Tensor Completion for Estimating Missing Values in Visual Data

Ji Liu[1], Przemyslaw Musialski [2], Peter Wonka[1], and Jieping Ye[1]
Arizona State University[1]
VRVis Research Center[2]
Ji.Liu@asu.edu, musialski@vrvis.at, Peter.Wonka@asu.edu, Jieping.Ye@asu.edu

Abstract

In this paper we propose an algorithm to estimate missing values in tensors of visual data. The values can be missing due to problems in the acquisition process, or because the user manually identified unwanted outliers. Our algorithm works even with a small amount of samples and it can propagate structure to fill larger missing regions. Our methodology is built on recent studies about matrix completion using the matrix trace norm. The contribution of our paper is to extend the matrix case to the tensor case by laying out the theoretical foundations and then by building a working algorithm. First, we propose a definition for the tensor trace norm, that generalizes the established definition of the matrix trace norm. Second, similar to matrix completion, the tensor completion is formulated as a convex optimization problem. Unfortunately, the straightforward problem extension is significantly harder to solve than the matrix case because of the dependency among multiple constraints. To tackle this problem, we employ a relaxation technique to separate the dependant relationships and use the block coordinate descent (BCD) method to achieve a globally optimal solution. Our experiments show potential applications of our algorithm and the quantitative evaluation indicates that our method is more accurate and robust than heuristic approaches.

1. Introduction

In computer vision and graphics, many problems can be formulated as the missing value estimation problem, e.g. image in-painting [4, 12, 11], video decoding, video in-painting, scan completion, and appearance acquisition completion. The core problem of the missing value estimation lies on how to build up the relationship between the known elements and the unknown ones. Some energy methods broadly used in image in-painting, e.g. PDEs [4] and belief propagation [11] mainly focus on the local relationship. The basic (implicit) assumption is that the missing entries mainly depend on their neighbors. The further apart two points are, the smaller their dependance is. However, sometimes the value of the missing entry depends on the entries which are far away. Thus, it is necessary to develop a tool to directly capture the global information in the data.

In the two-dimensional case, i.e. the matrix case, the "rank" is a powerful tool to capture the global information. In Fig. 1, we show a texture with 80% of its elements removed randomly on the left and its reconstruction using a low rank constraint on the right. This example illustrates the power of low rank approximation for missing data estimation. However, "rank(·)" is unfortunately not a convex function. Some heuristic algorithms were proposed to estimate the missing values iteratively [9, 13]. However, they are not guaranteed to find a globally optimal solution due to the non-convexity of the rank constraint.

Figure 1: The left figure contains 80% missing entries shown as white pixels and the right figure shows its reconstruction using the low rank approximation.

Recently, the trace norm of matrices was used to approximate the rank of matrices [14, 5, 16], which leads to a convex optimization problem. The trace norm has been shown to be the tightest convex approximation for the rank of matrices [16], and efficient algorithms for the matrix completion problem using the trace norm constraint were proposed in [14, 5]. Recently, Candés and Recht [6, 16] showed that if a certain restricted isometry property holds for the linear transformation defining the constraints, the minimum rank solution can be recovered by solving a convex optimization problem, namely the minimization of the trace norm over

the given affine space. Their work theoretically justified the validity of the trace norm to approximate the rank.

Although the low rank approximation problem has been well studied for matrices, there is not much work on tensors, which are a higher-dimensional extension of matrices. One major challenge lies in an appropriate definition of the trace norm for tensors. To the best of our knowledge, this has been not addressed in the literature. In this paper, we make two main contributions: 1) We lay the theoretical foundation of low rank tensor completion and propose the first definition of the trace norm for tensors. 2) We are the first to propose a solution for the low rank completion of tensors.

The challenge of the second part is to build a high quality algorithm. Similar to matrix completion, the tensor completion is formulated as a convex optimization problem. Unfortunately, the straightforward problem extension is significantly harder to solve than the matrix case because of the dependency among multiple constraints. To tackle this problem, we employ a relaxation technique to separate the dependant relationships and use the block coordinate descent (BCD) method to achieve a globally optimal solution. In addition, we present several heuristic algorithms, which involve non-convex optimization problems. Our experiments show our method is more accurate and robust than these heuristic approaches. We also give some potential applications in image in-painting, video compression and BRDF data estimation, using the tensor completion technique.

1.1. Notation

We use upper case letters for matrices, e.g. X, and lower case letters for the entries, e.g. x_{ij}. $\Sigma(X)$ is a vector, consisting of the singular values of X in descending order and $\sigma_i(X)$ denotes the i^{th} largest singular value. The Frobenius norm of the matrix X is defined as: $\|X\|_F := (\sum_{i,j} |x_{ij}|^2)^{\frac{1}{2}}$. The spectral norm is denoted as $\|X\|_2 := \sigma_1(X)$ and the trace norm as $\|X\|_{tr} := \sum_i \sigma_i(X)$. Let $X = U\Sigma V^\top$ be the singular value decomposition for X. The "shrinkage" operator $D_\tau(X)$ is defined as [5]:

$$D_\tau(X) = U\Sigma_\tau V^\top, \tag{1}$$

where $\Sigma_\tau = diag(\max(\sigma_i - \tau, 0))$. Let Ω be an index set, then X_Ω denotes the vector consisting of elements in the set Ω only. Define $\|X\|_\Omega = (\sum_{(i,j)\in\Omega} x_{ij}^2)^{\frac{1}{2}}$. A similar definition can be extended to the tensor case.

An n-mode tensor is defined as $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\ldots\times I_n}$. Its elements are denoted as $x_{i_1\ldots i_k\ldots i_n}$, where $1 \le i_k \le I_k, 1 \le k \le n$. For example, a vector is a 1-mode tensor and a matrix is a 2-mode tensor. It is sometimes convenient to unfold a tensor into a matrix. The "unfold" operation along the k-th mode on a tensor $\mathcal{X}$ is defined as $\text{unfold}_k(\mathcal{X}) := X_{(k)} \in \mathbb{R}^{I_k\times(I_1\ldots I_{k-1}I_{k+1}\ldots I_n)}$. The opposite operation "fold" is defined as $\text{fold}_k(X_{(k)}) := \mathcal{X}$. Denote $\|\mathcal{X}\|_F := (\sum_{i_1,i_2,\ldots i_n} |a_{i_1,i_2,\ldots i_n}|^2)^{\frac{1}{2}}$ as the Frobenius norm of a tensor. It is clear that $\|\mathcal{X}\|_F = \|X_{(k)}\|_F$ for any $1 \le k \le n$.

1.2. Organization

We review related work in Section 2, present the proposed tensor trace minimization in Section 3 and three heuristic algorithms in Section 4, report empirical results in Section 5, and conclude this paper in Section 6.

2. Related Work

The low rank or approximately low rank problem broadly occurs in science and engineering, e.g. computer vision [20], machine learning [1, 2], and bioinformatics [21]. Fazel et al. [9, 8] introduced a low rank minimization problem in control system analysis and design. They heuristically used the trace norm to approximate the rank of the matrix. They showed that the trace norm minimization problem can be reformulated as a semidefinite programming (SDP) problem via its dual norm (spectral norm). Srebro et al. [17] employed second-order cone programming (SCOP) to formulate a trace norm related problem in matrix factorization. However, many existing optimization methods such as SDPT3 [19] and SeDuMi [18] cannot solve a SDP or SOCP problem when the size of the matrix is much larger than 100×100 [14, 16]. This limitation prevented the usage of the matrix completion technique in computer vision and image processing. Recently, to solve the rank minimization problem for large scale matrices, Ma et al. [14] applied the fixed point and Bregman iterative method and Cai et al. [5] proposed a singular value thresholding algorithm. In both algorithms, one key building block is the existence of a closed form solution for the following optimization problem:

$$\min_{X\in\mathbb{R}^{p\times q}} : \ \frac{1}{2}\|X - M\|_F^2 + \tau\|X\|_{tr}, \tag{2}$$

where $M \in \mathbb{R}^{p\times q}$, and τ is a constant. Candés and Recht [6, 16] theoretically justified the validity of the trace norm to approximate the rank of matrices.

3. Algorithm

We first introduce the matrix completion problem in Section 3.1 and then we extend to the tensor case in Section 3.2. As we will see, the straightforward extension leads to a very complex optimization problem. In Section 3.3, we relax the original problem into a simple convex structure which can be solved by block coordinate descent. Section 3.4 presents the details of the proposed algorithm.

3.1. Matrix Completion

Our derivation starts with the well-known optimization problem [13] for matrix completion:

$$\begin{aligned} \min_{X} : \quad & \frac{1}{2}\|X - M\|_{\Omega}^2 \\ s.t. \quad & \text{rank}(X) \leq r, \end{aligned} \tag{3}$$

where $X, M \in \mathbb{R}^{p \times q}$, and the elements of M in the set Ω are given while the remaining elements are missing. We aim to use a low rank matrix X to approximate the missing elements. An equivalent reformulation of Eq. (3) is given by

$$\begin{aligned} \min_{X,Y} : \quad & \frac{1}{2}\|X - Y\|_F^2 \\ s.t. \quad & \text{rank}(X) \leq r \\ & Y_{\Omega} = M_{\Omega}. \end{aligned} \tag{4}$$

The optimization problems in Eq. (3) and Eq. (4) are not convex since the constraint $\text{rank}(X) \leq r$ is not convex. One common approach is to use the trace norm $\|.\|_{tr}$ to approximate the rank of matrices. The advantage of the trace norm is that $\|.\|_{tr}$ is the tightest convex envelop for the rank of matrices. This leads to the following convex optimization problem for matrix completion [3, 5, 14]:

$$\begin{aligned} \min_{X,Y} : \quad & \frac{1}{2}\|X - Y\|_F^2 \\ s.t. \quad & \|X\|_{tr} \leq c \\ & Y_{\Omega} = M_{\Omega}. \end{aligned} \tag{5}$$

3.2. Tensor Completion

The tensor is the generalization of the matrix concept. We generalize the completion algorithm for the matrix (i.e. 2nd-order tensor) case to higher-order tensors by solving the following optimization problem:

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y}} : \quad & \frac{1}{2}\|\mathcal{X} - \mathcal{Y}\|_F^2 \\ s.t. \quad & \|\mathcal{X}\|_{tr} \leq c \\ & \mathcal{Y}_{\Omega} = \mathcal{T}_{\Omega}, \end{aligned} \tag{6}$$

where $\mathcal{X}, \mathcal{Y}, \mathcal{T}$ are n-mode tensors with identical size in each mode. The first issue is the definition of the trace norm for the general tensor case. We propose the following definition for the tensor trace norm:

$$\|\mathcal{X}\|_{tr} := \frac{1}{n}\sum_{i=1}^{n} \|X_{(i)}\|_{tr}. \tag{7}$$

In essence, the trace norm of a tensor is the average of the trace norms of all matrices unfolded along each mode. Note that when the mode number n is equal to 2 (i.e. the matrix case), the definition of the trace norm of a tensor is consistent with the matrix case, because the trace norm of a matrix is equal to the trace norm of its transpose. Under this definition, the optimization in Eq. (6) can be written as:

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y}} : \quad & \frac{1}{2}\|\mathcal{X} - \mathcal{Y}\|_F^2 \quad \equiv \quad \frac{1}{2n}\sum_{i=1}^{n} \|X_{(i)} - Y_{(i)}\|_F^2 \\ s.t. \quad & \frac{1}{n}\sum_{i=1}^{n} \|X_{(i)}\|_{tr} \leq c \\ & \mathcal{Y}_{\Omega} = \mathcal{T}_{\Omega}. \end{aligned} \tag{8}$$

3.3. Simplified Formulation

The problem in Eq. (8) is difficult to solve due to the interdependent trace norm constraints. To simplify the formulation, we introduce additional matrices $M_1, ..., M_n$ and obtain the following equivalent formulation:

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y},M_i} : \quad & \frac{1}{2n}\sum_{i=1}^{n} \|M_i - Y_{(i)}\|_F^2 \\ s.t. \quad & \frac{1}{n}\sum_{i=1}^{n} \|M_i\|_{tr} \leq c \\ & M_i = X_{(i)}, \quad \text{for} \quad i = 1, 2, ..., n \\ & \mathcal{Y}_{\Omega} = \mathcal{T}_{\Omega}, \end{aligned} \tag{9}$$

In this formulation, the trace norm constraints are still not independent because of the equality constraint $M_i = X_{(i)}$ which enforces all M_i's to be identical. Thus, we relax the equality constraints $M_i = X_{(i)}$ by $\|M_i - X_{(i)}\|_F^2 \leq d_i$ as Eq. (10), so that we can independently solve each subproblem later on.

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y},M_i} : \quad & \frac{1}{2n}\sum_{i=1}^{n} \|M_i - Y_{(i)}\|_F^2 \\ s.t. \quad & \frac{1}{n}\sum_{i=1}^{n} \|M_i\|_{tr} \leq c \\ & \|M_i - X_{(i)}\|_F^2 \leq d_i, \quad \text{for} \quad i = 1, 2, ..., n \\ & \mathcal{Y}_{\Omega} = \mathcal{T}_{\Omega}. \end{aligned} \tag{10}$$

The above formula can be converted to its dual formulation for certain values of α_i's and γ:

$$\begin{aligned} \min_{\mathcal{X},\mathcal{Y},M_i} : \quad & \frac{1}{2n}\sum_{i=1}^{n} \alpha_i \|M_i - X_{(i)}\|_F^2 \\ & + \frac{1}{2n}\sum_{i=1}^{n} \|M_i - Y_{(i)}\|_F^2 \\ & + \frac{\gamma}{n}\sum_{i=1}^{n} \|M_i\|_{tr} \\ s.t. \quad & \mathcal{Y}_{\Omega} = \mathcal{T}_{\Omega}. \end{aligned} \tag{11}$$

We can consider a more general form of Eq. (11) by allowing for different weights to control the rank in each mode of the tensor:

$$\begin{aligned}\min_{\mathcal{X},\mathcal{Y},M_i}: \quad & \frac{1}{2}\sum_{i=1}^{n}\alpha_i\|M_i - X_{(i)}\|_F^2 \\ & + \frac{1}{2}\sum_{i=1}^{n}\beta_i\|M_i - Y_{(i)}\|_F^2 \\ & + \sum_{i=1}^{n}\gamma_i\|M_i\|_{tr} \\ s.t. \quad & \mathcal{Y}_\Omega = \mathcal{T}_\Omega.\end{aligned} \tag{12}$$

This is a convex but nondifferentiable optimization problem. Note that the term $\sum_{i=1}^{n}\|M_i - X_{(i)}\|_F^2$ is bounded by $\sum_{i=1}^{n}\|M_i - Y_{(i)}\|_F^2$, because $\mathcal{X}$ is totally free, while $\mathcal{Y}$ is constrained by $\mathcal{Y}_\Omega = \mathcal{T}_\Omega$. Thus, an alternative algorithm with faster computation time can be built by removing the term $\sum_{i=1}^{n}\|M_i - X_{(i)}\|_F^2$, if we relax the restriction. In our experiments we will compare both versions and show that the removal of this term has only a minor effect on the algorithm. Next, we show how to solve the optimization problem in Eq. (12).

3.4. The Main Algorithm

We propose to employ block coordinate descent (BCD) for the optimization. The basic idea of block coordinate descent is to optimize a group of variables while fixing the other groups. We divide the variables into $n+2$ blocks: $\mathcal{X}, \mathcal{Y}, M_1, M_2, ..., M_n$. Computing the M_is is the major challenge of the algorithm and will draw from current trace norm optimization literature to tackle the problem.

Computing $\mathcal{X}$: The optimal $\mathcal{X}$ with all other variables fixed is given by solving the following subproblem:

$$\min_{\mathcal{X}}: \quad \frac{1}{2}\sum_{i=1}^{n}\alpha_i\|M_i - X_{(i)}\|_F^2 \tag{13}$$

It is easy to check that the solution to Eq. (13) is given by

$$\mathcal{X} = \frac{\sum_{i=1}^{n}\alpha_i \text{fold}_i(M_i)}{\sum_{i=1}^{n}\alpha_i}. \tag{14}$$

Computing $\mathcal{Y}$: The optimal $\mathcal{Y}$ with all other variables fixed is given by solving the following subproblem:

$$\begin{aligned}\min_{\mathcal{Y}}: \quad & \frac{1}{2}\sum_{i=1}^{n}\beta_i\|M_i - Y_{(i)}\|_F^2 \\ s.t. \quad & \mathcal{Y}_\Omega = \mathcal{T}_\Omega.\end{aligned} \tag{15}$$

Similar to the first subproblem, the optimal $\mathcal{Y}$ in Eq. (15) is given by

$$\mathcal{Y}_{\bar{\Omega}} = \left(\frac{\sum_{i=1}^{n}\beta_i \text{fold}_i(M_i)}{\sum_{i=1}^{n}\beta_i}\right)_{\bar{\Omega}}. \tag{16}$$

Computing M_i: The optimal M_i with all other variables fixed is given by solving the following subproblem:

$$\begin{aligned}\min_{M_i}: \quad & \frac{\alpha_i}{2}\|M_i - Y_{(i)}\|_F^2 + \frac{\beta_i}{2}\|M_i - X_{(i)}\|_F^2 \\ & + \gamma_i\|M_i\|_{tr}\end{aligned} \tag{17}$$

The objective in Eq. (17) is nondifferentiable. The derivation of the solution is fairly complex. Here we give the main result and refer the interested reader to the appendix for the details. We give the solution in a general form and then explain how that translates to the specific choice of variables in Eq. (17).

Theorem 3.1. *The global optimal solution to the following optimization problem:*

$$\min_{M}: \quad \sum_{i=1}^{l}\frac{\theta_i}{2}\|M - Z_i\|_F^2 + \gamma\|M\|_{tr} \tag{18}$$

is given by

$$M^* = D_\tau(Z) \tag{19}$$

where $\theta = \sum_{i=1}^{l}\theta_i$, $\tau = \frac{\gamma}{\sum_{i=1}^{l}\theta_i}$, $Z = \frac{1}{\theta}\sum_{i=1}^{l}\theta_i Z_i$ *and the "shrinkage" operator* $D_\tau(Z)$ *is defined as in Eq. (1).*

Based on Theorem 3.1, the solution to Eq. (17) is given by setting

$$\tau = \frac{\gamma_i}{\alpha_i + \beta_i}, Z_i = \frac{\alpha_i X_{(i)} + \beta_i Y_{(i)}}{\alpha_i + \beta_i} \tag{20}$$

We call the proposed algorithm "LRTC", which stands for Low Rank Tensor Completion. The pseudo-code of the LRTC algorithm is given in Algorithm 1 below. As convergence criteria we compare the difference of $\mathcal{Y}$ in subsequent iterations to a threshold. Since the objective in Eq. (12) is convex and the first two terms are differentiable and the third term is separable, BCD is guaranteed to find the global optimal solution [22].

4. Three Heuristic Algorithms

We introduce several heuristic algorithms, which, unlike the one in the last section, involve non-convex optimization problems. A goal of introducing the heuristic algorithms is to establish some basic methods that can be used for comparison.

Tucker: One natural approach to extend the SVD to the tensor case is to use the Higher Order SVD (HOSVD) [7, pages 96-100]. Similar to the matrix case, we solve the following optimization:

$$\begin{aligned}\min_{\mathcal{X},C,U_1,U_2,...,U_n}: \quad & \frac{1}{2}\|\mathcal{X} - C \times_1 U_1 \times_2 U_2 \times_3 ... \times_n U_n\|_F^2 \\ s.t. \quad & \mathcal{X}_\Omega = \mathcal{T}_\Omega \\ & \text{rank}(U_i) \le r_i \quad \text{for } i = 1, 2, ..., n\end{aligned} \tag{21}$$

Algorithm 1 LRTC: Low Rank Tensor Completion

Input: $\mathcal{T}_\Omega$

Output: $\mathcal{X}, \mathcal{Y}, M_i$, from 1 to n

Set $\mathcal{Y}_\Omega = \mathcal{T}_\Omega, \mathcal{Y}_{\bar{\Omega}} = 0, \mathcal{X} = \mathcal{Y}, M_i = \mathcal{Y}_{(i)}$;

while no convergence **do**

  for $i = 1$ to n **do**

$$M_i = D_{\frac{\gamma_i}{\alpha_i+\beta_i}}\left(\frac{\alpha_i X_{(i)} + \beta_i Y_{(i)}}{\alpha_i + \beta_i}\right)$$

  end for

$$\mathcal{X} = \frac{\sum_{i=1}^{n} \alpha_i \text{fold}_i(M_i)}{\sum_{i=1}^{n} \alpha_i}$$

$$\mathcal{Y}_{\bar{\Omega}} = \left(\frac{\sum_{i=1}^{n} \beta_i \text{fold}_i(M_i)}{\sum_{i=1}^{n} \beta_i}\right)_{\bar{\Omega}}$$

end while

where $C \times_1 U_1 \times_2 U_2 \times_3 ... \times_n U_n$ is the HOSVD and $\mathcal{T}, \mathcal{X}$ are n mode tensors. Note that HOSVD implements the tensor decomposition based on the well-known Tucker model.

Parafac: Another natural approach is to use the parallel factor analysis (Parafac) model [10], resulting in the following optimization problem:

$$\begin{aligned} \min_{\mathcal{X}, U_1, U_2, ..., U_n} : \quad & \frac{1}{2}\|\mathcal{X} - U_1 \circ U_2 \circ ... \circ U_n\|_F^2 \\ s.t. \quad & \mathcal{X}_\Omega = \mathcal{T}_\Omega \\ & \text{rank}(U_i) \leq r \end{aligned} \tag{22}$$

where $\circ$ denotes the outer product and $U_1 \circ U_2 \circ ... \circ U_n$ is the Parafac model based decomposition.

SVD: The third alternative is to consider the tensor as multiple matrices and force the unfolding matrix along each mode of the tensor to be low rank as follows:

$$\begin{aligned} \min_{\mathcal{X}, M_1, M_2, ..., M_n} : \quad & \frac{1}{2}\sum_{k=1}^{n} \|X_{(k)} - M_k\|_F^2 \\ s.t. \quad & \mathcal{X}_\Omega = \mathcal{T}_\Omega \\ & \text{rank}(M_i) \leq r_i \quad \text{for } i = 1, 2, ..., n. \end{aligned} \tag{23}$$

To compute the approximate solution, we apply SVD on each unfolded matrix and estimate the missing values based on the average value of SVD results, and we repeat the same procedure until convergence.

5. Results

In this section, we compare the algorithms on synthetic and real-world data and show several applications.

5.1. Performance Comparison

We compare the proposed LRTC algorithm based on the tensor trace norm minimization in Section 3 with the three heuristic algorithms in Section 4 on both synthetic and real-world data. The synthetic data $\mathcal{T}$ is of size $40 \times 40 \times 40 \times 40$. The ranks of the tensor along all four modes are set to $[20, 20, 20, 20]$. The brain MRI data is of size $181 \times 217 \times 181$.

We choose the percentage of sample elements as 3%, 20% and 80% respectively. For a fair comparison, all algorithms are evaluated using the same ranks given by the result of our algorithm using the setting $\alpha = 0$ The relative square error (RSE) comparison is presented in Table 1 and Table 2, where $RSE = \|\mathcal{Y} - \mathcal{T}\|_F / \|\mathcal{T}\|_F$.

Results from these two tables show that the proposed convex formulation outperforms the three heuristic algorithms. The performance of the three heuristic algorithms is poor for high rank problems. We can also observe that the proposed convex formulation is able to recover the missing information using a small number of samples. Next we evaluate our algorithm using different parameters. We observe that the larger the α value, the more the solution approximates the original problem in Eq. (4). While the alpha setting makes some difference, the error is very small overall for all setting. An example slice of the MRI data is shown in Fig. 2.

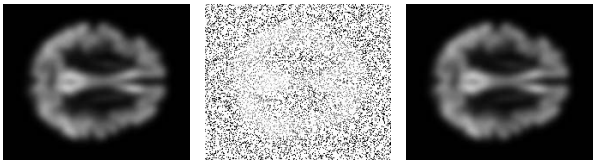

Figure 2: The left image (one slice of the MRI) is the original; we randomly select pixels for removal shown in white in the middle image; the right image is the result of the proposed completion algorithm.

5.2. Applications

In this section, we outline three potential application examples with three different types of data: Images, Videos, and reflectance data.

Images: Our algorithm can be used to estimate missing data in images and textures. For example, in Fig. 3 we show how missing pixels can be filled in a facade image. Note how our algorithm can propagate global structure even though a significant amount of information is missing.

Videos: The proposed algorithm may be used for video compression and video in-painting. The core idea of video compression is to remove individual pixels and to use tensor completion to recover the missing information. Similarly, a user can eliminate unwanted pixels in the data and use

Table 1: The RSE comparison on the synthetic data of size $40 \times 40 \times 40 \times 40$. P: Parafac model based heuristic algorithm; T: Tucker model heuristic algorithm; SVD: the heuristic algorithm based on the SVD; $\alpha 0$, $\alpha 10$ and $\alpha 50$ denote the proposed LRTC algorithm with three different values of the parameter: $\alpha = 0$, $\alpha = 10$ and $\alpha = 50$, respectively. The top, middle and bottom parts of the table respond to the sample percentage: 3%, 20% and 80%, respectively.

RSE Comparison (10^{-4})

Rank	T	P	SVD	$\alpha 0$	$\alpha 10$	$\alpha 50$
10,11,10,9	725	677	759	321	302	**257**
14,16,15,14	892	901	863	72.5	70.2	**65.3**
20,22,21,19	1665	1302	1474	50.1	46.9	**36.2**
24,25,25,26	2367	1987	2115	40.6	**38.7**	**38.7**
10,11,9,11	371	234	347	16.3	14.2	**12.7**
15,15,16,14	728	530	611	8.92	8.41	**8.23**
21,19,21,20	1093	982	895	**8.48**	8.56	**8.48**
24,25,26,26	1395	1202	1260	40.7	34.3	**13.7**
10,9,11,9	145	45	136	**3.08**	4.01	3.12
15,14,14,16	326	65	217	2.17	**2.05**	2.35
21,20,19,21	518	307	402	1.36	2.06	**1.27**
24,25,25,26	685	509	551	1.41	1.59	**1.04**

Table 2: The RSE comparison on the brain MRI data of size $181 \times 217 \times 181$. P: Parafac model based heuristic algorithm; T: Tucker model heuristic algorithm; SVD: the heuristic algorithm based on the SVD; $\alpha 0$, $\alpha 10$ and $\alpha 50$ denote the proposed LRTC algorithm with three different values of the parameter: $\alpha = 0$, $\alpha = 10$ and $\alpha = 50$, respectively. The top and bottom parts respond to the sample percentage: 20% and 80%, respectively.

RSE Comparison (10^{-4})

Rank	T	P	SVD	$\alpha 0$	$\alpha 10$	$\alpha 50$
21,24,23	311	234	274	210	193	**177**
38,41,37	1259	1001	1322	148	141	**121**
90,93,87	4982	3982	5025	61.0	53.7	**42.8**
21,24,23	12.3	8.64	**5.48**	29.4	12.8	13.5
35,42,36	179	153	99	**4.41**	5.32	5.69
39,48,41	279	345	199	**0.72**	1.05	1.26
45,55,47	606	523	513	1.22	1.35	**1.06**

the proposed algorithm to compute alternative values for the removed pixels. See Fig. 4 for an example frame of a video.

Reflectance data: The BRDF is the "Bidirectional Reflectance Distribution Function". The BRDF specifies the reflectance of a target as a function of illumination direction and viewing direction and can be interpreted as a 4 mode tensor. BRDFs of real materials can be acquired by complex appearance acquisition systems that typically require

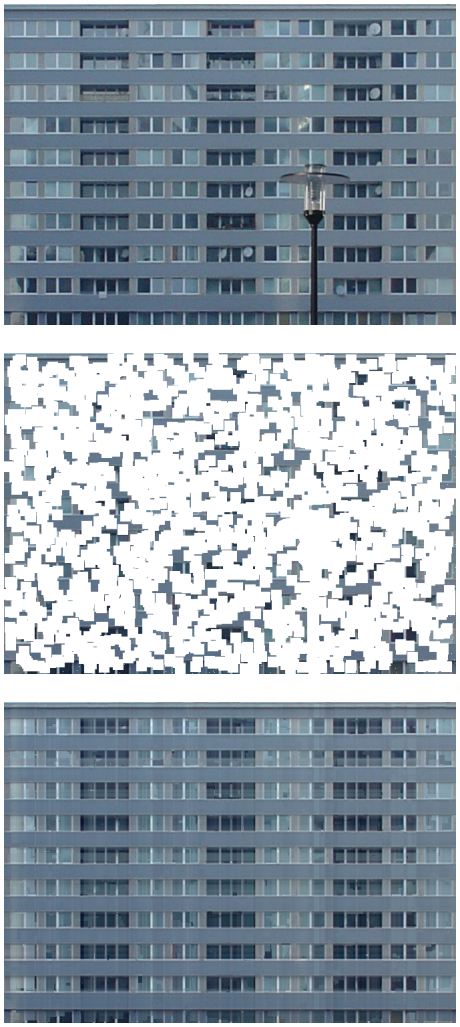

Figure 3: Facade in-painting. The top image is the original image; we select the lamp and satellite dishes together with a large set of randomly positioned squares as the missing parts shown in white in the middle image; the bottom image is the result of the proposed completion algorithm.

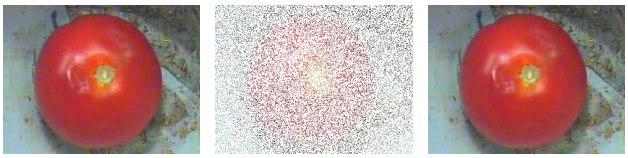

Figure 4: Video completion. The left image (one frame of the video) is the original; we randomly select pixels for removal shown in white in the middle image; the right image is the result of the proposed LTRC algorithm.

taking photographs of the same object under different lighting conditions. As part of the acquisition process, data can be missing or be unreliable, such as in the MIT BRDF data set. We use tensor completion to estimate the missing entries in reflectance data. See Fig. 5 for an example. More results are shown in the video accompanying this paper.

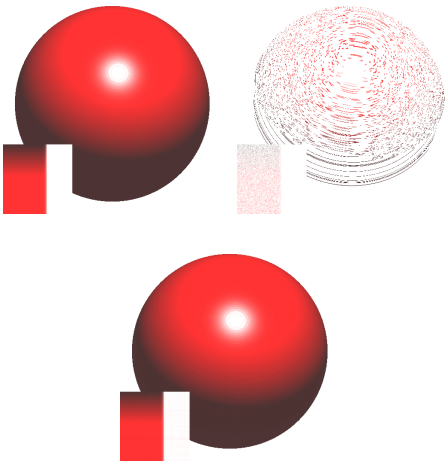

Figure 5: The top left image is a rendering of an original phong BRDF; we randomly select 90% of the pixels for removal shown in white in the top right image; the bottom image is the result of the proposed LRTC algorithm.

6. Conclusion

In this paper, we extend low rank matrix completion to low rank tensor completion. We propose theoretical foundations for tensor completion together with a strong convex optimization algorithm, called LRTC, to tackle the problem. Additionally, several heuristic algorithms are presented. We introduce an original definition of the trace norm for tensors in order to define a convex optimization for tensor completion. The experiments show that the proposed method is more stable and accurate in most cases, especially when the sample entries are very limited. Several application examples show the broad applicability of tensor completion in computer vision and graphics.

The proposed tensor completion algorithm assumes that the data is of low rank. This may not be the case in certain applications. We are currently investigating how to automatically estimate the values of the parameters in LRTC. We also plan to extend the theoretical results of Candés and Recht to the tensor case.

Acknowledgments

This work was supported by NSF IIS-0612069, IIS-0812551, CCF-0811790, and NGA HM1582-08-1-0016.

APPENDIX

The goal of the appendix is to derive the solution presented in Theorem 3.1. Before the actual proof, we first lay the foundation by introducing the subdifferential in Proposition 1 and Lemma 2.

Proposition 1. *Let $W = U\Sigma V^\top$ be the singular value decomposition (SVD) of $W \in \mathbb{R}^{m\times n}$, where $U \in \mathbb{R}^{m\times p}$, $V \in \mathbb{R}^{n\times p}$ and Σ is a full rank matrix, then the subdifferential of $\|W\|_{tr}$ is given by*

$$\partial\|W\|_{tr} = \{UV^\top + M \mid \|M\|_2 \leq 1, U^\top M = 0, MV = 0\}. \tag{24}$$

Lemma 2. *The subdifferential of the trace norm of $W \in \mathbb{R}^{m\times n}$ can also be written as*

$$\partial\|W\|_{tr} = \{UV^\top + U_\perp N V_\perp^\top \mid \|N\|_2 \leq 1\}, \tag{25}$$

where $U \in \mathbb{R}^{m\times p}, V \in \mathbb{R}^{n\times p}$ are defined as in Proposition 1 and $U_\perp \in \mathbb{R}^{m\times(m-p)}$ and $V_\perp \in \mathbb{R}^{n\times(n-p)}$ are orthogonal complements of U and V, respectively.

Proof. For convenience, we denote the set in Eq. (24) as S_1 and the set in Eq. (25) as S_2.

First, given any $g = UV^\top + M \in S_1$, there always exist an $N = U_\perp^\top M V_\perp$ such that $g = UV^\top + U_\perp N V_\perp^\top$ and $UV^\top + U_\perp N V_\perp^\top \in S_2$. Because $U_\perp N V_\perp^\top = U_\perp U_\perp^\top M V_\perp V_\perp^\top = M$, $g = UV_\top + U_\perp N V_\perp^\top$ holds. It follows from $\|M\|_2 \leq 1$ that $\|N\|_2 = \|U_\perp^\top M V_\perp\|_2 = \|M\|_2 \leq 1$. Thus, $g \in S_2$. Hence, $S_1 \subseteq S_2$.

Second, given any $g = UV^\top + N \in S_2$, let $M = U_\perp N V_\perp^\top$. We have $U^\top M = U^\top U_\perp N V_\perp^\top = 0$ and $MV = U_\perp N V_\perp^\top V = 0$. Similar to the proof above, $\|M\|_2 = \|U_\perp N V_\perp^\top\|_2 \leq 1$ holds. Thus, $g \in S_1$. We have $S_2 \subseteq S_1$.

□

Proof of Theorem 3.1

Proof. For convenience, we denote the objective in Eq. (18) as $f(M)$. As a convex nondifferentiable optimization problem, M^* is its optimal solution if and only if its subdifferential at M^* contains 0 [15, Thm. 3.1.15] . In other words, $0 \in \partial f(M^*)$. Any element $g \in \partial f(M)$ can be described as:

$$\begin{aligned} g =& \sum_{i=1}^{l} \theta_i (M - Z_i) + \partial\|M\|_{tr} \\ =& \sum_{i=1}^{l} \theta_i (M - Z_i) + \gamma(UV^T + U_\perp N V_\perp^\top) \\ =& \theta M - \theta Z + \gamma(UV^T + U_\perp N V_\perp^\top) \end{aligned} \tag{26}$$

where $N \in \mathbb{R}^{p\times p}, U \in \mathbb{R}^{m\times p}, V \in \mathbb{R}^{n\times p}$ are the factors of the SVD for M and $U_\perp \in \mathbb{R}^{m\times(m-p)}, V_\perp \in \mathbb{R}^{n\times(n-p)}$

are the orthogonal complements of U, V. For computing the optimal solution, we need to find an M and N such that $g = 0$.

$$\begin{aligned} \theta M - \theta Z + \gamma(UV^T + U_\perp N V_\perp^\top) &= 0 \\ M + \frac{\gamma}{\theta}(UV^T + U_\perp N V_\perp^\top) &= Z \end{aligned} \tag{27}$$

since

$$M = U\Sigma_M V^\top = \left[\begin{array}{cc} U & U_\perp \end{array}\right] \left[\begin{array}{cc} \Sigma_M & 0 \\ 0 & 0 \end{array}\right] \left[\begin{array}{c} V^\top \\ V_\perp^\top \end{array}\right] \tag{28}$$

and we define the SVD of Z as follows:

$$\begin{aligned} Z &= U_Z \Sigma_Z V_Z^\top \\ &= \left[\begin{array}{cc} \bar{U} & \bar{U}_\perp \end{array}\right] \left[\begin{array}{cc} \Sigma_{p\times p} & 0 \\ 0 & \Sigma_{(n-p)\times(m-p)} \end{array}\right] \left[\begin{array}{c} \bar{V}^\top \\ \bar{V}_\perp^\top \end{array}\right] \end{aligned} \tag{29}$$

Here, we let $\sigma(\Sigma_Z)$ be a vector consisting of the singular value in descending order. By incorporating Eq. (28) and Eq. (29) into Eq. (27), we obtain:

$$\begin{aligned} &\left[\begin{array}{cc} U & U_\perp \end{array}\right] \left[\begin{array}{cc} \Sigma_M & 0 \\ 0 & 0 \end{array}\right] \left[\begin{array}{c} V^\top \\ V_\perp^\top \end{array}\right] \\ &+\frac{\gamma}{\theta} \left[\begin{array}{cc} U & U_\perp \end{array}\right] \left[\begin{array}{cc} I_{p\times p} & 0 \\ 0 & N \end{array}\right] \left[\begin{array}{c} V^\top \\ V_\perp^\top \end{array}\right] \\ &= \left[\begin{array}{cc} \bar{U} & \bar{U}_\perp \end{array}\right] \left[\begin{array}{cc} \Sigma_{p\times p} & 0 \\ 0 & \Sigma_{(n-p)\times(m-p)} \end{array}\right] \left[\begin{array}{c} \bar{V}^\top \\ \bar{V}_\perp^\top \end{array}\right] \end{aligned}$$

It follows that

$$\begin{aligned} &\left[\begin{array}{cc} U & U_\perp \end{array}\right] \left[\begin{array}{cc} \frac{\gamma}{\theta} I_{p\times p} + \Sigma_M & 0 \\ 0 & \frac{\gamma}{\theta} N \end{array}\right] \left[\begin{array}{c} V^\top \\ V_\perp^\top \end{array}\right] \\ &= \left[\begin{array}{cc} \bar{U} & \bar{U}_\perp \end{array}\right] \left[\begin{array}{cc} \Sigma_{p\times p} & 0 \\ 0 & \Sigma_{(n-p)\times(m-p)} \end{array}\right] \left[\begin{array}{c} \bar{V}^\top \\ \bar{V}_\perp^\top \end{array}\right] \end{aligned} \tag{30}$$

Next, we set $U = \bar{U}, V = \bar{V}, U_\perp = \bar{U}_\perp, V_\perp = \bar{V}_\perp, \Sigma_M = \Sigma_{p\times p} - \frac{\gamma}{\theta} I_{p\times p}$ and $N = \frac{\gamma}{\theta}\Sigma_{(n-p)\times(m-p)}$, so that the Eq. (30) holds. To make sure $\|N\|_2 \leq 1$ and Σ_M is positive semidefinite, p is chosen as $p \in \{p \mid \sigma_p \geq \frac{\gamma}{\theta}, \sigma_{p+1} \leq \frac{\gamma}{\theta}$ or $p = \min(m,n)\}$. In summery, we get

$$M^* = U\Sigma_M V^\top = D_\tau(Z). \tag{31}$$

□

References

[1] Y. Amit, M. Fink, N. Srebro, and S. Ullman. Uncovering shared structures in multiclass classification. *ICML*, pages 17–24, 2007.

[2] A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. *NIPS*, 73:243–272, 2007.

[3] F. R. Bach. Consistency of trace norm minimization. *Journal of Machine Learning Research*, pages 1019–1048, 2008.

[4] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. *SIGGRAPH*, pages 414–424, 2000.

[5] J.-F. Cai, E. J. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *Submitted to Optimization and Control*, 2008.

[6] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. 2008.

[7] L. Elden. *Matrix Methods in Data Mining and Pattern Recgonition*. 2007.

[8] M. Fazel. Matrix rank minimization with applications. *PhD thesis, Stanford University*.

[9] M. Fazel, H. Hindi, and S. Boyd. A rank minimization heuristic with application to minimum order system approximation. *Proceedings of the American Control Conference*, pages 4734–4739, 2001.

[10] R. A. Harshman. Foundations of the parafac procedure: models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

[11] N. Komodakis and G. Tziritas. Image completion using global optimization. *CVPR*, pages 417–424, 2006.

[12] T. Korah and C. Rasmussen. Spatiotemporal inpainting for recovering texture maps of occluded building facades. *IEEE Transactions on Image Processing*, 16:2262–2271, 2007.

[13] M. Kurucz, A. A. Benczur, and K. Csalogany. Methods for large scale svd with missing values. *KDD*.

[14] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *To appear Mathematical Programming*.

[15] Y. Nesterov. Introductory lectures on convex programming. *Lecture Notes*, pages 119–120, 1998.

[16] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM*.

[17] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In *NIPS*, pages 1329–1336. MIT Press, 2005.

[18] J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones, 1998.

[19] K. C. Toh, M. J. Todd, and R. H. Tutuncu. Sdpt3: a matlab software package for semidefinite programming. *Optimization Methods and Software*, 11:545–581, 1999.

[20] M. Tomasi and T. Kanade. Shape and motion from image stream under orthography: a factorization method. *International Journal of Computer Vision*, 9:137–154, 1992.

[21] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17:520–525, 2001.

[22] P. Tseng. Convergence of block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory Application*, 109:475–494, 2001.