An Effective Tensor Completion Method Based on Multi-linear Tensor Ring Decomposition

Jinshi Yu*, Guoxu Zhou*, Qibin Zhao*† and Kan Xie*
* School of Automation, Guangdong University of Technology, Guangzhou, 510006, China
† Tensor Learning Unit, RIKEN Center for Advanced Intelligence Project (AIP), Japan

***Abstract*—By considering the balance unfolding scheme does help to catch the global information for tensor completion and the recently proposed tensor ring decomposition, in this paper a weighted multilinear tensor ring decomposition model is proposed for tensor completion and called MTRD. Utilizing the circular dimensional permutation invariance of tensor ring decomposition, a very balance matricization scheme $< k, d >$-unfolding is employed in MTRD. In order to evaluate MTRD, it is applied on both synthetic data and image tensor data, and the experiment results show that MTRD are able to achieve the desired relative square error by spending much less time than its compared methods, i.e. TMac-TT and TR-ALS. The results of image completion also show that MTRD outperforms its compared methods in relative square error. Specifically, TMac-TT and TR-ALS fails to get the same relative square error as MTRD and TR-ALS prevails TMac-TT but requiring a large amount of running time. To sum up, MTRD is more applicable than its compared methods.**

I. Introduction

In recent years, tensor completion has attracted considerable attention as the success of matrix completion [1] [2] [3] [4]. Tensor is the generalization of matrix and tensor decomposition [5] [6] [7] is often viewed as the strong tool for data representation. Lots of tensor completion methods is thus proposed [8] [9] [10] [11] [12] [13]. Among them, the incomplete data is often viewed to have the Tucker and tensor train (TT) structure, e.g. simple low rank tensor completion (SiLRTC) [11], fast low rank tensor completion (FaLRTC) [11], low-rank tensor completion by parallel matrix factorization (TMac) [12], SiLRTC with TT rank (SiLRTC-TT) [14] and TMac with TT rank (TMac-TT) [14].

In [11], SiLRTC and FaLRTC employ tensor nuclear-norm minimization and use the singular value decomposition (SVD) in the algorithm, which lead to the expensive computation and is not applicable for large-scale tensor data completion. To alleviate this problem, TMac [12] is proposed to recover a low-rank tensor by simultaneously performing low-rank matrix factorizations to all-mode matricizations of the underlying tensor. Although TMac is non-convex, but TMac performs consistently throughout the test and gets the better results than FaLRTC. There have been some progress in formulating low rank tensor completion (LRTC) by viewing the incomplete data with Tucker structure, however, its matricization is based on mode-k unfolding which leads to the matrix unbalance. Specifically the mode-k unfolding takes one mode for rows of matrix and the rest for columns. Considering the small upper bound of tensor rank may fail to describe the global information and the matrix rank minimization is only efficient when the matrix is more balance, [14] proposes SiLRTC-TT and TMac-TT. The key advantage of SiLRTC-TT and TMac-TT is adopting the more balance matricization scheme k-unfolding to improve the performance of the method. SiLRTC-TT and TMac-TT are shown outstanding than SiLRTC and TMac, and among them TMac-TT performs best. However, one crucial drawback is that the balances of the k-unfolding matrices $X_{[k]} \in \mathbb{R}^{I_1 \cdots I_k \times I_{k+1} \cdots I_N}$ with $k = 1, \ldots, N$ are very different, specifically the middle matrices are generally more balance than the border matrices.

The drawback of TMac-TT is in fact caused by the limitation of TT decomposition. In particular, as outlined in [15], TT decomposition has following limitations. i) TT model has rank-1 constraints to the border factors, i.e. $R_0 = R_N = 1$; ii) TT-ranks always have a fixed pattern, i.e., smaller for the border factors and larger for the middle factors; iii) the multiplications of TT factors are not permutation invariant. Taking into account these limitations, Zhao et al. [15] propose tensor ring (TR) decomposition which can be viewed as a linear combination of TT decomposition. By employing the trace operation, TR decomposition removes the rank-1 constraints for border factors and decomposes Nth-order tensor into N 3th-order factor tensors. In addition, TR model is circular dimensional permutation invariance due to the properties of trace operation.

Recently, [16] proposes a new tensor completion based on TR decomposition, namely TR-ALS. TT-ALS [17] with singular value decomposition (SVD) has been firstly used for initialization of TR-ALS, which will lead to a large amount of running time especially for large scale tensor data. Subsequently TR-ALS mainly applies an alternating least square method to update factor tensors, i.e. updating each factor tensor while remaining the rest factor tensors fixed. Unfortunately, TR-ALS requires a large amount of time to update factor tensors. Specifically, letting Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I \times \cdots \times I}$ and factor tensors $\mathcal{U}_k \in \mathbb{R}^{R \times I \times R}$ with $k = 1, \ldots, N$, the overall computational complexity of updating all factor tensors within one iteration is $\max(\mathcal{O}(NPR^4), \mathcal{O}(NR^6))$ [16], where P is the total number of observed entries. As the tensor order and rank increase, the high computational complexity significantly leads to a large amount of time, which has been shown in our experiment. Thus, TR-ALS is not applicable for large scale tensor data.

Considering the balance unfolding scheme does help to catch the global information [12] [14] and the recently proposed TR decomposition has a powerful and generalized representation abilities [17], we propose a novel tensor completion method by a weighted multilinear tensor ring decomposition in this paper, simply denoted as MTRD. Utilizing the circular dimensional permutation invariance, a very balance matrix scheme (about half of modes versus the rest) is proposed in MTRD, which is the most balance scheme to the best of our knowledge and denoted as $< k, d >$-unfolding, where generally $d = \lfloor \frac{N}{2} \rfloor$ and $\lfloor . \rfloor$ denotes the $floor$ operator. More details of MTRD will be described in the latter section. In order to evaluate the efficient of MTRD, MTRD is compared with TMac-TT and TR-ALS algorithms on both synthetic tensor data and images.

The rest of the paper is organized as follows. Section II provides some notations and preliminaries of tensors. In Section III, the formulation and algorithm for MTRD are described. Section IV presents the simulation results on synthetic data and images completion. Finally, Conclusion is provided in Section V.

II. Notations and Preliminaries of Tensors

In this paper, some notations and preliminaries of tensors [18] are adopted. Scalars, vectors and matrices are denoted by lowercase letter (e.g. x), boldface lowercase letters (e.g. $\mathbf{x}$) and capital letters (e.g. X) respectively. A tensor is a multidimensional array and its mode is the number of its dimension. A N-mode tensor is denoted by calligraphic letters, e.g. $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ where $I_k, k = 1, \ldots, N$ denotes the dimension along the mode k . An element of a tensor $\mathcal{X}$ is denoted by $\mathcal{X}(i_1, \ldots, i_N)$ or $\mathcal{X}_{i_1,\ldots,i_N}$, where $1 \le i_k \le I_k, k = 1, \ldots, N$. $\mathcal{X}(:, i_2, \ldots, i_N)$ is used to denote the fiber along mode 1, and $\mathcal{X}(:, :, i_3, \ldots, i_N)$ denotes the slice along mode 1 and mode 2 and so forth. The Frobenius norm of a tensor is the square root of the sum of the squares of all its elements, i.e., $\|\mathcal{X}\|_F = \sqrt{\sum_{i1,\ldots,i_N=1}^{I_1,\ldots,I_N} \mathcal{X}(i_1, \ldots, i_N)^2}$.

Definition 1: (k-unfolding [15]) Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, k-unfolding of $\mathcal{X}$ is denoted by $X_{[k]}$ of size $\prod_{j=1}^{k} I_j \times \prod_{j=k+1}^{N} I_j$ with its elements defined by

$$X_{[k]}(\overline{i_1 \ldots i_k}, \overline{i_{k+1} \ldots i_N}) = \mathcal{X}(i_1, i_2, \ldots, i_N) \qquad (1)$$

Definition 2: (Middle unfolding) Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, the middle unfolding of $\mathcal{X}$ is of size $\prod_{j=2}^{N-1} I_j \times I_1 I_N$, according the ordering of indices associated to the 2 modes, two types of middle unfolding operations are defined respectively, i.e. $M_f(\mathcal{X}) \in \mathbb{R}^{I_2 I_3 \ldots I_{N-1} \times I_N I_1}$, $M_b(\mathcal{X}) \in \mathbb{R}^{I_2 I_3 \ldots I_{N-1} \times I_1 I_N}$. Let $X_f = M_f(\mathcal{X})$ and $X_b = M_b(\mathcal{X})$, then their elements are defined respectively by

$$X_f(\overline{i_2 i_3 \ldots i_{N-1}}, \overline{i_N i_1}) = \mathcal{X}(i_1, i_2, \ldots, i_N) \qquad (2)$$
$$X_b(\overline{i_2 i_3 \ldots i_{N-1}}, \overline{i_1 i_N}) = \mathcal{X}(i_1, i_2, \ldots, i_N) \qquad (3)$$

Definition 3: Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, $\overleftarrow{k}(a)$ denotes the mode which is a away behind the mode k. Specifically, if $N = 5$, $\overleftarrow{3}(3) = 5$, $\overleftarrow{3}(1) = 2$ and $\overleftarrow{3}(0) = 3$ denote the 5th mode, 2th mode and 3th mode respectively.

Definition 4: Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, $\overrightarrow{k}(a)$ denotes the mode which is a away in front of the mode k. Specifically, if $N = 5$, $\overrightarrow{3}(0) = 3$, $\overrightarrow{3}(1) = 4$, $\overrightarrow{3}(3) = 1$ denote the 3th mode, 4th mode and 1th mode respectively.

Definition 5: ($< k, d >$-unfolding) Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, $< k, d >$-unfolding of $\mathcal{X}$ is denoted by $X_{<k,d>}$ of size $I_{\overleftarrow{k}(d-1)} \ldots I_{\overleftarrow{k}(0)} \times I_{\overrightarrow{k}(1)} \ldots I_{\overrightarrow{k}(N-d)}$ with its elements defined by

$$X_{<k,d>}(\overline{i_{\overleftarrow{k}(d-1)} \ldots i_{\overleftarrow{k}(0)}}, \overline{i_{\overrightarrow{k}(1)} \ldots i_{\overrightarrow{k}(N-d)}}) = \mathcal{X}(i_1, i_2, \ldots, i_N) \qquad (4)$$

Definition 6: (Tensor ring decomposition) Let a N-mode tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ with its TR rank $[R_1, R_2, \ldots, R_N]$, then the elements of $\mathcal{X}$ are represented by

$$\mathcal{X}(i_1, i_2, \ldots, i_N) = \mathrm{Tr}(\mathcal{U}_1(:, i_1, :)\mathcal{U}_2(:, i_2, :) \ldots \mathcal{U}_N(:, i_N, :)) \qquad (5)$$

where $\mathcal{U}_k \in \mathbb{R}^{R_{k-1} \times I_k \times R_k}$, $k = 1, \ldots, N$, $R_0 = R_N$ and Tr(.) denotes the trace operation. The tensor ring decomposition of tensor $\mathcal{X}$ is denoted by $\mathcal{X} = \Re(\mathcal{U}_1, \ldots, \mathcal{U}_N)$.

III. Formulation and Algorithm for MTRD

Considering the weighted multilinear matrix factorization model outperforms tensor nuclear-norm minimization model [12] [14] and TR decomposition outperforms TT decomposition [17], a novel tensor completion method called MTRD is proposed in this paper. In fact, MTRD is a weighted multilinear tensor ring decomposition model. Utilizing the circular dimensional permutation invariance of TR decomposition, $< k, d >$-unfolding is employed in MTRD, where generally $d = \lfloor \frac{N}{2} \rfloor$. It is easy to note that $< k, d >$-unfolding is the better balance scheme than k-unfolding in TMac-TT. Specifically, letting $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, $I_1 = I_2 = \cdots = I_N = I$, $N = 2d$, the balance of k-unfolding matrix $X_{[k]} \in \mathbb{R}^{I^k \times I^{N-k}}$ is obviously influenced by mode k, i.e., the poorer the balance is, the closer the mode k is to the two sides. While, the $< k, d >$-unfolding matrix $X_{<k,d>} \in \mathbb{R}^{I^d \times I^d}$ is invariance with k and the balances of $X_{<k,d>}$ with $k = 1, \ldots, N$ are identity and best.

Letting the low-rank tensor $\mathcal{M} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ and TR rank $[R_1, \ldots, R_N]$, MTRD employs TR decomposition to each $< k, d >$-unfolding matrix $M_{<k,d>}$ of $\mathcal{M}$ such that $M_{<k,d>} \approx \Re(\mathcal{W}_k, \mathcal{H}_k)$ for $k = 1, \ldots, N$. One common variable $\mathcal{X}$ is introduced to relate these tensor ring decomposition. Formally, MTRD recovers $\mathcal{M}$ by solving the following multilinear tensor ring decomposition model:

$$\min_{\mathcal{W}_k, \mathcal{H}_k, \mathcal{X}} : \quad \sum_{k=1}^{N} \frac{1}{2}\alpha_k \|\Re(\mathcal{W}_k, \mathcal{H}_k) - X_{<k,d>}\|_F^2 \qquad (6)$$
$$s.t. : \quad \mathcal{X}_\Omega = \mathcal{M}_\Omega.$$

where $\mathcal{W}_k \in \mathbb{R}^{R_{\overleftarrow{k}(d)} \times m_k \times R_{\overleftarrow{k}(0)}}$, $\mathcal{H}_k \in \mathbb{R}^{R_{\overleftarrow{k}(0)} \times n_k \times R_{\overleftarrow{k}(d)}}$, $X_{<k,d>} \in \mathbb{R}^{m_k \times n_k}$, $m_k = I_{\overleftarrow{k}(d-1)} \ldots I_{\overleftarrow{k}(0)}$, $n_k = I_{\overrightarrow{k}(1)} \ldots I_{\overrightarrow{k}(N-d)}$, $\alpha_k = \frac{\min(m_k, n_k)}{\sum_{k=1}^{N} \min(m_k, n_k)} > 0$, $\sum_{k=1}^{N} \alpha_k = 1$. $d = \lfloor \frac{N}{2} \rfloor$, $\lfloor . \rfloor$ denotes the $floor$ operator.

Let $\tilde{X}_{<k,d>} = \Re(\mathcal{W}_k, \mathcal{H}_k)$, then $\tilde{X}_{<k,d>}$ can be expressed in an elements-wise form given by

$$\tilde{X}_{<k,d>}(\overline{m}_k, \overline{n}_k) = \mathrm{Tr}(\mathcal{W}_k(:, \overline{m}_k, :)\mathcal{H}_k(:, \overline{n}_k, :)) \tag{7}$$

where $1 \leq \overline{m}_k \leq m_k, 1 \leq \overline{n}_k \leq n_k$. Equation (7) is equivalent to

$$\begin{aligned}
& \tilde{X}_{<k,d>}(\overline{m}_k, \overline{n}_k) \\
= & \textstyle\sum_{r_{\overleftarrow{k}(d)}=1, r_{\overleftarrow{k}(0)}=1}^{R_{\overleftarrow{k}(d)}, R_{\overleftarrow{k}(0)}} \mathcal{W}_k(r_{\overleftarrow{k}(d)}, \overline{m}_k, r_{\overleftarrow{k}(0)}) \\
& \mathcal{H}_k(r_{\overleftarrow{k}(0)}, \overline{n}_k, r_{\overleftarrow{k}(d)}) \\
= & \textstyle\sum_{\overline{r_{\overleftarrow{k}(d)} r_{\overleftarrow{k}(0)}}}^{\overline{R_{\overleftarrow{k}(d)} R_{\overleftarrow{k}(0)}}} W_k(\overline{m}_k, \overline{r_{\overleftarrow{k}(d)} r_{\overleftarrow{k}(0)}}) H_k(\overline{r_{\overleftarrow{k}(d)} r_{\overleftarrow{k}(0)}}, \overline{n}_k) \\
= & W_k(\overline{m}_k, :) H_k(:, \overline{n}_k)
\end{aligned} \tag{8}$$

where $W_k = M_b(\mathcal{W}_k) \in \mathbb{R}^{m_k \times R_{\overleftarrow{k}(d)} R_{\overleftarrow{k}(0)}}$, $H_k^T = M_f(\mathcal{H}_k) \in \mathbb{R}^{n_k \times R_{\overleftarrow{k}(d)} R_{\overleftarrow{k}(0)}}$. Equation (8) can be easily rewritten in the matrix form given by

$$\tilde{X}_{<k,d>} = W_k H_k \tag{9}$$

Thus,

$$\Re(\mathcal{W}_k, \mathcal{H}_k) = W_k H_k \tag{10}$$

By substituting (10) into (6), (6) can be rewritten as

$$\begin{aligned}
\min_{W_k, H_k, \mathcal{X}} : \quad & \textstyle\sum_{k=1}^{N} \frac{1}{2} \alpha_k \| W_k H_k - X_{<k,d>} \|_F^2 \\
s.t.: \quad & \mathcal{X}_\Omega = \mathcal{M}_\Omega.
\end{aligned} \tag{11}$$

Apparently, in order to solve the problem given by equation (11) we can first solve the following problem:

$$\begin{aligned}
\min_{W_k, H_k, \mathcal{X}} : \quad & \tfrac{1}{2} \| W_k H_k - X_{<k,d>} \|_F^2 \\
s.t.: \quad & \mathcal{X}_\Omega = \mathcal{M}_\Omega.
\end{aligned} \tag{12}$$

for $k = 1, \ldots, N$. Taking into account that (12) is convex with respect to each block of the variables W_k, H_k and $\mathcal{X}$ while the other two are fixed, we apply the alternating least squares method to (12) and have the following updates:

$$W_k^{t+1} = X_{<k,d>}^t (H_k^t)^T (H_k^t (H_k^t)^T)^\dagger, \tag{13}$$

$$H_k^{t+1} = ((W_k^{t+1})^T W_k^{t+1})^\dagger (W_k^{t+1})^T X_{<k,d>}^t, \tag{14}$$

$$X_{<k,d>}^{t+1} = W_k^{t+1} H_k^{t+1}. \tag{15}$$

where $\dagger$ denotes the Moore-Penrose pseudo-inverse. After updating W_k^{t+1}, H_k^{t+1} and $X_{<k,d>}^{t+1}$, $\mathcal{X}$ can be updated by following formulation:

$$\mathcal{X}_{i_1,\ldots,i_N}^{t+1} = \begin{cases} (\sum_{k=1}^{N} \alpha_k \mathrm{fold}(X_{<k,d>}^{t+1}))_{i_1,\ldots,i_N} & \\ \qquad\qquad (i_1, \ldots, i_N) \notin \Omega; & \\ \mathcal{M}_{i_1,\ldots,i_N} \qquad (i_1, \ldots, i_N) \in \Omega. & \end{cases} \tag{16}$$

As shown in [12] that, no matter how W_k is computed, only the products $W_k H_k, k = 1, \ldots, N$ affect $\mathcal{X}$ and thus the recovery $\mathcal{M}$. Hence, we shall update W_k in the following more efficient way

$$W_k^{t+1} = X_{<k,d>}^t (H_k^t)^T, \tag{17}$$

Algorithm 1 The MTRD algorithm.

Require: Missing entry zero filled tensor data $\mathcal{M} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ with observed index set Ω.

Parameters: $t = 0$, d, α_k where $k = 1, \ldots, N$.

```
Initialize: H_k^0, X^0 with X_Ω^0 = M_Ω.
repeat
    for k = 1 to N do
        W_k^{t+1} ← (17)
        H_k^{t+1} ← (14)
        X_{<k,d>}^{t+1} ← (15)
    end for
    X^{t+1} ← (16)
    t ← t + 1.
until A stopping criterion is met
```

which together with (14) gives the same products $W_k^{t+1} H_k^{t+1}$ as those by (13) and (14).

MTRD algorithm is summarized in Algorithm 1 and the algorithm can be stopped when the algorithm satisfies one of following conditions: i) relative error (RE) of the tensor $\mathcal{X}$ between two successive iteration achieves the desire accuracy; ii)the number of iteration reaches maximum. The maximum number of iteration is generally set as 1000. RE and RSE are respectively defined as follow:

$$\mathrm{RE} = \frac{\|\mathcal{X}^{t+1} - \mathcal{X}^t\|_F}{\|\mathcal{X}^t\|_F} \leq tol \tag{18}$$

where generally $tol = 1e^{-8}$.

IV. Experiment

In order to test the efficient of MTRD, both synthetic data and color images are used to evaluate its performance. MTRD will be compared with other methods in terms of relative square error (RSE) and running time . RSE between the approximately recovered tensor $\mathcal{X}$ and the original one $\mathcal{M}$ is defined as,

$$\mathrm{RSE} = \frac{\|\mathcal{X} - \mathcal{M}\|_F}{\|\mathcal{M}\|_F} \tag{19}$$

A. Completion of Low TR Rank Tensor

In this section, we consider the completion problem on synthetic tensor data. Specially, Nth-order tensor data $\mathcal{T} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ with tensor ring rank $[R_1^{TR}, \ldots, R_N^{TR}]$ is generated such that its elements are represented by a TR format. Specifically, its elements are $\mathcal{T}(i_1, \ldots, i_N) = Tr(\mathcal{U}_1(:, i_1, :) \ldots \mathcal{U}_N(:, i_N, :))$, where $\mathcal{U}_k \in \mathbb{R}^{R_{k-1}^{TR} \times I_k \times R_k^{TR}}$ with $k = 1, \ldots, N$ are generated by standard Gaussian distribution $\mathcal{N}(0, 1)$. For simplicity, we set $I_k = I, R_k^{TR} = R, k = 1, \ldots, N$ and the observed entries are chosen randomly from the tensor based on a uniform distribution. The 4th-order, 7th-order, 10th-order tensor data are generated by respectively setting mode dimension $I = 25, 10, 6$ and the corresponding TR rank $R = 8, 5, 3$. These three tensor data were denoted as $\mathcal{T}_{4D}, \mathcal{T}_{7D}, \mathcal{T}_{10D}$.

MTRD, TMac-TT and TR-ALS are respectively applied on these three tensor data and the RSEs with respect to observed ratio $p = 0.1, 0.3, 0.5, 0.7$ are shown in TABLE I. And the corresponding running time is shown in TABLE II. For MTRD and TR-ALS, the true rank is given, i.e. $R_1^{\text{MTRD}} = \cdots = R_N^{\text{MTRD}} = R$ and $R_1^{\text{TR-ALS}} = \cdots = R_N^{\text{TR-ALS}} = R$. For TMac-TT, the first TT rank is chosen as $R_1^{TT} = \cdots = R_{N-1}^{TT} = R$ and the completion with this rank is called TMac-TT-LR. The second TT rank is chosen as the double of the true rank, i.e. $R_1^{TT} = \cdots = R_{N-1}^{TT} = 2R$ and the completion with this rank is called TMac-TT-HR. The methods will stop when RSE $\leq$ 1e-6 or the number of iteration reaches 1000.

The results in TABLE I show that both MTRD and TR-ALS are successfully able to recover the tensor data and achieve the competitive results, while TMac-TT with low and high rank, i.e. TMac-TT-LR and TMac-TT-HR, fails to recover the tensor data in all the cases. This could be caused by the fact that TR decomposition can be viewed as the linear combination of TT decomposition. TABLE II shows the corresponding time for TABLE I. As shown in TABLE II, TR-ALS achieves the desired RSE by spending more time than MTRD, which is clearer as the tensor scale increases. This result may be caused by the SVD in initialization of factors and the high computational complexity $\max(\mathcal{O}(NPR^4), \mathcal{O}(NR^6))$ of updating all factors within one iteration, which significantly leads to the high running time for large scale tensor data. [16] has shown that the higher observed ratios, the fewer iteration TR-ALS needs. However, the computational complexity $\max(\mathcal{O}(NPR^4), \mathcal{O}(NR^6))$ increases with respect to the total amount of observed entries P, which significantly results in more running time for one iteration as observed ratio p increases. As a result, we find from TABLE II that the running time of TR-ALS tends to increase as the observed ratio increases in one case (e.g. $\mathcal{T}_{7D}$). In contrast, our method MTRD significantly decreases the running time as the observed ratio increases in one case. These results mean that MTRD is more applicable than TR-ALS when processing large high scale tensor data.

B. *Image completion*

In this section, the RGB images namely 'Lena' and 'Peppers' are used to test the performance of MTRD and its compared methods, i.e. TMac-TT and TR-ALS. All images are initially presented by 3th-order tensor with size of $256 \times 256 \times 3$. For methods with TT rank or TR rank, the low order tensors are often reshaped into high order tensors to improve performance in classification [19] and completion [14] [16] [20]. The 3th-order tensor is thus reshaped into 9th-order tensor of size $4 \times 4 \times \cdots \times 4 \times 3$. Observed entries of the images are chosen randomly according to a uniform distribution and the observed ratios of $p = 0.1, 0.3$ are considered. The methods are applied on the images and the performances of RSE and its running time are compared. For simplicity, we set all components of rank same, e.g. $R_1^{\text{MTRD}} = \cdots = R_N^{\text{MTRD}}$ for MTRD and so does for other methods. We stop the method when the method converges or its iteration number reaches 1000. For all the methods, the minimum rank is chosen such that the method gets the best performance. Fig. 1 and Fig. 2 respectively show the performance of the methods for 'Lena' and 'Peppers' images.

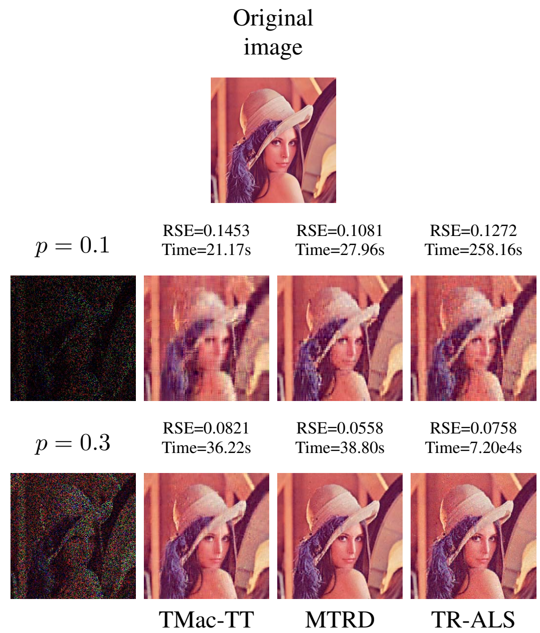

Fig. 1: Recover the 'Lena' image with 10% and 30% of observed entries using different methods. Top image represents the original image. Second row represents the copy of 'Lena' image with 10% observed entries and the recovery results of different methods. Third row represents the copy of 'Lena' image with 30% observed entries and the recovery results of different methods. Note that image is represented by 9th-order tensor for all the methods.

As shown in Fig. 1, compared with TMac-TT and TR-ALS, MTRD gets the best results in RSE in all case of p by spending acceptable running time. This result indicates that MTRD outperforms TMac-TT and TR-ALS in image completion. TR-ALS can achieve the better results in RSE than TMac-TT, however spending a large amount of running time, which could be caused by the SVD in initialization of factors and the high computational complexity $\max(\mathcal{O}(NPR^4), \mathcal{O}(NR^6))$ of updating all factors within one iteration. Same experiment is performed on the 'Peppers' image and the recovery results are shown in Fig. 2. The results also indicate that MTRD prevails against the other two algorithms in image completion. In summary, MTRD outperforms TMac-TT and TR-ALS in image completion.

V. Conclusion

In this paper, an efficient algorithm MTRD has been proposed based on a weighted multi-linear tensor ring decomposition model. By utilizing the circular dimensional permutation

TABLE I: TMac-TT, MTRD, TR-ALS are applied on 4th-order, 7th-order, 10th-order tensor data and the RSEs with respect to 10%, 30%, 50%, 70% observed ratio are shown as follows. Note that TMac-TT-LR is TMac-TT with low rank and TMac-TT-HR is TMac-TT with high rank.

	$\mathcal{T}_{4D}$				$\mathcal{T}_{7D}$				$\mathcal{T}_{10D}$			
	0.1	0.3	0.5	0.7	0.1	0.3	0.5	0.7	0.1	0.3	0.5	0.7
TMac-TT-LR	0.9973	0.7628	0.6276	0.4804	0.6913	0.6091	0.4959	0.3767	0.5976	0.4839	0.3947	0.2982
TMac-TT-HR	1.2731	0.7085	0.5631	0.4258	0.5519	0.4504	0.3618	0.2687	0.3412	0.2591	0.2046	0.1508
MTRD	9.91e-7	9.84e-7	7.83e-7	7.61e-7	9.54e-7	8.18e-7	7.05e-7	4.14e-7	9.63e-7	8.34e-7	9.16e-7	3.40e-7
TR-ALS	3.77e-7	4.61e-7	4.53e-7	1.89e-7	3.05e-8	7.47e-8	1.74e-8	8.19e-9	1.88e-7	7.35e-9	3.45e-7	7.42e-8

TABLE II: TMac-TT, MTRD, TR-ALS are applied on 4th-order, 7th-order, 10th-order tensor data and the running time (the unit is second) with respect to 10%, 30%, 50%, 70% observed ratio are shown as follows. Note that TMac-TT-LR is TMac-TT with low rank and TMac-TT-HR is TMac-TT with high rank.

	$\mathcal{T}_{4D}$				$\mathcal{T}_{7D}$				$\mathcal{T}_{10D}$			
	0.1	0.3	0.5	0.7	0.1	0.3	0.5	0.7	0.1	0.3	0.5	0.7
TMac-TT-LR	9.64	9.40	12.53	15.14	611.90	715.10	797.62	876.99	5.26e3	5.78e3	6.24e3	6.64e3
TMac-TT-HR	12.79	13.36	15.88	20.17	740.93	832.81	903.47	976.31	5.93e3	6.47e3	7.01e3	7.33e3
MTRD	27.62	4.93	2.47	1.52	203.74	65.31	35.72	23.75	2.12e5	711.83	338.55	223.33
TR-ALS	52.64	28.40	30.82	32.82	251.77	266.54	303.74	347.12	1.28e5	1.33e3	1.18e3	1.39e3

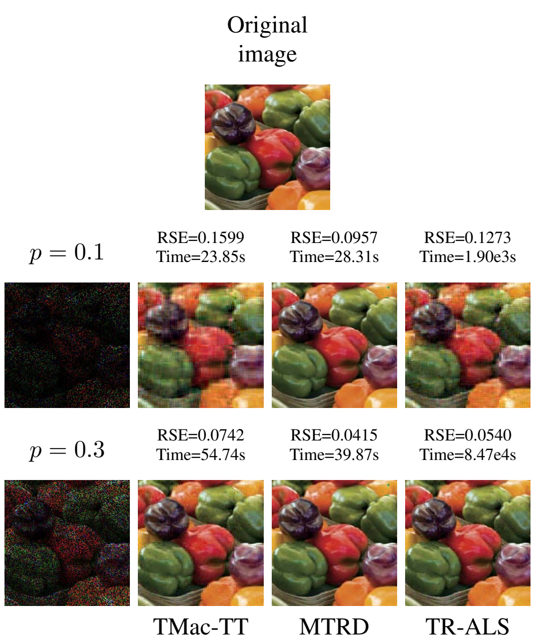

Fig. 2: Recover the 'Pepper' image with 10% and 30% of observed entries using different methods. Top image represents the original image. Second row represents the copy of 'Pepper' image with 10% observed entries and the recovery results of different methods. Third row represents the copy of 'Pepper' image with 30% observed entries and the recovery results of different methods. Note that image is represented by 9th-order tensor for all the methods.

invariance of tensor ring decomposition, a very balance matricization scheme $< k, d >$-unfolding is proposed. MTRD and its compared methods are applied on both synthetic data and images represented by high order tensor. The experiment results of synthetic data completion show that MTRD is able to achieve the desired relative square error by spending much less time and requires less time as the observed ratio increases. Moreover, MTRD gets the best performance in image completion. In summary, MTRD outperforms its compared methods and is more applicable in practice.

VI. ACKNOWLEDGEMENTS

This work was partly supported by the National Natural Science Foundation of China 61673124.

REFERENCES

[1] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Transactions on Information Theory*, vol. 56, pp. 2980–2998, June 2010.

[2] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, pp. 925–936, June 2010.

[3] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[4] E. J. Candes and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Transactions on Information Theory*, vol. 56, pp. 2053–2080, May 2010.

[5] G. Zhou, A. Cichocki, and S. Xie, "Fast nonnegative matrix/tensor factorization based on low-rank approximation," *IEEE Transactions on Signal Processing*, vol. 60, pp. 2928–2940, June 2012.

[6] Q. Zhao, G. Zhou, L. Zhang, A. Cichocki, and S. I. Amari, "Bayesian robust tensor factorization for incomplete multiway data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, pp. 736–748, April 2016.

[7] A. Cichocki, D. Mandic, L. D. Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. PHAN, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, pp. 145–163, March 2015.

[8] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, vol. 6, pp. 4734–4739 vol.6, 2001.

[9] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41 – 56, 2011.

[10] Q. Zhao, L. Zhang, and A. Cichocki, "Bayesian cp factorization of incomplete tensors with automatic rank determination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, pp. 1751–1763, Sept 2015.

[11] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 208–220, Jan 2013.

[12] Y. Xu, R. Hao, W. Yin, and Z. Su, "Parallel matrix factorization for low-rank tensor completion," *Inverse Problems and Imaging*, vol. 9, pp. 601–624, 1 2015.

[13] H. Tan, B. Cheng, W. Wang, Y.-J. Zhang, and B. Ran, "Tensor completion via a multi-linear low-n-rank factorization model," *Neurocomputing*, vol. 133, pp. 161 – 169, 2014.

[14] H. N. Phien, H. D. Tuan, J. A. Bengua, and M. N. Do, "Efficient tensor completion: Low-rank tensor train," *CoRR*, vol. abs/1601.01083, 2016.

[15] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, "Tensor ring decomposition," *CoRR*, vol. abs/1606.05535, 2016.

[16] W. Wang, V. Aggarwal, and S. Aeron, "Efficient low rank tensor ring completion," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5698–5706, Oct 2017.

[17] I. V. Oseledets, "Tensor-train decomposition," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.

[18] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[19] A. Novikov, D. Podoprikhin, A. Osokin, and D. Vetrov, "Tensorizing neural networks," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, (Cambridge, MA, USA), pp. 442–450, MIT Press, 2015.

[20] L. Yuan, Q. Zhao, and J. Cao, "High-order tensor completion for data recovery via sparse tensor-train optimization," *CoRR*, vol. abs/1711.02271, 2017.