Compact Matrix Factorization with Dependent Subspaces

Viktor Larsson[1]
[1]Centre for Mathematical Sciences
Lund University

Carl Olsson[1,2]
[2]Department of Signals and Systems
Chalmers University of Technology

{viktorl,calle}@maths.lth.se

Abstract

Traditional matrix factorization methods approximate high dimensional data with a low dimensional subspace. This imposes constraints on the matrix elements which allow for estimation of missing entries. A lower rank provides stronger constraints and makes estimation of the missing entries less ambiguous at the cost of measurement fit.

In this paper we propose a new factorization model that further constrains the matrix entries. Our approach can be seen as a unification of traditional low-rank matrix factorization and the more recent union-of-subspace approach. It adaptively finds clusters that can be modeled with low dimensional local subspaces and simultaneously uses a global rank constraint to capture the overall scene interactions. For inference we use an energy that penalizes a trade-off between data fit and degrees-of-freedom of the resulting factorization. We show qualitatively and quantitatively that regularizing both local and global dynamics yields significantly improved missing data estimation. [1]

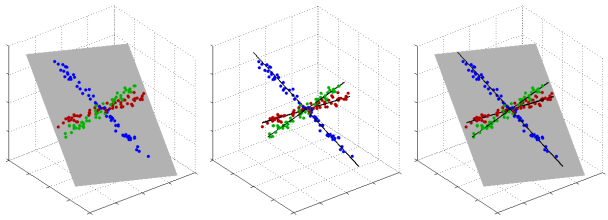
1. Introduction

Matrix factorization is a an important tool in many engineering applications. The assumption that data belongs to a low dimensional subspace has been proven useful in numerous computer vision applications, e.g. non-rigid and articulated structure from motion [6, 1, 41], photometric stereo [3], optical flow [15], face recognition [40, 34] and texture reparation [25].

Given an $m \times n$ matrix M containing m-dimensional measurements a low dimensional approximation $X \approx M$, where $\text{rank}(X) = r_0$, can be found using singular value decomposition (SVD). Since $\text{rank}(X) = r_0$ the matrix X can be written

$$X = BC^T, \tag{1}$$

where B is $m \times r_0$ and C is $n \times r_0$. The columns of B constitute a basis for the column-space of X. The matrix C contains coefficients used to form the columns of X from the basis. Alternatively one may think of the rows of X as n-dimensional data, C as a basis for the row-space and B as the coefficients. In both cases the data is approximated by an r_0-dimensional subspace, as illustrated in Figure 1(a).



(a) (b) (c)

Figure 1: 3D illustration of subspace representations. (a) - A 2D subspace is fitted to all the data (global model). (b) - A union of independent 1D subspaces is fitted to clustered data (local models). (c) - Our unified approach. 1D subspaces are fitted to clustered data and restricted to lie in a 2D subspace. (For this data $m = 3$, $n = 100$, $r_0 = 2$ and $r_k = 1$, see Section 2 for definitions.)

In a sense the factorization BC^T can be seen as a compressed representation of M where the mn elements have been reduced to $(m + n - r_0)r_0$ degrees of freedom (see Section 3.1). It is therefore possible to compute the factorization even if only a subset of the elements of M are known, by solving $W \odot M \approx W \odot (BC^T)$. Here $\odot$ denotes element-wise multiplication and the matrix W has elements $w_{ij} = 1$ for known data and 0 for missing data. Note that once computed, BC^T contains estimates of both known and missing data. In this way it is theoretically possible to "predict" at most $mn - (m + n - r_0)r_0$ missing elements.

In the presence of missing data the low rank approximation problem becomes very difficult, some variations of the problem even NP-hard [17]. However, due to its practical importance a lot of research have been directed at finding good algorithms. In [2] it is shown that under the spectral norm a closed form solution exist if the missing data forms a so called Young pattern. A recent trend has been to replace the rank function with the nuclear norm [31, 8, 30]. However, in many applications such as structure from motion, where missing entries are highly correlated, this approach has been shown to perform poorly (e.g. [24]).

If the rank of the sought matrix is known, the bilinear

[1]This work has been funded by the Swedish Research Council (grant no. 2012-4213) and the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots).

parametrization (1) can be locally optimized. Buchanan and Fitzgibbon [7] showed that alternating methods often exhibit very slow convergence and proposed a damped Gauss-Newton update. In [28] it was illustrated that the Wiberg elimination strategy [39] is very robust to local minima. For a recent comparison of different approaches to minimize the bilinear formulation see [19]. In [22] the ℓ_1 norm is used to address outliers. The proposed alternating approach is shown to converge slowly in [13]. Instead [13, 35] use generalizations of the Wiberg approach designed to handle the non-differentiable objective function while jointly updating the two factors.

Despite numerous recent developments in rank optimization missing data is still a problem plaguing vision algorithms. Dai *et al*. [10] argue that researchers have focused too much on optimization and ignored modeling issues. While the rank constraint provides a compact model representation it is limited by only measuring the overall complexity of the matrix even though individual sub-blocks may be less complex. Hence, there is no incentive to use fewer basis columns for sub-blocks than what the total rank admits. A relatively high overall model complexity is a particular problem when missing data needs to be estimated. As noted in [27, 18, 14] the availability of too many basis elements causes methods only optimizing a global rank constraint to over-fit giving very poor results.

A related model used in clustering is the union-of-subspace approach [44, 42]. Here data is clustered into similar groups that can be represented with independent low dimensional subspaces, see Figure 1(b). We refer to these as local subspaces since they are local to a particular cluster. In [26, 23] these are used to cluster frames into groups that allow simple deformation models. In principle these could also be used to address the missing data problem. In contrast to the global rank constraint, which constrains the whole matrix, each cluster has its own set of basis vectors and can only be constructed from these. This gives a data representation that is often (but not always, see Section 3.1) more compact. The overall idea of dividing the matrix into less complex parts and treating them separately is shared with the multi-body factorization methods [38, 9, 43] which typically perform clustering on the trajectories.

In this paper we address the missing data problem by presenting a new compact factorization formulation. Our approach unifies the local and global subspace approaches leveraging the benefits of them both. Our method adaptively clusters the data and fits local subspaces, but also enforces a low rank on the entire data matrix. This ensures that any potential interactions between clusters are identified by the model which increases the prediction capability. For example, if clusters correspond to rigid parts of an object, similar to [32], our model can predict occluded parts if a motion dependency exists. In contrast the union-of-subspace approach lacks the ability to learn global scene dependence since subspaces are treated independently. Figure 1(c) illustrates our approach for a simple 3D example.

Our main contributions are

- We analyze the performance of global and local models with respect to different types of missing data.
- We present a new factorization that incorporates both a global rank constraint and local subspace constraints and show how this reduces model complexity.
- For computing the factorization we propose an energy-based model fitting framework that is able to perform joint clustering and adaptive model selection.
- We show on real and synthetic experiments that the proposed approach handles missing data much more accurately than existing factorization models.

2. A Dependent Subspace Model

In this section we present our model. We make two assumptions on the data matrix; that the entire scene is explained well by a low rank model, and that it can be partitioned into clusters that are explained by simpler models. Let X be an $m \times n$ matrix. The model can then (possibly after column permutations) be written as

$$\begin{aligned} X &= \begin{bmatrix} X_1 & X_2 & \dots & X_K \end{bmatrix} \\ \text{rank}(X) &= r_0, \quad \text{rank}(X_k) = r_k \end{aligned} \tag{2}$$

where each X_k is an $m \times n_k$ matrix that contains the data points of a cluster. It is clear that $r_0 \geq r_k$ and typically we try to have $r_0 \ll \sum_{k=1}^{K} r_k$ since we want to model the dependence between the clusters. Here we have divided the matrix columns into clusters. Note however that the same model can be applied to the rows by transposing.

Since X_k is of rank r_k it can be factorized into $X_k = B_k C_k^T$, where B_k is $m \times r_k$ and C_k is $n_k \times r_k$. The matrix B_k contains a basis for the subspace spanned by the columns of X_k. The full matrix X can thus be written

$$X = \begin{bmatrix} B_1 C_1^T & B_2 C_2^T & \dots & B_K C_K^T \end{bmatrix}. \tag{3}$$

Note that if the global rank constraint $\text{rank}(X) = r_0$ is ignored then $B_1, B_2, ..., B_K$ are assumed to be independent and this expression constitutes a union of subspace representation of X.

Now, assuming $r_0 < \sum_{k=1}^{K} r_k$ there is a dependence between the cluster subspaces. Since the columns of X are spanned by the columns of $\begin{bmatrix} B_1 & B_2 & \dots & B_K \end{bmatrix}$ this matrix must also be of rank r_0. Therefore we may factor it into

$$\begin{bmatrix} B_1 & B_2 & \dots & B_K \end{bmatrix} = B \begin{bmatrix} U_1 & U_2 & \dots U_K \end{bmatrix}, \tag{4}$$

where B is $m \times r_0$ and U_k is $r_0 \times r_k$. Here B is a basis of the column space of $\begin{bmatrix} B_1 & B_2 & \dots & B_K \end{bmatrix}$ and therefore also of X. Inserting into (3) gives our model

$$X = B \begin{bmatrix} U_1 C_1^T & U_2 C_2^T & \dots & U_K C_K^T \end{bmatrix}. \quad (5)$$

We can think of the $r_0 \times r_k$ matrices U_k as selecting a r_k-dimensional basis within the r_0-dimensional space spanned by the columns of B. While the union-of-subspace model (3) treat subspaces independently by allowing arbitrary selection of the bases $B_1, B_2, ..., B_k$ our model forces these to be selected in the global subspace spanned by B. Figure 2 shows an example of the three model factorizations when $r_0 = 5$ and $r_k = 3$ for $k = 1, 2, 3$.

In the above description of our model we have assumed that the subspaces are linear. Note however that it is easy to use affine subspaces by restricting the last row of C_k^T to be all ones. If $B_k = \begin{bmatrix} A & t \end{bmatrix}$ and $C_k^T = \begin{bmatrix} C^T \\ \mathbb{1}^T \end{bmatrix}$ then $B_k C_k^T = AC^T + t\mathbb{1}^T$, which is an affine function in C.

3. Benefits of Dependent Models

In this section we discuss the benefits of using both local and global subspace constraints. We compare three formulations: the *global model* (1), *local models* (3) and our *unified model* (5).

3.1. Degrees of Freedom

We first compute the degrees of freedom (DOF) of the three models. Note that it is clear that the unified model will have fewer DOF than both the local and the global models since (5) is a special case of both (1) and (3). Having an accurate model with few DOF makes matrix completion more well posed and reduces the space of feasible matrices.

Linear Subspace Models Under the global model the data matrix X can be factorized as in (1). The matrices B and C have mr_0 and nr_0 elements respectively. However due to the gauge freedom $X = BC^T = BGG^{-1}C^T$, where G is an unknown invertible $r_0 \times r_0$ matrix the DOF for the global model are

$$mr_0 + nr_0 - r_0^2. \quad (6)$$

For cluster k in (3) the matrices B_k and C_k have mr_k and $n_k r_k$ elements respectively. Similarly to the global model B_k and C_k are only determined up to an invertible $r_k \times r_k$ matrix G_k. We therefore get

$$\sum_{k=1}^{K} mr_k + n_k r_k - r_k^2 \quad (7)$$

DOF for the local models.

For the unified model we first consider the term $BU_kC_k^T$. Since B is $m \times r_0$, U_k is $r_0 \times r_k$ and C_k $n_k \times r_k$ this term has $mr_0 + r_0 r_k + n_k r_k$ elements. However, since

$$X_k = BU_kC_k^T = BGG^{-1}U_kG_kG_k^{-1}C_k^T, \quad (8)$$

there are two ambiguities here. The first subtracts r_0^2 DOF once and the second r_k^2 DOF for each cluster. Summing over k we thus get

$$mr_0 - r_0^2 + \sum_{k=1}^{K} r_0 r_k + r_k n_k - r_k^2. \quad (9)$$

Note that for independent clusters this reduces to (7). However when $r_0 < \sum_k r_k$ (and typically $r_0 \ll \sum_k r_k$) it is easy to see that the unified model is at least as compact as the local model. To compare to the global model we note that $\sum_k n_k = n$ and subtract (9) from (6). This gives

$$nr_0 - \sum_{k=1}^{K}(r_0 r_k + r_k n_k - r_k^2) = \sum_{k=1}^{K}(r_0 - r_k)(n_k - r_k). \quad (10)$$

Since we can't form clusters with fewer columns than their rank both terms of the product are positive, which confirms that the unified model is always at least as compact as the global model.

Affine Subspace Models In our applications we will typically use affine subspaces since this removes some scale ambiguities. In this case the matrix C_k^T is required to have one row of all ones, which reduces the DOF in this matrix to $n_k(r_k - 1)$. Furthermore, this requires the last row of G_k^{-1} to be $\begin{bmatrix} 0 & 0 & \dots & 1 \end{bmatrix}$ which therefore has $r_k(r_k - 1)$ DOF. The unified model then has

$$mr_0 - r_0^2 + \sum_{k=1}^{K} r_0 r_k + (r_k - 1)n_k - (r_k - 1)r_k \quad (11)$$

DOF. Note that the dimension of the affine subspace is $r_k - 1$ while the rank of its matrix $BU_kC_k^T$ is still r_k.

3.2. Predicting Missing Data

In this section we discuss the prediction capabilities of the unified model and illustrate how the global and local models complement each other when recovering missing data. To gain some intuition about the model we first consider the situation where a new column is added to each of the three factorizations, see Figure 2. In SfM this corresponds to estimation of a point track from a motion model. To generate a new column we need to specify coefficients in the C and C_k matrices (for some $k \in \{1, ..., K\}$), that is, the elements marked with c in Figure 2. In this example the global model needs to determine 5 parameters and therefore require at least 5 known elements in the new column.

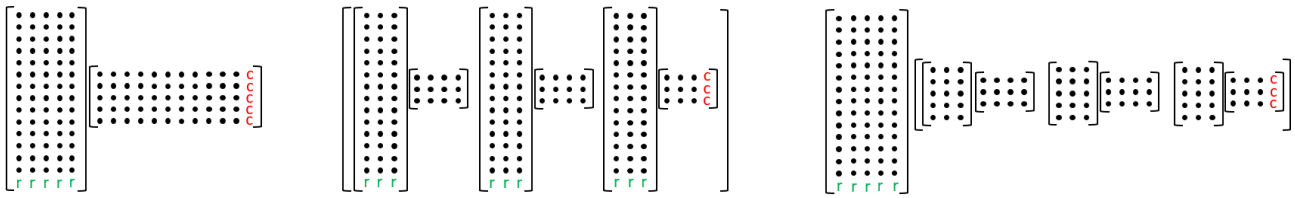


Figure 2: Three factorizations: *Left* - global model. *Middle* - union of subspace model. *Right* - unified model. Here $r_0 = 5$ and $r_i = 3$, $i = 1, 2, 3$. The r and c markers highlight elements that need to be estimated when adding a new row or column.

For the local and unified models we only have 3 unknowns. (Additionally we may need a 4th known element to determine which cluster the new column belongs to.) Hence, in this situation the local and unified models require less data than the global model to predict missing elements.

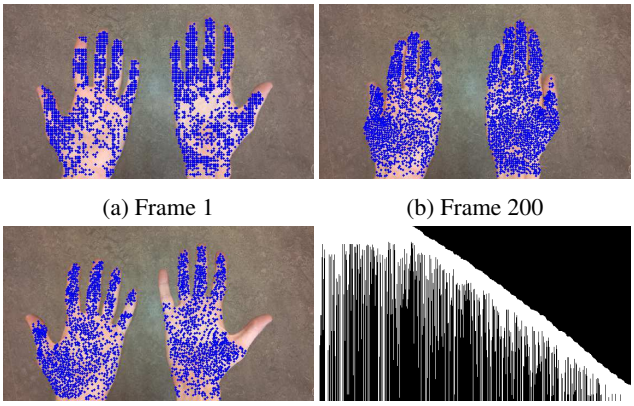
Interestingly, when we consider rows instead of columns (see Figure 2) the relation is different. In SfM this situation corresponds to estimating a new scene shape from a shape model. For the global and the unified models there are 5 coefficients that needs to be determined. For the local model there are 9 since the cluster bases are independent. Hence the global and unified models can recover the entire row using 5 available measurements while the local model requires 9. Furthermore, note that the local model needs at least three measurements for each cluster since these are estimated independently. In contrast, the unified model could theoretically predict the entire row from measurements in a subset of the clusters. Specifically, if X_{new} is the new row (with missing data) we want to find a row B_{new} by solving

$$X_{\text{new}} = B_{\text{new}} \begin{bmatrix} U_1 C_1^T & U_T C_2^T & \dots & U_K C_K^T \end{bmatrix} \qquad (12)$$

(possibly in a least squares sense). This is possible if the columns of $\begin{bmatrix} U_1 C_1^T & U_2 C_2^T & \dots & U_K C_K^T \end{bmatrix}$ that correspond to known data entries of X_{new} span a r_0-dimensional space. In the example of Figure 2 each $U_i V_i^T$ is of at most rank 3 hence it is not possible to completely determine B_{new} from only one cluster. However two clusters could be enough if their columns span the entire column space of B.

Next we show a real example that illustrates the benefits of using the unified model. The sequence consists of images containing two hands flexing, see Figure 3. Using the method of [36] we tracked points on the hands throughout the sequence. The dataset contains 7899 point trajectories in 441 frames with 67% missing data due to tracking failures. Figure 3 shows three of the 441 images together with the tracked points as well as the missing data pattern.

The point trajectories were manually partitioned into 14 approximately rigid components (see Figure 4d). Since each rigid component essentially only undergoes planar rotation and translation we restrict each cluster to a two-dimensional affine subspace (i.e. $r_k = 3$). For the global model we used $r_0 = 5$.



(a) Frame 1 (b) Frame 200

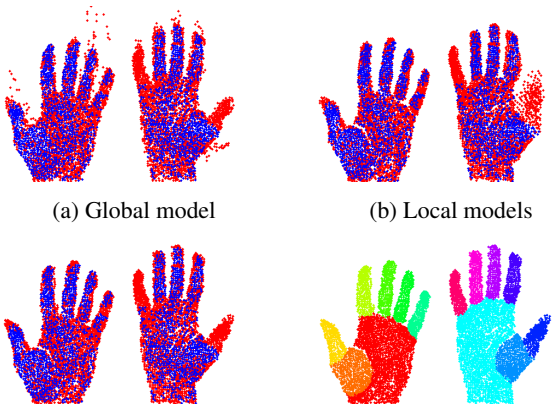
(c) Frame 441 (d) The missing data pattern.

Figure 3: Frames 1, 200 and 441 of the hand sequence. Note that in the last frame the right thumb has no tracks. Bottom right shows the missing data pattern. The observed entries of the measurement matrix are shown in white.

Figure 4 shows the result for the last frame of the sequence. In this frame the right thumb has almost no point trajectories due to tracking failures. Using only the global model (Figure 4a) we can successfully recover the unobserved thumb but each rigid part is over-parameterized leading to over-fitting and noisy tracks. Table 1 (first column) shows the number of parameters for the three alternatives.

Using only the local models (Figure 4b) it is difficult to recover the correct track locations at the thumb when there are only a few visible tracks. Combining both the global and local models (Figure 4c) allows us to deal with the missing observations without over-parameterizing each rigid part.

Figure 5 illustrates how the unified model can estimate new poses (rows) from only 5 known point positions (since $r_0 = 5$). Note that since the hands move together throughout the sequence the learned model can infer the pose of the right hand (for which there are no measurements) from the left. If the clusters were treated independently each cluster would need at least 3 measurements for successful estimation. On the other hand our model would not fit well to a



(a) Global model (b) Local models

(c) Unified model (d) Partitioning for local models.

Figure 4: The reconstructed tracks in the last frame of the sequence. The tracks which have observations in the current frame are shown in blue.

Dataset	Hand	Paper	Back	Heart
Global model	43880	3311	166824	547576
Local models	52758	2750	63472	163134
Unified model	20309	1686	43908	138814

Table 1: DOF for the three type of models for various datasets used in the experiments (see Section 5.2 for more information.)

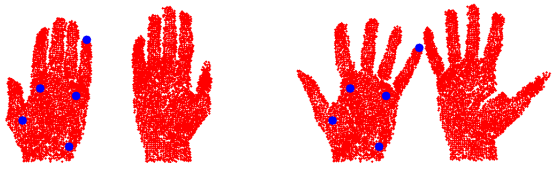


Figure 5: The constraint $r_0 = 5$ allows us to generate new shapes. Here the position of the five blue points where specified while the red points where predicted by the model.

new image where for example the distance between the two hands is significantly different from what has been previously observed.

4. Inference

In this section we present an energy-based optimization framework for computing compact factorizations. Given a measurement matrix M we seek a factorization

$$W \odot M \approx W \odot \left(B \begin{bmatrix} U_1 C_1^T & \dots & U_K C_K^T \end{bmatrix} P\right), \quad (13)$$

where P is a permutation matrix that switches the order of columns and W is a binary matrix with element $w_{ij} = 1$ if m_{ij} is known and 0 otherwise. Changing column order using P corresponds to assigning a column of M to a particular cluster. Note that the overall rank r_0 (and thereby the size of B) is assumed to be known (otherwise it is possible that rank estimation methods similar to [21] could be used). However, the cluster number K, the ranks r_k and assignments are estimated by penalizing a trade-off between data fit and complexity.

For a fixed B determining the factorization can be seen as a model fitting problem where we assign affine subspaces to the columns of M. In the discrete setting, it is well known that these problems are NP hard [20]. However, [20, 11] has demonstrated that move making approaches such as α-expansion typically provide good solutions.

4.1. Energy Formulation

The approach we take essentially follows [20, 11] which generates a large but finite number of proposal subspaces and fuses them into a complete clustering by optimizing a discrete labeling energy using α-expansion [5].

Let l be a labeling of the matrix columns. Then given a finite set of proposal subspaces $\{BU_k\}$, letting $l_p = k$ corresponds to assigning column p to cluster k. Note that once a column is assigned to a local subspace the coefficients C_k can be determined solving a simple least squares problem.

From the proposals we compute the cluster assignment by minimizing the discrete function

$$E(l) = \sum_p D_p(l_p) + \sum_k h_k \delta_k(l). \quad (14)$$

The data term D_p consists of two components. The first is a standard least squares term that measures the fit to the measurement matrix. The second component counts the number of elements required for representing the column in the factorization. Specifically, we use

$$D_p(k) = \min_c \|W_p \odot (M_p - BU_k c)\|_F^2 + \lambda(r_k - 1), \quad (15)$$

where W_p and M_p denote the p:th column of W and M respectively. Summing over the columns in the cluster the second term contributes $\lambda n_k(r_k - 1)$, which is the DOF in the C_k matrix of (11) times a weight λ. The weight λ controls the trade-off between data-fit and DOF.

The second term in (14) is a label cost term which we use to encode the remaining part of the model-complexity in (11) by setting

$$h_k = \lambda \left(r_0 r_k - (r_k - 1) r_k\right). \quad (16)$$

The function δ_k returns one if any of the columns is assigned to proposal k and zero otherwise. Thus using both the data term and the label cost we can achieve an adaptive penalization of the complexity of the factorization. Since we assume that r_0 is known the first term of (11) is constant and ignored.

Note that a pairwise Potts terms $V_{pq}(l_p, l_q)$ [5] can easily be introduced to (14) to add geometric context. From a practical point of view this can help to resolve ambiguous assignments in the vicinity of subspace intersections and therefore typically yield visually more appealing clusters. However this requires a neighborhood system and a number of additional parameters. For the experiments we therefore only use (14). In the supplementary material we perform experiments with the pairwise term.

4.2. Optimization

It is clear from [11] that the above energy yields submodular α-expansions. Note however that the dimensionality of the search space is typically very large, which makes efficient proposal generation difficult. For example, to compute a 3-dimensional affine subspace we need to specify the elements of 4 columns, that is $4m$ elements, where m is the number of rows of M. Furthermore, because of missing data we cannot expect to be able to sample complete columns directly from M. To address this issue we maintain estimates of the B, U_k and C_k matrices and use these to fill in the measurement matrix. Using the completed measurement matrix we sample subsets of columns M_s and use these to estimate new U_k such that $M_s \approx BU_k$. If there is no application specific prior on the dimension of the local subspaces, the number of sampled columns is also selected at random in order to ensure that subspaces of different dimensions are generated. We employ the above proposal generation with α-expansion as outlined in [20]. In each iteration re-estimation is performed individually for the B, U_k and C_k matrices by solving the corresponding linear least squares problems. For initialization we find one r_0-dimensional subspace for the whole matrix using local optimization.

5. Experiments

In this section we will evaluate the performance of our method both quantitatively and qualitatively on different datasets and compare to several state-of-the-art methods. In order to obtain ground truth data we use a number of publicly available data sets and remove random entries from these. Figure 6 shows the data patterns that we consider. In the left pattern entries were discarded with a uniform probability. It is well known from compressed sensing that nuclear norm optimization works well (and even has performance guarantees [8]) for this kind of data. We argue that this setup is of limited interest since it does not occur in tracking based applications and further results in easier problem instances. Therefore we only test this type of data in Section 5.3 for completeness.

To construct more realistic patterns we simulate tracking failure by randomly selecting (with uniform probability) if a track should have missing data. We then select (with uniform probability after the first few frames) in which image tracking failure occurs. No track is restarted after it has been lost. This results in data patterns such as the on displayed in the middle of Figure 6. In Section 5.2 we further simulate occlusion by removing a complete block of the matrix, see the right pattern of Figure 6.

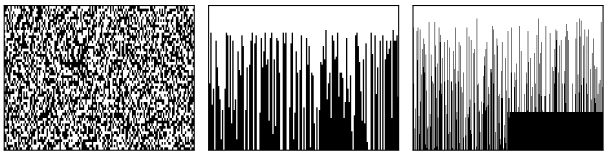


Figure 6: Examples of synthetic missing data patterns used for the experiments. Observed entries are shown in white. *Left:* Uniformly missing entires. *Middle:* Trajectories exhibiting tracking failure. *Right:* Tracking failure and occlusion.

5.1. Effects of the Trade-off Parameter λ

Our energy contains one parameter λ that controls the trade-off between model fit and DOF. To evaluate the behavior of our energy for different λ we use one instance from the CMU Motion Capture dataset. We used subject 10, which contains 5 sequences of a person kicking a soccer ball and one sequence of walking. These were selected since they were approximately the same size and they all provided about 330 point trajectories. The 3D points were projected into an orthographic camera slowly rotating around the subject. Some example frames can be seen in Figure 7. We generated missing data patterns as displayed in the middle image of Figure 6. Figure 8 shows how the resulting errors

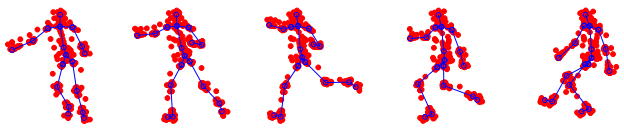


Figure 7: Some example frames from one of the soccer kick instances. Blue skeleton added for visualization.

on the observed and missing data as well as the model complexity varies for different λ. For low values of λ the model fit term dominates the energy giving almost perfect fit to the available measurements. On the other hand model complexity is high which limits the ability to accurately predict missing data. The best results are achieved for mid range values of λ (in this case between 1 and 10). When λ is high the DOF of the model becomes too low to be able to capture the full scene dynamics resulting in poor prediction.

5.2. Occlusion and Tracking Failures

In this section we show some result on trajectories from three public image sequences: The *paper sequence* [33]

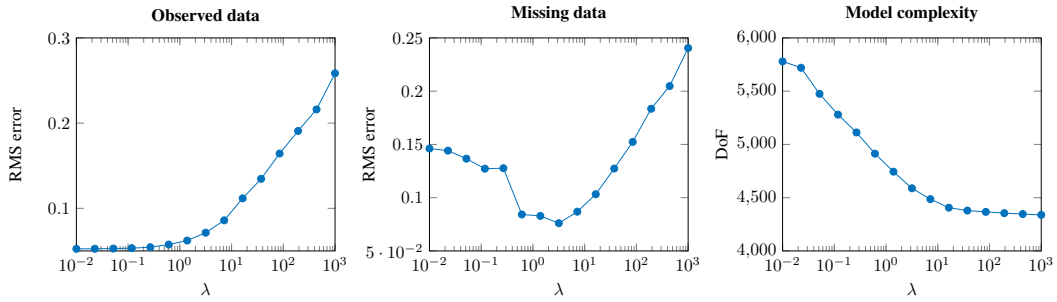


Figure 8: For different values of λ the plots show; *Left:* the RMS error on the observed data, *Middle:* the RMS error on the missing data, *Right:* the degrees of freedom in the resulting model.

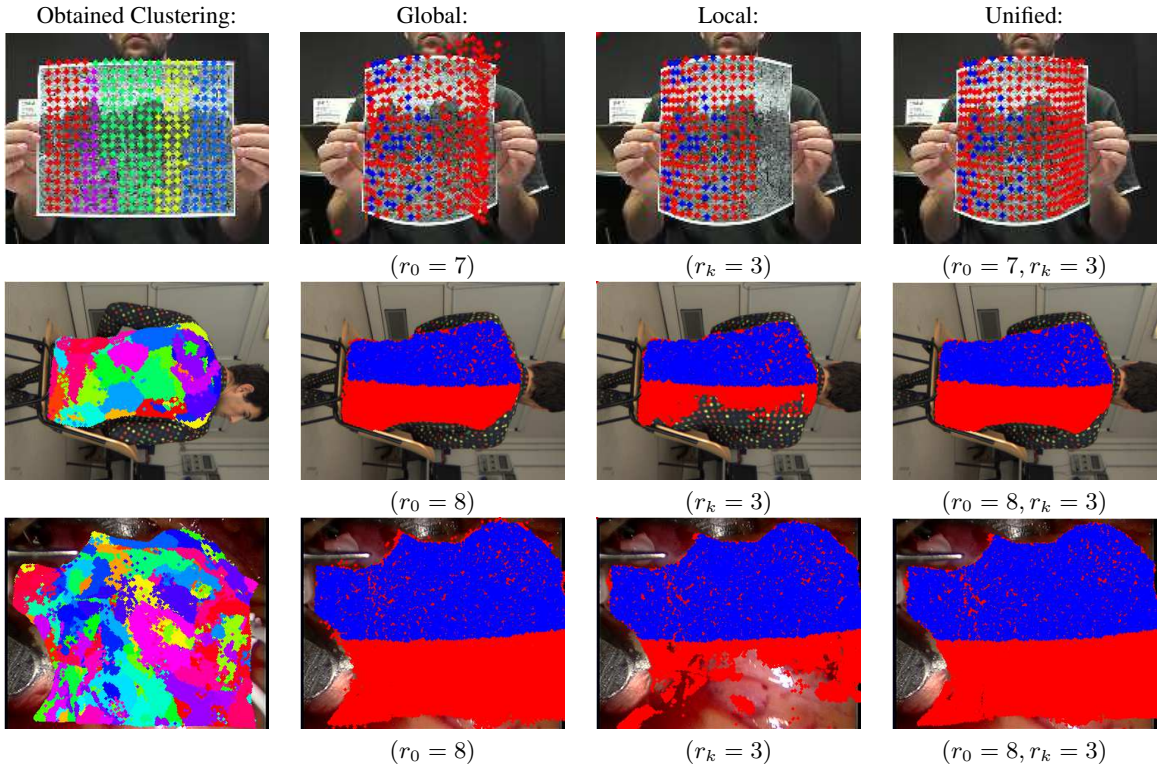


Figure 9: Paper, Back and Heart sequences from [33, 16]. The left column shows the clustering obtained using our method. The rest of the columns show the visible points (blue) and the reconstructed points (red) in one frame.

containing 340 points in 70 frames, the *back sequence* [16] containing 20561 points in 150 frames and the *heart sequence* [16] containing 68295 points in 80 frames. To these we generated missing data as illustrated in the right image of Figure 6. (For occlusion we remove all trajectories in one half of the image for the last 25% of the frames.) Since these sequences are roughly locally planar we only sample affine rank 3 subspaces. We used $\lambda = 500$ in all three cases. Figure 9 shows the obtained clusterings and one frame from each sequence with the visible (blue) points and the reconstructed (red) points. Here we compare the local, global and unified models. For the local model we used the clustering computed by our method. Note clusters that do not have any visible points due to occlusion are not reconstructed by the local method. Table 2 shows the reconstruction errors for both missing and visible points. Table 1 shows the DOF of the resulting factorizations.

5.3. Quantitative Comparisons

Next we compare our approach to a number of state-of-the-art methods. We test four methods that are based a single global rank model:

- **LM-r0** and **Wiberg**: Fitting a rank r_0 matrix by minimizing $f(B, C) = \left\| W \odot (BC^T - M) \right\|_F^2$ using

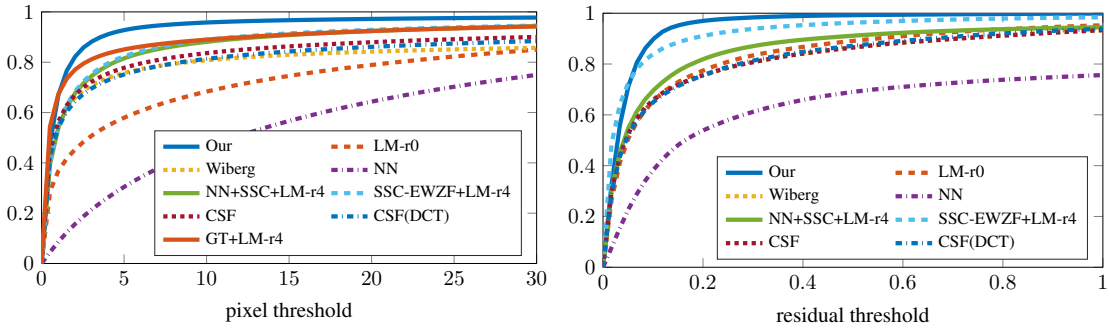


Figure 10: The fraction of residuals across all instances less than a threshold for the Hopkins (left) and MOCAP data (right).

	Observed data			Missing data		
	Paper	Back	Heart	Paper	Back	Heart
Global	3.95e1	5.3e2	1.1e3	3.27e3	1.9e3	4.4e3
Local	**1.04e2**	**4.9e2**	**1.1e3**	2.79e4	1.1e5	2.2e5
Unified	1.07e2	6.9e2	1.5e3	**2.86e2**	**1.4e3**	**3.6e3**

Table 2: Reconstruction errors for visible and missing data. For each column the smallest errors highlighted in bold.

Levenberg-Marquardt and the damped Wiberg method from [29] respectively.

- **CSF** and **CSF-DCT**: The column space fitting method from [18], both with and without using the DCT basis.
- **NN**: Nuclear norm minimization $f(X) = \lambda \|X\|_* + \|W \odot (X - M)\|_F^2$ using ADMM [4].

We also test using two approaches from [42] for clustering the columns followed by fitting local models to each cluster. For these methods we use rank 4 affine models since these correspond to rigid 3D objects.

- **SSC-EZWF+LM-r4**: The Entry-wise Zero-Fill method from [42], followed by fitting rank 4 affine models to each cluster.
- **NN+SSC+LM-r4**: Nuclear norm minimization (as in **NN**) followed by regular SSC [12]. Affine rank 4 models are fitted to the resulting clusters.

For the competing methods the available parameters were tuned for each dataset to give the best results. In our comparisons we measure the fraction of matrix elements that have reconstruction error less than a given threshold. This is because over-fitting to noise may lead to highly unstable tracks which results in unpredictable ℓ_2 errors.

Hopkins155 and CMU Motion Capture. We first consider the Hopkins155 dataset [37] which is commonly used for motion segmentation. It contains 155 sequences with multiple rigidly moving objects. Since the dataset contains a ground truth clustering of the trajectories, we include a comparison with fitting local models to this partition. This is denoted **GT+LM-r4** in the results. For the methods using a global rank constraint we use $r_0 = 4K$ where K is the number of scene motions.

	Missing data	
Method	Uniform	Tracking failure
Our	**155 (100.0%)**	**148 (95.5%)**
LM-r0	143 (92.3%)	26 (16.8%)
Wiberg	152 (98.1%)	75 (48.4%)
NN	**155 (100.0%)**	6 (3.9%)
NN+SSC+LM-r4	**155 (100.0%)**	97 (62.6%)
SSC-EWZF+LM-r4	**155 (100.0%)**	93 (60.0%)
CSF	118 (76.1%)	70 (45.2%)
CSF(DCT)	154 (99.4%)	54 (34.8%)
GT+LM-r4	**155 (100.0%)**	88 (56.8%)

Table 3: Number of instances where 90% of the missing entries have less than 10px error.

Here we generated both uniform missing entries and random tracking failure (left and middle of Figure 6). Table 3 shows the number of instances where more than 90% of the missing data was reconstructed with less than a 10 pixel error. In Figure 10 we vary the pixel threshold and show the fraction of residuals, across all instances, that are reconstructed with a lower error. (Here we did not consider the uniform missing entry pattern.) It is a bit surprising that our approach outperforms competing methods, even **GT+LM-r4**, on this dataset since the ground truth clusters are typically independently moving objects. Closer inspection reveals that our method often splits objects into smaller dependent models, whose interactions are captured by the global rank constraint. Thus a likely explanation is that affine rank 4 models with ground truth clusters is still an over-parametrization. To the right in Figure 10 we show the results obtained when performing the same experiment on subject 10 of the MOCAP data set.

References

[1] Ijaz Akhter, Yaser Ajmal Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *Neural Information Processing Systems*, 2008. 1

[2] Pedro M. Q. Aquiar, Marko Stosic, and Joao M. F. Xavier. Spectrally optimal factorization of incomplete matrices. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 1

[3] Ronen Basri, David Jacobs, and Ira Kemelmacher. Photometric stereo with general, unknown lighting. *Int. J. Comput. Vision*, 72(3):239–257, May 2007. 1

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011. 8

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Trans. Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001. 5, 6

[6] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2000. 1

[7] A. M. Buchanan and A. W. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 2

[8] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):11:1–11:37, June 2011. 1, 6

[9] N.P. da Silva and J.P. Costeira. The normalized subspace inclusion: Robust clustering of motion subspaces. In *IEEE International Conference on Computer Vision*, pages 1444–1450, 2009. 2

[10] Yuchao Dai and Hongdong Li. Rank minimization or nuclear-norm minimization: Are we solving the right problem? In *DICTA*, 2014. 2

[11] Andrew Delong, Anton Osokin, Hossam Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 96:1–27, Jan. 2012. 5, 6

[12] Ehsan Elhamifar and Rene Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2765–2781, 2013. 8

[13] A. Eriksson and A. Hengel. Efficient computation of robust weighted low-rank matrix approximations using the L_1 norm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(9):1681–1690, 2012. 2

[14] Katerina Fragkiadaki, Marta Salas, Pablo Arbelaez, and Jitendra Malik. Grouping-based low-rank trajectory completion and 3d reconstruction. In *Advances in Neural Information Processing Systems 27*. 2014. 2

[15] Ravi Garg, Anastasios Roussos, and Lourdes Agapito. A variational approach to video registration with subspace constraints. *Int. J. Comput. Vision*, 104(3):286–314, 2013. 1

[16] Ravi Garg, Anastasios Roussos, and Lourdes de Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 7

[17] N. Gillis and F. Glinuer. Low-rank matrix approximation with weights or missing data is np-hard. *SIAM Journal on Matrix Analysis and Applications*, 32(4), 2011. 1

[18] Paulo FU Gotardo and Aleix M Martinez. Non-rigid structure from motion with complementary rank-3 spaces. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3065–3072. IEEE, 2011. 2, 8

[19] Je Hyeong Hong and Andrew Fitzgibbon. Secrets of matrix factorization: Approximations, numerics, manifold optimization and random restarts. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4130–4138, 2015. 2

[20] Hossam Isack and Yuri Boykov. Energy-based geometric multi-model fitting. *International journal of computer vision*, 97(2):123–147, 2012. 5, 6

[21] Pan Ji, Mathieu Salzmann, and Hongdong Li. Shape interaction matrix revisited and robustified: Efficient subspace clustering with corrupted and incomplete data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4687–4695, 2015. 5

[22] Q. Ke and T. Kanade. Robust L_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *CVPR*, pages 739–746, San Diego, USA, 2005. 2

[23] Chen Kong and Simon Lucey. Prior-less compressible structure from motion. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2016. 2

[24] Viktor Larsson, Erik Bylow, Carl Olsson, and Fredrik Kahl. Rank minimization with structured data patterns. In *European Conference on Computer Vision*, 2014. 1

[25] Xiao Liang, Xiang Ren, Zhengdong Zhang, and Yi Ma. *Repairing Sparse Low-Rank Texture*. 2012. 1

[26] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013. 2

[27] Manuel Marques and João Costeira. Estimating 3d shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, 113(2):261 – 272, 2009. 2

[28] T. Okatani and K. Deguchi. On the wiberg algorithm for factorization with missing data. *International Journal of Computer Vision*, 72(3):329–337, 2007. 2

[29] Takayuki Okatani, Takahiro Yoshida, and Koichiro Deguchi. Efficient algorithm for low-rank matrix factorization with missing components and performance comparison of latest algorithms. In *International Conference on Computer Vision*, 2011. 8

[30] Carl Olsson and Magnus Oskarsson. A convex approach to low rank matrix approximation with missing data. In *Scandinavian Conference on Image Analysis*, 2009. 1

[31] Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.*, 52(3):471–501, August 2010. 1

[32] Chris Russell, Rui Yu, and Lourdes Agapito. Video pop-up: Monocular 3d reconstruction of dynamic scenes. In *European Conference on Computer Vision*, pages 583–598. 2014. 2

[33] Mathieu Salzmann, Richard Hartley, and Pascal Fua. Convex optimization for deformable surface 3-d tracking. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007. 6, 7

[34] Qinfeng Shi, Anders Eriksson, Anton Van Den Hengel, and Chunhua Shen. Is face recognition really a compressive sensing problem? In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 553–560. IEEE, 2011. 1

[35] D. Strelow. General and nested Wiberg minimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 2

[36] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *Computer Vision–ECCV 2010*, pages 438–451. Springer, 2010. 4

[37] Roberto Tron and René Vidal. A benchmark for the comparison of 3D motion segmentation algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 8

[38] René Vidal, Roberto Tron, and Richard Hartley. Multiframe motion segmentation with missing data using powerfactorization and gpca. *International Journal of Computer Vision*, 79(1):85–105, 2008. 2

[39] T. Wiberg. Computation of principal components when data are missing. In *Proc. Second Symp Computational Statistics*, 1976. 2

[40] John Wright, Allen Y Yang, Arvind Ganesh, S Shankar Sastry, and Yi Ma. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–227, 2009. 1

[41] Jingyu Yan and Marc Pollefeys. A factorization-based approach for articulated nonrigid shape, motion and kinematic chain recovery from video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(5):865–877, 2008. 1

[42] Congyuan Yang, Daniel Robinson, and René Vidal. Sparse subspace clustering with missing entries. In *Proc. of The 32nd International Conference on Machine Learning*, 2015. 2, 8

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

[44] Yingying Zhu, Dong Huang, Fernando De La Torre, and Simon Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2