

Nonnegative Low-Rank Tensor Completion via Dual Formulation with Applications to Image and Video Completion

Tanmay Kumar Sinha*
IIIT, Hyderabad

tanmay.kumar@research.iiit.ac.in

Jayadev Naram*
IIIT, Hyderabad

jayadev.naram@research.iiit.ac.in

Pawan Kumar
IIIT, Hyderabad

pawan.kumar@iiit.ac.in

Abstract

Recent approaches to the tensor completion problem have often overlooked the nonnegative structure of the data. We consider the problem of learning a nonnegative low-rank tensor, and using duality theory, we propose a novel factorization of such tensors. The factorization decouples the nonnegative constraints from the low-rank constraints. The resulting problem is an optimization problem on manifolds, and we propose a variant of Riemannian conjugate gradients to solve it. We test the proposed algorithm across various tasks such as colour image inpainting, video completion, and hyperspectral image completion. Experimental results show that the proposed method outperforms many state-of-the-art tensor completion algorithms.

1. Introduction

Recent years have seen an increase in the quantity of multidimensional data available, such as colour images, video sequences, and 3D images. Flattening multidimensional data to matrices usually leads to loss of information as matrices cannot capture the inherent structures present in most multidimensional data. This has led to increased research on tensor-based techniques for handling such data.

The low-rank tensor completion problem aims to recover an original tensor from partial observations. A well-known [6] formulation for such problems is

$$\min_{\mathcal{W} \in \mathbb{R}^{n_1 \times \dots \times n_K}} C L(\mathcal{W}, \mathcal{Y}_\Omega) + R(\mathcal{W}), \quad (1)$$

where $\mathcal{Y}_\Omega \in \mathbb{R}^{n_1 \times \dots \times n_K}$ is a partially observed tensor for indices given in the set Ω , $L : \mathbb{R}^{n_1 \times \dots \times n_K} \rightarrow \mathbb{R}$ is a loss

function, $C > 0$ denotes the cost parameter, and R is a regularizer enforcing low-rank constraint.

In many applications of tensor reconstruction such as color image recovery, video completion, recommendation systems, and link prediction, the data is nonnegative. Problem (1) does not enforce this structural constraint, and as such, the recovered tensors might contain negative entries.

To incorporate these constraints, we consider the non-negative low-rank tensor learning problem of the form:

$$\begin{aligned} \min_{\mathcal{W} \in \mathbb{R}^{n_1 \times \dots \times n_K}} \quad & C \|\mathcal{W}_\Omega - \mathcal{Y}_\Omega\|^2 + R(\mathcal{W}) \\ \text{subject to} \quad & \mathcal{W} \geq 0, \end{aligned} \quad (2)$$

where $(\mathcal{W}_\Omega)_{i_1, \dots, i_K} = \mathcal{W}_{i_1, \dots, i_K}$ if $(i_1, \dots, i_K) \in \Omega$. We convert the problem (2) into a minimax problem by constructing a partial dual similar to [15]. This leads to a factorization of the tensor \mathcal{W} in a form with separate factors for the nonnegative and low-rank constraints. The minimax problem has a rich geometric structure. We employ a Riemannian conjugate gradient algorithm to exploit this structure and develop an efficient solution.

The main contributions of the paper are listed below.

- We propose a novel factorization for modeling nonnegative low-rank tensors.
- We develop an algorithm exploiting the inherent geometric structure of this factorization.
- Experiments carried out on several real-world datasets show that the proposed algorithm outperforms state-of-the-art tensor completion algorithms.

The rest of the paper is organized as follows. In Section 2, we introduce the notation used in the paper. In Section 3, we review previous work related to the tensor completion problem. In Sections 4 and 5, we develop the dual framework

*Equal contribution

and present our algorithm. Section 6 details experiments carried out to compare our algorithm with several state-of-the-art algorithms. In Section 7, we end with concluding remarks.

2. Notation

For a full treatment of tensors, we refer to [2]. Here, we outline the basic notation we use for tensors. We denote tensors by calligraphic capital letters and matrices by capital letters. For a matrix $X \in \mathbb{R}^{m \times n}$, the nuclear norm of X , denoted by $\|X\|_*$, is the l_1 -norm of the singular values of X . The inner product of two same-sized tensors $\mathcal{W}, \mathcal{U} \in \mathbb{R}^{n_1 \times \dots \times n_K}$ is the sum of the products of their entries

$$\langle \mathcal{W}, \mathcal{U} \rangle = \sum_{i_1}^{n_1} \sum_{i_2}^{n_2} \dots \sum_{i_K}^{n_K} \mathcal{W}_{i_1, \dots, i_K} \mathcal{U}_{i_1, \dots, i_K}.$$

A mode- k fiber of a tensor $\mathcal{W} \in \mathbb{R}^{n_1 \times \dots \times n_K}$, denoted by $\mathcal{W}_{i_1, \dots, i_{k-1}, :, i_{k+1}, \dots, i_K}$, is a vector obtained by fixing all but k -th index of \mathcal{W} . The mode- k unfolding of a tensor $\mathcal{W} \in \mathbb{R}^{n_1 \times \dots \times n_K}$ is a matrix $W_k \in \mathbb{R}^{n_k \times n_1 \dots n_{k-1} n_{k+1} \dots n_K}$ formed by arranging the mode- k fibers to be the columns of the resulting matrix, i.e.,

$$W_k = [\mathcal{W}_{i_1, \dots, i_{k-1}, :, i_{k+1}, \dots, i_K}] \forall i_j, j \neq k.$$

The reverse of unfolding operation is called the folding operation which converts a given matrix to a tensor of specific order. We also represent the mode- k unfolding by the map $unfold_k : \mathbb{R}^{n_1 \times \dots \times n_K} \rightarrow \mathbb{R}^{n_k \times n_1 \dots n_{k-1} n_{k+1} \dots n_K}$ such that $unfold_k(\mathcal{W}) = W_k$, and the mode- k folding by the map $fold_k : \mathbb{R}^{n_k \times n_1 \dots n_{k-1} n_{k+1} \dots n_K} \rightarrow \mathbb{R}^{n_1 \times \dots \times n_K}$. The k -mode product of a tensor $\mathcal{W} \in \mathbb{R}^{n_1 \times \dots \times n_K}$ with a matrix $X \in \mathbb{R}^{m \times n_k}$ is denoted by $\mathcal{W} \times_k X \in \mathbb{R}^{n_1 \times \dots \times n_{k-1} \times m \times n_{k+1} \times \dots \times n_K}$, defined element-wise as follows:

$$(\mathcal{W} \times_k X)_{i_1, \dots, i_{k-1}, j, i_{k+1}, \dots, i_K} = \sum_{i_k}^{n_k} \mathcal{W}_{i_1, \dots, i_K} X_{j, i_k}.$$

Then we have

$$U = \mathcal{W} \times_k X \iff U_k = X W_k.$$

3. Previous Work

Tensor completion for visual data recovery was introduced in [8], building on the framework for low-rank matrix completion using matrix trace norm regularizer. The trace norm for tensors can be defined in several ways, and as such there exist multiple formulations for trace norm regularized tensor completion. [8], [13], and [12] use the regularizer $R(\mathcal{W}) = \sum_{k=1}^K \|W_k\|_*$, known as the overlapped trace norm, which promotes a lower Tucker (multilinear) rank in the recovered tensors.

Another popular trace norm regularizer is the latent trace norm regularizer. Methods that use this formulation, model the tensor as a sum of K individual tensors, and the latent trace norm amounts to an l_1 norm regularizer that promotes sparsity. A few examples of such methods are [16], which uses a Frank-Wolfe algorithm for optimization, and [14], which uses a scaled variant of the latent trace norm.

[6] uses a formulation that models the recovered tensor as a sum of non-sparse tensors and proposes a regularizer that uses l_2 norm regularizer as opposed to an l_1 norm. This allows for the development of a dual framework for tensor completion, which is solved using methods from Riemannian optimization.

Another class of tensor completion algorithms attempt to exploit the smoothness properties present in real-world tensor data like hyperspectral images and 3D images. [18] integrates the smooth PARAFAC decompositions for partially observed tensors and develops two variants using the total variation and quadratic variation. [17] adopts total variation(TV) regularizer to formulate the model, and [19] uses smooth matrix factorizations to incorporate tensor smoothness constraints.

Tensor decomposition methods form another class of algorithms. Tensor decompositions like Tucker and CP decompositions act as generalizations of the familiar notion of singular value decomposition of matrices. [9] and [10] exploit the Riemannian geometry of the set of fixed multilinear rank tensors to efficiently learn the Tucker decomposition. [28] employs a Bayesian probabilistic CP decomposition model to recover the incomplete tensors.

Other methods include [22], which uses another form of tensor singular value decomposition to define a tensor rank known as the tubal rank. [20] enforces the low-rank by factorizing the unfoldings of the tensor as low-rank matrices.

In [11] an algorithm is proposed that uses a block coordinate descent method for nonnegative tensor completion, utilizing the CP decomposition. [23] performs nonnegative tensor completion based on low-rank Tucker decomposition. Most of the research considering nonnegative tensors is devoted to learning nonnegative tensor decompositions. A few examples are [24], [25], [27], [26], etc. These methods cannot perform tensor completion task on incomplete data.

4. Dual Framework

Problem (2) models the nonnegative tensor completion using a regularizer that promotes low-rank solutions. We seek to learn \mathcal{W} as the sum $\sum \mathcal{W}^{(k)}$ of K tensors, as detailed in [6]. For our formulation, we use the regularizer

$$R(\mathcal{W}) = \sum_{k=1}^K \frac{1}{\lambda_k} \|\mathcal{W}^{(k)}\|_*^2,$$

Following [6], we develop a dual formulation for problem (2), incorporating the structural constraint of nonnegativity into the formulation. We do this following a similar approach developed in [15] for nonnegative matrix completion.

A key lemma [7] used in the development of the formulation is given below.

Lemma 1. For a matrix $X \in \mathbb{R}^{d \times T}$, the nuclear norm of X satisfies the following relation:

$$\|X\|_*^2 = \min_{\substack{\Theta \in \mathcal{P}^d \\ \text{range}(X) \subseteq \text{range}(\Theta)}} \langle \Theta^\dagger X, X \rangle$$

where $\mathcal{P}^d = \{S \in \mathbb{R}^{d \times d} : S \succeq 0, \text{tr}(S) = 1\}$, $\text{range}(\Theta) = \{\Theta z : z \in \mathbb{R}^d\}$, Θ^\dagger denotes the pseudo-inverse of Θ . For a given X , the optimal Θ is $\bar{\Theta} = \sqrt{XX^T} / \text{tr}(\sqrt{XX^T})$.

Using the above lemma, we can write (2) as

$$\begin{aligned} \min_{\substack{\Theta_k \in \mathcal{P}^{n_k}, \mathcal{W}^{(k)} \\ k \in \{1, \dots, K\}}} C \|\mathcal{W}_\Omega - \mathcal{Y}_\Omega\|^2 + \sum_{k=1}^K \frac{1}{2\lambda_k} \langle \Theta_k^\dagger W_k^{(k)}, W_k^{(k)} \rangle \\ \text{subject to} \quad \mathcal{W} \geq 0. \end{aligned} \quad (3)$$

The following theorem provides the dual framework for the nonnegative low-rank tensor completion problem. It is a direct generalization of Theorem 1 in [6] to the case with nonnegative constraints.

Theorem 2. An equivalent partial dual formulation of the problem (3) is

$$\begin{aligned} \min_{\substack{\Theta_k \in \mathcal{P}^{n_k}, \\ k \in \{1, \dots, K\}}} \max_{\substack{\mathcal{Z} \in \mathcal{C} \\ \mathcal{S} \in \mathbb{R}_+^{n_1 \times \dots \times n_K}}} \langle \mathcal{Z}, \mathcal{Y}_\Omega \rangle - \frac{1}{4C} \|\mathcal{Z}\|^2 \\ - \sum_{k=1}^K \frac{\lambda_k}{2} \langle (Z_k + S_k), \Theta_k(Z_k + S_k) \rangle, \end{aligned} \quad (4)$$

where $\mathcal{C} = \{\mathcal{Z} \in \mathbb{R}^{n_1 \times \dots \times n_K} : \mathcal{Z} = \mathcal{Z}_\Omega\}$. \mathcal{Z} is the dual tensor variable corresponding to the primal problem (3), \mathcal{S} is the dual tensor variable corresponding to the nonnegative constraints.

Proof. Consider the inner problem of (3) over $\mathcal{W}^{(k)}$. We introduce auxiliary variables U_k with the associated constraints $U_k = W_k^{(k)}$. The Lagrangian of this problem will be

$$\begin{aligned} \mathcal{L}(\mathcal{W}^{(1)}, \dots, \mathcal{W}^{(K)}, U_1, \dots, U_K, \Lambda_1, \dots, \Lambda_K, \mathcal{S}) = \\ C \left\| \left(\sum_{k=1}^K \mathcal{W}^{(k)} \right)_\Omega - \mathcal{Y}_\Omega \right\|^2 + \sum_{k=1}^K \frac{1}{2\lambda_k} \langle \Theta_k^\dagger U_k, U_k \rangle \\ + \sum_{k=1}^K \langle \Lambda_k, W_k^{(k)} - U_k \rangle - \langle \mathcal{S}, \mathcal{W} \rangle \end{aligned} \quad (5)$$

The dual function of the above will be given by

$$\mathcal{Q}(\Theta_1, \dots, \Theta_K, \Lambda_1, \dots, \Lambda_K, \mathcal{S}) = \min_{\substack{U_k, \mathcal{W}^{(k)} \\ k \in \{1, \dots, K\}}} \mathcal{L} \quad (6)$$

Applying the first-order KKT conditions, we get the following equations:

$$\text{fold}_k(\Lambda_k) = \mathcal{Z} + \mathcal{S}, \quad (7a)$$

$$U_k = \lambda_k \Theta_k \Lambda_k. \quad (7b)$$

where $\mathcal{Z} / (2C) = \mathcal{Y}_\Omega - \left(\sum_{k=1}^K \mathcal{W}^{(k)} \right)_\Omega$. It can be seen from the definition of \mathcal{Z} that $\mathcal{Z} = \mathcal{Z}_\Omega$.

Using (7a) and (7b), we compute each term of (5) to be

$$\begin{aligned} C \left\| \left(\sum_{k=1}^K \mathcal{W}^{(k)} \right)_\Omega - \mathcal{Y}_\Omega \right\|^2 = C \left(\frac{\|\mathcal{Z}\|^2}{4C^2} \right) = \frac{1}{4C} \|\mathcal{Z}\|^2, \\ \sum_{k=1}^K \frac{1}{2\lambda_k} \langle \Theta_k^\dagger U_k, U_k \rangle - \langle \Lambda_k, U_k \rangle \\ = - \sum_{k=1}^K \frac{\lambda_k}{2} \langle (Z_k + S_k), \Theta_k(Z_k + S_k) \rangle, \end{aligned}$$

$$\sum_{k=1}^K \langle \Lambda_k, W_k^{(k)} \rangle - \langle \mathcal{S}, \mathcal{W} \rangle = \langle \mathcal{Z}, \mathcal{Y}_\Omega \rangle - \frac{1}{2C} \|\mathcal{Z}\|^2.$$

Summing the terms, we obtain the expression for the dual function as

$$\mathcal{Q} = \langle \mathcal{Z}, \mathcal{Y}_\Omega \rangle - \frac{\|\mathcal{Z}\|^2}{4C} - \sum_{k=1}^K \frac{\lambda_k}{2} \langle (Z_k + S_k), \Theta_k(Z_k + S_k) \rangle.$$

This gives the minimax problem (4). From (7a) and (7b), we can deduce the relation between optimal points of primal and minimax problems. \square

If $\{\bar{\Theta}_1, \dots, \bar{\Theta}_K, \bar{\mathcal{Z}}, \bar{\mathcal{S}}\}$ is the optimal solution of (4), then the reconstructed tensor is given by $\bar{\mathcal{W}} = \sum_{k=1}^K \bar{\mathcal{W}}^{(k)}$ where $\bar{\mathcal{W}}^{(k)} = \lambda_k (\bar{\mathcal{Z}} + \bar{\mathcal{S}}) \times_k \bar{\Theta}_k$ for all k . This factorization gives us a decoupling of the low-rank and nonnegative constraints enforced on \mathcal{W} in (3) - the low-rank constraint is enforced by Θ_k , the nonnegative constraints are encoded in \mathcal{S} , and \mathcal{Z} corresponds to the dual variables of the primal problem.

5. Proposed Algorithm

Since $\Theta_k \in \mathcal{P}^{n_k}$ we can enforce the rank constraint explicitly by factorizing Θ_k as $\Theta_k = U_k U_k^T$, $U_k \in \mathcal{S}_{r_k}^{n_k}$,

where $\mathcal{S}_r^n = \{U \in \mathbb{R}^{n \times r} : \|U\|_F = 1\}$. We rewrite (4) as

$$\min_{U \in \mathcal{S}_{r_1}^{n_1} \times \dots \times \mathcal{S}_{r_K}^{n_K}} g(U), \quad (8)$$

where $U = (U_1, \dots, U_K)$, and $g(U)$ is the optimal value of the problem

$$g(U) = \max_{\substack{\mathcal{Z} \in \mathcal{C} \\ \mathcal{S} \in \mathbb{R}_+^{n_1 \times \dots \times n_K}}} \langle \mathcal{Z}, \mathcal{Y}_\Omega \rangle - \frac{\|\mathcal{Z}\|^2}{4C} - \sum_{k=1}^K \frac{\lambda_k}{2} \|U_k^T (Z_k + S_k)\|^2. \quad (9)$$

5.1. Convex Optimization Problem

The optimization problem (9) is a convex optimization problem over the variables \mathcal{Z} and \mathcal{S} , for a given U , hence it has a unique solution.

The problem (9) is solved separately for \mathcal{Z} and \mathcal{S} using an alternating minimization method. Equating the gradient of objective with respect to \mathcal{Z} to zero, we get

$$\frac{\mathcal{Z}_\Omega}{2C} + \sum_{k=1}^K \lambda_k (\mathcal{Z} \times_k U_k U_k^T)_\Omega = \mathcal{Y}_\Omega - \sum_{k=1}^K \lambda_k (\mathcal{S} \times_k U_k U_k^T)_\Omega. \quad (10)$$

This is a sparse linear system in \mathcal{Z} , which can be solved using linear conjugate gradient method.

Problem (9) has only one term involving \mathcal{S} . Hence, the optimization problem over \mathcal{S} reduces to

$$\min_{\mathcal{S} \in \mathbb{R}_+^{n_1 \times \dots \times n_K}} \sum_{k=1}^K \frac{\lambda_k}{2} \|U_k^T Z_k + U_k^T S_k\|^2. \quad (11)$$

This is a nonnegative least squares (NNLS) problem. We use the method detailed in [21], modified to suit our objective.

5.2. Riemannian Optimization Problem

Given the optimizer $(\hat{\mathcal{Z}}, \hat{\mathcal{S}})$ of (9), we compute g at a point U as

$$g(U) = \langle \hat{\mathcal{Z}}, \mathcal{Y}_\Omega \rangle - \frac{\|\hat{\mathcal{Z}}\|^2}{4C} - \sum_{k=1}^K \frac{\lambda_k}{2} \|U_k^T (\hat{Z}_k + \hat{S}_k)\|^2. \quad (12)$$

The set \mathcal{S}_r^n is a Riemannian manifold, known as the spectrahedron manifold. The constraint set $\mathcal{S}_{r_1}^{n_1} \times \dots \times \mathcal{S}_{r_K}^{n_K}$, therefore forms a product manifold and problem (8) is an optimization problem on a manifold.

To develop optimization algorithms on manifolds, we need a few geometric tools. We delegate development of the specific tools to [5] and [6]. For an introduction to optimization on general manifolds, we refer [4] and [1].

For our case, the Euclidean gradient for g can be computed as

$$\nabla g(U) = -(\lambda_1 A_1, \dots, \lambda_K A_K),$$

where $A_k = (\hat{Z}_k + \hat{S}_k)(\hat{Z}_k + \hat{S}_k)^T U_k$, for $1 \leq k \leq K$. We use a generalization of non-linear conjugate gradient algorithm to Riemannian manifolds [29] to solve problem (8).

The proposed algorithm is detailed in Algorithm 1. The reconstructed tensor is given by

$$\hat{\mathcal{W}} = \sum_{k=1}^K \lambda_k (\hat{\mathcal{Z}} + \hat{\mathcal{S}}) \times_k (U_k U_k^T).$$

Algorithm 1 Proposed Algorithm for Nonnegative Tensor Completion

Require: \mathcal{Y}_Ω , $\text{rank}=(r_1, \dots, r_K)$, τ , $(\lambda_1, \dots, \lambda_K)$ ▷ Input parameters
1: **for** $t = 1, 2, \dots$ **do**
2: Check Termination: if $\|\nabla g(U^{(t)})\| \leq \tau$ then break
3: Compute $\hat{\mathcal{Z}}^{(t)}$ in (10) using conjugate gradient algorithm
4: Compute $\hat{\mathcal{S}}^{(t)}$ in (11) using NNLS solver
5: Compute cost $g(U^{(t)})$ and gradient $\nabla g(U^{(t)})$
6: Update U : $U^{(t+1)} = \text{RiemannianCG-update}(U^{(t)})$
7: **end for**
8: **Output:** $\hat{\mathcal{W}} = \sum_{k=1}^K \lambda_k (\hat{\mathcal{Z}} + \hat{\mathcal{S}}) \times_k (U_k U_k^T)$

5.3. Complexity

- Step 3 (Computing $\hat{\mathcal{Z}}$): We use the linear conjugate gradient algorithm to solve the linear system (10). The major cost in this step is to compute the matrix products $U_k^T Z_k$ and $U_k^T S_k$, for $k \in \{1, \dots, K\}$. We can exploit the sparse structure of the problem to compute the products in $O(|\Omega| r_k)$ steps, and hence, if the linear solver takes T_{cg} iterations, the total cost of this step is $O\left(\sum_{k=1}^K T_{cg} |\Omega| r_k\right)$.
- Step 4 (Computing $\hat{\mathcal{S}}$): For each iteration of the NNLS algorithm, we need to compute the cost function in (11) and its gradient with respect to \mathcal{S} . Both of these operations can be computed in a similar manner as done for \mathcal{Z} , and the total cost of this step is $O\left(\sum_{k=1}^K T_{nnls} |\Omega| r_k\right)$, where T_{nnls} is the number of iterations of NNLS algorithm.
- Step 5 (Computing cost and gradient): We can compute $g(U)$ from (12) given $\hat{\mathcal{Z}}$ and $\hat{\mathcal{S}}$ computed in previous steps. This can be done in $O(K|\Omega|)$. The gradient requires computing the matrix products

Type	Dataset	Dimensions
Hyperspectral	Ribeira	$203 \times 268 \times 33$
Hyperspectral	Braga	$203 \times 268 \times 33$
Hyperspectral	Ruivaes	$203 \times 268 \times 33$
Video	Tomato	$320 \times 242 \times 167$
Video	Container	$144 \times 176 \times 150$
Video	Hall	$144 \times 176 \times 150$
Video	Highway	$144 \times 176 \times 150$
Color Image	Baboon	$256 \times 256 \times 3$
Color Image	Splash	$512 \times 512 \times 3$

Table 1: Description of datasets.

$(Z_k + S_k)(Z_k + S_k)^T U_k$, and we can do this in $O(|\Omega|r_k)$. Hence, total cost for computing the gradient is $O(\sum_{k=1}^K |\Omega|r_k)$.

- Step 6 (Riemannian Conjugate Gradient): Search direction and step length are computed in this step. Then the current solution $U^{(t)}$ is updated to $U^{(t+1)}$ by performing retraction at the $U^{(t)}$ along the search direction. This step ensures that the update remains on the product manifold. These operations can be done in $O(\sum_{k=1}^K n_k r_k^2 + \sum_{k=1}^K r_k^3)$.

Therefore, the overall per-iteration complexity of the proposed algorithm is

$$O\left((T_{cg} + T_{nnls})|\Omega| \sum_{k=1}^K r_k + \sum_{k=1}^K n_k r_k^2 + \sum_{k=1}^K r_k^3\right).$$

We store all the tensors in the sparse format, and perform operations accordingly. Hence, the overall space complexity of the proposed algorithm is

$$O\left(|\Omega| + \sum_{k=1}^K n_k r_k\right).$$

6. Numerical Experiments

6.1. Experimental setup

We have performed experiments on several publicly available datasets (see Table 1). We compare the performance of our algorithm to other state-of-the-art tensor completion algorithms. The baseline algorithms used for comparison are given below. Note that, with the exception of NCPC, all the other baseline algorithms do not enforce non-negativity in the completed tensors.

1. Dual [6]: A dual framework for low-rank tensor completion using a variant of the latent trace norm regularizer.

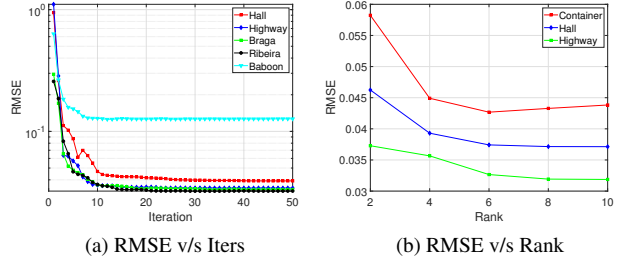


Figure 1: Variation of RMSE with iterations and rank. In the iterations plot, RMSE is in log scale. In the rank plot, the rank is taken to be value of X-label times $[1, 1, 1]$.

2. RPrecon [10]: A low-rank tensor completion algorithm with a multi-linear rank constraint using Riemannian preconditioning.
3. geomCG [9]: An algorithm for tensor completion using optimization on the manifold of fixed multi-linear rank tensors.
4. NCPC [11]: A nonnegative tensor completion method using the CP decomposition.
5. TMac [20]: An alternating minimization algorithm that uses parallel matrix factorization.
6. LRTC-TV [17]: An ADMM based algorithm that uses total variation regularization to enforce smoothness.
7. SMF-LRTC [19]: An algorithm that enforces smoothness constraint on factor matrices.
8. FFW [16]: An algorithm with scaled latent nuclear norm using the Frank-Wolfe algorithm.

We randomly sample 10% of the tensor entries and use it as training data. The metric we use for evaluation is the RMSE between the reconstructed and original tensors

$$\text{RMSE} = \sqrt{\frac{\|\mathcal{W} - \mathcal{W}_{true}\|_F^2}{n_1 n_2 n_3}}.$$

The proposed method is implemented based on Dual code. It uses MANOPT library [3] for implementing outer problem (8) on manifolds. For the nonnegative least squares problem in (9), we use the NNLS code [21] modified to work with our objective.

6.2. Hyperparameters

The ranks (r_1, r_2, r_3) are chosen as $(10, 10, 5)$ for all datasets, except color images where we chose $(10, 10, 3)$ since the dimension in mode-3 is less than 5. We chose the

Dataset	Prop	Dual	RPrecon	geomCG	NCPC	TMac	LRTC-TV	SMF-LRTC	FFW
Ribeira	0.03090	<u>0.03093</u>	0.0454	0.06593	0.16465	0.1644	0.04984	0.04159	0.0696
Braga	0.02817	<u>0.03054</u>	0.0939	0.07348	0.07348	0.20226	0.20227	0.06445	0.06691
Ruivaes	0.029211	0.04969	0.0352	0.07146	0.14059	0.14073	<u>0.02955</u>	0.040223	0.05437
Tomato	0.04282	0.04286	0.0589	0.05895	0.44175	0.44203	0.04638	0.053002	0.11463
Container	0.044908	<u>0.04693</u>	0.0645	0.06452	0.5433	0.54413	0.09773	0.05894	0.15116
Hall	0.03696	<u>0.03702</u>	0.0687	0.06879	0.53629	0.53717	0.09409	0.06327	0.06327
Highway	0.03255	<u>0.03652</u>	0.0405	0.04055	0.6416	0.64261	0.04171	0.03723	0.11204
Baboon	<u>0.11943</u>	<u>0.33258</u>	0.1563	2.7629	0.53315	0.5210	0.08729	0.13456	0.14222
Splash	<u>0.06371</u>	0.33475	0.3291	1.6897	0.50346	0.50014	0.05262	0.09331	0.07779

Table 2: RMSE of various methods. The best result among all methods is in bold and second best are underlined.

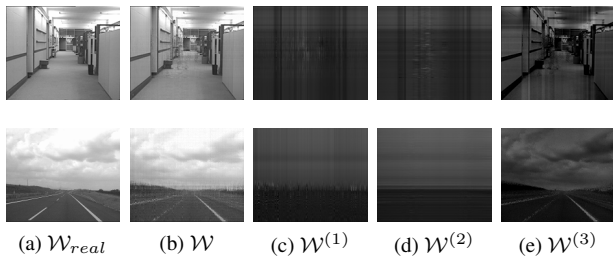


Figure 2: Original frame, reconstructed frame and components of reconstructed frame from Hall and Highway videos.

regularization constants λ_k 's according to [6]. The maximum iterations for outer optimization problem (8) was set to 200 as no improvement in RMSE is seen after 100 iterations on most of the datasets. For the baseline algorithms using rank as a hyperparameter, we have chosen the same rank as in our case since it is sufficient for a variety of datasets (see [6]). Additional hyperparameters for each baseline were set as indicated in the code provided by the authors.

We consider the effect of RMSE on the variation in hyperparameters. Fig. 1(a) shows the variation of RMSE over iterations of the proposed algorithm. We see that the RMSE decreases as the algorithm proceeds, and the decrease is rapid in the initial iterations. The RMSE decreases monotonically, so we have chosen 200 iterations as the threshold to guarantee good solutions. Fig. 1(b) shows the variation of RMSE with the rank. Increasing rank decreases the RMSE, but it quickly saturates to 10, which can be due to the inherent rank of the dataset. This justifies our choice of hyperparameters.

6.3. Image Completion

The task of image completion is to reconstruct the original image tensor, given only partial observations. As mentioned earlier, we have randomly sampled 10% of observations for training. We have experimented with sev-

eral hyperspectral images (see Table 1, [30]) where each data tensor contains stack of images measured at different wavelengths. Following [6] we resized these datasets to $203 \times 268 \times 33$ using bilinear interpolation. We have also considered two color images (see Table 1, [31]) which are naturally represented as third-order tensors.

We report the RMSE in Table 2 and some of the reconstructed images in Fig. 3. Our proposed algorithm outperforms the baseline algorithms in all hyperspectral datasets considered. The reconstructed images are of good quality, given only 10% of data for training. In color image datasets, LRTC-TV performs best. We expect that this is because the original images have the local smoothness property, which is exploited by LRTC-TV through the smoothness constraints it enforces. However, the low-rank and nonnegative structure does not preserve such smoothness, which explains the performance of other algorithms. Nevertheless, the proposed algorithm achieves the best RMSE next to LRTC-TV. We believe this indicates the usefulness of nonnegative constraints.

For hyperspectral images, LRTC-TV performs badly. On Braga, RMSE of LRTC-TV is 10 times that of the proposed algorithm. By comparing the reconstructed images, it can be seen that the smooth image produced by LRTC-TV is an imperfect reconstruction, suggesting the lack of local smoothness property in this dataset.

The effect of nonnegativity is more pronounced in color images, where the proposed algorithm achieves 3 times lower RMSE on Baboon and 5 times lower RMSE on Splash compared to Dual. We see this effect in the reconstructed image of Baboon and Splash, where, perhaps due to negative entries, the reconstructed images appear darker.

6.4. Video Completion

Video completion task is the reconstruction of the frames of the video from the partial observations given. We considered several gray-scale videos (see Table 1, [32]) which form third-order tensors.

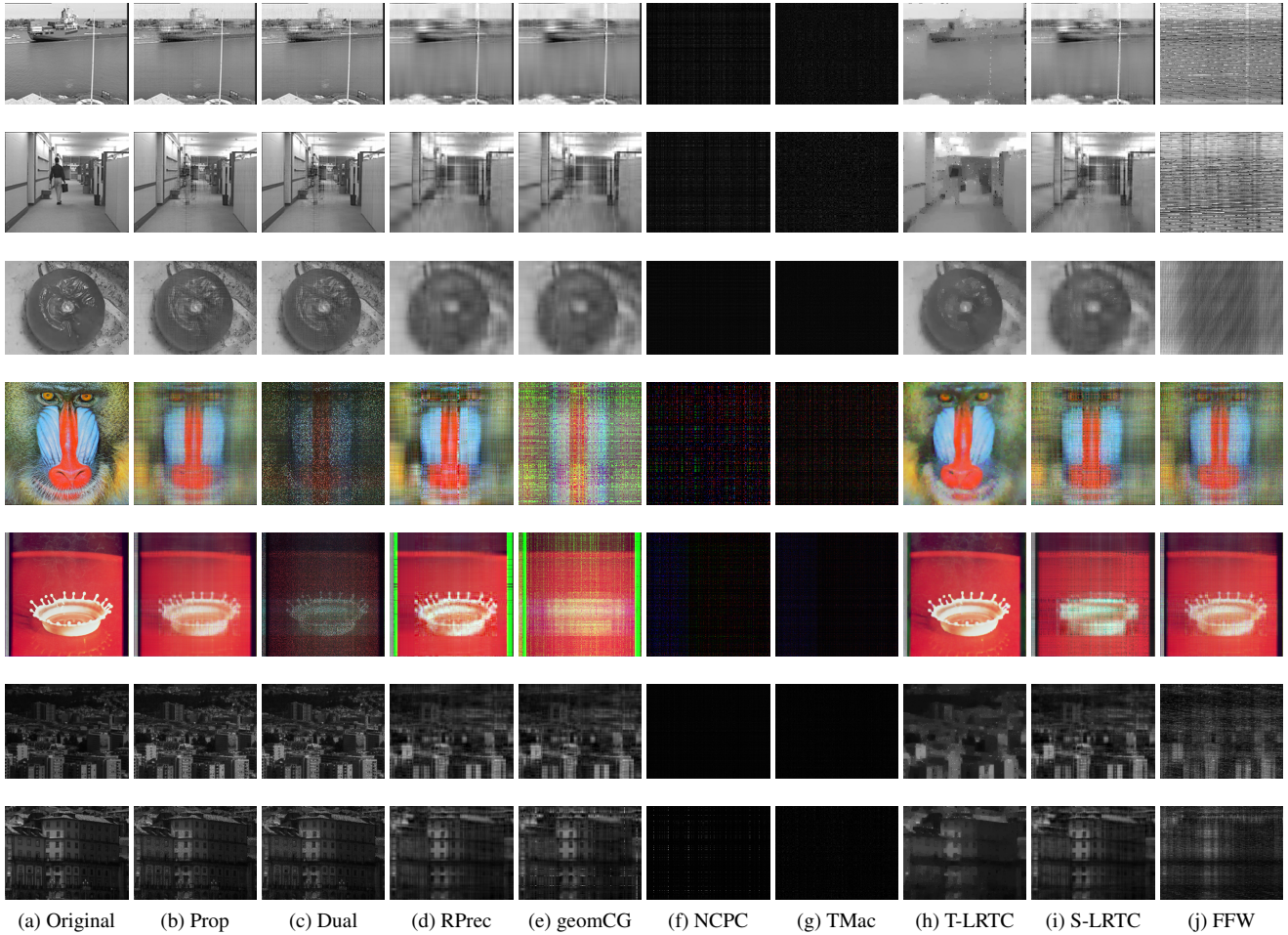


Figure 3: Original and Reconstructed Images for Different Algorithms given 10% of the fraction as training data. The datasets shown from top to bottom are Container, Hall, Tomato, Baboon, Splash, Braga, Ribeira respectively. For videos, a random frame was chosen. From left to right: Original, Proposed, Dual [6], RPrecon [10], geomCG [9], NCPC [11], TMac [20], LRTC-TV(T-LRTC) [17], SMF-LRTC(S-LRTC) [19] and FFW-LRTC [16].

Fig. 2 shows the component frames, $\mathcal{W}^{(k)}$'s, of the reconstructed frames, \mathcal{W} , of the proposed algorithm. For the video data, most of the information varies along the frames (i.e., along mode-3 rather than the other modes). Consequently, we see the frames of $\mathcal{W}^{(1)}$ and $\mathcal{W}^{(2)}$ to have less information, whereas the frame of $\mathcal{W}^{(3)}$ is close to the original frame. As we enforce the low-rank constraint on mode- k of $\mathcal{W}^{(k)}$, each component has a compact representation that captures the original scene very well.

The proposed algorithm achieves the least RMSE compared to the baselines (see Table 2). In Hall, the RMSE scores of all baseline algorithms, except Dual, is at least two times that of the proposed algorithm. Despite the increase in dimensions of the tensor compared to hyperspectral images, choosing the same rank (10, 10, 5) gives the best RMSE scores. The reconstructed image shown in Fig. 3

are significantly clear as indicated by the RMSE scores. As mentioned earlier, in Tomato and Hall, we believe that the lack of local smoothness property leads to the failure of LRTC-TV algorithm.

7. Conclusion

We have proposed a novel factorization for nonnegative low-rank tensor completion, $\mathcal{W} = \sum_{k=1}^K (\mathcal{Z} + \mathcal{S}) \times_k U_k U_k^T$. The factorization decouples the nonnegative constraint and low-rank constraint on \mathcal{S} and $U_k U_k^T$ respectively. The resultant problem has a geometric structure in the constraints. We exploit this structure to propose a Riemannian optimization algorithm to solve the problem. On several real-world datasets, our proposed algorithm outperforms the state-of-the-art tensor completion algorithms.

Acknowledgement

Tanmay Kumar Sinha was supported by IIIT seed grant. Jayadev Naram thanks IHub-Data, IIIT Hyderabad for a research fellowship.

References

- [1] Boumal, N.: An introduction to optimization on smooth manifolds, (2020). Accessed online from <http://www.nicolasboumal.net/book>
- [2] Kolda, T. G. and Bader, B. W.: Tensor decompositions and applications, *SIAM Review* 51 (2009), no. 3, 455–500.
- [3] Boumal, N. and Mishra, B. and Absil, P.-A. and Sepulchre, R.: Manopt, a Matlab Toolbox for Optimization on Manifolds. In: *Journal of Machine Learning Research*, vol. 15, pp. 1455–1459 (2014). <https://www.manopt.org>
- [4] Absil, P.-A., Mahony, R. and Sepulchre, R.: *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ (2008). Accessed online from <https://press.princeton.edu/absil>
- [5] Journée, M., Bach, F., Absil, P.A. and Sepulchre, R., 2010. Low-rank optimization on the cone of positive semidefinite matrices. *SIAM Journal on Optimization*, 20(5), pp.2327-2351.
- [6] Nimishakavi, M., Jawanpuria, P. and Mishra, B.: A Dual Framework for Trace Norm Regularized Low-rank Tensor Completion, *Conference on Neural Information Processing Systems (NeurIPS)*, 2018. <https://github.com/madhavcsa/Low-Rank-Tensor-Completion>
- [7] Argyriou, A., Evgeniou, T. and Pontil, M.: Multi-task feature learning, *NIPS*, 2006.
- [8] Liu, J., Musialski, P., Wonka, P., and Ye, J.: Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence*, 35(1), 208-220, 2012.
- [9] Kressner, D., Steinlechner, M. and Vandereycken, B.: Low-rank tensor completion by Riemannian optimization, *BIT Numerical Mathematics* 54, no. 2 (2014): 447-468. <https://www.epfl.ch/labs/anchp/index-html/software/geomcg>
- [10] Kasai, H. and Mishra, B., 2016, June. Low-rank tensor completion: a Riemannian manifold preconditioning approach. In *International conference on machine learning* (pp. 1012-1021). PMLR. <https://bamdevmishra.in/codes/tensorcompletion>
- [11] Xu, Y. and Yin, W., 2013. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3), pp.1758-1789. <https://xu-yangyang.github.io/BCD>
- [12] Signoretto, M., Dinh, Q. T., De Lathauwer, L., and Suykens, J. A.: Learning with tensors: a framework based on convex optimization and spectral regularization. *Machine Learning*, 94(3), 303-351, 2014.
- [13] Cheng, H., Yu, Y., Zhang, X., Xing, E., and Schuurmans, D.: Scalable and sound low-rank tensor learning. In *Artificial Intelligence and Statistics* (pp. 1114-1123). PMLR, 2016.
- [14] Wimalawarne, K., Sugiyama, M. and Tomioka, R.: Multitask learning meets tensor factorization: task imputation via convex optimization. *Advances in neural information processing systems*, 27, 2014, pp.2825-2833.
- [15] Jawanpuria, P. and Mishra, B.: A unified framework for structured low-rank matrix learning, *ICML*, 2018.
- [16] Guo, X., Yao, Q. and Kwok, J. T.: Efficient sparse low-rank tensor completion using the frank-wolfe algorithm, *AAAI*, 2017. <https://github.com/quanmingyao/FFWTensor>
- [17] Li, X., Ye, Y. and Xu, X.: Low-Rank Tensor Completion with Total Variation for Visual Data inpainting. *AAAI*, 2017.
- [18] Yokota, T., Zhao, Q. and Cichocki, A., 2016. Smooth PARAFAC decomposition for tensor completion. *IEEE Transactions on Signal Processing*, 64(20), pp.5423-5436.
- [19] Zheng, Y., Huang, T., Ji, T., Zhao, X., Jiang, T., Ma, T.: Low-rank tensor completion via smooth matrix factorization, *Applied Mathematical Modelling*, Volume 70, 2019, Pages 677-695. https://github.com/uestctensorgroup/code_SMFLRTC
- [20] Xu, Y., Hao, R., Yin, W. and Su, Z., 2013. Parallel matrix factorization for low-rank tensor completion. *arXiv preprint arXiv:1312.1254*. <https://xu-yangyang.github.io/TMac>
- [21] Kim, D., Sra, S. and Dhillon, I. S.: A non-monotonic method for large-scale non-negative least squares. *Optimization Methods and Software*, 28(5):1012–1039, 2013. <http://optml.mit.edu/work/soft/npls.html>
- [22] Zhang, Z. and Aeron, S., 2016. Exact tensor completion using t-SVD. *IEEE Transactions on Signal Processing*, 65(6), pp.1511-1526.
- [23] Chen, B., Sun, T., Zhou, Z., Zeng, Y. and Cao, L., 2019. Nonnegative tensor completion via low-rank Tucker decomposition: model and algorithm. *IEEE Access*, 7, pp.95903-95914.
- [24] Veganzones, M. A., Cohen, J. E., Farias, R. C., Chanussot, J. and Comon, P.: Nonnegative tensor CP decomposition of hyperspectral data, *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 2577–2588, May 2016.
- [25] Li, X., Ng, M. K., Cong, G., Ye, Y. and Wu, Q.: MR-NTD: Manifold regularization nonnegative Tucker decomposition for tensor data dimension reduction and representation, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 8, pp. 1787–1800, Aug. 2017.
- [26] Lee, N., Phan, A., Cong, F. and Cichocki, A.: Nonnegative tensor train decompositions for multi-domain feature extraction and clustering, *NIPS*, 2016, pp. 87–95.
- [27] Xu, Y.: Alternating proximal gradient method for sparse nonnegative Tucker decomposition, *Mathematical Programming Computation*, vol. 7, no. 1, pp. 39–70, 2015.

- [28] Zhao, Q., Zhang, L. and Cichocki, A.: Bayesian CP factorization of incomplete tensors with automatic rank determination, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (2015), no. 9, 1751–1763.
- [29] Boumal, N. and Absil, P.-A.: Low-rank matrix completion via preconditioned optimization on the Grassmann manifold. In: *Linear Algebra and its Applications*, vol. 475, pp. 200–239 (2015).
- [30] Foster, D. H., Amano, K., Nascimento, S. M. C., and Foster, M. J.: Frequency of metamerism in natural scenes. *Journal of the Optical Society of America A*, 23, 2359-2372, 2006.
- [31] SIPI Image Database. <http://sipi.usc.edu/database/database.php?volume=misc>
- [32] YUV Video Sequences. <http://trace.eas.asu.edu/yuv/index.html>