Optimum Subspace Learning and Error Correction for Tensors

Yin Li, Junchi Yan, Yue Zhou⋆, and Jie Yang

Institute of Image Processing and Pattern Recogntion,
Shanghai Jiaotong University
{happyharry,yanster,zhouyue,jieyang}@sjtu.edu.cn

Abstract. Confronted with the high-dimensional tensor-like visual data, we derive a method for the decomposition of an observed tensor into a low-dimensional structure plus unbounded but sparse irregular patterns. The optimal rank-$(R_1, R_2, ...R_n)$ tensor decomposition model that we propose in this paper, could automatically explore the low-dimensional structure of the tensor data, seeking optimal dimension and basis for each mode and separating the irregular patterns. Consequently, our method accounts for the implicit multi-factor structure of tensor-like visual data in an explicit and concise manner. In addition, the optimal tensor decomposition is formulated as a convex optimization through relaxation technique. We then develop a block coordinate descent (BCD) based algorithm to efficiently solve the problem. In experiments, we show several applications of our method in computer vision and the results are promising.

1 Introduction

As the size of data and the amount of redundancy increase fast with dimensionality, the recent explosion of massive amounts of high-dimensional visual data presents a challenge to computer vision. Most of the existing high-dimensional visual data either has the natural form of tensor (e.g. multi-channel images and videos) or can be grouped into the form of tensor (e.g. tensor face [1]). On one side, one may seek a compact and concise low-dimensional representation of the data, such as dimension reduction [2–4] or image compression [5]. On the other side, one may seek to detect the irregular patterns of the data, such as saliency detection [6] or foreground segmentation [7]. As a consequence, it is desirable to develop tools that can find and exploit the low-dimensional structure in a high-dimensional tensor-like visual data.

In the two-dimensional case, i.e. the matrix case, the "rank" plays an important part in capturing the global information of visual data. One simple and useful assumption is that the data lie near certain low-dimensional subspace,

Fig. 1. Result of our method on a color facade (left). The method automatically seek a low dimensional representation (middle) and separate the sparse irregular patterns (right). Better viewed in color and zoom in for details.

which is closely related to the notation of rank. Although the "rank" itself is nonconvex, it can be approximated by its convex envelop, namely the trace norm. The validation of this approximation is justified in theory [8]. Among all the trace norm minimization problems, matrix completion may be a well-known one [8, 9]. Recently, [10] extends the matrix completion problem to the tensor case and develops an efficient solution.

The "sparsity" is also a useful tool for visual data analysis. One common observation is that the irregular patterns often occupy a small portion of the data. This sparse prior has demonstrated a wide range of applications including image denoising [11], error correction [12] and face recognition [13]. It was not until very recently that had much attention been focused on the rank-sparsity problem for matrix [14, 15], namely the Principal Component Pursuit (PCP) or the Robust Principal Component Analysis (RPCA). These work seek to directly decompose a matrix into a low-rank part plus a sparse part. Theoretic analysis [15] shows that under rather weak assumptions, the problem can be solved by the joint minimization of trace norm and l_1 norm.

We consider the decomposition of an observed tensor data into a low dimensional structure and an additive (sparse) irregular pattern. Analogy to the PCP problem in the matrix case, the optimal rank-$(R_1, R_2, ...R_n)$ tensor decomposition model that we propose in the paper, could automatically explore the low-dimensional structure of the tensor data, seeking optimal dimension and basis for each mode and separating the irregular patterns (See Fig.1 for an example and the core idea). Our method is an multilinear extension of the PCP problem and subsumes the matrix PCP problem as a special case. The optimal tensor decomposition is formulated as a convex optimization through relaxation technique. In addition, we develop a efficient block coordinate descent (BCD) based solution. We show several applications of our method in computer vision and the results are promising.

The rest of the paper is organized as follows: Section 2 briefly reviews related work. Section 3 provides the foundations of tensor algebra that are relevant to our approach. Section 4 formulates our proposed optimal rank-$(R_1, R_2, ..., R_n)$ tensor decomposition model together with its solution. Section 5 reports experimental

results of our algorithm for several computer vision tasks. Finally, Section 6 concludes the paper.

2   Related Work

Prior research on subspace analysis is abundant, including Principal Component Analysis (PCA) [2], Linear Discriminant Analysis (LDA) [3], Locality Preserving Projection (LPP) [4], etc. These models are widely adopted in computer vision problems. They usually treat an image as a vector and consider only one factor of the problem (e.g. only the face identity is considered in face recognition task). Various researchers have attempted to overcome the shortcomings of these methods by considering the image as a 2-mode tensor (i.e. matrix), including 2DPCA [7], tensor subspace analysis (tensor LPP) [16], tensor LDA [17], etc.

   Much effort has been focused on the tensor representation and analysis of visual data. Vasilescu and Terzopoulos [1] introduce a multilinear tensor framework to the analysis of face ensembles that explicitly accounts for each of the multiple factors implicit in image formation. Possible applications of the multilinear approach cover face recognition [18–20], facial expression decomposition [20, 21] and face super-resolution [22]. These methods are based on the higher order singular value decomposition [23], i.e. the Tucker decomposition, leading to best rank-$(R_1, R_2, ...R_n)$ approximations of higher-order tensors (See Section 2 for details).

   Shashua and Levin [24] propose 3-way tensor decomposition for the images as a 3D cube. They develop compression algorithms for images and video, that take advantage of spatial and temporal redundancies. The method is further extended to non-negative 3D tensor factorization [22] for the purpose of establishing a local parts feature decomposition from an object class of images. The non-negative tensor factorization is also applied to hypergraph clustering [25] to study a series of vision problems including 3D multi-body segmentation and illumination-based face clustering. These methods are based on the PARAFAC decomposition [26], leading to best (non-negative) rank-R approximations of higher-order tensors.

   The optimal rank-$(R_1, R_2, ...R_n)$ tensor decomposition model that we propose in the paper seeks a best n-rank condition for the tensor data, yielding a rather different approach from previous work. Our model could simultaneously find the optimal dimension and basis for each mode and separate the irregular patterns in an automatic manner. As a result, by rather weak prior, our method can account for the implicit multi-factor structure of tensor-like visual data in an explicit and concise manner.

3   Tensor Basics

A tensor, or n-way array, is a higher-order generalization of matrix. We use lower case letters (a, b,...) for scalars, bold lower case letters ($\boldsymbol{a}$, $\boldsymbol{b}$,...) for vectors, upper case letters (A, B,...) for matrix, and calligraphic upper case letters ($\mathcal{A}$, $\mathcal{B}$,...) for

higher order tensors. Formally, a n-mode tensor is defined as $\mathcal{A} \in R^{I_1 \times I_2 \times ... \times I_n}$, with its elements $a_{i_1}...a_{i_k}...a_{i_n} \in R$. Therefore, a vector can be seen as a 1-mode tensor and a matrix can be seen as a 2-mode tensor.

It is often convenient to flatten a tensor into a matrix, also called matricizing or unfolding. The "unfold" operation along the kth mode on a tensor $\mathcal{A}$ is defined as $unfold(\mathcal{A}, k) := \mathcal{A}_{(k)} \in R^{I_k \times (I_1...I_{k-1}I_{k+1}...I_n)}$. Accordingly, its inverse operator $fold$ can be defined as $fold(\mathcal{A}_{(k)}, k) := A$. Moreover, the k-rank of tensor $\mathcal{A}$, denoted by r_k, is defined as the rank of the matrix $\mathcal{A}_{(k)}$:

$$r_k = rank_k(\mathcal{A}) = rank(\mathcal{A}_{(k)}) \tag{1}$$

The Frobenius norm of a tensor is defined as $\|\mathcal{A}\|_F := (\sum_{i_1,i_2,...i_n} |a_{i_1}a_{i_2}...a_{i_n}|^2)^{\frac{1}{2}}$. Besides, denote the l_0 norm $\|\mathcal{A}\|_0$ as the number of non-zero entities in $\mathcal{A}$ and the l_1 norm $\|\mathcal{A}\|_1 := \sum_{i_1,i_2,...i_n} |a_{i_1}a_{i_2}...a_{i_n}|$ respectively. Then, we have $\|\mathcal{A}\|_F = \|A_{(k)}\|_F$, $\|\mathcal{A}\|_0 = \|A_{(k)}\|_0$ and $\|\mathcal{A}\|_1 = \|A_{(k)}\|_1$ for any $1 \leq k \leq n$.

A generalization of the product of two matrix is the product of a tensor and a matrix. The mode-k product of a tensor $\mathcal{A} \in R^{I_1 \times I_2 \times ... \times I_n}$ by a matrix $M \in R^{J_k \times I_k}$, denoted by $\mathcal{A} \times_k M$, is a tensor $\mathcal{B} \in R^{I_1 \times ... I_{k-1} \times J_k \times I_{k+1} \times ... \times I_n}$ with its elements given by

$$b_{i_1 \times ... i_{k-1} \times j_k \times i_{k+1} \times ... \times i_n} = \sum_{i_k} a_{i_1 \times ... i_{k-1} \times i_k \times i_{k+1} \times ... \times i_n} m_{j_k i_k} \tag{2}$$

The mode-k product can be expressed in tensor notation, or in terms of flattened matrix:

$$\mathcal{B} = \mathcal{A} \times_k M = fold(MA_{(k)}, k) \tag{3}$$

The notion of rank for tensors with order greater than two is subtle. There are two types of higher-order tensor decompositions, but neither of them has all the nice properties of the matrix SVD. The PARAFAC decomposition [26] represents the n-mode tensor $\mathcal{A} \in R^{I_1 \times I_2 \times ... \times I_n}$ as the outer product of vectors $\boldsymbol{u}_k^j \in R^{I_k}$ (Fig.2).

$$A = \sum_{j=1}^{R} \lambda_j \boldsymbol{u}_1^j \circ \boldsymbol{u}_2^j \circ ... \circ \boldsymbol{u}_n^j \tag{4}$$

where $\boldsymbol{u}_k^j$ are unit length vectors. Under mild conditions, the rank-R decomposition is essentially unique [26]. The *rank* of a n-mode tensor $\mathcal{A}$, is the minimal number of R, indicating the optimal rank-R decomposition. It is a natural extension of the matrix rank-R decomposition, but it does not compute the orthonormal subspace associated with each mode.

The Tucker decomposition, in the other hand, does not reveal the rank of the tensor, but it naturally generalizes the orthonormal subspaces corresponding to the left/right singular matrix computed by the matrix SVD [23]. The n-mode tensor $\mathcal{A} \in R^{I_1 \times I_2 \times ... \times I_n}$ can be decomposed as

$$\mathcal{A} = \mathcal{Z} \times_1 U_1 \times_2 U_2, ... \times_k U_k... \times_n U_n \tag{5}$$
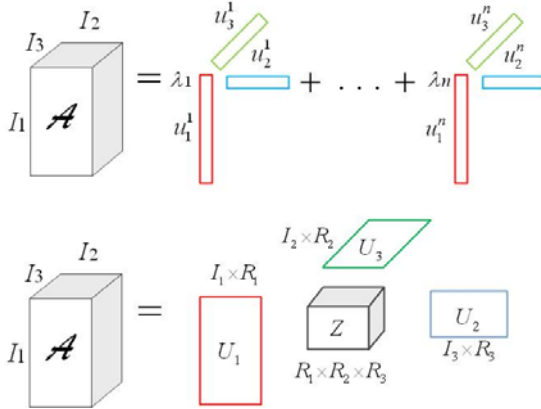
Fig. 2. Comparison of different decomposition for 3D tensor; Top: Rank-R decomposition. Bottom: Rank-$(R_1, R_2, \ldots R_N)$.

where $U_i \in R^{I_i \times R_i}$ are n orthogonal matrix. U_i spans the R_i dimensional subspace of the original R^{I_i} space, with its orthonormal columns as the basis. U_i accounts for the implicit factor of the ith-mode dimension of tensor $\mathcal{A}$. $\mathcal{Z}$ is the (dense) core tensor associating each of the n subspace (Fig.2).

4   Optimum Rank-$(R_1, R_2, \ldots R_N)$ Tensor Decomposition

4.1   The Model

To begin with, we give a brief introduction to the best Rank-$(R_1, R_2, \ldots R_N)$ approximation (decomposition) problem in [17, 20, 23]. Consider a real n-mode tensor $\mathcal{A} \in R^{I_1 \times I_2 \times \ldots \times I_n}$, the best rank-$(R_1, R_2, \ldots R_N)$ approximation is to find a tensor $\tilde{\mathcal{A}} \in R^{I_1 \times I_2 \times \ldots \times I_n}$ with pre-specified $rank_k(\tilde{A}) = R_k$, that minimizes the least-squares cost function:

$$\min_{\tilde{\mathcal{A}}} \quad f(\tilde{\mathcal{A}}) = \|\mathcal{A} - \tilde{\mathcal{A}}\|_F^2$$
$$s.t. \quad rank_i(\tilde{\mathcal{A}}) = R_i \qquad \forall i \tag{6}$$

The n-rank conditions imply that $\tilde{\mathcal{A}}$ should have the Tucker decomposition as (5): $\tilde{\mathcal{A}} = \mathcal{Z} \times_1 U_1 \times_2 U_2, \ldots \times_k U_k \ldots \times_n U_n$. The decomposition is discussed in [23] and Higher Order Orthogonal Iteration (HOOI) has been proposed to solve the problem.

HOOI requires strong prior knowledge of the tensor $\mathcal{A} \in R^{I_1 \times I_2 \times \ldots \times I_n}$, namely $rank_i(A) = R_i$, to find the (local) minimum solution. However, for visual data in real applications (e.g. a video clip or CT data), such prior knowledge is hardly available. Problem arises that if only weak prior knowledge is known (e.g. the

configuration of the tensor data), can one design a method that could automatically find the optimal n-rank condition of the given tensor $\mathcal{A}$. To simplify the problem, we consider a ideal model that the corruption is produced by additive irregular patterns $\mathcal{S}$.

$$\mathcal{A} = \mathcal{L} + \mathcal{S} \tag{7}$$

where $\mathcal{A}$, $\mathcal{L}$ and $\mathcal{S}$ are n-mode tensors with identical size in each mode. $\mathcal{A}$ is the observed data tensor. $\mathcal{L}$ and $\mathcal{S}$ represent the correspondent structured part and irregular part, respectively.

The underlining assumption of (7) is that the tensor data $\mathcal{A}$ is generated by a highly structured tensor $\mathcal{L}$, and then corrupted by an additive irregular patterns $\mathcal{S}$. One straightforward assumption may be that the n-rank of $\mathcal{L}$ should be small and the corruption $\mathcal{S}$ is bounded, leading to the formulation:

$$
\begin{aligned}
\min_{\mathcal{L}} \quad & \sum_i \lambda_i \ rank_i(\mathcal{L}) \\
s.t. \quad & \|\mathcal{L} - \mathcal{A}\|_F^2 \leq \varepsilon^2
\end{aligned}
\tag{8}
$$

where $U_i \in R^{I_i \times rank_i(\mathcal{L})}$. Intuitively, the weights λ_i indicates the preference towards different "unfold" operation, i.e. the configuration of the tensor. For example, we would prefer to explain the tensor representation of a video as the collection of frames.

(8) imposes constraints on the least square errors, suggesting that the corruption of the irregular patterns $\mathcal{S}$ is bounded. The constraint could be the case in certain situations. However, the irregular patterns in real world visual data is unknown and unbounded in general. A reasonable observation is that the irregular patterns $\mathcal{S}$ usually occupy only a small portion of the data. Therefore, we could impose l_0 norm penalization on $\mathcal{S}$ and form the problem as follows:

$$
\begin{aligned}
\min_{\mathcal{L},\mathcal{S}} \quad & \sum_i \lambda_i \ rank_i(\mathcal{L}) + \eta \|\mathcal{S}\|_0 \\
s.t. \quad & \|\mathcal{L} + \mathcal{S} - \mathcal{A}\|_F^2 \leq \varepsilon^2
\end{aligned}
\tag{9}
$$

The constant η balances between the low-dimensional structure and sparse irregularity. In addition, it is easy to check that (7) is a special case of (9) if we force $\mathcal{S} = 0$. Thus, we will focus on problem (9) in the rest of the paper.

When the optimal $\mathcal{L}$ is achieved, similar to the Tucker Decomposition, the core tensor $\mathcal{Z}$ can be computed by [23]

$$\mathcal{Z} = \mathcal{L} \times_1 U_1^T \times_2 U_2^T ... \times_n U_n^T \tag{10}$$

where U_i is the left singular matrix of $\mathcal{L}_i$. Accordingly, we can get the rank-$(R_1, R_2,\dots R_N)$ decomposition of $\mathcal{L} = \mathcal{Z} \times_1 U_1 \times_2 U_2, ... \times_i U_i ... \times_n U_n$. We call the correspondent decomposition in (11)

$$\mathcal{A} \sim \mathcal{Z} \times_1 U_1 \times_2 U_2, ... \times_i U_i ... \times_n U_n \tag{11}$$

to be **the optimal rank-$(R_1, R_2, \ldots R_N)$ decomposition of tensor** $\mathcal{A}$ under the sense of l_1 norm . The term "**optimal**" means that the model could automatically exploit the low-dimensional structure of the n-mode tensor $\mathcal{A}$, finding optimal dimension and basis for each mode and separating the sparse irregular patterns. The unknown support of the errors makes the problem more difficult than the tensor completion problem that has been recently much studied [10]. In the next section, we discuss the solution toward the optimization problem and propose the rank sparsity tensor decomposition (RSTD) algorithm.

4.2   Simplified Formulation

Equation (9) provides a promise for simultaneously exploring the low-dimensional structure and separating the irregular patterns of given tensor data $\mathcal{A} \in R^{I_1 \times I_2 \times \ldots \times I_n}$. However, (9) as the combination of two NP hard problem (matrix rank and l_0 norm), is highly nonconvex optimization. Given the fact that the trace norm $\|\mathcal{L}_{(i)}\|_{tr}$ and l_1 norm $\|\mathcal{S}\|_1$ are the tightest convex approximation of $rank_i(L)$ and $\|\mathcal{S}\|_0$ respectively, one can relax $rank_i(\mathcal{L})$ and $\|\mathcal{S}\|_0$ by $\|\mathcal{L}_{(i)}\|_{tr}$ and $\|\mathcal{S}\|_1$. Therefore, we could obtain a tractable optimization problem:

$$\min_{\mathcal{L}, \mathcal{S}} \quad \sum_i \lambda_i \|\mathcal{L}_{(i)}\|_{tr} + \eta \|\mathcal{S}\|_1$$
$$s.t. \quad \|\mathcal{L} + \mathcal{S} - \mathcal{A}\|_F^2 \leq \varepsilon^2 \tag{12}$$

where the trace norm, or the nuclear norm of matrix $\mathcal{L}_{(i)}$ is defined as the sum of its singular values σ_j, i.e. $\|\mathcal{L}_{(i)}\|_{tr} = \sum_j \sigma_j(\mathcal{L}_{(i)})$. If $rank_i\mathcal{L} \ll I_i$ and $\|\mathcal{S}\|_0 \ll \Pi_{i=1}^n I_i$, i.e. tensor $\mathcal{L}$ is highly structured and tensor $\mathcal{S}$ is sparse enough, under rather mild conditions, the approximation can be highly accurate [8, 15]. Empirically, for general visual data with high redundancy, the approximation produces good results.

   Problem (12) is still hard to solve due to the interdependent trace norm and l_1 norm constraint. To simplify the problem, we introduce additional auxiliary matrix $M_i = \mathcal{L}_{(i)}$ and $N_i = \mathcal{S}_{(i)}$. Thus, we obtain the equivalent formulation:

$$\min_{\mathcal{L}, \mathcal{S}, M_i, N_i} \quad \frac{1}{n} \sum_{i=1}^n \lambda_i \|M_i\|_{tr} + \frac{\eta}{n} \sum_{i=1}^n \|N_i\|_1$$
$$s.t. \quad M_i = \mathcal{L}_{(i)} \qquad N_i = \mathcal{S}_{(i)} \qquad \forall i$$
$$\|M_i + N_i - \mathcal{A}_{(i)}\|_F^2 \leq \varepsilon^2 \qquad \forall i \tag{13}$$

In (13), the constrains $M_i = \mathcal{L}_{(i)}$ and $N_i = \mathcal{S}_{(i)}$ still enforce the consistency of all M_i and N_i. Thus, we further relax the equality constrains $M_i = \mathcal{L}_{(i)}$ and $N_i = \mathcal{S}_{(i)}$ by $\|M_i - \mathcal{L}_{(i)}\|_F \leq \varepsilon_1$ and $\|N_i - \mathcal{S}_{(i)}\|_F \leq \varepsilon_2$. Then, it is easy to check that the dense noise term by $\|M_i + N_i - \mathcal{A}_{(i)}\|_F \leq \varepsilon_3$ corresponds to the stable Principle Component Pursuit(sPCP) in the matrix case [27]. Then, we get the relaxed form:

$$\min_{\mathcal{L},\mathcal{S},M_i,N_i} \quad \frac{1}{n}\sum_{i=1}^{n}\lambda_i\|M_i\|_{tr} + \frac{\eta}{n}\sum_{i=1}^{n}\|N_i\|_1$$

$$s.t. \quad \|M_i - \mathcal{L}_{(i)}\|_F^2 \le \varepsilon_1^2 \qquad \|N_i - \mathcal{S}_{(i)}\|_F^2 \le \varepsilon_2^2 \qquad \forall i$$

$$\|M_i + N_i - \mathcal{A}_{(i)}\|_F^2 \le \varepsilon_3^2 \qquad \qquad \forall i \tag{14}$$

For certain α_i, β_i and γ_i, (14) can be converted to its equivalent form by Lagrange multiplier.

$$\min_{\mathcal{L},\mathcal{S},M_i,N_i} \quad F(\mathcal{L},\mathcal{S},M_i,N_i) = \frac{1}{2n}\sum_{i=1}^{n}\alpha_i\|M_i - \mathcal{L}_{(i)}\|_F^2 + \frac{1}{2n}\sum_{i=1}^{n}\beta_i\|N_i - \mathcal{S}_{(i)}\|_F^2$$

$$+ \frac{1}{2n}\sum_{i=1}^{n}\gamma_i\|M_i + N_i - \mathcal{A}_{(i)}\|_F^2 + \frac{1}{n}\sum_{i=1}^{n}\lambda_i\|M_i\|_{tr} + \frac{\eta}{n}\sum_{i=1}^{n}\|N_i\|_1 \tag{15}$$

Intuitively, the weights α_i, β_i and γ_i indicate the preference towards different "unfold" operation similar to λ_i. The optimization problem in (15) is convex but nondifferentiable. Next, we show how to solve this problem.

4.3   The Proposed Algorithm

We propose to employ the alternating direction method (ADM) for the optimization (15), leading to an block coordinate descent (BCD) algorithm. The core idea of the BCD is to optimize a group of variables while fixing the other groups. The variables in the optimization are $N_1,..., N_n$, $M_1,..., M_n$, $\mathcal{L}$, $\mathcal{S}$, which can be divided into $2n + 2$ blocks. To achieve the optimal solution, we estimate N_i, M_i, $\mathcal{L}$ and $\mathcal{S}$ sequentially, followed by certain refinement in each iteration. For clarity, we first define the "shrinkage" operator $D_\tau(x)$ with $\tau > 0$ by

$$D_\tau(x) = \begin{cases} x - \tau & \text{if } x > \tau \\ \tau - x & \text{if } x < -\tau \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

The operator can be extended to the matrix or tensor case by performing the shrinkage operator towards each element. Then, we introduce the solution towards each subproblem.

Computing N_i: The optimal N_i with all other variables fixed is the solution to the following subproblem

$$\min_{N_i} \quad \frac{\beta_i}{2}\|N_i - \mathcal{S}_{(i)}\|_F^2 + \frac{\gamma_i}{2}\|N_i + M_i - \mathcal{A}_{(i)}\|_F^2 + \eta\|N_i\|_1 \tag{17}$$

By the well-known l_1 minimization [28], the global minimum of the optimization problem in (17) is given by

$$N_i^* = D_{\frac{\eta}{\beta_i+\gamma_i}}\left(\frac{\beta_i S_{(i)} + \gamma_i(A_{(i)} - M_i)}{\beta_i + \gamma_i}\right) \tag{18}$$

where D_τ is the "shrinkage" operation.

Computing M_i: The optimal M_i with all other variables fixed is the solution to the following subproblem:

$$\min_{M_i} \quad \frac{\alpha_i}{2}\|M_i - \mathcal{L}_{(i)}\|_F^2 + \frac{\gamma_i}{2}\|M_i + N_i - \mathcal{A}_{(i)}\|_F^2 + \lambda_i\|M_i\|_{tr} \qquad (19)$$

As shown in [9], the global minimum of the optimization problem in (19) is given by

$$M_i^* = U_i D_{\frac{\lambda_i}{\alpha_i+\gamma_i}}(\Lambda)V_i^T \qquad (20)$$

where $U_i\Lambda V_i^T$ is the singular value decomposition given by

$$U_i\Lambda V_i^T = \frac{\alpha_i L_{(i)} + \gamma_i(A_{(i)} - N_i)}{\alpha_i + \gamma_i} \qquad (21)$$

Computing $\mathcal{S}_i$: The optimal $\mathcal{S}$ with all other variables fixed is the solution to the following subproblem

$$\min_{\mathcal{S}} \quad \frac{1}{2}\sum_{i=1}^{n}\beta_i\|N_i - \mathcal{S}_{(i)}\|_F^2 \qquad (22)$$

It is easy to show that the solution to (22) is given by

$$\hat{\mathcal{S}}^* = \frac{\sum_{i=1}^{n}\beta_i fold(N_i, i)}{\sum_{i=1}^{n}\beta_i} \qquad (23)$$

Computing $\mathcal{L}_i$: The optimal $\mathcal{L}$ with all other variables fixed is the solution to the following subproblem

$$\min_{\mathcal{L}} \quad \frac{1}{2}\sum_{i=1}^{n}\alpha_i\|M_i - \mathcal{L}_{(i)}\|_F^2 \qquad (24)$$

Similar to (22), the solution to (24) is given by

$$\hat{\mathcal{L}}^* = \frac{\sum_{i=1}^{n}\alpha_i fold(M_i, i)}{\sum_{i=1}^{n}\alpha_i} \qquad (25)$$

We choose the difference of L and S in successive iterations against a certain tolerance as the stopping criterion. N_i^*, M_i^*, $\mathcal{L}^*$ and $\mathcal{S}^*$ are estimated iteratively until the convergence. We call the proposed algorithm Rank Sparsity Tensor Decomposition (RSTD). The pseudo-code of RSTD is summarized in Algorithm 1. We can further show that accelerated BCD for RSTD is guaranteed to reach the global optimum of (15), since the first three terms in (15) are differentiable and the last two terms are separable [29].

Algorithm 1. RSTD for Optimum Rank-$(R_1...R_N)$ Tensor Approximation

---

Input : n-mode tensor $\mathcal{A}$

Parameters : $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\lambda}$, $\boldsymbol{\eta}$

Output : n-mode tensor $\mathcal{L}$, $\mathcal{S}$, $\mathcal{Z}$, matrix U_i from 1 to n

1. Set $\mathcal{L}^{(0)} = \mathcal{A}$, $\mathcal{S}^{(0)} = 0$, $M_i = \mathcal{L}_{(i)}$, $N_i = 0$, $k = 1$, $t^{(0)} = 1$

2. ***while*** no convergence

3.   ***for*** $i = 1$ to n

4. $N_i^* = D_{\frac{\eta}{\beta_i + \gamma_i}} \left(\frac{\beta_i S_{(i)} + \gamma_i (A_{(i)} - M_i)}{\beta_i + \gamma_i} \right)$

5. $M_i^* = U_i D_{\frac{\lambda_i}{\alpha_i + \gamma_i}}(\Lambda) V_i^T \quad where \quad U_i \Lambda V_i^T = \frac{\alpha_i L_{(i)} + \gamma_i (A_{(i)} - N_i)}{\alpha_i + \gamma_i}$

6.   ***end for***

7. $\mathcal{S}^* = \frac{\sum_{i=1}^{n} \beta_i fold(N_i, i)}{\sum_{i=1}^{n} \beta_i}$

8. $\mathcal{L}^* = \frac{\sum_{i=1}^{n} \alpha_i fold(M_i, i)}{\sum_{i=1}^{n} \alpha_i}$

9. ***end while***

10. $\mathcal{Z} = \mathcal{L} \times_1 U_1^T \times_2 U_2^T ... \times_n U_n^T$

---

5   Experiments

5.1   Implementation Details

In the implementation, we adopt the Lanczos bidiagonalization algorithm with partial reorthogonalization [30] to obtain a few singular values and vectors during each iteration. The prediction rule for the dimension of the principal singular space is the same as [15]. A major challenge of our method is the selection of parameters. As the redundancy usually grows with the dimension, we simply set $\alpha = \beta = \gamma = [I_1/I_{max}, I_2/I_{max}, ..., I_n/I_{max}]^T$ for all experiments, where $I_{max} = max\{I_i\}$. Similarly, we set $\lambda = [sv_1/sv_{max}, sv_2/sv_{max}, ..., sv_n/sv_{max}]$, where sv_i is the 95% singular value of $A_{(i)}$ and $sv_{max} = max\{sv_i\}$. Finally, we choose $\eta = 1/\sqrt{I_{max}}$ as suggested in [15]. During the experiments, we observe that for most of the samples our implementation is able to converge in less than 100 iterations with a tolerance equal to 10^{-6}.

5.2   Image Restoration

As shown in Fig.1, our algorithm can be used to separate unbounded sparse noise in visual data. One straightforward application of our method is the image restoration. However, we must point out that our algorithm assumes the tensor be well structured. This assumption would not be reasonable for some natural images, but it should be applicable for many visual data such as structured object (e.g. the facade), CT/fMRI data, multi-spectral image, etc. Therefore, we apply our algorithm on a set of MRI data including 181 brain images, which is also used in [10]. We add different percent of unbounded random noise to the image and demonstrate some of the results produced by our method in Fig.3.
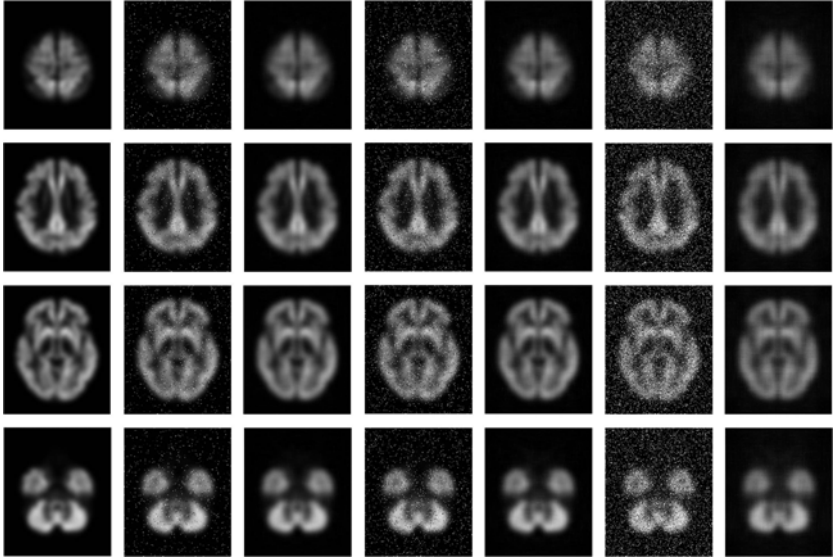
Fig. 3. Demonstration of the results produced by our algorithm (from left to right): original image, 5% corrupted image, recovered image from 5% corrupted noise; 10% corrupted image, recovered image from 10% noise; 30% corrupted image, recovered image from 30% corrupted noise

Our algorithm is able to find the structured data and separate the noise without the location of the corruption (about 30 percent of the data). Table.1 further provides quantitative results of our algorithm.

Table 1. Error correction for the brain MRI data

Percentage of Corruption	5%	10%	15%	20%	30%
Average PSNR (dB)	37.41	34.41	30.70	28.95	20.35

5.3   Background Subtraction

Another possible application of our algorithm is the background subtraction problem. Background substraction establishes a background model (the structured part) and segments the foreground object (sparse irregular pattern). For most of the video clip, redundancy is abundant. We conduct experiments on several video clips. Fig.4 demonstrates some of our results in one of the highly dynamic scenes. The results are comparable to the-state-of-art background subtraction algorithms.

Fig. 4. Background subtraction by our method (no filter is performed on the results)

5.4 Face Representation and Recognition

By the TensorFace in [1], we test our algorithm on the CMU PIE dataset, which contains 68 person under various viewpoints, expressions and illuminations. We use the same data set as [16] with the resolution at 64×64. For simplicity, only the five near frontal view under 21 different illuminations (105 images) of one person are used as training and the rest (65 images including the expressions) is for testing. Thus, we get a $5 \times 21 \times 68 \times 64 \times 64$ tensor. Then, the method learns a $5 \times 5 \times 68 \times 23 \times 22$ core tensor. Fig.5 compares the reconstructed faces with the original ones. We can see that the shadows have been removed. As a consequence, we achieve a competitive 94.3% accuracy by the recognition method in [19].

Fig. 5. Original Face (left) v.s. Reconstructed Face (right): the shadow caused by different illumination has been removed

6   Conclusion and Future Work

In this paper, we propose the optimal rank-$(R_1, R_2, ...R_n)$ tensor decomposition model. The model could automatically explore the low-dimensional structure of the tensor data seeking optimal dimension and basis for each mode and separating the irregular patterns. We are currently working on parameters and the

optimization method of our model (e.g the proximal gradient), which may lead to better efficiency. We would also like to further explore additional applications and to investigate the theoretic side of our method in the future work.

References

1. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: Tensorfaces. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 447–460. Springer, Heidelberg (2002)
2. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cognitive Neuroscience 3, 71–86 (1991)
3. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 711–720 (1997)
4. He, X., Niyogi, P.: Locality preserving projections. In: Neural Information Processing Systems, NIPS (2004)
5. Lewis, A., Knowles, G.: Image compression using the 2-d wavelet transform. IEEE Transactions on Image Processing 1, 244–250 (1992)
6. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 1254–1259 (1998)
7. Sheikh, Y., Shah, M.: Bayesian modeling of dynamic scenes for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 1778–1792 (2005)
8. Candès, E.J., Recht, B.: Exact matrix completion via convex optimization. Found. of Comput. Math. 9, 717–772 (2008)
9. Cai, J.F., Candes, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion (2008) (preprint)
10. Liu, J., Musialski, P., Wonka, P., Ye, J.: Tensor completion for estimating missing values in visual data. In: International Conference on Computer Vision, ICCV (2009)
11. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Transactions on Image Processing 15, 3736–3745 (2006)
12. Wright, J., Ma, Y.: Dense error correction via ℓ^1-minimization (2008) (preprint), http://perception.csl.uiuc.edu/~jnwright/
13. Wright, J., Yang, A., Ganesh, A., Sastry, S., Ma, Y.: Robust face recognition via sparse representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 31, 210–227 (2009)
14. Wright, J., Ganesh, A., Rao, S., Peng, Y., Ma, Y.: Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In: Neural Information Processing Systems, NIPS (2009)
15. Cands, E., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? (2009) (preprint)
16. He, X., Cai, D., Niyogi, P.: Tensor subspace analysis. In: Neural Information Processing Systems, NIPS (2005)
17. Xu, D., Yan, S., Zhang, L., Zhang, H.J., Liu, Z., Shum, H.Y.: Concurrent subspaces analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 203–208 (2005)

18. Vasilescu, M., Terzopoulos, D.: Multilinear subspace analysis of image ensembles. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. II – 93–II –9 (2003)
19. Vasilescu, M., Terzopoulos, D.: Multilinear projection for appearance-based recognition in the tensor framework. In: International Conference on Computer Vision (ICCV), pp. 1–8 (2007)
20. Wang, H., Ahuja, N.: A tensor approximation approach to? dimensionality reduction. International Journal of Computer Vision 76, 217–229 (2008)
21. Wang, H., Ahuja, N.: Facial expression decomposition. In: International Conference on Computer Vision (ICCV), vol. 2, pp. 958–965 (2003)
22. Jia, K., Gong, S.: Multi-modal tensor face for simultaneous super-resolution and recognition. In: International Conference on Computer Vision (ICCV), vol. 2, pp. 1683–1690 (2005)
23. Lathauwer, L.D., Moor, B.D., Vandewalle, J.: On the best rank-1 and rank-(r1,r2,.,rn) approximation of higher-order tensors. SIAM J. Matrix Anal. Appl. 21, 1324–1342 (2000)
24. Shashua, A., Levin, A.: Linear image coding for regression and classification using the tensor-rank principle. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. I–42–I–49 (2001)
25. Shashua, A., Zass, R., Hazan, T.: Multi-way clustering using super-symmetric non-negative tensor factorization. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 595–608. Springer, Heidelberg (2006)
26. Kruskal, J.B.: Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. Linear Algebra and its Applications 18, 95–138 (1977)
27. Zhou, Z., Li, X., Wright, J., Candes, E., Ma, Y.: Stable principal component pursuit. In: International Symposium on Information Theory (2010)
28. Hale, E.T., Yin, W., Zhang, Y.: Fixed-point continuation for l1-minimization: Methodology and convergence. SIAM Journal on Optim. 19, 1107–1130 (2008)
29. Tseng, P.: Convergence of a block coordinate descent method for nondifferentiable minimization. J. Optim. Theory Appl. 109, 475–494 (2001)
30. Simon, H.D.: The lanczos algorithm with partial reorthogonalization. Math. Comp. 42, 115–142 (1984)