

On the Minimal Problems of Low-Rank Matrix Factorization

Fangyuan Jiang Magnus Oskarsson Kalle Åström
Centre for Mathematical Sciences
Lund University, Sweden

{fangyuan, magnuso, kalle}@maths.lth.se

Abstract

Low-rank matrix factorization is an essential problem in many areas including computer vision, with applications in e.g. affine structure-from-motion, photometric stereo, and non-rigid structure from motion. However, very little attention has been drawn to minimal cases for this problem or to using the minimal configuration of observations to find the solution. Minimal problems are useful when either outliers are present or the observation matrix is sparse. In this paper, we first give some theoretical insights on how to generate all the minimal problems of a given size using Laman graph theory. We then propose a new parametrization and a building-block scheme to solve these minimal problems by extending the solution from a small sized minimal problem. We test our solvers on synthetic data as well as real data with outliers or a large portion of missing data and show that our method can handle the cases when other iterative methods, based on convex relaxation, fail.

1. Introduction

Assume that we have a matrix $X \in \mathbb{R}^{m \times n}$ of rank $r < \min(m, n)$. Low-rank matrix factorization aims at finding two matrices $U \in \mathbb{R}^{r \times m}$ and $V \in \mathbb{R}^{r \times n}$, such that X can be written as

$$X = U^T V. \quad (1)$$

In geometric computer vision, several problems can be formulated as low-rank matrix factorization. For example, for affine structure-from-motion (SfM), the observation matrix containing feature tracks can be factorized into the camera motions and the 3D structure. For photometric stereo, the directions of light sources and the surface normals are separated by factorizing the measurement matrix composed of pixel intensities under a Lambertian model. More examples of applications can be found in [2, 3, 11, 30].

For real data, with noise or outliers, the observation matrix will in general not have the exact rank as given. So the

problem is usually formulated as minimizing the following cost function,

$$\min_{U, V} \|X - U^T V\|, \quad (2)$$

where $\|\cdot\|$ is a matrix norm – typically the L_1 - or L_2 -norm. This is under the assumption that all entries in X are known. When missing data is present, a binary matrix $W \in \{0, 1\}^{m \times n}$ is used to indicate if a certain entry $X(i, j)$ is present ($W(i, j) = 1$) or missing ($W(i, j) = 0$). In this case, the problem is formulated as

$$\min_{U, V} \|W \odot (X - U^T V)\|, \quad (3)$$

where \odot is the Hadamard product, that is, the element-wise product.

Related work Early work on low-rank matrix factorization solve (2) using the L_2 -norm. Truncating the singular value decomposition of X has been shown to give the optimal solution under the L_2 -norm when X is complete, see [9]. The work in [29] was the first to consider missing data.

However, the L_2 -norm is sensitive to outliers. An early work that aims for robustness to outliers is [1]. Here they use an iteratively re-weighted least squares approach to optimize the objective function. In [4], a damped Newton method is proposed to handle the missing data. Since [16], more robust norms have been considered. In [16], algorithms based on alternating optimization are introduced under the Huber-norm and the L_1 -norm. In [10], the Wiberg algorithm [29] is generalized from the L_2 -norm to the L_1 -norm. A number of recent works put extra constraints on the factor matrices U and V . In [5], the constraints that the matrix U should lie on a certain manifold for different applications are considered and incorporated in the formulation. In [31], orthogonal constraints on the columns of U and a nuclear norm regularizer on V are incorporated.

Most methods mentioned above are based on alternating optimization and are prone to get trapped in local minima. Recently, several works [8, 7, 21, 11, 25] re-formulate the problem to minimize the convex surrogate of the rank

function, that is, the nuclear norm. This makes it possible to use convex optimization to find the global optimum of the approximated objective function. These approaches can handle the problems when the rank is not known a priori. However, for applications with a given rank, the nuclear norm based methods usually perform inferior to the bilinear formulation-based methods [6]. Another drawback of the nuclear norm method is that it is sensitive to outliers. The performance is also affected by an increasing number of missing data.

A few recent works [23, 15, 20, 19] also explore the idea to divide the whole matrix into overlapping sub-blocks and combine the sub-block solutions. One motivation is due to the structured missing data patterns in the measurement matrix. For example, for SfM datasets the observations normally span a diagonal band while off-diagonal entries are mainly missing data. However, these methods do not consider both outliers and missing data at the same time. We use a similar block-partition strategy, but give a more general solution by considering both outliers and missing data.

In this paper, we investigate the minimal problems in low-rank matrix factorization. It turns out that there are many of them and that the structure of these minimal problems is extremely rich and complicated. Nevertheless, we will give some results concerning characterization and generation of such problems and also provide an initial study on solving them. It is worth noting that a few works are relevant to this paper. For the specific SfM problem, estimation of the minimal cases for missing data were investigated in [26], but here we look at a much more general problem. In [14], a combinatorial method is proposed by searching among the linearly solvable minimal cases, while in this paper, we provide a unified view on all the minimal problems and give solvers for more general minimal problems, not only restricted to linearly solvable ones. The concepts based on rigid graphs have also been previously investigated for multidimensional scaling problems with missing data in [27].

Our main contributions in this paper are: (i) generation and characterization of the minimal problems, (ii) solvers to a number of these minimal problems and (iii) algorithms using minimal solvers to factorization problem with missing data and outliers in L_1 -norm or L_2 norm.

2. Generating the minimal problems

We can define the following partial order relationship " \leq " between two index matrices W and W' .

Definition 2.1. Given two index matrices W and W' , we say that $W \leq W'$ if $W(i, j) = 1 \implies W'(i, j) = 1$. Here W is submatrix of W' . $W < W'$ if W is a strict submatrix of W' .

An index matrix W is said to be *rigid* if for general data,

the low-rank matrix factorization problem given by

$$W \odot (X - U^T V) = 0, \quad (4)$$

is *locally well defined*.

The notion of rigidity is invariant under the permutation of the rows or columns. We define the following equivalent relationship between two index matrices

Definition 2.2. Given two index matrices W and W' , we say that W is equivalent to W' if $W(i, j) = W'(P_1(i), P_2(j))$ where P_1 and P_2 are permutations of row/column indices for all $\{i, j\}$

Before giving the definition of the minimal problems in matrix factorization, we look at the degrees of freedom (DoF) for a low-rank matrix. For a rank- r factorization of a matrix $X \in \mathbb{R}^{m \times n}$ we have that U has mr DoF and V has nr DoF. There is a total coordinate ambiguity of size $r \times r$ as

$$X = U^T V = U^T Q Q^{-1} V, \quad (5)$$

where $Q \in \mathbb{R}^{r \times r}$ is a full rank matrix. Thus a matrix $X \in \mathbb{R}^{m \times n}$ with rank r , has $mr + nr - r^2$ degrees of freedom. This means that we need at least $d = mr + nr - r^2$ measurements to recover a rank- r matrix X of size $m \times n$.

A minimal problem for low-rank matrix factorization is characterized using a minimal index matrix, which is defined as

Definition 2.3. An index matrix W for a rank- r problem is said to be minimal if it is rigid and satisfies $\sum_{ij} W(i, j) = mr + nr - r^2$. W is said to be overdetermined if $\sum_{ij} W(i, j) > mr + nr - r^2$.

It is trivial to see that if W is rigid and if $W \leq W'$ then W' is also rigid. It also can be shown that if W' is rigid and overdetermined, then there is at least one $W < W'$ that is rigid and minimal.

A minimal low-rank matrix factorization problem is finding two factor matrices U and V that exactly solves (4), where W is a minimal index matrix and X is the measurement matrix. For the minimal problem, with general coefficients, characterized by a minimal index matrix W , there is a finite number $n_W > 0$ of solutions, where n_W only depends on the index matrix W .

2.1. Characterizing minimal index matrices

For every minimal index matrix W of size $m \times n$ there is a corresponding minimal index matrix W' of size $n \times m$ such that $W(i, j) = W'(j, i)$. Without loss of generality we may thus in the discussion assume that $n \geq m$. Each minimal index matrix has $mr + nr - r^2$ non-zero elements. Since by assumption $m \leq n$, we have at most $2nr - r^2$ non-zero elements, which are distributed among n columns. Thus there are never enough non-zero elements to fill up $2r$

non-zero elements of each column, in another word, there is at least one column which has fewer than $2r$ non-zeros. Furthermore it is obvious that for rank- r problems, the minimal index matrices must have at least r non-zero elements in each column, otherwise the corresponding column \mathbf{v} of V has too few constraints to be solvable. So for the column with the smallest number k of non-zero elements we must have $r \leq k < 2r$.

We also notice that since we assume $r < \min(m, n)$ then $m-r \geq 1$. This means that for a minimal index matrix, the number of non-zeros $mr + nr - r^2 \geq nr + r$. If we distribute these non-zeros in n columns, each of which has at least r zeros, then there is at least one column which has k non-zeros, where $k \geq r + 1$.

2.2. Henneberg-like extensions

We will now describe how to generate the minimal problems in low-rank matrix factorization. The inspiration comes from rigid graph theory, where the Henneberg construction is used to generate the Laman graph, see [18, 12]. The idea is that one starts with the smallest index matrix, and by a series of extensions every index matrix can be generated. For example, for $r = 2$, the smallest index matrix is

$$W = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}. \quad (6)$$

In the following we will distinguish between *constructive* extensions and *non-constructive* extensions. For a constructive extension from W to W' , one can infer the number of solutions $n_{W'}$ from n_W and construct the solver, denoted by $f_{W'}$ from f_W . For non-constructive extensions, it can be shown that W is minimal if and only if W' is minimal. However, we can in general neither infer the number of solutions $n_{W'}$ from n_W nor derive a solver $f_{W'}$ from f_W . We propose the following extensions and reductions which are denoted *Henneberg- k extensions/reductions*. Of these Henneberg-1 is constructive, whereas Henneberg- k are generally non-constructive.

Henneberg-1 extension Given a minimal index matrix W for a rank- r problem of size $m \times n$, an extended minimal index matrix W' of size $m \times (n + 1)$ is formed by adding a column with exactly r indices (non-zero elements). The number of solutions is identical, i.e. $n_W = n_{W'}$. Extending an algorithm from f_W to $f_{W'}$ is straightforward. A similar extension can be done by adding a row with r indices.

Henneberg-1 reduction Given a minimal index matrix W for the rank- r problem of size $m \times n$ where there is a column \mathbf{w}_j with exactly r indices, a reduced minimal index matrix W' of size $m \times (n - 1)$ is formed by removing column j . The number of solutions is preserved under the

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \xrightarrow{\text{Hen I}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \xrightarrow{\text{Hen I}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \xrightarrow{\text{Hen II}} \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

Figure 1. An example of generating rank-2 index matrices of increasing size using a sequence of Henneberg-1 and Henneberg-2 extensions.

reduction and the solution for W' can be obtained straightforward from the solution for W . A similar extension can be done by removing a row with r indices.

Henneberg-2 extension Given a minimal index matrix W for the rank- r problem of size $m \times n$, where there is a column \mathbf{w}_j with at least $r + 1$ non-zero elements at rows i_1, \dots, i_{r+1} (such a column must exist from Section 2.1), an extended index matrix W' of size $m \times (n + 1)$ can be formed by first adding a column \mathbf{w}' with exactly $r + 1$ non-zero elements at rows i_1, \dots, i_{r+1} , then setting one of the non-zeros of \mathbf{w}_j to be zero. The resulting index matrix W' will be minimal. A similar extension can be done in a row-wise manner.

Similarly one can define the Henneberg-2 Reduction. This can be generalized to Henneberg- k extension and reduction for $k > 2$. We illustrate an example of how the index matrices for rank-2 problem can be generated using both Henneberg-1 and Henneberg-2 extensions in Fig. 1.

As shown in [12], every minimally rigid graph can be formed using a sequence of Henneberg-1 and Henneberg-2 construction in the context of rigid graphs. In the following we will show that every minimal index matrix can be generated using a series of Henneberg- k extension defined above for rank $r = 1$ and $r = 2$ case. We also make a conjecture that this is the case for the general rank- r problems with $r > 2$.

Theorem 2.4. *Each index matrix for a minimal rank-1 problem can be formed by a series of Henneberg-1 extensions from the 1×1 index matrix $W = [1]$ as the base case.*

Proof. The proof is by induction over size. If the matrix is of size 1×1 then we are finished. Assume that it is true for all matrices of size $m \times n$ with $m + n \leq K$. Take a minimal index matrix with $m + n = K + 1$. From Section 2.1, we know that there always exists a column with at least k non-zero elements where $r \leq k < 2r$. In this case, there is always a column with exactly one non-zero element. After a Henneberg-1 reduction on that column, we obtain a minimal index matrix with $m + n = K$, which can be constructed as our assumption. So the original matrix is a Henneberg-1 extension of a smaller index matrix that is in the assumption, which proves the theorem. A similar proof exists for a row-wise extension. \square

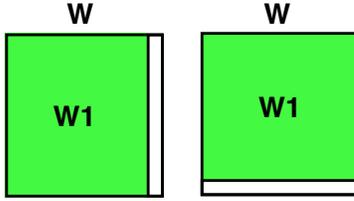


Figure 2. Column-wise and row-wise Henneberg-1 extensions.

Theorem 2.5. *Each index matrix for a minimal rank-2 problem can be formed by a series of Henneberg-1 and Henneberg-2 extensions from the 2×2 index matrix in (6) as the base case.*

Proof. The proof is similarly by induction over size. If the matrix is of size 2×2 then we are finished. Assume that it is true for all matrices of size $m \times n$ with $m + n \leq K$. Take a minimal index matrix with $m + n = K + 1$. From Section 2.1, we know that there is always a column with exactly k non-zero elements where $r \leq k < 2r$, i.e. $k = 2$ or 3 in this case. If the column has two non-zeros then the index matrix can be constructed using the Henneberg-1 extension from the one that is in the assumption. If the column has three non-zeros then it can be reduced to an index matrix of size $m + n \leq K$ using a Henneberg-2 reduction. In either case, we have shown that the index matrix can be constructed using a Henneberg-1 or Henneberg-2 extension from a smaller index matrix given in the assumption, which proves the theorem. The proof for a row-wise extension can be shown similarly. \square

Conjecture 2.1. *Each minimal index matrix for the rank r problem can be formed by a series of Henneberg-1 to Henneberg- r extensions.*

Thus it is possible to generate all the minimal index matrices using a sequence of Henneberg extensions.

3. Minimal solvers

In this section, we will describe the solvers to the minimal problems generated from different Henneberg extensions.

3.1. Solvers for Henneberg-1 extensions

Suppose we are given a rank- r minimal problem with index matrix W and measurements X . Assume that the solution to $\{X, W\}$ is given by $\{U, V\}$ as

$$W \odot (X - U^T V) = \mathbf{0}. \quad (7)$$

Now we apply a Henneberg-1 extension to W to get $W' = [W | \mathbf{w}]$ where the column vector \mathbf{w} has r ones. Correspondingly the measurement matrix is extended to $X' = [X | \mathbf{x}]$

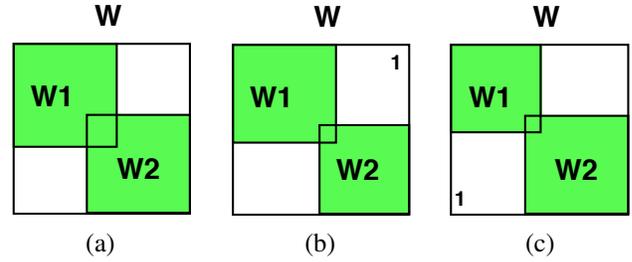


Figure 3. Other constructive extensions. In (a) the overlap of size $r \times r$ gives sufficient constraints. In (b) and (c) some extra constraints are needed when the overlap is smaller than $r \times r$.

where \mathbf{x} has r observations. To find the solution to the extended minimal problem, it is obvious that we only need to solve for the extra column \mathbf{v} of V to satisfy the following equation

$$W' \odot (X' - U^T [V | \mathbf{v}]) = \mathbf{0}. \quad (8)$$

Now assume the positions for the r observations of \mathbf{x} are at $\mathcal{I} = \{n_1, \dots, n_r\}$. Then we have the following equation

$$U_{\mathcal{I}}^T \mathbf{v} = \mathbf{x}_{\mathcal{I}}, \quad (9)$$

where $U_{\mathcal{I}} \in \mathbb{R}^{r \times r}$ and $\mathbf{x}_{\mathcal{I}} \in \mathbb{R}^r$ denotes taking the corresponding rows at \mathcal{I} from U and \mathbf{x} respectively. This is a linear system with a unique solution. For the row-wise Henneberg-1 extension, we will keep V unchanged and solve for the extra row \mathbf{u}^T of U using a similar strategy.

3.2. Solvers for other constructive extensions

Henneberg- k extensions for $k \geq 2$ are non-constructive, which means that for $W \rightarrow W'$ one cannot construct the solution to W' from the solution to W . However, we can define some other constructive extensions. The intuitive idea is that given two index matrices W_1 and W_2 , one can construct a new index matrix W by "glueing" W_1 and W_2 together. By "glueing", we mean that W contains both W_1 and W_2 with overlapping rows and columns as illustrated in Fig. 3.

In the following, we will first describe a general parametrization for these types of constructive extensions, that is independent of the rank r . We derive several constraints from the parametrization that can be used to solve the minimal problem. A few examples for rank-2 and rank-3 are then illustrated.

Parametrization Consider that we have a minimal problem with index matrix W and measurement X . W is constructed by "glueing" two index matrices W_1 and W_2 as in Fig. 3a. We use I_1 and J_1 to denote the row and column indices of W_1 in W , similarly I_2, J_2 for W_2 . Then we have

$W_1 = W(I_1, J_1)$ and $W_2 = W(I_2, J_2)$. Accordingly for measurement matrices X_1 and X_2 associated with W_1 and W_2 , we have $X_1 = X(I_1, J_1)$ and $X_2 = X(I_2, J_2)$. We also use I_{12} and J_{12} to denote the indices of overlapping rows and columns as $I_{12} = I_1 \cap I_2$ and $J_{12} = J_1 \cap J_2$.

Assume that the solutions to the sub-problem $\{W_1, X_1\}$ and $\{W_2, X_2\}$ are given by $\{U_1, V_1\}$ and $\{U_2, V_2\}$ respectively. To construct the solution to $\{W, X\}$, the idea is to find a transformation matrix $H \in \mathbb{R}^{r \times r}$ to transform the subspace U_2 to the same coordinate framework as the subspace U_1 . Using this transformation we have

$$U_2^T V_2 = (U_2)^T H^T H^{-T} (V_2) = (H U_2)^T (H^{-T} V_2). \quad (10)$$

Now $H U_2$ and $H^{-T} V_2$ are in the same coordinate framework as U_1 and V_1 respectively. The remaining problem is to solve for H . We have the following constraint, that states that U_1 and $H U_2$ should coincide for the overlapping columns as

$$U_1(i, I_{12}) = H U_2(i, I_{12}), \quad i = 1, \dots, r. \quad (11)$$

Similarly we have the overlapping constraints for V_1 and $H^{-T} V_2$ as

$$V_1(i, J_{12}) = H^{-T} V_2(i, J_{12}), \quad i = 1, \dots, r. \quad (12)$$

This can be written as

$$H^T V_1(i, J_{12}) = V_2(i, J_{12}), \quad i = 1, \dots, r, \quad (13)$$

which is linear with respect to H .

If we have enough constraints from (11) and (13), H can be solved linearly. In that case, the overlap should be of size $r \times r$ as in Fig. 3a. For the cases where the overlap doesn't give sufficiently many constraints, we need some extra constraint outside W_1 and W_2 to solve for the transformation matrix H as in Fig. 3b-c.

To solve the case with extra constraints, we know that each extra measurement $X(i, j)$ directly gives a constraint as

$$\mathbf{u}_i^T \mathbf{v}_j - X(i, j) = 0, \quad (14)$$

where \mathbf{u}_i and \mathbf{v}_j are i -th and j -th column of U and V respectively. We need to express the constraint using U_1 , U_2 , V_1 and V_2 from which we assume $\{U_1, V_1\}$ are in the same coordinate system as $\{U, V\}$. There are generally two cases. When $X(i, j)$ is in the top-right as in Fig. 3b, one needs to use U_1 and V_2 as

$$U_1(:, i_1)^T H^{-T} V_2(:, j_2) - X(i, j) = 0. \quad (15)$$

where i_1 denotes the local index of column i in U_1 and j_2 the local index of column j in V_2 . When $X(i, j)$ is in the bottom-left as in Fig. 3c, one needs to rewrite (14) using U_2 and V_1 instead as

$$(H U_2(:, i_2))^T V_1(:, j_1) - X(i, j) = 0. \quad (16)$$

where i_2 denotes the local index of column i in U_2 and j_1 the local index of column j in V_1 .

All these constraints from (11), (13), (15) and (16) form either a linear system or a simple polynomial system, from which one can solve the transformation H and thus get a solution to the original problem. In the following we will illustrate a few examples for rank-2 and rank-3 problem, all of which can be parametrized and solved in a similar way.

Examples of rank-2 constructive extensions Here we will present a few constructive extensions for rank-2 problems, which are illustrated in Fig. 4. The overlap for rank-2 problems is at most 2×2 , otherwise the extension is non-minimal. For a 2×2 overlap, H can be solved linearly using only the overlap constraints. For a 1×1 overlap with an extra constraint as in Fig. 4, the overlap in U gives two equations from (11) and the overlap in V gives two equations from (13). Among the four equations, one is redundant as it is automatically satisfied when the other three hold. So one can parametrize H using a single variable, and we can solve using a single extra constraint.

One could of course generate a minimal problem by "glueing" more index matrices. In Fig. 4b-c, each sub-part needs a transformation, namely H_1 , H_2 and H_3 of which one can fix the gauge by setting $H_2 = I$. Two extra constraint are needed to solve the problem. For the case in Fig. 4d, four transformation H_1 , H_2 , H_3 and H_4 are needed. By setting $H_2 = I$, one can parametrize H_1 , H_3 and $H_3^{-1} H_4$ using z_1 , z_2 and z_3 respectively. Each of three extra constraints provide a quadratic equation in z_1 , z_2 and z_3 respectively, which could be solved giving three solutions.

Examples of rank-3 constructive extensions For rank-3 problems, the maximum overlap is 3×3 which can then be solved linearly. The 2×2 overlap case, illustrated in Fig. 5a, H can be parametrized using a single variable z , and can be solved linearly using one extra constraint. When the overlap is 1×1 as in Fig. 5b, the overlap provides three equations in (11) and three equations in (13). However one of them is redundant, which means it reduces the DoF of H from nine to four. Thus four extra constraints are needed to solve for H .

4. Algorithms for structured data patterns

For real problems, a measurement matrix might be more dense than that for a minimal problem. However, one could apply a random sampling strategy similar to [14] to sample minimal configurations from measurements and solve a minimal problem in each RANSAC-like iteration.

In applications where the locations of the missing data are highly correlated and structured – for example

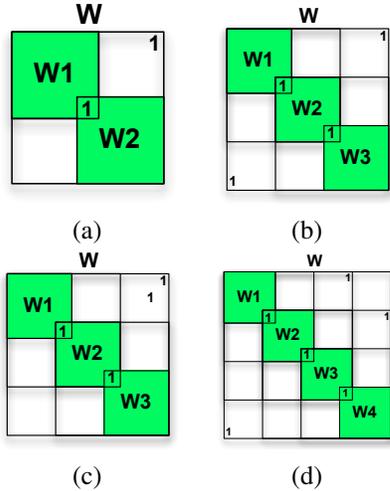


Figure 4. Examples of rank-2 constructive extensions. In each of these four examples several smaller minimal cases are glued together with a few extra constraints. For these extensions the multiplicity of solutions grows as follows: for (a) $n_W = n_{W_1}n_{W_2}$, for (b) $n_W = 2n_{W_1}n_{W_2}n_{W_3}$, for (c) $n_W = 2n_{W_1}n_{W_2}n_{W_3}$, and for (d) $n_W = 3n_{W_1}n_{W_2}n_{W_3}n_{W_4}$. For each of these extensions we have implemented the method of extending the algorithms, i.e. $\{f_{W_1}, f_{W_2}, \dots, f_{W_n}\} \rightarrow f_W$.

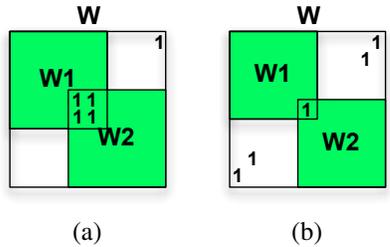


Figure 5. Examples of rank-3 constructive extensions.

affine structure-from-motion – we can use a similar block-partition idea as in [20] to divide the measurement matrix into overlapping sub-blocks. The solution to each sub-block is estimated separately using the random sampling method. Once we obtain solutions to all the sub-blocks, we can use a null space matching method [24] to combine those solutions and recover the full factorization.

The block partition in our method differs from the strategy used in [20] in that using minimal solvers, missing data can be handled in the sub-blocks, while in [20] each sub-block must be complete without missing data based on their convex formulation. As we will show in the experiment, being able to cope with missing data in each sub-block leads to a more flexible way of partition which could cover all the observations.

5. Experiments

We have conducted a number of experiments on both synthetic and real data. For synthetic data and affine structure-from-motion, we use the solvers for both the Henneberg-1 extension and the constructive extensions from Section 3.2. The other constructive extension is useful when the measurement matrix is sparse, especially when it is not possible to find a minimal configuration using only the Henneberg-1 extension. For the shape basis estimation, we use only the Henneberg-1 extension solver, as the matrix is more dense.

5.1. Synthetic data

For synthetic data, we generate random rank-3 matrices of size 100×100 with entries uniformly drawn from $[-1, 1]$. All entries are then perturbed with noise drawn from a normal distribution $\mathcal{N}(0, \sigma)$ where σ takes $(0, 10^{-3}, 10^{-2})$. The rank constraint is enforced using a truncated SVD.

A structured data pattern is formed by removing some measurements in the generated matrices. We are especially interested in the band-diagonal structure which appears in many structure-from-motion problems. A sparse matrix D is said to be band-diagonal with bandwidth k if the following equations hold

$$d_{ij} = 0 \quad \text{for } j < i - k \quad \text{or } j > i + k. \quad (17)$$

Here we generate random band-diagonal matrices of size 100×100 with varied bandwidth k from 20 down to 4. The corresponding proportion of missing data ranges from 64% up to 92%. For comparison, we consider two state-of-the-art methods, namely Truncated Nuclear Norm Regularization (TNNR-ADMM) [13] and OptSpace [17]. OptSpace was initialized using truncated SVD.

We plot the \log_{10} error versus the bandwidth of the matrix in Fig. 6, with the error defined as $\|W \odot (X - U^T V)\|_F$. When the bandwidth decreases (the missing data increases), the performance of both TNNR-ADMM and OptSpace are affected, especially when the bandwidth $k < 10$ which corresponds to around 80% missing data. In the noise-free case, our method achieves significantly lower error. With low level noise, that is $\sigma = 10^{-3}$, our method remains stable with respect to the rate of missing data. With medium level noise, $\sigma = 10^{-2}$, our method will also be affected when $K < 10$, but will still perform better compared to TNNR-ADMM and OptSpace.

5.2. Real data

Affine structure-from-motion We evaluate our method on the well-known dinosaur sequence for affine structure-from-motion. The original dinosaur sequence contains 2D point tracks from the projection of in total 3184 3D points

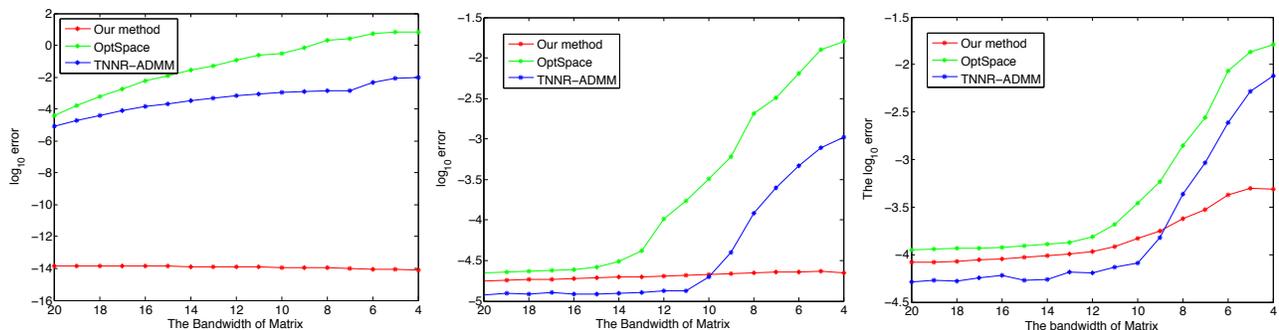


Figure 6. The synthetic data with band-diagonal structure. The bandwidth of matrix vs. The $\log_{10} L_2$ -error. Left: Noise-free case. Middle: Noise level $\sigma = 10^{-3}$, Right: Noise level $\sigma = 10^{-2}$.

in 36 cameras. Each 3D point is only visible in a few consecutive views and missing for the rest of the views due to self-occlusion. We consider rank-4 factorization which does not use any initial estimate of the translation, see e.g. [28].

To reduce processing time, we take a subset of 3D points which are visible in at least 8 views. This forms an observation matrix X of size 72×116 with 87.8% missing data. We first compare with TNNR and OptSpace, neither of which handle outliers. In this case, we divide the measurement matrix into 4 overlapping blocks with varied size, see Fig. 7a. The partition of the measurement matrix is a trade-off between the complexity and the numerical accuracy. For a too large block, the numerical error within the block accumulates for the Henneberg-1 extension solver. If we divide the matrix into too many sub-blocks, it unnecessarily increases the complexity. We ran our method with 1000 iterations followed by a non-linear least square optimization in the final step.

The results are shown in Fig. 7. Both TNNR and OptSpace failed to recover the 2D tracks. We plot the recovered tracks for our method and TNNR in Fig. 7c and 7d respectively. When the data matrix is sparse, the recovered 2D tracks from TNNR are stretched towards the origin (top right corner) of the image.

In the presence of outliers, both TNNR and OptSpace will fail. We thus compare with two L_1 -norm based methods, namely the Wiberg- L_1 in [10] and Regularized L_1 augmented Lagrange multiplier method (Reg L_1 -ALM) in [31]. We add 10% outliers to the measurement matrix that in the range $[-50, 50]$. The only change in our method is that in each iteration we compute the error in L_1 -norm as $\|W \odot (X - U^T V)\|_1$ instead of the Frobenius norm. We run 1000 iterations with our minimal solvers within each sub-block. The average L_1 error for our method is 1.314 pixels and 1.231 for the Wiberg- L_1 and 2.223 for Reg L_1 -ALM.

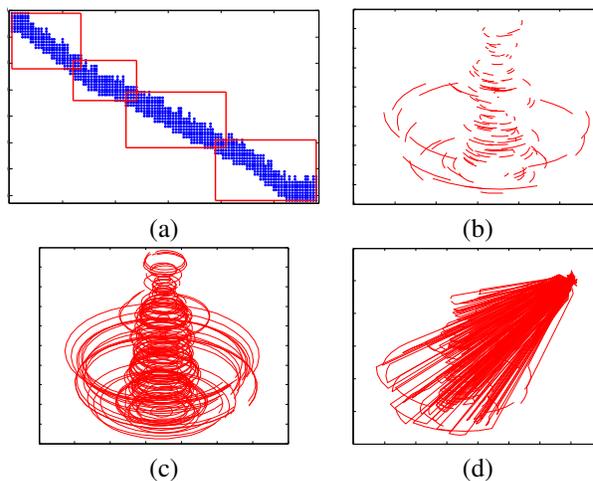


Figure 7. Affine structure from motion on the dinosaur sequence. (a) The band-diagonal structure and block partition. (b) The input 2D tracks. (c) The recovered 2D tracks using our method. (d) The recovered 2D tracks using TNNR.

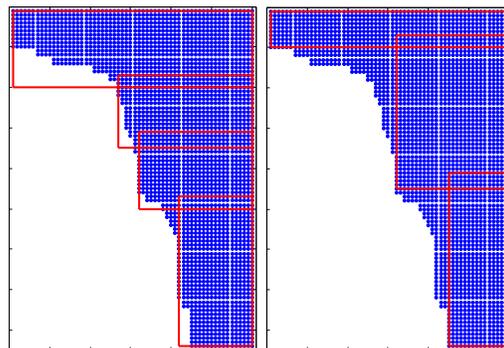


Figure 8. The missing data and block partition on the *Book* dataset. Left: our method. Right: [20]

Linear shape basis For non-rigid structure-from-motion, a linear shape basis model is commonly used to model the

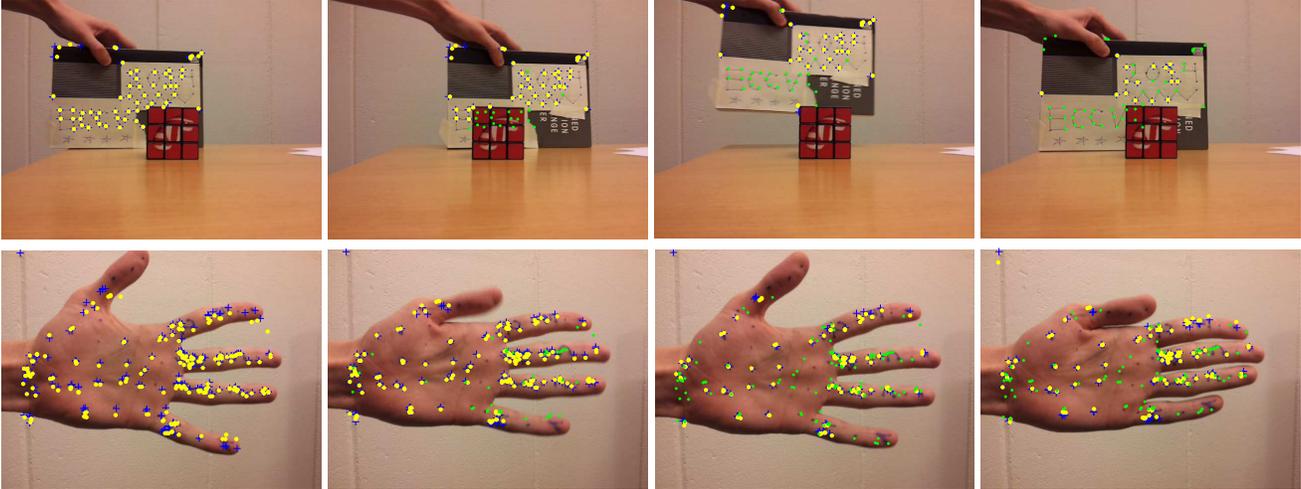


Figure 9. The results on the *Book* (top) and *Hand* (bottom) dataset using our method with an extra 10% missing data added in the blocks. From left to right: The 1st, 10th, 20th, 30th frame. The blue crosses are the measurements. The yellow dots are the recovered measurements (usually coinciding with the actual measurement). The green dots are the recovered missing data.

shape of a non-rigid object. It assumes that any non-rigid deformation of an object can be represented as a linear combination of a set of shapes. Normally the size of the shape basis is much smaller than either the number of frames or the tracked points, so the measurement matrix containing the point tracks can be factorized into a coefficient matrix and a shape basis matrix.

Two datasets – *Book* and *Hand* – from [20] are used. The image points are tracked using a standard *Kanade-Lucas-Tomasi* (KLT) tracker [22]. Due to occlusions, the tracker fails after a number of frames for a subset of points, which leads to the missing data pattern for the *Book* dataset shown in Fig. 8. In our experiments, we use a subset of 42 frames with 60 tracked points from the *Book* and 38 frames with 203 points from the *Hand* dataset. Following the setup in [20], we seek a rank-3 and rank-5 factorization on the two datasets respectively.

The block partition of both our method and [20] are illustrated in Fig. 8. Contrary to [20], missing data within blocks are handled in our method, giving a more flexible partition with wider coverage of measurements. We run our method with 1000 iterations for each block. To further show that our method is capable to handle random missing data, we conduct a second experiment by randomly adding an extra 10% of missing data in the measurement matrix and run both methods with the same settings as before.

We summarize the error $\|W \odot (X - U^T V)\|_F$ in Table 1. Our method achieves smaller error in both dataset with or without adding extra missing data. When random missing data is added into the blocks, [20] fails with large errors. In Fig. 9, the recovered tracked points are plotted for both datasets using our method with extra missing data. Without

Dataset	Algorithm		10 % missing data	
	[20]	<i>Our</i>	[20]	<i>Our</i>
<i>Book</i>	0.3522	0.1740	8.0436	0.1772
<i>Hand</i>	0.8613	0.6891	1.5495	0.7297

Table 1. The result on linear shape basis estimation on the *Book* and *Hand* dataset, where the second experiment contains an extra 10% missing data. The numbers depict the Frobenius errors using our method compared to the method of [20].

extra missing data, both our method and [20] achieve quite similar visual results. The running time for our method is 12s on *Book* and 28s on *Hand*. For [20], it is 2.5s on *Book* and 4.5s on *Hand*. However, the running time for our method can be further reduced as it is highly parallelizable.

6. Conclusion

In this paper, we have introduced theory to characterize and generate the minimal problems of low rank matrix factorization, inspired by the Henneberg extensions from rigid graph theory. We have shown that for rank-1 and rank-2, the proposed Henneberg extensions can generate all minimal problems. We conjecture that using additional Henneberg extensions, all minimal problems can be generated for any given rank. Several solvers are proposed to solve minimal problems using constructive extensions. With block partition and a random sampling scheme, minimal solvers can be used in a number of real applications when the data matrix is sparse and contains outliers.

Acknowledgement

This work has been financed by ELLIIT and eSSENCE.

References

- [1] H. Aanaes, R. Fisker, K. Åström, and J. Carstensen. Robust factorization. *Trans. Pattern Analysis and Machine Intelligence*, 2002. 1
- [2] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72(3):239–257, 2007. 1
- [3] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *Conf. Computer Vision and Pattern Recognition*, 2000. 1
- [4] A. M. Buchanan and A. W. Fitzgibbon. Damped newton algorithms for matrix factorization with missing data. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 316–322, Washington, DC, USA, 2005. IEEE Computer Society. 1
- [5] A. D. Bue, J. M. F. Xavier, L. de Agapito, and M. Paladini. Bilinear modeling via augmented lagrange multipliers (balm). *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(8):1496–1508, 2012. 1
- [6] R. Cabral, F. D. la Torre, J. P. Costeira, and A. Bernardino. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *Int. Conf. Computer Vision*, 2013. 2
- [7] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of ACM*, 2009. 1
- [8] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *CoRR*, abs/0805.4471, 2008. 1
- [9] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936. 1
- [10] A. Eriksson and A. Hengel. Efficient computation of robust weighted low-rank matrix approximations using the L_1 norm. *Trans. Pattern Analysis and Machine Intelligence*, 2012. 1, 7
- [11] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *Computer Vision and Pattern Recognition (CVPR)*, 2013 *IEEE Conference on*, pages 1272–1279. IEEE, 2013. 1
- [12] L. Henneberg. *Die graphische Statik der starren Systeme*, volume 31. BG Teubner, 1911. 3
- [13] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He. Fast and accurate matrix completion via truncated nuclear norm regularization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(9):2117–2130, 2013. 6
- [14] F. Jiang, O. Enqvist, and F. Kahl. A combinatorial approach to l_1 -matrix factorization. *Journal of Mathematical Imaging and Vision*, 2014. 2, 5
- [15] C. Julià, A. D. Sappa, F. Lumberras, J. Serrat, and A. López. An iterative multiresolution scheme for sfm with missing data. *Journal of Mathematical Imaging and Vision*, 34(3):240–258, 2009. 2
- [16] Q. Ke and T. Kanade. Robust L_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *Conf. Computer Vision and Pattern Recognition*, 2005. 1
- [17] R. H. Keshavan, S. Oh, and A. Montanari. Matrix completion from a few entries. *CoRR*, abs/0901.3150, 2009. 6
- [18] G. Laman. On graphs and rigidity of plane skeletal structures. *Journal of Engineering mathematics*, 4(4):331–340, 1970. 3
- [19] V. Larsson and C. Olsson. Convex envelopes for low rank approximation. In *EMMCVPR 2015*, 2015. 2
- [20] V. Larsson, C. Olsson, E. Bylow, and F. Kahl. Rank minimization with structured data patterns. In *ECCV*, 2014. 2, 6, 7, 8
- [21] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of a corrupted low-rank matrices. *CoRR*, abs/1009.5055, 2009. 1
- [22] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI '81, pages 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc. 8
- [23] L. W. Mackey, M. I. Jordan, and A. Talwalkar. Divide-and-conquer matrix factorization. In *Advances in Neural Information Processing Systems*, pages 1134–1142, 2011. 2
- [24] S. I. Olsen and A. Bartoli. Implicit non-rigid structure-from-motion with priors. *Journal of Mathematical Imaging and Vision*, 31(2-3):233–244, 2008. 6
- [25] C. Olsson and M. Oskarsson. A convex approach to low rank matrix approximation with missing data. In *Scandinavian Conf. on Image Analysis*, 2011. 1
- [26] M. Oskarsson, K. Astrom, and N. C. Overgaard. Classifying and solving minimal structure and motion problems with missing data. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 628–634. IEEE, 2001. 2
- [27] M. Oskarsson, K. Astrom, and A. Torstensson. Prime rigid graphs and multidimensional scaling with missing data. In *Pattern Recognition (ICPR)*, 2014 *22nd International Conference on*, pages 750–755, Aug 2014. 2
- [28] G. Wang, J. S. Zelek, Q. J. Wu, and R. Bajcsy. Robust rank-4 affine factorization for structure from motion. In *Applications of Computer Vision (WACV)*, 2013 *IEEE Workshop on*, pages 180–185. IEEE, 2013. 7
- [29] T. Wiberg. Computation of principal components when data are missing. In *Proc. Second Symp. Computational Statistics*, 1976. 1
- [30] J. Yan and M. Pollefeys. A factorization-based approach for articulated nonrigid shape, motion and kinematic chain recovery from video. *Trans. Pattern Analysis and Machine Intelligence*, 2008. 1
- [31] Y. Zheng, G. Liu, S. Sugimoto, S. Yan, and M. Okutomi. Practical low-rank matrix approximation under robust L_1 -norm. In *Conf. Computer Vision and Pattern Recognition*, 2012. 1, 7