. The 3D shape recovered with SVD is delineated with solid lines, making it clear that the noisy row (i.e., the outlier) biases the whole result. In Fig. 1(b) we show the 3D shape recovered by SVD when the noisy row is deleted from $\widehat{\mathbf{W}}$. Here, we see that the recovered result is almost perfect. Note that this would in fact apply to any other method based on least-squares not just SVD.

Unfortunately, the amount of noise in each feature point is a priori unknown. Had the same amount of noise been uniformly distributed over all the points (i.e., following a Gaussian distribution), least-squares would have been optimal. However, in practice, for each given sequence, noise is attached to a particular

set of points, and no prior information on these or their structure is known. Even when the noise is i.i.d., all we know is that, on average, all points will be equally affected. This, however, does not provide much knowledge of the noise term in each image or video sequence. Hence, our *main goal* is to provide a good and useful estimate of which set of column vectors is (on average) less affected by the same noise term. This is *key* to enhancing the accuracy of the final result, because if we had such information, we would know which columns of the data matrix to use to reconstruct our object. Note that this approach does not require that we know the actual noise term in each column, but rather how the noise affects the recovery.

In this paper, we demonstrate that *the same amount of noise does not affect every data matrix* (or submatrix) *equally*. When a data matrix includes columns or rows defining very distinct vectors, the effect of noise is minimal. When the matrix rows or columns define very similar vectors though, the same amount of noise has a very large effect. We will prove this result formally and derive an upper-bound estimate of the effect of the noise term. This does not provide an optimal mechanism to reduce noise. Without any knowledge of the noise term, one can only hope to optimize a criterion. Our result shows that when no a priori information on the noise term is given, it is convenient to select those submatrices that are less swayed by the noise term.

Thus far, we have dealt with the problem caused by noise. The other typical problem addressed by researchers working in SFM is that of missing data caused by self-occlusions or by the incapacity of the tracking algorithm to successfully locate one or several of the feature points in some of the frames of our image sequence. In these cases, the measurement matrix becomes incomplete. To make this worse, and as already stated above, this incomplete data matrix will generally contain noise. Hence, we should extend the definition given in (1) to include the missing data case. To formally state this, we will define a set Γ representing all those cells in the data matrix $\widehat{\mathbf{W}}$ that correspond to the feature points which are visible and properly detected (tracked) in each of the frames of the image sequence. This set is thus defined as

$$\Gamma = \{(i,j)|(i,j)\text{successfully detected 2D visible points}\}. \tag{2}$$

Here, the j component specifies the column of $\widehat{\mathbf{W}}$, corresponding to one of the coordinates of the feature point visible in frame i. The non-missing elements in the measurement matrix $\widehat{\mathbf{W}} = [\hat{w}_{ij}]$ can now be formally defined as

$$\hat{w}_{ij} = w_{ij} + e_{ij}, \forall (i,j) \in \Gamma,$$

where $\mathbf{E} = [e_{ij}]$ is the noise matrix, and $[(\cdot)_{ij}]$ denotes a $m \times n$ matrix with the $(i,j)^{th}$ entry as $(\cdot)_{ij}$. Recall that in projective SFM, the third coordinate of the feature points is the homogenous coordinate. We note that these do not carry a noise term. In our formulation given above, this means that the e_{ij} representing a homogenous coordinate will always need to be zero.

To resolve the problem caused by missing data, Jacobs [16] proposes to construct a set of submatrices by randomly selecting three columns from the data matrix $\widehat{\mathbf{W}}$. If there was no missing data, any three columns would define the solution space. However, if some of the cells of the three randomly selected columns are missing, several solutions will be possible. By combining the solutions observed with a sufficient number of different triple-column submatrices, we can obtain a full reconstruction. This process can also be used to initialize global approaches, generally resulting in more accurate global fits. The main problem with this approach is that, due to the randomness of the triple-column selection process, the algorithm does not always produce a consistent and accurate recovery. Also, since the selection of the three columns is random, there is no mechanism to know whether the selected columns carry most of the noise or not. It is our contention that the selection of the columns constituting the submatrix should be carried out on the basis of how noise affects them and on the number of missing entries. In this paper, we propose to first sort the r-column submatrices based on an estimate of the effect that noise and occlusion have on them. Then, we select a sufficient number of submatrices to reconstruct the entire object.

After a more formal, in-depth presentation of the problem in Section II, we provide detailed derivations of our approach in Section III. This is an extension of our preliminary work presented in [17]. In Section IV, we show how the derived algorithm can be efficiently used to resolve the problems of noise and

missing data in affine and projective SFM. We show that our SFM algorithm can recover the position of the 2D image points and corresponding 3D feature points with high accuracy. This is so even when the point has a large noise term or when it has been temporarily occluded. Experimental results are in Section V. We conclude in Section VI.

II. Fitting a Low-rank Matrix with Missing Data

As summarized in the preceding section, missing matrix entries require that we redefine approaches such as SVD. Several solutions to this problem have been proposed over the years. We start by summarizing two of the main approaches defined to date and state the necessary techniques we need to use to formulate the proposed solution.

A. The direct fitting method

In SFM we need to find an appropriate low-rank matrix which contains the information of the structure and motion of the object being tracked. This suggests a direct approach where we search for those missing entries that maintain a low rank measurement matrix. This low-rank constraint means that only the first r singular values of $\mathbf{W}$ can be non-zero. Since r is usually known (e.g., four in the general SFM problem), one can derive approaches for filling in the missing elements [6]. Two such solutions are proposed by Friedland et al. [9] and Troyanskaya et al. [35]. In these algorithms, the authors first fill in the missing elements with zeros, and then utilize SVD to find that r-dimensional subspace that best fits the data. This allows the authors to project the data matrix onto the subspace, resulting in a new low-rank representation. The measurement matrix can now be reconstructed using the basis vectors selected by this process. This algorithm can be iterated, yielding better least-squares estimates of the missing elements. In a related paper, Shum et al. [29] propose to first carry the classical decomposition of the measurement matrix $\widehat{\mathbf{W}}$ into two matrices, one describing the camera motion $\mathbf{P}$ and the other the object's shape $\mathbf{Q}$, using standard SVD but with zeros or average or random values in place of the missing elements. This decomposition allows for the definition of two optimization approaches. One is to optimize $\mathbf{P}$ by keeping $\mathbf{Q}$ fixed. The second optimization requires to fix $\mathbf{P}$ and solve for $\mathbf{Q}$. This trick reduces the original bilinear problem to two linear ones where the goal is to minimize the norm of the difference between the measurement matrix and its SVD reconstruction when only using the non-missing elements of the data matrix. This solutions reduces to a weighted least-squares problem, which can be iterated until convergence.

The methods described in this section are related to earlier extensions of SVD with missing elements. One particular case is defined by Wiberg [39], whose derivations also provide the minimum number of observations needed to get a unique solution. More recently, EM-based extensions of SVD and PCA have been proposed to address the problem of missing elements [27], [33], [36], [11]. In [5], this is further improved with robust statistics. And, it has been shown that projection pursuit could also be employed, since this compares favorably with other robust estimators [19]. Unfortunately, in general, the approaches defined in this section can only guarantee convergence to a local minimum.

Random Sample Consensus (RANSAC) [7] is a well-known, robust method to deal with outliers. In RANSAC-like procedures, a fixed number of data units are randomly selected to fit a model which will be measured over all the data. If there are no missing components, any r-column submatrix will span a r-dimensional subspace. However, in the case with missing data, the fixed number of columns which can fit a complete r-dimensional subspace is no longer available. In such a case, we would need to first eliminate the rows with missing elements.

B. The subspace constraint

To resolve the issues that arise from the methods described in the preceding section, one can use additional constraints. For example, the factorization of the low-rank matrix, $\mathbf{W} = \mathbf{P} \cdot \mathbf{Q}$, implies another possible solution by defining a subspace constraint: the spanning space of the column vectors of $\mathbf{W}$ and

$\mathbf{P}$ should be identical. However, this constraint only makes sense when we have noise-free entries. To deal with noise, Jacobs [15], [16] proposes an approach where the subspace constraint can be derived from several submatrices. Jacob's approach is to then combine these local solutions to find the global one. To achieve this, each column of the measurement matrix is regarded as the coordinates of a point in a m-dimensional space. SVD can then be used to find the best 3D subspace, $\mathcal{W}$, fitting the n available points. When there is neither noise nor missing data, $\mathcal{W}$ is the space spanned by any three linearly independent columns. If there are some missing elements in the measurement matrix, each column spans an affine subspace that accounts for all the possible missing elements. In this case, $\mathcal{W}$ lies in the space spanned by three such affine subspaces. By following this argument and letting $\mathcal{B}_k$ be the space spanned by the k^{th} column triplet, we have $\mathcal{W} \subseteq \mathcal{B}_k$. Then, $\mathcal{W}$ should be a subset of the intersection of all possible $\mathcal{B}_k$,

$$\mathcal{W} \subseteq \mathcal{G} = \cap \mathcal{B}_k, \quad k = 1, 2, \cdots, l,$$

where l is the number of all possible $\mathcal{B}_k$.

If noise is introduced into the equation and we follow our previous notation, which uses the symbol $\widehat{\ }$ to specify the corresponding noisy versions (e.g., $\widehat{\mathcal{G}}$ and $\widehat{\mathcal{W}}$), then, $\widehat{\mathcal{G}}$ will become empty because our target $\widehat{\mathcal{W}}$ cannot accurately lie in any $\widehat{\mathcal{B}}_k$. A null-space based method is used to solve this problem. All the matrix representations of the orthogonal complementary space of $\widehat{\mathcal{B}}_k$ are packed together to form a matrix representation of $\widehat{\mathcal{G}}^{\perp}$. This can now be decomposed using standard SVD, providing the least-squares solution. The three singular vectors corresponding to the three smallest singular values are selected to form a 3D linear space $\widehat{\mathcal{W}}$ to be orthogonal to the matrix which is considered to be closest to $\mathcal{W}$. The affine shape of the original structure is thus recovered from $\widehat{\mathcal{W}}$ [15], [16]. This approach falls within the area of "subset selection," where a set of columns is selected to generate a solution. A review and variants of this approach can be found in [23].

Jacobs' solution is an elegant way to deal with missing data. However, in practice, we see that the recovered results vary extensively when measured by the Mean Square Error (MSE). The reason for this is simple. When the triple-columns carrying the least amount of noise are selected, the recovered shape will generally be very close to the ground-truth and the MSE will be small. Unfortunately, if one or more of the triple-columns carrying large amounts of noise are used, the result will be far from optimal, leading to a large MSE. Therefore, the remaining problem to be addressed within this framework is to find a criterion that determines which triple-columns are associated to less noise and thus are the best candidates for the algorithm. Chen and Suter propose one such criterion in [4], where the fitness of each column is inversely proportional to the number of missing elements. This means that those columns with less missing elements are better candidates for reconstruction. This works well when the noise is evenly distributed. However, in many instances the columns with more missing entries are precisely those carrying the least amount of noise or, equivalently, those less affected by it. In these cases, Chen and Suter's approach would result in large MSEs. Our experimental results will show that indeed, many times, the recovery obtained from those columns with more missing elements carry a lower MSE. This is not to cast any aspersions of Jacobs and Chen and Suter approaches. These are very general and may be used in a large variety of problems. Our goal in the remaining of the paper is to propose an alternate approach and show that it generally provides better fits in SFM problems. The proposed criterion is based on the subset selection approach [23] as it was done in [16], and does not necessarily extend to the other methods described above. Also, the derivations provided below are for the singular value decomposition of the submatrices. As detailed in [40], several of these results should extend to the eigen-decomposition as well.

III. Deviation Parameter Criterion

A new method of determining which subset of the data is most suitable for fitting the low-rank matrix is introduced. This method only depends on the subset itself and does not require of a precalculation of the result as it would be the case in robust statistics [41].

A. Selecting the appropriate submatrices

The problem we need to address is that of determining which of the (triple-column) submatrices are less affected by the noise term. We first note that each of these submatrices defines a subspace. If the 3D subspace given by the complete (in the sense of not having any missing entries), noise-free data matrix $\mathbf{W}$ were known, then our task would simplify to finding those submatrices spanning subspaces similar to that of $\mathbf{W}$. In such an idealistic case, we would still need to define a mechanism that can compute the distance between two spaces – that of the ground-truth $\mathcal{W}$ and that of the i^{th} submatrix $\widehat{\mathcal{B}}_i$ (with $i = 1, \ldots, l$). Since the dimensionality of these subspaces is identical, we can compute the actual distance between them, $dist(\mathcal{W}, \widehat{\mathcal{B}}_i)$, by looking at the largest principal angle. More formally, if the dimensionality of our subspaces is r, then the principal angles $0 \leq \theta_{i,1} \leq \cdots \leq \theta_{i,r} \leq \pi/2$ between $\mathcal{W}$ and $\widehat{\mathcal{B}}_i$ can be obtained recursively from

$$cos(\theta_{i,k}) = \max_{\mathbf{u} \in \mathcal{W}} \max_{\mathbf{v} \in \widehat{\mathcal{B}}_i} \mathbf{u}^T \mathbf{v} = \mathbf{u}_{i,k}^T \mathbf{v}_{i,k},$$

with the added constraints $\|\mathbf{u}\| = \|\mathbf{v}\| = 1$, $\mathbf{u}^T \mathbf{u}_{i,h} = 0$, $\mathbf{v}^T \mathbf{v}_{i,h} = 0$, $h = 1, \ldots, k-1$, $1 \leq k \leq r$. Then, the distance between subspaces is $dist(\mathcal{W}, \widehat{\mathcal{B}}_i) = \sin(\theta_{i,r})$ [10]. For simplicity of notation, we will use $\theta(\mathbf{X}, \mathbf{Y})$ to specify the largest principal angle between the spaces defined by the matrices $\mathbf{X}$ and $\mathbf{Y}$. Also, we will refer to the space spanned by $\mathbf{X}$ as $\mathcal{X}$, and the largest principal angle between two spaces as $\theta(\mathcal{X}, \mathcal{Y})$.

In actuality, the space $\mathcal{W}$ defining the ground-truth is not known. Therefore, we need to find another mechanism to determine how noise influences each subspace $\widehat{\mathcal{B}}_i$. To this end, we first note that the same amount of noise does not affect every submatrix defining a given subspace $\widehat{\mathcal{B}}_i$ equally. In fact, when the vectors given by the columns of our submatrix are separated by a large angle (e.g., close to 90°), additive noise will have a limited effect. However, *when the same noise is added to a submatrix with similar column vectors, the new resulting (noisy) subspace will be more different from the original noise-free version than the effect observed in submatrices with very distinct column vectors*.

To clarify this point, one can think of the effect that noise has in a stereo vision system. When using two images describing a similar view of the same scene, noise will have a greater sway than that observed when the views are far apart. In other words, the 3D reconstruction will be generally less affected by noise when the vectors describing the scene correspond to sufficiently different views. This is so because a small amount of noise will correspond to a large percentage of the difference between two similar vectors but a small percentage in those that are far apart. This result will be formally proven next.

B. Subspace perturbation analysis

Since our approach is that of determining which submatrices are best to be employed based on their robustness to noise, our criterion needs to be related to the matrix sensitivity to noise. Subspace perturbation analysis [31] provides a way to formally derive a solution.

To this end, let the matrix $\widehat{\mathbf{W}}$ be a perturbed version of the noise-free matrix $\mathbf{W} \in \mathbb{R}^{m \times n}$ with some missing elements, and let the rank of $\mathbf{W}$ be $r \ll \min(m, n)$. The perturbing matrix $\mathbf{E}$ is considered to be the additive noise in the observation of $\mathbf{W}$ in (1). As stated above, a classical approach to recover $\mathbf{W}$ is to find that matrix $\mathbf{W}_r$ of rank r which minimizes the difference between itself and $\mathbf{W}$. A convenient norm to calculate this difference is the Frobenius norm calculated over all non-missing elements of the matrix, $\|\mathbf{W} - \mathbf{W}_r\|_{F\_nonmissing}$.

If we select a r-column submatrix $\widehat{\mathbf{B}}_i$ $(i = 1, 2, \cdots, l)$ from $\widehat{\mathbf{W}}$ and follow the steps of Jacobs algorithm [16] presented in Section II-B, we get the null space of $\widehat{\mathbf{B}}_i$ which, if correct, should be orthogonal to the desired low-dimensional space $\widehat{\mathcal{W}}$. After selecting a sufficient number of such submatrices, we build a large matrix which is composed of all these null-spaces. SVD can then be used to select the r singular vectors corresponding to the r smallest singular values. This generates a r-dimensional linear subspace

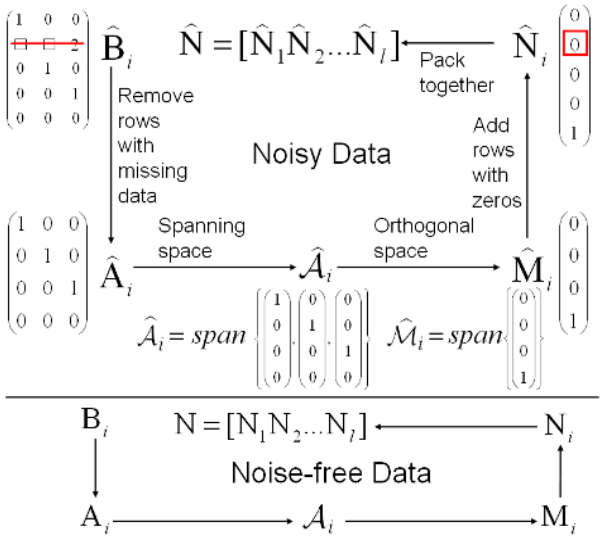


Fig. 2. Shown here is the relation between each pair of matrices in the process followed by the subspace constraint approach. The top part illustrates the relation between the noisy matrices, while the bottom part illustrates the noise-free case. In this figure, the empty squares in the matrix correspond to missing elements. Note that $\widehat{\mathbf{M}}_i^T \widehat{\mathbf{A}}_i = \mathbf{0}$ and $\widehat{\mathbf{N}}_i^T \widehat{\mathbf{B}}_i = \mathbf{0}$.

$\widehat{\mathcal{W}}$. This linear subspace can be utilized to recover the missing data and reconstruct a rank-r matrix, as we will see shortly.

For each $\widehat{\mathbf{B}}_i$, let $\widehat{\mathbf{A}}_i \in \mathbb{R}^{p \times r}$ $(p \leq m)$ be the corresponding reduced form of $\widehat{\mathbf{B}}_i$ with no missing entries. This, we accomplish by removing all the rows having at least one missing entry. The spanning space of $\widehat{\mathbf{A}}_i$ is $\widehat{\mathcal{A}}_i$, with $\widehat{\mathcal{M}}_i$ its null space and $\widehat{\mathbf{M}}_i$ the $p \times (p - r)$ matrix defining it. We can then expand $\widehat{\mathbf{M}}_i$ by simply adding zeros in all the rows which were removed from $\widehat{\mathbf{B}}_i$. This provides us with an original size matrix, $\widehat{\mathbf{N}}_i \in \mathbb{R}^{m \times (p-r)}$, representing the null space. This process is illustrated in Fig. 2. All such $\widehat{\mathbf{N}}_i$ will then be packed together to form the matrix $\widehat{\mathbf{N}} = [\widehat{\mathbf{N}}_1 \widehat{\mathbf{N}}_2 \cdots \widehat{\mathbf{N}}_l]$. Our low-dimensional linear space $\widehat{\mathcal{W}}$ will be orthogonal to the rank-$(n - r)$ matrix closest to $\widehat{\mathbf{N}}$ as given by the Frobenius norm. Note that the subspace will be correctly recovered only when all the rows are considered at least in one of the $\widehat{\mathbf{A}}_i$ matrices.

Had we applied the process just defined to the noise-free condition, we would have obtained the matrices $\mathbf{B}_i$, $\mathbf{A}_i$, $\mathbf{M}_i$, $\mathbf{N}_i$, and their corresponding space (e.g., $\mathcal{A}_i$ and $\mathcal{M}_i$), Fig. 2. Since $\widehat{\mathbf{N}}$ is the final matrix to be decomposed by SVD, the difference between $\widehat{\mathbf{N}}$ and $\mathbf{N}$ should be small. Similarly, we also require the difference between each $\widehat{\mathbf{N}}_i$ and $\mathbf{N}_i$ to be as small as possible. Since $\mathbf{N}_i$ and $\mathbf{M}_i$ and $\widehat{\mathbf{N}}_i$ and $\widehat{\mathbf{M}}_i$ are the same except for added zeros, the principal angles between $\mathbf{N}_i$ and $\widehat{\mathbf{N}}_i$ are the same as that between $\mathbf{M}_i$ and $\widehat{\mathbf{M}}_i$. Also, since $\mathcal{A}_i$ and $\mathcal{M}_i$ and $\widehat{\mathcal{A}}_i$ and $\widehat{\mathcal{M}}_i$ are both orthogonal to each other, the principal angle between $\mathcal{M}_i$ and $\widehat{\mathcal{M}}_i$ will be the same as that between $\mathcal{A}_i$ and $\widehat{\mathcal{A}}_i$. Therefore, we have $\theta(\mathbf{N}_i, \widehat{\mathbf{N}}_i) = \theta(\mathbf{M}_i, \widehat{\mathbf{M}}_i) = \theta(\mathbf{A}_i, \widehat{\mathbf{A}}_i)$ and we can calculate the distance between subspaces directly from $\mathbf{A}_i$ and $\widehat{\mathbf{A}}_i$, i.e., $dist(\mathbf{N}_i, \widehat{\mathbf{N}}_i) = dist(\mathbf{A}_i, \widehat{\mathbf{A}}_i) = \sin\theta(\mathcal{A}_i, \widehat{\mathcal{A}}_i)$.

The process defined in the preceding paragraph allows us to calculate the distance between the noise-free and noisy versions of the original data matrix with missing elements, by concentrating on spaces spanned by the matrices $\mathbf{A}_i$ and $\widehat{\mathbf{A}}_i$ as

$$pb(\mathbf{B}_i, \widehat{\mathbf{B}}_i) = \sin\theta(\mathcal{R}(\widehat{\mathbf{A}}_i), \mathcal{R}(\mathbf{A}_i)) = \sin\theta(\mathcal{A}_i, \widehat{\mathcal{A}}_i), \tag{3}$$

where $\mathcal{R}(\mathbf{X})$ denotes the range space $\mathcal{X}$ spanned by the column vectors in $\mathbf{X}$, and pb stands for perturbation.

We note that, in the process described above, the farther the space of $\widehat{\mathbf{N}}$ is from that of $\mathbf{N}$, the farther apart the recovered low-dimensional subspace $\widehat{\mathcal{W}}$ will be from the ground-truth $\mathcal{W}$. Hence, we want

to choose those $\widehat{\mathbf{N}}_i$ that defines the smallest of all possible largest principal angles $\theta(\mathbf{N}_i, \widehat{\mathbf{N}}_i)$. *This is equivalent to selecting those r-column submatrices $\widehat{\mathbf{B}}_i \in \mathbb{R}^{m\times r}$ with the smallest $pb(\mathbf{B}_i, \widehat{\mathbf{B}}_i)$ values.*

Since the ground-truth is not known, we need to resort to some other type of comparison. As argued in the previous section, those submatrices $\widehat{\mathbf{B}}_i$ with most dissimilar column vectors will be generally less affected by additive noise. The framework derived in this section enables us to prove this result formally. Moreover, note the deleted rows do not directly relate to the calculated pb value. In fact, this is not necessary because these rows will not enter in the computation of $\widehat{N}_i$ and, hence, have no effect in the final result.

C. Upper-bound for the subspace distance

Let $\widehat{\mathbf{B}} \in \mathbb{R}^{m\times r}$ represent one of the r-column submatrices from $\widehat{\mathbf{W}}$, and let $\widehat{\mathbf{A}} \in \mathbb{R}^{p\times r}$ ($p \leq m$) denote its complete part, which, as above, we construct by deleting all rows with at least one missing element. Further, let $\mathbf{E}_{\mathbf{A}} \in \mathbb{R}^{p\times r}$ be the noise matrix defining the noise term on $\widehat{\mathbf{A}}$. The matrix $\mathbf{E}_{\mathbf{A}}$ can be obtained directly from $\mathbf{E}$ by deleting the same rows and columns that were eliminated to convert $\widehat{\mathbf{W}}$ into $\widehat{\mathbf{A}}$.

We begin by determining a bound on the distance between $\mathcal{R}(\widehat{\mathbf{A}})$ and $\mathcal{R}(\mathbf{A})$, where $\mathbf{A}$ denotes the corresponding noise-free version of $\widehat{\mathbf{A}}$, in terms of some $f(\widehat{\mathbf{A}})$ and $\|\mathbf{E}_{\mathbf{A}}\|_2$, where $\|\cdot\|_2$ denotes the 2-norm.

To do this, we extend on a perturbation theorem provided by Wedin [38] to determine a bound on the pb value between $\widehat{\mathbf{B}}$ and $\mathbf{B}$, which is equal to the sine of the largest principal angle between the spanning spaces of $\mathbf{A}$ and $\widehat{\mathbf{A}}$. Before the theorem is presented, we need to introduce some definitions. Let

$$\widehat{\mathbf{A}} = \mathbf{A} + \mathbf{E}_{\mathbf{A}},$$

with SVDs $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ and $\widehat{\mathbf{A}} = \widehat{\mathbf{U}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{V}}^T$. $\mathbf{A}$ and $\widehat{\mathbf{A}}$ can be decomposed as

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix} \begin{bmatrix}\mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2\end{bmatrix}^T \\ \widehat{\mathbf{A}} &= \begin{bmatrix}\widehat{\mathbf{U}}_1 & \widehat{\mathbf{U}}_2\end{bmatrix} \begin{bmatrix}\widehat{\mathbf{\Sigma}}_1 & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{\Sigma}}_2 \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \begin{bmatrix}\widehat{\mathbf{V}}_1 & \widehat{\mathbf{V}}_2\end{bmatrix}^T \quad , \end{aligned} \tag{4}$$

where $\mathbf{U}_1, \widehat{\mathbf{U}}_1 \in \mathbb{R}^{p\times s}$, $\mathbf{U}_2, \widehat{\mathbf{U}}_2 \in \mathbb{R}^{p\times(p-s)}$, $\mathbf{V}_1, \widehat{\mathbf{V}}_1 \in \mathbb{R}^{r\times s}$, $\mathbf{V}_2, \widehat{\mathbf{V}}_2 \in \mathbb{R}^{r\times(r-s)}$, $s \leq r$, $\mathbf{\Sigma}_1 = \text{diag}(\sigma_1, \cdots, \sigma_s)$, $\mathbf{\Sigma}_2 = \text{diag}(\sigma_{s+1}, \cdots, \sigma_r)$, $\widehat{\mathbf{\Sigma}}_1 = \text{diag}(\hat{\sigma}_1, \cdots, \hat{\sigma}_s)$ and $\widehat{\mathbf{\Sigma}}_2 = \text{diag}(\hat{\sigma}_{s+1}, \cdots, \hat{\sigma}_r)$.

In the theorem that follows, the representation of $\mathbf{E}_{\mathbf{A}}$ in the orthonormal subspace $\widehat{\mathbf{V}}_1$ and $\widehat{\mathbf{U}}_1$ is used, rather than $\mathbf{E}_{\mathbf{A}}$ directly, since we are defining bounds for subspaces. As such, define

$$\begin{aligned} \mathbf{R} &= \mathbf{A}\widehat{\mathbf{V}}_1 - \widehat{\mathbf{U}}_1\widehat{\mathbf{\Sigma}}_1 = -\mathbf{E}_{\mathbf{A}}\widehat{\mathbf{V}}_1 \\ \mathbf{D} &= \mathbf{A}^T\widehat{\mathbf{U}}_1 - \widehat{\mathbf{V}}_1\widehat{\mathbf{\Sigma}}_1 = -\mathbf{E}_{\mathbf{A}}^T\widehat{\mathbf{U}}_1. \end{aligned}$$

From these definitions we note that

$$\begin{aligned} \|\mathbf{R}\| &= \|\mathbf{E}_{\mathbf{A}}\widehat{\mathbf{V}}_1\| \leq \|\mathbf{E}_{\mathbf{A}}\| \\ \|\mathbf{D}\| &= \|\mathbf{E}_{\mathbf{A}}^T\widehat{\mathbf{U}}_1\| \leq \|\mathbf{E}_{\mathbf{A}}\|, \end{aligned} \tag{5}$$

where $\|\cdot\|$ represents an appropriate norm, such as the 2-norm or the Frobenius norm. We can now state Wedin's theorem [38] as follows.

Theorem 1: If $\exists \alpha, \delta > 0$ such that

$$\min \sigma(\widehat{\mathbf{\Sigma}}_1) \geq \alpha + \delta \quad \textit{and} \quad \max \sigma(\mathbf{\Sigma}_2) \leq \alpha,$$

then

$$\max\{\|\sin \mathbf{\Phi}\|, \|\sin \mathbf{\Theta}\|\} \leq \frac{\max\{\|\mathbf{R}\|, \|\mathbf{D}\|\}}{\delta},$$

where $\mathbf{\Phi}$ *is a matrix of angles between* $\mathcal{R}(\mathbf{U}_1)$ *and* $\mathcal{R}(\widehat{\mathbf{U}}_1)$, $\mathbf{\Theta}$ *is a matrix of angles between* $\mathcal{R}(\mathbf{V}_1)$ *and* $\mathcal{R}(\widehat{\mathbf{V}}_1)$, *and the operator* $\sigma(\cdot)$ *denotes the singular value spectrum. Here,* $\sin\mathbf{\Phi} = \mathbf{U}_2^T\widehat{\mathbf{U}}_1$ *and* $\sin\mathbf{\Theta} = \mathbf{V}_2^T\widehat{\mathbf{V}}_1$.

Returning to the original problem, to determine a bound on the distance between the spanning spaces $\mathcal{R}(\widehat{\mathbf{A}})$ and $\mathcal{R}(\mathbf{A})$, we use the sine of the largest principal angle between $\mathcal{R}(\widehat{\mathbf{A}})$ and $\mathcal{R}(\mathbf{A})$ to measure the distance between two submatrices, defined as $\sin\theta(\mathcal{R}(\widehat{\mathbf{A}}), \mathcal{R}(\mathbf{A}))$. Using the inequality in (5), we have

$$\begin{aligned}\sin\theta(\mathcal{R}(\widehat{\mathbf{A}}), \mathcal{R}(\mathbf{A})) &\leq \max\{\|\sin\mathbf{\Phi}\|, \|\sin\mathbf{\Theta}\|\} \\ &\leq \frac{\max\{\|\mathbf{R}\|, \|\mathbf{D}\|\}}{gap} \leq \frac{\|\mathbf{E_A}\|}{gap},\end{aligned} \tag{6}$$

where $\mathbf{\Phi}$ now represents the matrix of angles between $\mathcal{R}(\mathbf{A})$ and $\mathcal{R}(\widehat{\mathbf{A}})$, and $\mathbf{\Theta}$ is a matrix of angles between $\mathcal{R}(\mathbf{A}^T)$ and $\mathcal{R}(\widehat{\mathbf{A}}^T)$. From the conditions in Theorem 1, it follows $gap = \min\sigma(\widehat{\mathbf{\Sigma}}_1) - \max\sigma(\mathbf{\Sigma}_2)$, with $\mathbf{\Sigma}_2$ and $\widehat{\mathbf{\Sigma}}_1$ the diagonal matrices of the singular values of $\mathbf{A}$ and $\widehat{\mathbf{A}}$ as shown in (4).

Finally, for the case where $p > r$, the first r left singular vectors of $\widehat{\mathbf{U}}$ will span $\mathcal{R}(\widehat{\mathbf{A}})$ and the remaining singular vectors will have corresponding singular values equal to 0. To illustrate, consider the following SVD for $p > r$, rewritten with a square matrix of singular values

$$\mathbf{A} = \begin{bmatrix}\mathbf{U}_1' & \mathbf{U}_2'\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma}_1' & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2'\end{bmatrix}\begin{bmatrix}\mathbf{V}_1' & \mathbf{V}_2'\end{bmatrix}^T, \tag{7}$$

where $\mathbf{U}_1' \in \mathbb{R}^{p\times r}$, $\mathbf{U}_2' \in \mathbb{R}^{p\times(p-r)}$, $\mathbf{V}_1' \in \mathbb{R}^{r\times r}$, $\mathbf{V}_2' \in \mathbb{R}^{r\times(p-r)}$, $\mathbf{\Sigma}_1' = \text{diag}(\sigma_1, \cdots, \sigma_r)$, and $\mathbf{\Sigma}_2' = \text{diag}(\sigma_{r+1}, \cdots, \sigma_p)$. Similarly the decomposition of $\widehat{\mathbf{A}}$ is

$$\widehat{\mathbf{A}} = \begin{bmatrix}\widehat{\mathbf{U}}_1' & \widehat{\mathbf{U}}_2'\end{bmatrix}\begin{bmatrix}\widehat{\mathbf{\Sigma}}_1' & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{\Sigma}}_2'\end{bmatrix}\begin{bmatrix}\widehat{\mathbf{V}}_1' & \widehat{\mathbf{V}}_2'\end{bmatrix}^T, \tag{8}$$

where $\widehat{\mathbf{U}}_1' \in \mathbb{R}^{p\times r}$, $\widehat{\mathbf{U}}_2' \in \mathbb{R}^{p\times(p-r)}$, $\widehat{\mathbf{V}}_1' \in \mathbb{R}^{r\times r}$, $\widehat{\mathbf{V}}_2' \in \mathbb{R}^{r\times(p-r)}$, $\widehat{\mathbf{\Sigma}}_1' = \text{diag}(\hat{\sigma}_1, \cdots, \hat{\sigma}_r)$ and $\widehat{\mathbf{\Sigma}}_2' = \text{diag}(\hat{\sigma}_{r+1}, \cdots, \hat{\sigma}_p)$.

Comparing (7) and (8) to (4) with $s = r$, yields $\mathbf{U}_1' = \mathbf{U}_1$, $\mathbf{U}_2' = \mathbf{U}_2$, $\mathbf{\Sigma}_1' = \mathbf{\Sigma_1}$, $\mathbf{\Sigma}_2' = \mathbf{0}$, $\mathbf{V}_1' = \mathbf{V}_1$ and $\mathbf{V}_2' = \mathbf{I}$. From this result, we see that the condition in Wedin's theorem is $\max\sigma(\mathbf{\Sigma}_2') = 0$ and $\min\sigma(\widehat{\mathbf{\Sigma}}_1') = \min\sigma(\widehat{\mathbf{A}})$. And, hence, $gap = \min\sigma(\widehat{\mathbf{\Sigma}}_1') - \max\sigma(\mathbf{\Sigma}_2') = \min\sigma(\widehat{\mathbf{A}})$. Finally, from (6) we get our expression for the bound on the distance between $\mathcal{R}(\mathbf{A})$ and $\mathcal{R}(\widehat{\mathbf{A}})$ as

$$\sin\theta(\mathcal{R}(\widehat{\mathbf{A}}), \mathcal{R}(\mathbf{A})) \leq \frac{\|\mathbf{E_A}\|}{\min\sigma(\widehat{\mathbf{A}})}. \tag{9}$$

D. The deviation parameter criterion

In order to use the perturbation bound derived in (9), we must have some knowledge of the nature of the noise matrix $\mathbf{E}$. Since this cannot be measured directly, we will resort to a statistical model. Specifically, we assume that the elements of the noise matrix are Gaussian distributed according to $N(0, \sigma^2)$ and are independent of each other. This is a reasonable characterization of noise that arises from inaccurate measurements, i.e., it is most probable that the measurement will be close to the actual value.

Let $\mathbf{X}$ be a $p \times r$ Gaussian random matrix with entries distributed according to $N(0, 1)$. As shown by Johnstone [18], the mean and variance of the largest eigenvalue λ_1 of the covariance matrix $\mathbf{X}^T\mathbf{X}$ can be approximated by

$$\begin{aligned}\mu_{\lambda_1} &= (\sqrt{p-1} + \sqrt{r})^2 \\ \sigma_{\lambda_1} &= (\sqrt{p-1} + \sqrt{r})\left(\frac{1}{\sqrt{p-1}} + \frac{1}{\sqrt{r}}\right)^{\frac{1}{3}}.\end{aligned} \tag{10}$$

Given the fact that λ_1 is the square of the largest singular value of $\mathbf{X}$ and the largest singular value of a matrix is its 2-norm, the expected value of the 2-norm of our error matrix $\mathbf{E_A} \in \mathbb{R}^{p\times r}$ with elements distributed as $N(0,\sigma^2)$ is

$$\mu_{\|\mathbf{E_A}\|} \approx (\sqrt{p-1}+\sqrt{r})\sigma. \tag{11}$$

It is important to note from (10) that while the estimate of the mean increases almost linearly with the dimensionality of the matrix, the standard deviation increases at a lower rate. This means that the ratio of standard deviation over mean, which indicates the percentage of error in the estimate of the mean, decreases as the size of the matrix increases. This is relevant because, in most applications, the size of our matrices is quite large. And, in such cases, it becomes appropriate to substitute the 2-norm of our matrix by the mean derived in (10) as was done in (11).

Now considering the results in (3), (9) and (11) together, we obtain the expectation of the upper-bound of $pb(\mathbf{B},\widehat{\mathbf{B}})$. We refer to this value as the Deviation Parameter (DP) of submatrix $\widehat{\mathbf{B}} \in \mathbb{R}^{m\times r}$ with b rows of missing data ($b = m - p$), and define it as

$$DP(\widehat{\mathbf{B}}) = \frac{(\sqrt{m-b-1}+\sqrt{r})\sigma}{\min(\sigma(\widehat{\mathbf{A}}))}. \tag{12}$$

In this result, σ represents the variance of the noise given by the 2-norm of the noise matrix $\mathbf{E_A}$. As mentioned earlier, it is common to assume that the norm of the noise in each submatrix is identical – although the distribution of the noise in each column/row can vary considerably from matrix to matrix. Under this condition, all σ's are the same and can thus be eliminated from the computation of the DP criterion presented above.

Given a measurement of a low-rank matrix with noise and missing data, the difference between it and the ground-truth can be estimated with (12). This DP value gives a measure of the sensitivity of the low-rank matrix to perturbation due to i.i.d. additive Gaussian noise. If a matrix has a larger DP value, the corrupted matrix will generally be farther from the original one under the same noise situation than that with a smaller DP value. Note that this result is based on an upper-bound and may thus not lead to optimal solutions. Nonetheless, the Deviation Parameter provides an appropriate mechanism to select the candidate submatrices in the problem of finding a low-dimensional linear space representation of the data matrix.

E. Analysis of the DP criterion

In the preceding sections, we have given an estimate of the possible noise carried on an incomplete, noisy matrix based on results borrowed from matrix perturbation [31] and random matrix theory. In our derivations provided thus far, we use the expectation of the upper-bound to measure the distance between $\mathcal{R}(\mathbf{A})$ and $\mathcal{R}(\widehat{\mathbf{A}})$ instead of the actual distance because this cannot be calculated. Although in our previous section we provided grounded arguments for such a definition, we now turn to a study of the behavior of this newly defined criterion to demonstrate its effectiveness.

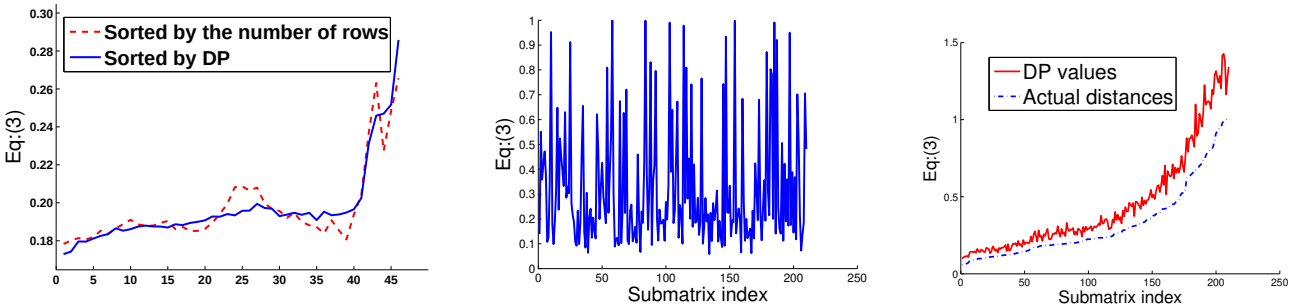
We start with a $m \times n$ matrix $\mathbf{W}$ of rank r and with every entry in the matrix in $[0, 100]$. This matrix is then contaminated with additive Gaussian noise, $N(0,\sigma^2)$, followed by a random occlusion mask with $d\%$ of missing entries. Let the resulting matrix be $\widehat{\mathbf{W}}$. The deviation parameter DP_i for each of the possible submatrices $\widehat{\mathbf{B}}_i$ (each constructed with r columns) is computed. Now, the resulting DP_i values need to be compared to the actual distance θ_i between the submatrix $\widehat{\mathbf{A}}_i$ and its corresponding noise-free submatrix $\mathbf{A}_i$.

Our first study will test how many times the deviation parameter criterion correctly selects that submatrix leading to a closer estimate of the ground-truth. For this to happen, the DP_i and DP_j computed from two submatrices $\widehat{\mathbf{B}}_i$ and $\widehat{\mathbf{B}}_j$ should be in the same order as θ_i and θ_j. That is, if $DP_i > DP_j$, then $\theta_i > \theta_j$, and vice-versa. This can be readily computed as the percentage ($\rho\%$) of times that $\widehat{\mathbf{B}}_i$ and $\widehat{\mathbf{B}}_j$ yield $(DP_i - DP_j)(\theta_i - \theta_j) > 0$. This is shown in Table I. These results are calculated from a total of

m	n	r	σ	$d\%$	$\rho\%$
10	8	2	1	10%	91.80%
20	10	4	2	20%	87.14%
20	20	6	2	20%	85.98%
30	20	4	2	10%	93.31%
30	20	4	2	40%	84.16%
30	20	4	5	40%	82.78%

TABLE I

PERCENTAGE OF TIMES THE DEVIATION PARAMETER CRITERION CORRECTLY PREDICTS THE ORDERING OF THE EFFECTS OF NOISE. PERCENTAGES ARE GIVEN FOR A VARIETY OF NOISE TERMS AND OCCLUSIONS.



(a) (b) (c)

Fig. 3. The effect of noise over each possible submatrix. (a) Actual distance between subspaces when the submatrices are sorted according to the number of deleted rows or DP value. In (b), the submatrices are ordered from having less to more missing elements. The abscissa corresponds to the index of the submatrices sorted by the numbers of missing entries, and the ordinate is the corresponding pb value. In (c) the submatrices are sorted by the actual distance between subspaces (dashed curve). The solid curve specifies the value of the deviation parameter derived in this paper.

30 trials with the matrix size, noise and missing data as specified in the table. From these results, we see that even for reasonably large amounts of missing elements and noise, the ordering provided by our deviation parameter criterion is almost always consistent with that of the ground-truth.

The DP criterion thus provides a convenient estimate of how useful each submatrix is. This is further illustrated in Fig. 3(a). In this case, we first generate a 50×10 rank 4 matrix $\mathbf{A}$ and its corresponding noisy version $\widehat{\mathbf{A}}$ (with zero-mean Gaussian noise). Then, we obtained the 210 possible 4-column submatrices. Now, for each of these 210 submatrices, we generate another subset of 46 submatrices by deleting $0, 1, 2, \ldots, 45$ rows. Each resulting set of 46 submatrices is sorted from smallest to largest DP_i. A plot of the actual distance as given by (3) is shown in Fig. 3(a) as a solid curve. This plot is the average over all possible 210 sets of 46 submatrices. As seen in the figure, the DP criterion results in the monotonically increasing function we need. This result is further compared to the one obtained when the submatrices are sorted from less to more missing rows (dashed curve in the figure). Adding rows is equivalent to adding samples to the least-squares fitting, generally resulting in better estimates. This result is however not as accurate as that of the DP criterion. Therefore, it is expected to provide better recoveries in practice, a point we will demonstrate in Section V.

The problem with the above result is that it uses the same submatrices to generate the results with different percentages of missing entries. In actuality, the r-column submatrix with more missing entries will be generally constructed with different columns than those used to build a submatrix with less missing entries. Therefore, our criterion should also work under this condition. To test this, a 50×10 random matrix of rank 4 with 10% missing elements (also randomly selected) and 1% random Gaussian noise is divided into all possible 210 submatrices $\widehat{\mathbf{B}}_i$. First, the submatrices $\widehat{\mathbf{B}}_i$ are sorted from those with the smallest to those with the largest number of missing elements. This means that $\widehat{\mathbf{B}}_1$ corresponds to the

matrix with the least number of missing elements, while $\widehat{\mathbf{B}}_{210}$ is that with the most. This index corresponds to the x-axis in Fig. 3(b). Then, we use Eq. (3) to calculate the actual distance between the ground-truth and each of the perturbed submatrices, which is shown in the y-axis in Fig. 3(b). We see that the selection of those matrices with less missing entries does not always help choose an appropriate set.

We now order the submatrices according to the actual distance between them and the ground-truth, dashed curve in Fig. 3(c). Here, we also plot the DP_i values for each of the submatrices. As it can be seen in this figure, the DP_i values follow the plot of the true distance very closely and thus result in a very convenient and efficient way to order the submatrices according to their sensitivity to noise. This result is contrasted against the previous one shown in Fig. 3(b) where the submatrices were sorted according to the missing elements. Clearly, the estimate provided by the DP criterion is much preferred. This is because the deviation parameter provides information about both, the sensitivity to noise and the amount of missing elements. If the number of the non-complete rows (b) increases, both $(\sqrt{m-b-1}+\sqrt{r})$ and $\min(\sigma(\widehat{\mathbf{A}}))$ will decrease. In the Supplementary Documentation we provide the proof for this result and show that although both the numerator and denominator decrease as the number of missing elements increases, the value of DP increases with it. This means that we favor matrices with less missing entries in general.

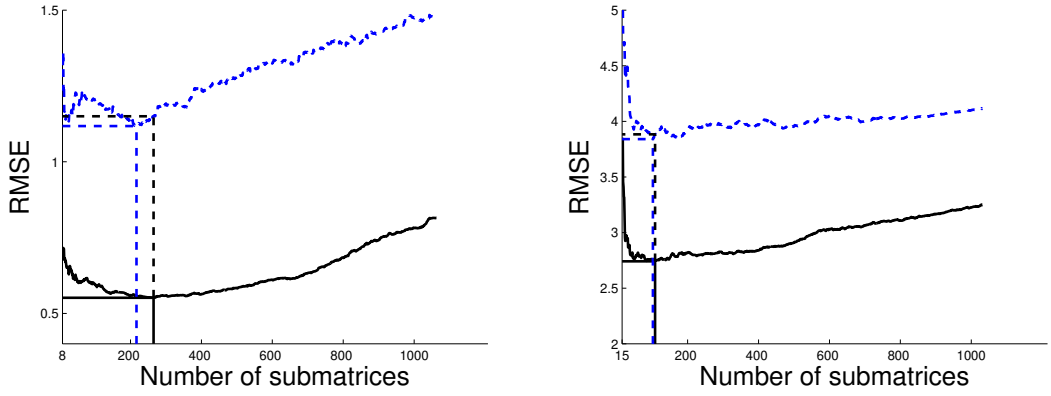
As a final note, it is important to understand the complexity of the algorithm defined in this section. Recall that for a $m \times n$ matrix, there is a total of C_n^r r-column submatrices. If the percentage of missing elements in this matrix is p, each r-column submatrix has $m(1-p)^r$ full rows on average (that is, assuming the occlusions are uniformly distributed). This mean that SVD needs to be performed on the resulting $m(1-p)^r \times r$ submatrix. The computational complexity of SVD is $O(m(1-p)^r r^2 + r^3)$, which is polynomial (of degree 3) in each iteration. The computational complexity of our criterion is thus given by $C_n^r O(m(1-p)^r r^2 + r^3)$. For large matrices this is a cost to be considered. In these cases, we can divide the large data matrix into more tractable submatrices. We will address this issue in the section to follow.

F. Low-rank matrix fitting with DP

Before the data selection criterion described above can be employed in an actual low-rank matrix fitting approach, we have to mention two points which are essential to its successful implementation. The first practical issue to attend to is given by the process we have selected to eliminate the missing entries from the original data matrix. In our approach, we eliminate all rows that have at least one missing element. The problem is that we need to guarantee that there is at least one submatrix in $\widehat{\mathbf{N}}$ which contains the information of each row. In our algorithm, we first sort the submatrices according to their DP_i value – from smallest to largest. Then, as to how many submatrices to select, we will choose the minimum number of submatrices needed to include the information of every row.

The second problem is that of determining the appropriate number of columns for the final matrix $\widehat{\mathbf{N}}$. Since the number of possible r-column submatrices is C_n^r and this is much larger than n, it is not necessary to include all these submatrices to construct $\widehat{\mathbf{N}}$. In [16], the width of the matrix $\widehat{\mathbf{N}}$ is set to a fixed size, e.g. $10m$ or $100m$. Here, we go one step further and propose a method which is based on the performance of the algorithm in recovering the matrix defining the null-space. To illustrate this, let us look at a couple of examples.

Let a $m \times n$ matrix of rank r have $d\%$ of its elements missing, and contain additive Gaussian noise at level σ. Also, as above, let the value of each entry be bounded by zero and 100. In our first case study, we generate a matrix with the parameters $m = 10$, $n = 16$, $r = 4$, $\sigma = 1$, and $d = 20$. This provides us with the ground-truth matrix $\mathbf{W}$ and its noisy version $\widehat{\mathbf{W}}$. We can use our algorithm (as described above) to find the best, minimum number of submatrices needed to recover $\mathbf{W}_r$. This allows us to analyze how good the recovery is when the minimum number of submatrices $\widehat{\mathbf{B}}_i$ is used and how much improvement one gets when we keep adding additional submatrices. This is illustrated in Fig. 4, where the x axis specifies the number of submatrices used to compute $\mathbf{W}_r$ and the y axis indicates the Root Mean Square



(a) (b)

Fig. 4. Two example curves of the RMSE with (a) $[m, n, r, \sigma, d\%] = [10, 16, 4, 1.00, 20\%]$ and (b) $[m, n, r, \sigma, d\%] = [15, 20, 3, 4.00, 30\%]$. The dashed curve is the true error (i.e., the difference between $\mathbf{W}$ and $\mathbf{W}_r$), while the solid curve corresponds to that we can calculate (i.e., $\widehat{\mathbf{W}}$ to $\mathbf{W}_r$). The dashed lines indicate the number of submatrices used in each case and their corresponding RMSE. We can see that the RMSEs obtained with the true measure and with our estimate are practically identical.

Error (RMSE). The dashed curve in the figure corresponds to the RMSE between $\mathbf{W}$ and $\mathbf{W}_r$ obtained by using *all* the matrix elements, even those that were occluded to the algorithm and had to be recovered by it. Hence, this first measure provides the holy grail of measures, because it will show how our estimate relates to it. The solid curve in the figure corresponds to the actual estimate, given by the RMSE between $\widehat{\mathbf{W}}$ and $\mathbf{W}_r$ over the *non-missing* elements of the matrices. The global minima of these two curves are shown in the figure as the lines travelling from those minima to the x and y axes. These specify the optimal number of submatrices needed to achieve the minimum RMSE.

We see in Fig. 4(a) that the true RMSE (dashed curve) and the one we can calculate (solid curve) have a similar behavior. They both decrease at first and then start increasing (indicating the noisy submatrices have started to overcome the recovery). This is also the case in the other example provided in Fig. 4(b), where $m = 15$, $n = 20$, $r = 3$, $\sigma = 4$, and $d = 30$. Most importantly, we see that the global minima in these two curves are located at a very proximal x value, i.e., a very similar number of submatrices is needed to minimize our computed measure (i.e., the difference between $\widehat{\mathbf{W}}$ and $\mathbf{W}_r$) and that provided by the ground-truth. We have further observed this pattern in a large number of simulations we have carried out. Therefore, the number of submatrices needed to carry out the recovery of $\mathbf{W}_r$ is conveniently given by the global minima of our estimate (solid line).

A final point needs to be made about large data matrices too. When $\widehat{\mathbf{W}}$ is very large, the number of possible r-column submatrices grows very fast. While this may be computationally demanding, in actuality, there is no need for creating all the possible submatrices. Here, we propose a three-step iterative DP method, which can improve the performance of the DP method in the case of a large measurement matrix. First, the large measurement matrix is divided into several overlapping submatrices (e.g., in the "dinosaur" sequence to be used in Section V, we employed three overlapping submatrices). Second, the original DP method is applied to each of these (overlapping) submatrices, and the rows with the largest reconstruction error are iteratively removed. This process continues until the average reprojection error over all the visible data begins to increase. This process is similar to that in [22]. The difference is that in [22] a threshold representing the highest tolerable reprojection error has to be pre-determined by the user. In our approach, this is automatically determined by the algorithm. In the third and final step, the whole recovery will be obtained by combining all these sub-recoveries. To do this, we work as follow. The data matrix $\widehat{\mathbf{W}}$ is partitioned into two overlapping submatrices, $\widehat{\mathbf{W}}_1$ and $\widehat{\mathbf{W}}_2$. This process is done to ensure that the number of overlapping columns in these two submatrices is at least r. Hence, $\widehat{\mathbf{W}}_1 = [\widehat{\mathbf{W}}_{11}, \widehat{\mathbf{W}}_{12}]$ and $\widehat{\mathbf{W}}_2 = [\widehat{\mathbf{W}}_{21}, \widehat{\mathbf{W}}_{22}]$, where $\widehat{\mathbf{W}}_{12}$ and $\widehat{\mathbf{W}}_{21}$ come from the same part of the measurement matrix of rank

r. The matrix representations of the recovered r-dimensional (row) spaces, $\widehat{\mathbf{F}}_1$ and $\widehat{\mathbf{F}}_2$, can be written as $\widehat{\mathbf{F}}_1 = [\widehat{\mathbf{F}}_{11}, \widehat{\mathbf{F}}_{12}]$ and $\widehat{\mathbf{F}}_2 = [\widehat{\mathbf{F}}_{21}, \widehat{\mathbf{F}}_{22}]$. Then, to find the $r \times r$ matrix $\mathbf{K}$ which minimizes $\|\widehat{\mathbf{F}}_{12} - \mathbf{K}\widehat{\mathbf{F}}_{21}\|_F$, we can follow a simple linear least-squares method where $\mathbf{K} = \widehat{\mathbf{F}}_{12}\widehat{\mathbf{F}}_{21}^T(\widehat{\mathbf{F}}_{21}\widehat{\mathbf{F}}_{21}^T)^{-1}$. For the case with overlapping rows, a similar process can be followed.

Note that the division process described in this section works best when the occlusion follows a continuous pattern – meaning that the points occluded in a first set of images are visible in another set and vice-versa. This is the case, for example, when one moves the face left to right or up and down or when an object is placed on a turntable.

IV. Application to SFM

The approach described thus far can be directly applied to the problem of affine SFM. In projective SFM, the measurement matrix needs to be scaled by a set of proper projective depths. The projective depth can be recovered either using the fundamental matrix in epipolar geometry [32] or using the iterative estimation approach [13], [21]. The iterative estimation method has many advantages but requires a good low-rank matrix fitting solution to ensure the convergence of the projective depths. The DP approach described in this paper provides such a solution.

A. Projective SFM with missing data

Assume that we have q views of a scene, each with n 3D points generated from different projective projections. We want to recover the projective structure of the scene as well as the camera motion (or projection) for all q views. Denote the 3×4 projection matrix of view i by $\mathbf{P}_i$, $i = 1, 2, \cdots, q$, and the 3D point j by $\mathbf{Q}_j$; in homogeneous coordinates $\mathbf{Q}_j = [x_j, y_j, z_j, 1]^T$, $j = 1, 2, \cdots, n$. Then, the projection equation for point j in view i can be written as $\lambda_{ij}\mathbf{q}_{ij} = \mathbf{P}_i\mathbf{Q}_j$, where $\mathbf{q}_{ij} = [x_{ij}, y_{ij}, 1]^T$ is the homogeneous coordinate of point j in image i, and λ_{ij} is the corresponding projective depth. Further, let $\mathbf{p}_{ij} = [x_{ij}, y_{ij}]^T$ be the inhomogeneous 2D coordinate of point j in image i.

If we write all the projective matrices and all the 3D point coordinates in a single matrix, we have $\mathbf{P} = [\mathbf{P}_1^T, \mathbf{P}_2^T, ..., \mathbf{P}_q^T]^T$ and $\mathbf{Q} = [\mathbf{Q}_1, \mathbf{Q}_2, ..., \mathbf{Q}_n]$. Then the sequence of all $q \times n$ tracked 2D points with the scaling factors λ_{ij} can be represented as a scaled measurement matrix,

$$\mathbf{S} = [\lambda_{ij}\mathbf{q}_{ij}] = \mathbf{P} \cdot \mathbf{Q}.$$

The size of the scaled measurement matrix $\mathbf{S}$ is $m \times n$, where $m = 3q$. If there is no noise and all the 3D points are visible in all views, it is clear that the scaled matrix $\mathbf{S}$ is of rank 4 [32].

In general, however, we only have the non-scaled measurement matrix $\mathbf{W}$ or its homogeneous version $\mathbf{H}$ obtained from the actual 2D tracked points,

$$\mathbf{W} = \left[\mathbf{p}_{ij}\right] \quad \text{and} \quad \mathbf{H} = \left[\mathbf{q}_{ij}\right].$$

Had we known the correct projective depths λ_{ij}, the low-rank matrix fitting method could have recovered $\mathbf{P}$ and $\mathbf{Q}$ up to a 4×4 homography.

The other major problem is that the (non-scaled) measurement matrix usually comes with some associated noise and missing elements, resulting in

$$\widehat{\mathbf{W}} = \left[\widehat{\mathbf{p}}_{ij}\right] \quad \text{and} \quad \widehat{\mathbf{H}} = \left[\widehat{\mathbf{q}}_{ij}\right],$$

where for all $(i, j) \in \Gamma$, $\widehat{\mathbf{p}}_{ij} = [\widehat{x}_{ij}, \widehat{y}_{ij}]^T$ and $\widehat{\mathbf{q}}_{ij} = [\widehat{x}_{ij}, \widehat{y}_{ij}, 1]^T$ are the inhomogeneous and homogeneous coordinates with unknown additive noise. The goal is to recover the full, noise-free matrices $\mathbf{P}$ and $\mathbf{Q}$ from the incomplete and inaccurate measurement matrix.

B. DP-based projective SFM

The proposed DP-based approach will be used iteratively to recover the projective depths, which will allow us to fit a rank 4 matrix to our current result. To facilitate convergence, the minimization criteria used in these two steps should have a similar form. Mahamud and Hebert [21] introduce a projective depth update where they propose to minimize the angle between each column vector and the low dimensional linear space. This method is based on a complete measurement matrix. The convergence of this algorithm has been proven under the case without missing data. We now extend this method to work in the missing data case.

For all those points that are occluded, we set $\widehat{\mathbf{q}}_{ij} = [0,0,0]^T$, i.e., if $(i,j) \notin \Gamma$. In the k^{th} iterative step, we have the current projective depth $\Lambda^{(k)}$ as

$$\Lambda^{(k)} = [\lambda_{ij}^{(k)}] \quad \text{and} \quad \lambda_j^{(k)} = [\lambda_{1j}^{(k)}, \lambda_{2j}^{(k)}, \cdots, \lambda_{mj}^{(k)}]^T,$$

where $j = 1, 2, \cdots, n$. The scaled measurement matrix $\widehat{\mathbf{S}}^{(k)}$ satisfies

$$\widehat{\mathbf{S}}^{(k)} = \widehat{\mathbf{H}} \odot \Lambda^{(k)},$$

where $\odot$ indicates the Hadamard product ($[a_{ij}] \odot [b_{ij}] = [a_{ij} \cdot b_{ij}]$). The proposed method is used on the current scaled measurement matrix $\widehat{\mathbf{S}}^{(k)}$ to find its best rank-4 fitting and factorize it into a product,

$$\widehat{\mathbf{S}}^{(k)} \rightarrow \widehat{\mathbf{P}}^{(k)} \cdot \widehat{\mathbf{Q}}^{(k)},$$

where $\widehat{\mathbf{P}}^{(k)}$ is a matrix composed by 4 orthonormal vectors.

The second part of this step is to update the current projective depths. First, we need to fill in the missing data based on the current projective depths, and then we have the fill-in version of $\widehat{\mathbf{H}}$, denoted as $\widehat{\mathbf{H}}^{(k)}$. We want to find $\Lambda^{(k+1)}$, such that the range space of $\widehat{\mathbf{S}}^{(k+1)} = \widehat{\mathbf{H}}^{(k)} \odot \Lambda^{(k+1)}$ is closest to that of $\widehat{\mathbf{P}}^{(k)}$, which is given by the sine of the largest principal angle. Here we use $\widehat{\mathbf{P}}^{(k)}$ to update each column of $\Lambda^{(k)}$ and $\widehat{\mathbf{Q}}^{(k)}$ to update the rows of $\Lambda^{(k)}$. Each column vector in $\widehat{\mathbf{S}}^{(k+1)}$ should be as close as possible to the space $\mathcal{R}(\widehat{\mathbf{P}}^{(k)})$. The j^{th} column of $\widehat{\mathbf{S}}^{(k+1)}$ is given by

$$s_j \equiv [\lambda_{1j} \cdot \widehat{\mathbf{q}}_{1j}^T, \lambda_{2j} \cdot \widehat{\mathbf{q}}_{2j}^T, \cdots, \lambda_{mj} \cdot \widehat{\mathbf{q}}_{mj}^T]^T,$$

where $[\lambda_{1j}, \lambda_{2j}, \cdots, \lambda_{mj}]^T = \lambda_j$ is a column vector with some projective depths. Now, let θ_j be the angle between s_j and $\mathcal{R}(\widehat{\mathbf{P}}^{(k)})$. We then have

$$\begin{aligned} \lambda_j^{(k+1)} &= \arg\min_{\lambda_j} \theta_j = \arg\max_{\lambda_j} \cos^2\theta_j = \arg\max_{\lambda_j} \frac{\|\widehat{\mathbf{P}}^{(k)}\widehat{\mathbf{P}}^{(k)T} s_j\|^2}{\|s_j\|^2} \\ &= \arg\max_{\lambda_j} \frac{s_j^T \widehat{\mathbf{P}}^{(k)}\widehat{\mathbf{P}}^{(k)T} s_j}{s_j^T s_j}. \end{aligned}$$

This result can be rewritten as

$$\lambda_j^{(k+1)} = \arg\max_{\lambda_j} \frac{\lambda_j^T \mathbf{C}_j \mathbf{C}_j^T \lambda_j}{\lambda_j^T \mathbf{T}_j \lambda_j}, \tag{13}$$

where the i^{th} row of the $m \times 4$ matrix $\mathbf{C}_j$ is given by $\widehat{\mathbf{q}}_{ij}^T \widehat{\mathbf{P}}_i^{(k)}$, $\widehat{\mathbf{P}}_i^{(k)}$ is a 3×4 matrix constructed with the i^{th} triplet of rows of $\widehat{\mathbf{P}}^{(k)}$, and $\mathbf{T}_j$ is a diagonal matrix with the i^{th} diagonal entry equal to $\widehat{\mathbf{q}}_{ij}^T \widehat{\mathbf{q}}_{ij}$. Eq. (13) is in fact a generalized eigenvalue-decomposition problem, where the correct projective depth $\lambda_j^{(k+1)}$ correspond to the eigenvector associated with the largest eigenvalue. This result directly provides a solution for the scale matrix $\Lambda^{(k+1)}$ at each iteration. While iterative methods such as [21] have recently been found to lead to trivial solutions in some cases [25], the DP-based algorithm just presented is shown to converge to the correct solution in a large number of experiments detailed in the section to follow.

$\sigma, d\%$	$diff_1$		$diff_2$		$diff_3$	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
$0.5, 10\%$	0.417	1.413	0.308	1.096	0.332	1.147
$0.5, 30\%$	0.435	1.569	0.387	1.466	0.559	2.396
$1.0, 10\%$	0.836	2.773	0.606	2.195	0.665	2.580
$1.0, 20\%$	0.823	2.762	0.676	2.528	0.825	3.356
$1.0, 30\%$	0.843	2.841	0.763	2.740	1.054	4.082
$2.0, 20\%$	1.661	5.910	1.361	5.114	1.665	6.549
$4.0, 20\%$	2.693	5.874	2.559	5.663	3.151	8.545
$5.0, 40\%$	6.900	10.26	6.756	10.08	8.125	26.36

TABLE II

DENOISING ABILITY OF THE PROPOSED FACTORIZATION METHOD.

V. EXPERIMENTAL RESULTS

We now provide extensive experimental validation for the proposed approach. A statistical analysis is first presented using synthetic data. We then conclude with the application of the proposed approach on four real datasets.

A. Fitting low-rank matrices

We begin by testing the denoising ability of the proposed method. For this, we generate a 30×20 matrix $\mathbf{W}$ of rank 4 with the absolute values of its entries set to no more than 100, and then add Gaussian noise with variance σ and randomly occlude $d\%$ of the matrix entries, $\widehat{\mathbf{W}}$. We recover the low-rank matrix $\mathbf{W}_r$ and then provide several measures of performance: *i*) the difference over the non-occluded entries between $\mathbf{W}_r$ and $\widehat{\mathbf{W}}$ (which we denote as $diff_1$), *ii*) the difference over the visible data between $\mathbf{W}_r$ and $\mathbf{W}$ ($diff_2$), and *iii*) the difference between $\mathbf{W}_r$ and $\mathbf{W}$ over all entries ($diff_3$). All these measures are given in RMSE and MAE (Maximum Absolute Error). The averages over a total of 30 trials are listed in Table II for each of the specified values of noise and missing elements. In this table we have added a result with a very large noise term ($\sigma = 5$) and extreme occlusion (40%), which results in a large $diff_3$. In this case, we see that the deletion of the rows with missing entries is problematic because it may eliminate other useful (visible) information. This was not a problem when the noise and occlusion were smaller, because these columns were included in other submatrices. However, in this extreme conditions, it is common to have missing elements in many rows and too much noise in the remaining ones. These are thus the limits of the algorithm.

As we can see from this table, the recovered low-rank matrix $\mathbf{W}_r$ is closer to the noise-free version, $\mathbf{W}$, than to its noisy version $\widehat{\mathbf{W}}$. This is indeed a most desirable property, since it shows the algorithm is capable of denoising the data matrix. Furthermore, we see that the missing elements recovered by our algorithm do not include much additional error; demonstrating that the proposed approach does a good job in recovering the missing information too. (This point will also be shown to hold true for real data.) By plotting these results as a function of the noise parameter and the RMSE, Fig. 5, we see that the error increases linearly with the amount of noise that is added to the data matrix. This is also a very desirable property.

The comparison to other data selection criteria is provided in Table III. In this table, we compare the RMSE, as given by $diff_1$, $diff_2$, $diff_3$, of the proposed algorithm and those obtained with a random selection of the submatrices [16] (indicated in the table as RAND) and the selection of the columns with less missing elements (MME, Minimal Missing Elements). In Table III, we tabulate the RMSE results of random selection for RAND, MME and DP, averaging over 30 trials. We generate 30×20 rank-3 matrices with additive Gaussian noise (σ) and missing data ($d\%$). As expected, the larger the noise and the amount of missing elements, the more sense it makes to use the criterion presented in this paper.

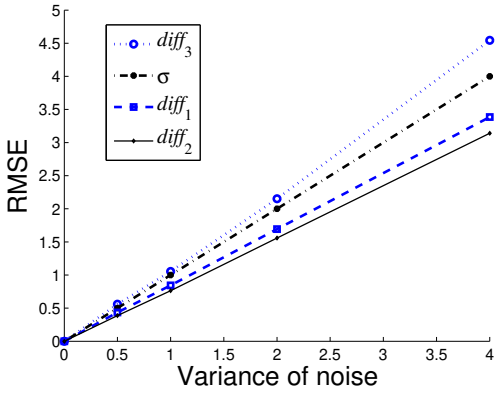


Fig. 5. The average recovered error with 30% missing data and different noise level, $\sigma = 0.5, 1, 2, 4$.

$\sigma, d\%$	RAND			MME			DP		
	$diff_1$	$diff_2$	$diff_3$	$diff_1$	$diff_2$	$diff_3$	$diff_1$	$diff_2$	$diff_3$
0.5, 10%	0.5032	0.3623	0.3770	0.4712	0.3189	0.3311	**0.4370**	**0.2658**	**0.2731**
1.0, 20%	1.1491	0.9312	1.1117	1.0133	0.7682	0.8419	**0.8692**	**0.5706**	**0.6032**
2.0, 20%	2.2586	1.8609	2.9495	1.9885	1.5264	1.7351	**1.7337**	**1.2012**	**1.3550**
2.0, 40%	3.1655	2.9333	34.068	2.5910	2.3620	9.1349	**1.9260**	**1.6840**	**2.6918**
4.0, 40%	5.0738	4.4706	33.697	4.6420	4.0388	13.695	**4.4141**	**3.8565**	**7.2933**
5.0, 50%	5.9846	5.3507	76.851	5.5914	5.1408	31.121	**5.4652**	**5.0143**	**16.321**

TABLE III
COMPARISON BETWEEN RAND, MME AND DP.

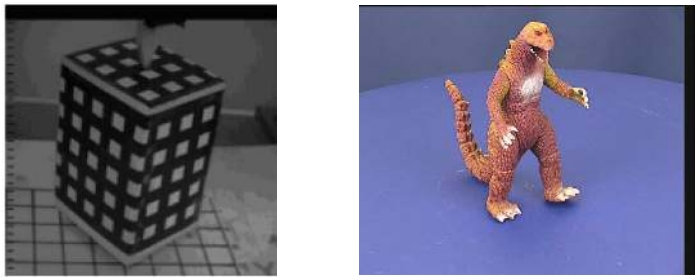
B. Affine SFM

In our first experimental result with real data, we employed a publicly available sequence[1] of 8 frames where a box with a calibration grid drawn onto it is shown, Fig. 6(a). This dataset comes with a total of 40 points tracked over each of the 8 frames with no occlusions.

In Fig. 7, we compare the results of the DP affine SFM algorithm and that of Jacobs' [16] using the $diff_2$ measure given above. To generate these results we randomly occluded a percentage of the image points. This is specified by the index in the x axis, while the y axis represents the RMSE in (a) and the MAE in (b). In these results, Jacobs algorithm has been labelled Jacobs$_{trans}$ because it includes a row of all ones. This is based on the observation that when the translation is included in the formulation of the affine model, a row with all entries equal to one should always be present in the solution space [16]. Hence, in general, using this approach results in better estimates. We have also extended our method to include this step. This extension is labelled DP$_{trans}$ in the figure.

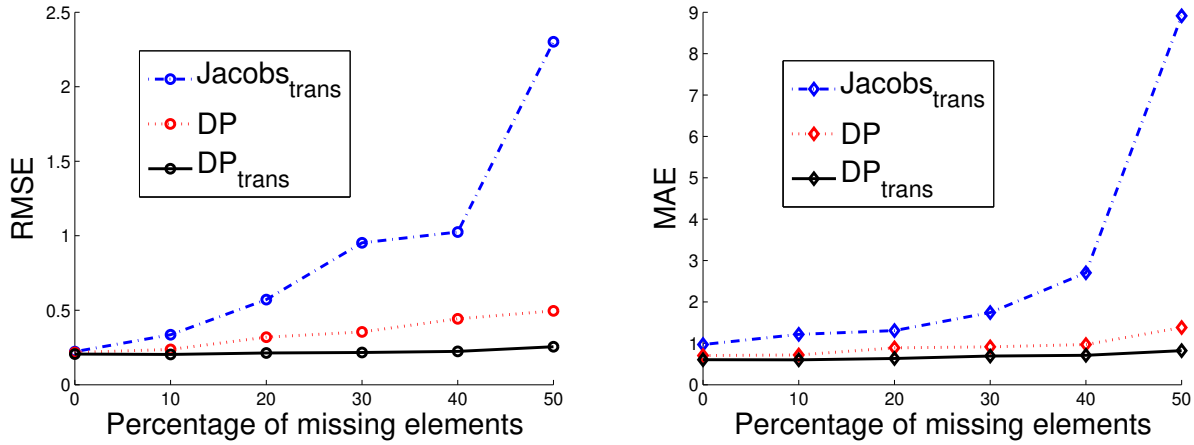
Fig. 8 illustrates how the RMSE increases as the amount of occlusion and noise increase. In (a) and (b) we show the RMSE as a function of both, noise and occlusion, for each of the two algorithms. From this result, we see that the sway noise performs over the DP-based approach is minimal. Most importantly, this effect is constant regardless of the occlusion term. The two algorithms are further compared in Fig. 8(c) for the particular case of 40% occlusion. Since the algorithm precision is consistently equated with the additive noise term, this could be further used as an initialization of a linear iterative optimization algorithm, such as, the bi-linear method defined in [29] or the iteration-refining step given in [4]. These results can now be contrasted to a global method such as RANSAC [7]. In RANSAC one could randomly choose r columns, eliminate the rows with missing entries, and then obtain the subspace directly (without the need to compute the null space). We can then calculate how each of the unused columns fits to this

[1]http://www.cs.umd.edu/~djacobs/missing-data.tar



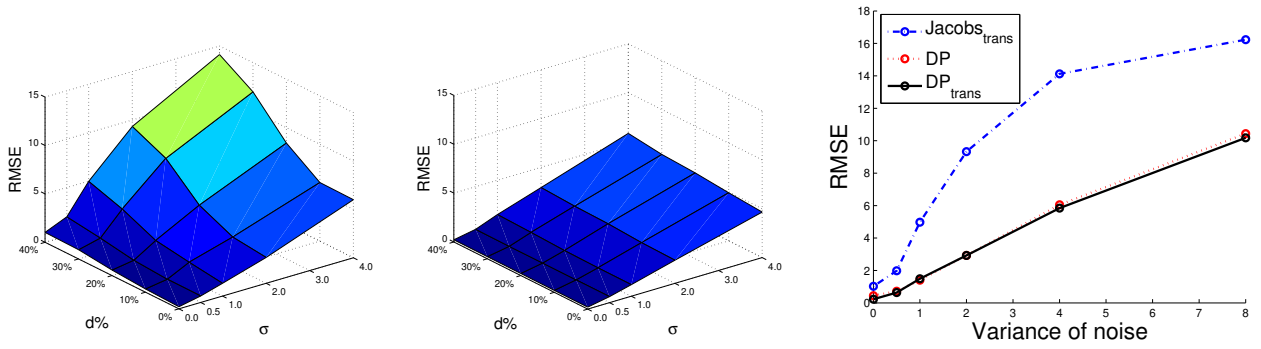
(a) (b)

Fig. 6. Shown here is (a) a frame of the box sequence, and (b) a frame of the dinosaur sequence.



(a) (b)

Fig. 7. Plotted here are the reprojection errors obtained with Jacobs algorithm and the DP-based affine SFM. In (a) we show the RMSE over a total of 30 runs for each of the occluding percentages. In (b) we plot the MAE of each algorithm for each of the occlusions.



(a) (b) (c)

Fig. 8. Shown here are the reprojection errors for the proposed approach and Jacobs algorithm as functions of the noise term and the amount of occlusion. RMSE on (a) Jacobs algorithm [16] and (b) DP affine SFM. In (c) we show a slice of the plots in (a) and (b) at 40% occlusion but with varying noise. This last plot allows for a one to one comparison.

TABLE IV

RMSE RESULTS AS GIVEN BY $diff_3$ AND AVERAGE COMPUTATION TIME.

$\sigma, d\%$	DP	RAND	MME	RANSAC	CF	RPCA
$0.5, 30\%$	**0.9738**	2.8692	2.1526	2.1666	3.3184	35.667
$1.0, 30\%$	**2.3501**	5.6598	3.9688	6.6115	2.3672	34.098
$2.0, 30\%$	**3.6384**	16.963	8.6395	9.8347	3.8571	35.816
$4.0, 30\%$	**6.4969**	36.960	13.369	11.686	15.040	37.602
$0.5, 40\%$	**1.9068**	10.256	3.2569	3.6903	14.561	39.539
$1.0, 40\%$	**3.1088**	11.261	7.5623	8.4500	18.077	46.470
$2.0, 40\%$	**6.3400**	23.653	15.320	13.009	22.906	45.234
$4.0, 40\%$	**8.5015**	36.523	18.965	17.528	46.200	45.469
time (s)	100	0.1	100	100	30	0.3

The comparison is given between the following approaches: DP, RAND (i.e., Random Selection) [16], MME (Minimal Missing Elements) [4], the variant of RANSAC described in the text, CF (the Closed-Form solution of [2]), RPCA (Robust PCA) [5].

subspace result and sort the r-column submatrices according to the fitting error. This approach results in higher RMSE than DP. Comparisons between the proposed approach and this variant of RANSAC are in Table IV. This table also includes comparative results with the robust approach presented in [5] and the closed-form solution of [2]. Additional details are in the Supplementary Documentation.

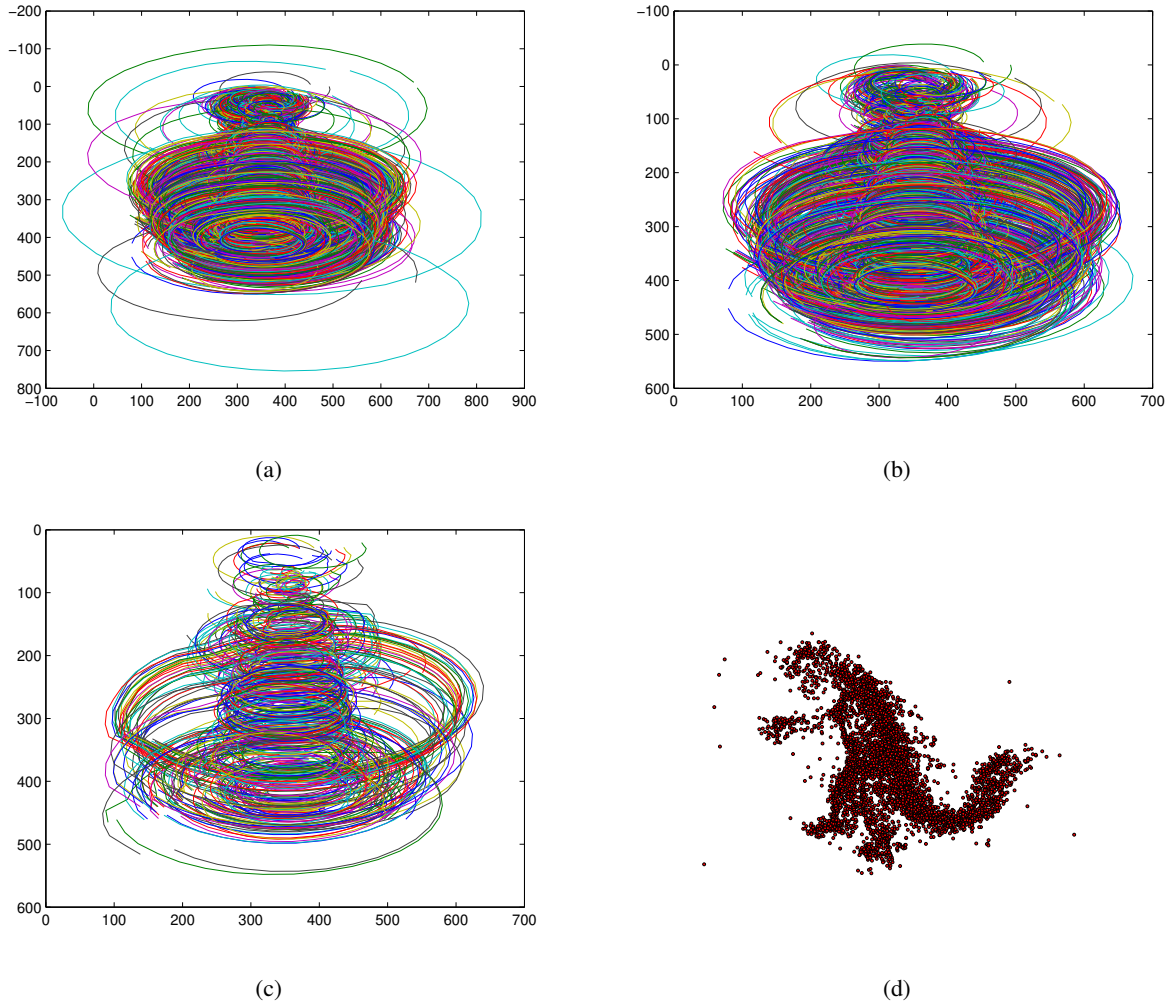
The other real dataset used in this section is the Dinosaur sequence [8]. This sequence has 36 frames and $4,983$ tracked feature points which become occluded for the duration of several frames. One of these frames is shown in Fig. 6(b). What makes this sequence of interest is the large amount of missing (occluded) elements. Overall, the matrix has 90.84% of its entries missing (occluded). In fact, $2,300$ of the feature points appear in only two frames. To facilitate convergence, at the end of each iteration of our algorithm, we will employ the optimization of [29] described in Section II-A. To provide a direct comparison with the results given in [4], we provide the results of the DP-based affine SFM with: all the data points, a subset of $2,683$, and a yet smaller set of 336. These results are shown in Fig. 9(a-c). The 3D reconstruction of the Dinosaur's shape when using all the data points is shown in Fig. 9(d). In [4], the authors provide a reprojection error by first computing the reconstruction on the smallest set and then extending this result to the two other (larger) sets. Their average reprojection errors and their maximum errors are (represented here as average/maximum): $1.8438/72.4467$ for the set with $4,983$ (i.e., all the data), and $2.4017/72.4467$ when using the subset of $2,683$ feature points. By repeating this procedure with the approach presented in this paper, we obtain the following lower errors: $1.6419/39.9988$, and $1.9340/39.9988$, respectively.

C. Projective SFM

To test the DP-based projective SFM algorithm, we will use a synthetic data-set to provide quantitative results and two sets of real data to show actual applications of the method.

The synthetic data-set consists of fifty randomly selected 3D points in the range of a sphere of radius 100 centered at the origin. Eight views of the resulting structure are generated. The cameras are located at random positions outside that sphere within the range of 200 to 500. The relatively large size of the scene produces large perspective distortions that could not be correctly recovered by affine SFM algorithms. Next, we add Gaussian noise with different variances σ and randomly occlude $d\%$ of the data points. The average distance between the recovered 2D coordinates (obtained with the proposed algorithm) and the (2D) ground-truth of each of the images is given in Fig. 10. Similar to our results in affine SFM, here too the average recovered errors remain almost unchanged for different quantities of noise.

The RMSE between the recovered 3D structure obtained with our algorithm and the ground-truth is in Table V. Since this is a 3D matching problem, the table illustrates the percentage of error added to our reconstruction. In this table, we also provide the RMSE obtained using the MME criterion in the projective SFM approach defined above. Since Jacbos' random selection cannot guarantee a consistent



(a) (b) (c) (d)

Fig. 9. The recovered tracks of the Dinosaur sequence: (a) on $4,983$ points, (b) on $2,683$ points and (c) on 336 points. (d) The 3D reconstruction of the dinosaur.

$(\sigma, d\%)$	(0.5, 20%)	(1.0, 20%)	(2.0, 20%)	(2.0, 40%)
MME	0.32%	0.85%	1.70%	3.85%
DP	0.12%	0.31%	0.53%	0.61%

TABLE V

COMPARISON BETWEEN THE MME AND DP CRITERIA IN PROJECTIVE SFM

recovery result at each iteration, it is not an appropriate candidate to be embedded in an iterative framework and was excluded from this comparison. In the table, the results are averaged over 30 trials.

We now apply the DP-based projective SFM algorithm to two real image sequences. The first example has 7 real images of a small wooden object shown in Fig. 11(a). Here, we manually select 32 feature points and track them for the duration of the video sequence with 25% of the points missing (see Supplementary Documentation). The other sequence we will use is the Model House sequence, which includes 10 images and 672 3D feature points. The percentage of missing points in this second sequence is 57.65%. One of the frames is shown in Fig. 12(a). The projective effect on this second sequence is larger than in the first one. This is because the distance between the camera and the turntable used to take the images is comparable to the size of the object.

The 3D reconstructions obtained with the proposed algorithm are in Figs. 11(b) and 12(b). In the first

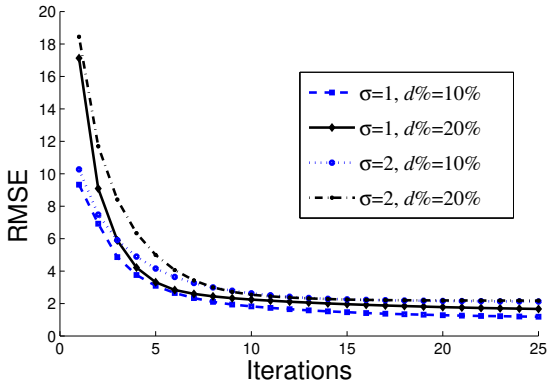
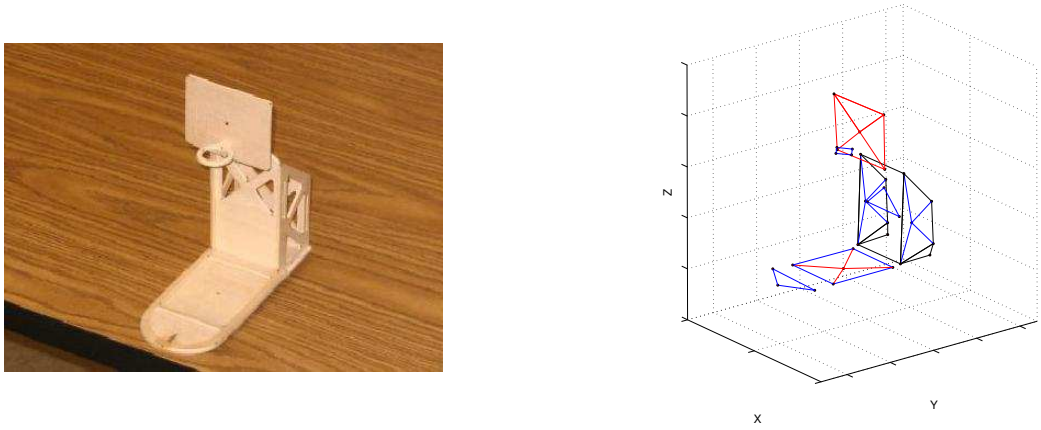


Fig. 10. The average 2D reconstruction error after 25 iterations with the noise term as specified in the plot.



(a) (b)

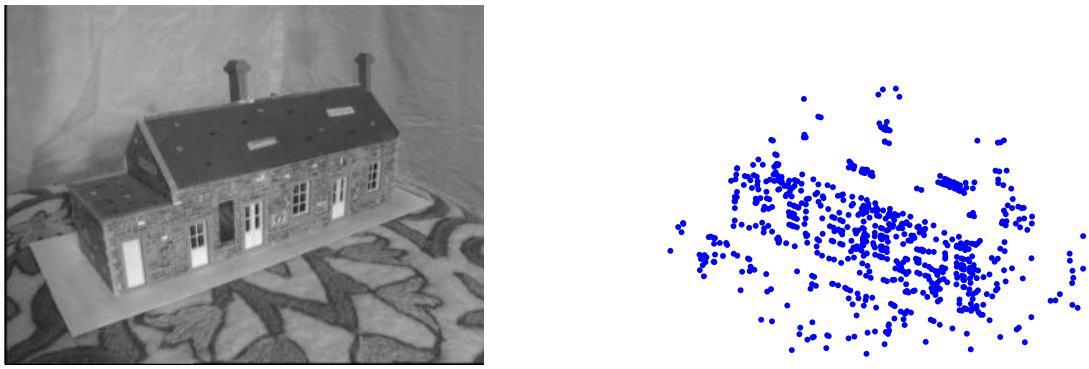
Fig. 11. (a) One image in the wooden toy sequence. (b) The 3D structure of the object recovered by the DP-based projective SFM algorithm.

sequence (wooden object), the 2D locations of all feature points are recovered at pixel accuracy. The precision of the projective recovery is clear in this case from the figure too. In the second sequence, the RMSE and the MAE between the recovered and the given data is $0.6479/11.9946$.

VI. Conclusions

Many problems in computer vision, pattern recognition and related areas reduce to finding that low-rank matrix that best fits an original data matrix with noisy and missing entries. A classical example is the SFM problem, where the 3D shape and motion of the object need to be recovered from a sequence of 2D images. In SFM, many of the points are generally imprecisely detected (noisy), while others are occluded in some of the frames.

In this paper, we have shown that the missing and noise problems can be simultaneously addressed by first dividing the data matrix into an appropriate set of submatrices with no missing elements and, then, using a criterion that determines which of these submatrices are less affected by the noise term. Our key result was to provide a formal proof for the relation between the effect of noise in a submatrix and the similarity of its column vectors. That is, when the vectors given by the columns of one of our submatrices are separated by a large angle (say, close to $90°$), additive noise has a limited influence. However, when the same noise is added to a submatrix with similar column vectors, the resulting subspace will be more different from its original noise-free version than the sway observed in submatrices with very distinct column vectors. The reason for this is grounded in the fact that dissimilar measurements do not get



(a) (b)

Fig. 12. (a) One of the images in the model house sequence. (b) The recovered 3D structure of the scene.

affected as much by a relatively small error term as do very similar ones. Note that in the latter case the dimensions of the subspace of the noisy matrix that correspond to the error term have a deviation from the original basis vectors similar to that seen in the original bases. In this case, it is unclear which small variations (between vectors) correspond to the noise term and which define the underlying subspace.

We have then shown how we can employ this formulation and a noise model to derive an upper-bound for the effects of noise in each of the submatrices. The derived criterion, referred to as DP (for Deviation Parameter), has been shown to be a very consistent and reliable criterion for estimating low-rank matrices from synthetic and real data. In particular, we have shown how the criterion can be successfully applied to the problems of affine and projective SFM. In these cases, our criterion was able to work under large occlusions (about 40%) and noise terms (with variances around 5).

The criterion presented in this article is however very general and can be employed in any other problem where a low-rank fitting step is required. This is the case, for example, in the problem of face and object recognition, in optical flow, and in the modeling and classification of microarray data in bioinformatics. Further research will determine how the proposed criterion compares to previously defined approaches in these other domains. Extensions of this approach should also include other estimates of the upper-bound and extensions to other types of noise.

ACKNOWLEDGMENTS

We thank the reviewers for their constructive comments. Thanks also go to Jeff Fortuna for discussion and involvements in the early developments of the DP criterion. We thank David Jacobs for making his code available to us. This research was supported in part by NSF grant IIS 0713055 and NIH grant R01 DC 005241.

REFERENCES

[1] S.S. Beauchemin and J.L. Barron, "The Computation of Optical Flow," *ACM Computing Surveys (CSUR)*, vol. 27, no. 3, pp. 433-467, 1995.

[2] S. Brandt, "Closed-form Solutions for Affine Reconstruction under Missing Data," *Proceedings Stat. Methods for Video Processing (ECCV02 workshop),* pp. 109-114, 2002.

[3] A.M. Buchanan and A.W. Fitzgibbon, "Damped Newton Algorithms for Matrix Factorization with Missing Data," *Proc. Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 316-322, 2005

[4] P. Chen and D. Suter, "Recovering the Missing Components in a Large Noisy Low-Rank Matrix: Application to SFM," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 26, no. 8, pp. 1051-1063, 2004.

[5] F. de la Torre and M. Black, "Robust Principal Component Analysis for Computer Vision," *Proc. Int. Conf. Computer Vision,* vol. 1, pp. 362-369, 2001.

[6] Y. Dodge, "Analysis of Experiments with Missing Data," Wiley, 1985.

[7] M. Fischler and R. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *CACM,* vol. 24, no. 6, pp. 381-395, 1981.
[8] A. Fitzgibbon, G. Cross, and A. Zisserman, "Automatic 3D Model Construction for Turn-Table Sequences, 3D Structure from Multiple Images of Large-Scale Environments," *Lecture Notes in Computer Science,* vol. 1506, pp. 155-170, 1998.
[9] S. Friedland, M. Kaveh; A. Niknejad, and H. Zare, "An Algorithm for Missing Value Estimation for DNA Microarray Data," *Proc. ICASSP*, vol. 2, pp. 1092-1095, 2006.
[10] G.H. Golub and C.F. von Loan, *Matrix Computations,* Johns Hopkins University Press, 1996.
[11] A. Gruber and Y. Weiss, "Multibody Factorization with Uncertainty and Missing Data Using the EM Algorithm," *Proc. Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 707-714, 2004.
[12] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2^{nd} edition, Cambridge University Press, 2003.
[13] Y.S. Hung and W.K. Tang, "Projective Reconstruction from Multiple Views with Minimization of 2D Reprojection Error," *International Journal of Computer Vision,* vol. 66, no. 3, pp. 305-317, 2006.
[14] M. Irani, "Multi-Frame Correspondences Estimation Using Subspace Constraints," *International Journal of Computer Vision,* vol. 48, no. 3, pp. 173-194, 1999.
[15] D.W. Jacobs, "Linear Fitting with Missing Data for Structure-from-Motion," *Proc. Conf. Computer Vision and Pattern Recognition,* pp. 206-212, 1997.
[16] D.W. Jacobs, "Linear Fitting with Missing Data for Structure-from-Motion," *Computer Vision and Image Understanding*, vol. 82, no. 1, pp. 57-81, 2001.
[17] H. Jia, J. Fortuna and A.M. Martinez, "Perturbation Estimation of the Subspaces for Structure from Motion with Noisy and Missing Data," *Proceedings of the Third International Symposium on 3D Data Processing, Visualization and Transmission,* Chapel Hill (NC), pp. 1101-1107, 2006.
[18] I.M. Johnstone, "On the Distribution of the Largest Eigenvalue in Principal Components Analysis," *The Annals of Statistics*, vol. 29, no. 2, pp. 295-327, 2001.
[19] G. Li and Z. Chen, "Projection-pursuit Approach to Robust Dispersion Matrices and Principal Components: Primary Theory and Monte Carlo," *J. Amer. Stat. Assoc.*, vol. 80, no. 391, pp. 759-766, 1985.
[20] H.C. Longuet-Higgins, "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature*, vol. 293, pp. 133-135, 1981.
[21] S. Mahamud and M. Hebert, "Iterative Projective Reconstruction from Multiple Views," *Proc. Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 430-437, 2000.
[22] D. Martinec and T. Pajdla, "Structure from Many Perspective Images with Occlusions," *Proc. European Conf. Computer Vision,* pp. 355-369, 2002.
[23] A.J. Miller, "Subset Selection in Regression," Champman & Hall, 2002.
[24] H. Murase and S. Nayar, "Visual Learning and Recognition of 3D Objects from Appearance", *International Journal of Computer Vision*, vol. 14, pp. 5-24, 1995.
[25] J. Oliensis and R. Hartley, "Iterative Extensions of the Sturm/Triggs Algorithm: Convergence and Nonconvergence",*IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, no. 12, pp. 2217-2233, 2007.
[26] C.J. Poelman and T. Kanade, "A Paraperspective Factorization Method for Shape and Motion Recovery," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19, no. 3, pp. 206-218, 1997.
[27] S. Roweis, "Em Algorithm for PCA and SPCA. Neural Information," *Proc. NIPS*, pp. 626-632, 1997.
[28] Y. Sato and K. Ikeuchi, "Reflectance Analysis for 3D Computer Graphics Model Generation," *Graphical Models and Image Processing,* vol. 58, no. 5, pp. 437-451, 1996.
[29] H.Y. Shum, K. Ikeuchi and R. Reddy, "Principal Component Analysis with Missing Data and Its Application to Polyhedral Object Modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 17, no. 9, pp. 854-867, 1995.
[30] L. Sirovich and M. Kirby, "Low-dimensional Procedure for the Characterization of Human Faces," *Journal of the Optical Society of America,* vol. 4, no. 3, pp. 519-524, 1987.
[31] G. W. Stewart and Ji-Guang Sun, *Matrix Perturbation Theory,* Academic Press, 1990.
[32] P. Sturm and B. Triggs. "A Factorization Based Algorithm for Multi-image Projective Structure and Motion," *Proc. European Conf. Computer Vision,* pp. 709-720, 1996.
[33] M. Tipping and C. Bishop, "Probabilistic Principal Components Analysis," *J. R. Stat. SOC. B,* vol. 61, pp. 611-622, 1999.
[34] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams under Orthography: A Factorization Method," *International Journal of Computer Vision,* vol. 9, no. 2, pp. 137-154, 1992.
[35] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, R.B. Altman, "Missing Value Estimation Methods for DNA Microarrays," *Bioinformatics,* vol. 17, pp. 520-525, 2001.
[36] R. Vidal and R. Hartley, "Motion Segmentation with Missing Data using PowerFactorization and GPCA," *Proc. Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 310-316, 2004.
[37] R.M. Voyles, J.D. Morrow and P.K. Khosla, "The Shape from Motion Approach to Rapid and Precise Force/Torque Sensor Calibration," *J. Dynamic Sys. Measurement Control,* vol. 119, pp. 229-235, 1997.
[38] P. Wedin, "Perturbation Bounds in Connection with Singular Value Decomposition," *BIT Numerical Mathematics,* vol. 12, no. 99-111, 1972.
[39] T. Wiberg, "Computation of Principal Components When Data is Missing," In *Proc. 2^{nd} Symp. Computational Statistics,* pp. 229-236, 1976.
[40] J.H. Wilkinson, *The Algebraic Eigenvalue Problem,* Clarendon Press, 1965.
[41] Y. Zhang and A.M. Martinez, "A Weighted Probabilistic Approach to Face Recognition from Multiple Images and Video Sequences," *Image and Vision Computing,* vol. 24, no. 6, pp. 626-638, 2006.