平成３０年度

博士後期学位論文

Study on Tensor Factorization and Its Applications to Computer Vision

テンソル因子分解に関する研究及びコンピュータービジョンへの応用

貴 麗華

埼玉工業大学大学院 博士後期課程

工学研究科 電子工学専攻

指導教員 曹 建庭 教授

Abstract

Tensor, also called multiway array, is a generalization of vector, matrix to the higher-dimensional cases. In many real-world applications, the data recorded from multiple conditions and multidimensional structured data is frequently occurred, which is very suitable to be represented efficiently by using tensors instead of matrices. To process such multi-dimensional data, tensor decomposition/factorization and multilinear algebra is the fundamental tools, which is still under development. Tensor factorization can capture the multilinear latent factors effectively and take the structure information into account explicitly. The theory and algorithms of tensor factorization have been widely studied during the past decade, and demonstrated to be successful for feature extraction, dictionary learning, dimension reduction, efficient algorithm, compressive representation and large scale data analysis. Hence, it has been applied to many real-world applications, such as image/video recognition/classification, social network analysis, image completion, speech processing, natural language processing and brain signal processing.

In this study, we focus on the probability formulate of tensor factorization and Bayesian inference for learning algorithm. The Bayesian tensor factorization has several advantages. The first one is that the rank of tensor can be inferred automatically from the given data, which thus avoids the time-consuming procedure of tuning parameter. Secondly, the uncertainty information of latent factors is considered in our model and thus is more robust to prevent the overfitting problem. To handling the missing value problem, the Bayesian tensor factorization methods are extended to incomplete tensor data and the corresponding algorithms are developed.

Based on these Bayesian tensor factorization methods, we study how it can be applied to solve several real-world problems which mainly focuses

on the denoising and completion of image, video and MRI data. We investigate the non-local tensor denoising framework for multi-dimensional data by using 3D tensor patches instead of 2D patches, which is more useful for image/video and MRI data denoising. By using Bayesian tensor factorization for low-rank approximation of similar patches, our method enables us to preserve the spatial and time structure and also automatically find the noise variance parameter, which is thus more practical as compared to the traditional denoising methods. Furthermore, we introduce the Bayesian Tucker decomposition method, which is able to predict the missing values by using partially observed tensor data. Our method can effectively find the optimal multilinear ranks given a specific missing ratio. The experimental results on image/video denoising and image/video/MRI completion demonstrate the effectiveness of our methods in terms of flexibility and performance, as compared to other tensor-based denoising, and tensor-based completion methods.

The organization of my thesis is as follows: Chapter 1 introduces the basic notations and operations of tensor and multilinear algebra. The most popular tensor factorization models are also presented; Chapter 2 presents a tensor denoising framework by using Bayesian CP factorization method. The formulation of Bayesian CP factorization together with the detailed algorithm are described. The experiments on image/video and MRI denoising are performed with comparisons with other related methods; Chapter 3 presents a tensor completion framework by using Bayesian Tucker model. The forumulation of Bayesian Tucker decompositon togher with the detailed algorithm are described. The experiments on image/video and MRI completion, i. e., prediction of missing values by using only a small portion of data, are performed and compared with other related methods; Chapter 4 summarizes the previous studies and presents some future trends and directions of tensor related methods in machine learning.

概　要

テンソルは、高次元配列とも呼ばれ、ベクトル、行列を高次の場合に一般化したものである。実際の多くのアプリケーションでは、複数の条件と高次元構造化データから記録されたデータが頻繁に発生し、行列の代わりにテンソルは効率的良く表現するのに適している。このような高次元データを処理するために、テンソル分解/因数分解と多重線形代数は基本的なツールであり、これからも沢山の方法を開発している。テンソル因子分解は、多重線形潜在因子を効果的に捕捉し、構造情報を明示的に考慮に入れることができる。テンソル因子分解の理論とアルゴリズムは、過去10年間に広く研究され、特徴抽出、辞書学習、次元削減、効率的なアルゴリズム、圧縮表現、および大規模なデータ分析に成功したことが示された。従って、画像/ビデオ認識/分類、ソーシャルネットワーク分析、画像補完、音声処理、自然言語処理、脳信号処理などの多くの現実世界のアプリケーションに適用されて来た。

本研究では、テンソル分解の確率定式化とベイズアン推論による学習アルゴリズムに焦点に当てる。ベジジンソル分解にはいくつかの利点がある。まず、与えられたデータからテンソルのランクを自動的に推測することができ、チューニングパラメータの時間的な手続きを回避することができる。次に、潜在因子の不確かさの情報が我々のモデルで考慮されているため、オーバーフィッシングの問題を防ぐためによりロベストである。欠損値の問題を処理するために、ベイズテンソル分解法が不完全テンソルデータに拡張され、対応するアルゴリズムを開発している。

応用面としてベイジアンテンソル分解法に基づいて、主に画像、ビデオとMRIデータの補完やノイズ除去に焦点に当て、いくつかの現実世界の問題を解決する方法を研究する。2Dパッチの代わりに3Dテンソルパッチを用いて、高次元データの非局所テンソルノイズ除去フレームワークを調べる。これは、画像/ビデオおよびMRI

データノイズ除去にさらに役たつ。似てるパッチの低ランク近似にベジアンアンテンソル分解を用いることにより、空間および時間構造を保存し、ノイズ分散パラメータを自動的に見つけることを可能にする。従来の雑音除去方法と比較して、より実用的である。さらに、部分的に観測されたテンソルデータを用いて欠損値を予測できるベイジアンタッカー分解法を紹介する。提案した方法は、特定の欠損率が与えられた場合、最適な多重線形ランクを効果的に見つけることができる。画像/ビデオノイズ除去および画像/ビデオ/ MRI補完に関する実験結果は、他のテンソルベースのノイズ除去法およびテンソルベースの補完法と比較して検証を行う。

本論文では，以下に示す全4章から構成されている。第1章では、テンソルと多重線形代数の基本的な表記法と操作について述べ、最も一般的なテンソル因子分解モデルも提示されている。第2章では、ベイジアンCP分解法を用いたテンソルノイズ除去フレームワークを紹介し、提案したアルゴリズムと一緒に、ベイジアンCP因子分解について述べる。画像/ビデオおよびMRIノイズ除去に関する実験は、他の関連する方法との比較して検証を行う。第3章では、ベイジアンタッカーモデルを用いてテンソル補完フレームワークを提案する。アルゴリズムを用いたベイジアンタッカー分解法を述べる。また、画像/ビデオおよびMRI補完に関する実験は、データのわずかな部分だけを使用して欠損値の予測を実行し、他の関連する方法と比較して検証を行う。第4章では、前述した研究をまとめ、機械学習におけるテンソル関連の方法の将来の動向と方向性を述べる。

Contents

List of Figures

List of Tables

Chapter 1

Multilinear Algebra and Tensor Decomposition

1.1 Background

Tensor is a multidimensional array which is a generalization of vectors and matrices to higher dimensions. First-order tensor is a vector, second-order tensor is a matrix, and third and higher order tensor are called a tensor. The tensors of first, second, third-order are shown in Fig. 1.1.

Tensor decompositions originated from Hitchcock (Hitchcock, 1927)(Hitchcock, 1928). Under the work of Tucker (Tucker, 1963b) (Tucker, 1964) (Tucker, 1966), Carroll and Chang (Carroll and Chang, 1970), Harshman (Harshman, 1970), Appellof and Davidson (Appellof and Davidson, 1981), the tensor theory and tensor decompositions (factorizations) algorithms have been successfully applied to various fields, the examples include signal processing, computer vision and etc.

1.2 Notations

Tensor is a multidimensional array, the order of a tensor is the number of dimensions (Kolda and Bader, 2009). Tensor of order one (vector) is denoted by boldface lowercase letters, e.g., $\mathbf{a}$, the i-th element of a one-order tensor

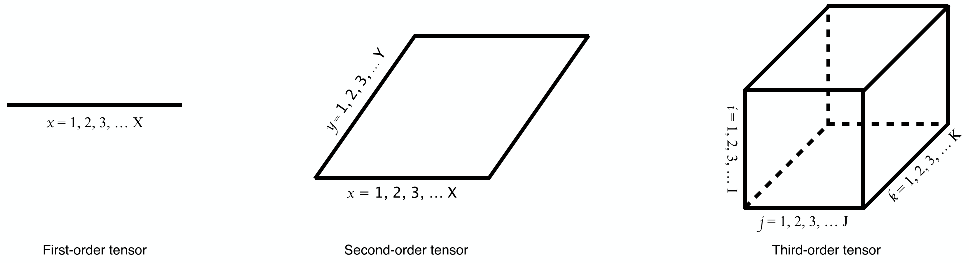

FIGURE 1.1: First, second, thrid-order tensor

is denoted by a_i. Tensor of order two (matrix) is denoted by boldface capital letters, e.g., **A**, the (i, j) element of a two-order tensor is denoted by a_{ij}. Tensor of order three or higher (higher-order tensor) is denoted by boldface Euler script letters, e.g., $\boldsymbol{\mathcal{X}}$, the (i, j, k) element of a three-order tensor is denoted by x_{ijk}. Indices typically range from 1 to their capital version, e.g., $i = 1, ..., I$.

1.3 Multilinear Algebra

The *Frobeniusnorm* of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times .... \times I_N}$, is the square root of the sum of the square of all elements (1.1)

$$\|\boldsymbol{\mathcal{X}}\|_F = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} x^2_{i_1 i_2 \cdots i_N}}. \tag{1.1}$$

The *inner product* of two same sized tensors $\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is defined by

$$\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} ... \sum_{i_N=1}^{I_N} x_{i_1 i_2 \cdots i_N} y_{i_1 i_2 ... i_N}. \tag{1.2}$$

It follows immediately that $\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}} \rangle = \|\boldsymbol{\mathcal{X}}\|_F^2$.

The *Hadamard product* is an elementwise product between two tensors that must be same sizes. Given $\mathbf{A} \in \mathbb{R}^{I \times J}$ and $\mathbf{B} \in \mathbb{R}^{I \times J}$, the Hadamard product is denoted by $\mathbf{A} \circledast \mathbf{B} \in \mathbb{R}^{I \times J}$, which is computed by

$$\mathbf{A} \circledast \mathbf{B} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2J}b_{2J} \\ a_{31}b_{31} & a_{32}b_{32} & \cdots & a_{3J}b_{13} \\ \cdots & \cdots & & \cdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \cdots & a_{IJ}b_{IJ} \end{bmatrix}. \tag{1.3}$$

The Hadamard product of $N \geq 3$ items is defined as

$$\circledast_{n=1}^{N} \mathbf{A}^{(n)} = \mathbf{A}^{(1)} \circledast \mathbf{A}^{(2)} \circledast \cdots \circledast \mathbf{A}^{(N)}. \tag{1.4}$$

The *Kronecker product* of matrices $\mathbf{A} \in \mathbb{R}^{I \times J}$ and $\mathbf{B} \in \mathbb{R}^{K \times L}$ becomes a matrix of size $IK \times JL$, denoted by $\mathbf{A} \otimes \mathbf{B}$ and computed by

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ a_{31}B & a_{32}B & \cdots & a_{3J}B \\ \cdots & \cdots & & \cdots \\ a_{I1}B & a_{I2}B & \cdots & a_{IJ}B \end{bmatrix}. \tag{1.5}$$

The *Khatri* $-$ *Rao product* of matrices $\mathbf{A} \in \mathbb{R}^{I \times K}$ and $\mathbf{B} \in \mathbb{R}^{J \times K}$ is a matrix of size $IJ \times K$, denoted by $\mathbf{A} \odot \mathbf{B}$. In particular, the Khatri-Rao product of $N \geq 3$ matrices in a reverse order is defined by

$$\bigodot_{n=1}^{N} \mathbf{A}^{(n)} = \mathbf{A}^{(N)} \odot \mathbf{A}^{(N-1)} \odot \cdots \odot \mathbf{A}^{(1)}. \tag{1.6}$$

The Khatri-Rao product of a group of matrices, except the nth matrix is denoted by $\mathbf{A}^{(\backslash n)}$ and computed by

$$\begin{aligned}\bigodot_{k=1,k\neq n}^{N} \mathbf{A}^{(k)} = \mathbf{A}^{(N)} \odot \cdots \odot \mathbf{A}^{(n+1)} \\ \odot \mathbf{A}^{(n-1)} \odot \cdots \odot \mathbf{A}^{(1)}.\end{aligned} \tag{1.7}$$

1.4 CP Decomposition

CANDECOMP/PARAFAC (CP) decomposition method is proposed by Carroll and Chang (Carroll and Chang, 1970) and PARAFAC (parallel factors) proposed by Harshman (Harshman, 1970). Usually, we refer to the CANDECOMP/PARAFAC decomposition as CP (Kiers, 2000). CP decomposition is to represent a tensor as a sum of rank-one tensors. For instance, given a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I\times J\times K}$, we wish to represent it by

$$\boldsymbol{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]. \tag{1.8}$$

The element-wise form of (1.8) is written as

$$\begin{aligned} x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}, \\ \forall\, i = 1,..., I, \forall\, j = 1, ..., J, \forall\, k = 1, ..., K. \end{aligned} \tag{1.9}$$

where $\mathbf{a}_r \in \mathbb{R}^I$, $\mathbf{b}_r \in \mathbb{R}^J$ and $\mathbf{c}_r \in \mathbb{R}^K$, $\forall r$ = 1,...,R. The rank of a tensor $\boldsymbol{\mathcal{X}}$, denoted R = rank($\boldsymbol{\mathcal{X}}$), is define as the smallest number of rank-one tensors that can exactly represent $\boldsymbol{\mathcal{X}}$. The scheme of CP decompositions is illustrated in Fig. 1.2.

1.5 Tucker Decomposition

The Tucker decomposition was proposed in 1963 (Tucker, 1963a), and refined in subsequent articles by Levin (Appellof and Davidson, 1981) and Tucker

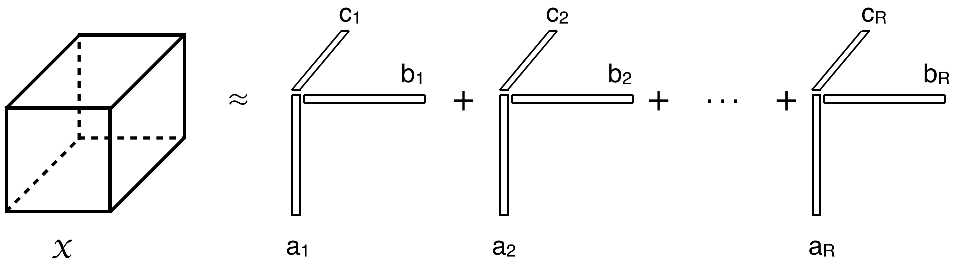

FIGURE 1.2: CP decomposition of a third-order tensor

(Tucker, 1964; Tucker, 1966). Tucker decomposition can be considered as an extension of PCA (Principal Components Analysis) to a high order tensor, which decomposes a tensor into a core tensor multiplied (or transformed) by several matrices along each mode. For instance, given a three-way tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, Tucker decomposition is written as

$$\begin{aligned} \mathcal{X} &= \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \\ &= \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr} \circ \mathbf{a_p} \circ \mathbf{b_q} \circ \mathbf{c_r} \\ &= [\![\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]. \end{aligned} \tag{1.10}$$

The element in tensor can thus be computed and represented by

$$\begin{gathered} x_{ijk} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr} a_{ip} b_{jq} c_{kr}, \\ \forall\, i = 1, ..., I, \forall\, j = 1, ..., J, \forall\, k = 1, ..., K. \end{gathered} \tag{1.11}$$

Here, $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$ and $\mathbf{C} \in \mathbb{R}^{K \times R}$ are the factor matrices (which are usually orthogonal) and can be considered as the principal components in each mode. Tensor $\mathcal{G} \in \mathbb{R}^{P \times Q \times \ R}$ is called the *core tensor* and its entries show the level of interaction between the different components. The last equality in (1.10) using the shorthand $[\![\mathcal{G}; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$ was introduced in (Kolda and Bader, 2009). The scheme of Tucker decompositions is illustrated in Fig. 1.3.

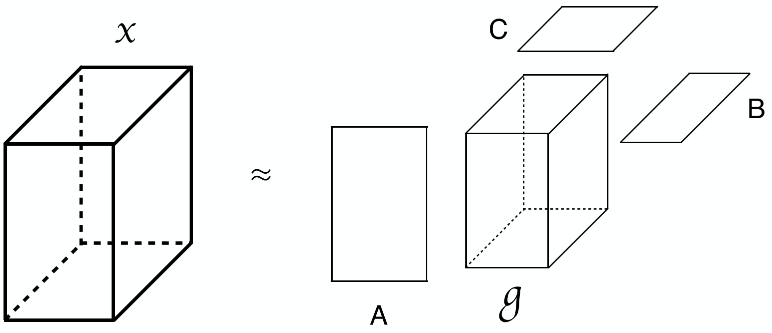

FIGURE 1.3: Tucker decomposition of a third-order tensor

Chapter 2

Non-Local Tensor Denoising Using Bayesian Low-rank Tensor Factorization on High-order Patches

Removing the noise from an image is vitally important in many real-world computer vision applications. One of the most effective method is block matching collaborative filtering, which employs low-rank approximation to the group of similar patches gathered by searching from the noisy image. However, the main drawback of this method is that the standard deviation of noises within the image is assumed to be known in advance, which is impossible for many real applications. In this chapter, we propose a non-local filtering method by using the low-rank tensor decomposition method. For tensor decomposition, we choose CP model as the underlying low-rank approximation. Since we assume the noise variance is unknown and need to be learned from data itself, we employ the Bayesian CP factorization that can learn CP-rank as well as noise variance solely from the observed noisy tensor data, The experimental results on image and MRI denoising demonstrate the superiorities of our method in terms of flexibility and performance, as compared to other tensor-based denoising methods.

2.1 Background

Image denoising is an important task in image processing field, many techniques try to solve this problem. Recently, non-local filtering techniques have attracted a lot of interest and demonstrated the superiority in terms of performance (Buades, Coll, and Morel, 2005; Wang and Zhang, 1999; Dabov et al., 2007; Rajwade, Rangarajan, and Banerjee, 2011). The key technique in image denoising is to infer the optimal bases from a group of similar patches. More specifically, for any reference patch, the bases can be learned from a set of patches selected within a specific distant range, which are similar to that patch. These image denoising methods have been also extended to video denoising, which enables the patches from adjacent video frames to be considered in gathering the similar patches. However, the existing methods are all based on 2D patches, thus is not effective for multiway data that is naturally represented as a tensor.

Multidimensional data is natural represent by tensor, compared to matrix tensor factorization can capture the multilinear latent factors effectively and take the structure information into account explicitly. The theory and algorithms of tensor factorization have been widely studied during the past decade and were successfully applied to many real-world applications, such as image completion (Gui, Zhao, and Cao, 2017; Yuan, Zhao, and Cao, 2017b; Liu et al., 2013; Yuan, Zhao, and Cao, 2017a; Yuan et al., 2018b; Zhao, Zhang, and Cichocki, 2015; Filipović and Jukić, 2015; Yuan et al., 2018c; Yuan et al., 2018a), signal processing (De Lathauwer and Castaing, 2008; Gui, Zhao, and Cao, 2016; Cichocki et al., 2015; Gui et al., 2017; Muti and Bourennane, 2005; De Lathauwer and De Moor, 1998), brain machine interface (BMI) (Liu et al., 2014; Mocks, 1988; Zhang et al., 2016), image classification (Shashua and

Levin, 2001), face recognition (Geng et al., 2011), machine learning (Zhao et al., 2016), etc.

The higher order singular value decomposition (HOSVD) is an extension of the matrix SVD technique to the multiway tensor (De Lathauwer, De Moor, and Vandewalle, 2000). Recently, the HOSVD has been successfully applied to image and video denoising (Rajwade, Rangarajan, and Banerjee, 2011) (Rajwade, Rangarajan, and Banerjee, 2013) (Zhang et al., 2015) as a multilinear transform basis. However, the limitations of HOSVD-based denoising are that the noise standard deviation must be known in advance, which results in difficulties in practical applications.

To solve this problem, we leverage the Bayesian approach to learn the noise variance from original data without using priori knowledge. In contrast to HOSVD model which is a Tucker tensor decomposition model, we apply the CP decomposition which has a more compact representation ability than Tucker decomposition. Since the computation of CP Rank of a tensor is proven to be a NP hard problem, we specify the sparsity priors over the latent components, which can thus obtain the minimum number of components via Bayesian inference. Similarly, we also place a non-informative hyper-prior over the noise precision parameter which leads to the possibility of inferring it from data.

2.2 Non-Local Tensor Denoising

We consider a given tensor $\mathcal{T}$ corrupted by Gaussion noise $\mathcal{N}(0,\sigma)$, our objective is to recover the underlying clean tensor $\mathcal{V}$. The main procedure includes three steps that are

- At each tensor element and for a fixed sub-tensor size, a group of similar sub-tensors is selected and constructed to be a higher order tensor.

- The proposed Bayesian CP factorization is employed for each stack to obtain an estimate of a denoised stack.

- The sub-tensors are reassembled in original location to obtain a denoised tensor.

Given a reference sub-tensor $\boldsymbol{\mathcal{P}}$ from the noisy tensor $\boldsymbol{\mathcal{T}}$, we choose other sub-tensors in the tensor $\boldsymbol{\mathcal{T}}$ that are similar to $\boldsymbol{\mathcal{P}}$. The similarity can be simply measured by Euclidean distance. There are two choices for selecting similar sub-tensors. One is to use a distance threshold $\tau_d = 3\sigma^2 s$, where σ^2 denotes the noise variance and s denotes the size of each sub-tensor. The other one is to use a fixed number of sub-tensors ordered by the distance with the reference sub-tensor. Assume that there are K such sub-tensors (including $\boldsymbol{\mathcal{P}}$) which are labeled as $\{\boldsymbol{\mathcal{P}}_i\}$ where $1 \leq i \leq K$. These sub-tensors were assumed to be noise corrupted versions of $\boldsymbol{\mathcal{P}}$. If a set of sub-tensors are similar to each another, denoising can be performed by leveraging this fact and filter them jointly. Based on this, we group together similar sub-tensors and organize them as a higher order tensor $\boldsymbol{\mathcal{Y}} = \{\boldsymbol{\mathcal{P}}_i | i = 1, \ldots, K\}$.

Now we consider how the filtering of $\boldsymbol{\mathcal{Y}}$ can be performed. The concept of jointly filtering multiple patches has been implemented in the BM3D algorithm but with fixed bases. However, we extend this concept to learn the spatially adaptive bases. By assuming that the group of similar sub-tensors were generated from a same clean sub-tensor, we can easily identify the low-rank properties of $\boldsymbol{\mathcal{Y}}$. Therefore, the low-rank tensor factorization can be employed to learn the bases independently for each group of similar sub-tensors. One straightforward way is to apply HOSVD to solve this problem. However, the truncated HOSVD method requires that the parameter for thresholding the transform coefficients must be known in advance, which results in difficulties in the practical applications. Hence, in this paper, we propose a Bayesian tensor factorization based on CP model and the low-rank assumption. In addition, we assume that noise variance is unknown and must be learned from noisy data automatically. After leaning the latent multiliear factor matrices from Bayesian CP factorization, we can reconstruct the group of similar sub-tensors as the denoised results for $\boldsymbol{\mathcal{Y}}$. Then, all the sub-tensors in $\boldsymbol{\mathcal{Y}}$ are jointly denoised. This procedure will be repeated for each reference sub-tensor $\boldsymbol{\mathcal{P}}_n$ in a sliding window fashion and the denoised sub-tensors are averaged to obtain the denoised result for tensor $\boldsymbol{\mathcal{T}}$.

2.3 Bayesian Low-Rank Tensor Factorization

2.3.1 Model Specification

We introduce the Bayesian CP factorization for jointly filtering of multiple sub-tensors in $\boldsymbol{\mathcal{Y}}$. Without loss of generality, let $\boldsymbol{\mathcal{Y}}$ be an Nth-order tensor of size $I_1 \times I_2 \times \cdots \times I_N$. We assume $\boldsymbol{\mathcal{Y}}$ is a noisy observation of true tensor $\boldsymbol{\mathcal{X}}$, that is, $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{E}}$, where the noise term is assumed to be an i.i.d. Gaussian distribution, i.e., $\boldsymbol{\mathcal{E}} \sim \prod_{i_1,\ldots,i_N} \boldsymbol{\mathcal{N}}(0, \tau^{-1})$, and the latent tensor $\boldsymbol{\mathcal{X}}$ is generated by a CP model, defined by

$$\boldsymbol{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \circ \cdots \circ \mathbf{a}_r^{(N)} = [\![\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}]\!]. \tag{2.1}$$

where $\circ$ denotes the outer product of vectors and $[\![\cdot \cdots]\!]$ is a shorthand notation, also termed as the Kruskal operator. CP factorization can be interpreted as a sum of R rank-one tensors, and the smallest integer R is defined as *CP rank*. $\{\mathbf{A}^{(n)}\}_{n=1}^{N}$ denote a group of factor matrices. For clarity, we denote mode-n factor matrix $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R}$ by row-wise or column-wise vectors (2.2)

$$\begin{aligned}\mathbf{A}^{(n)} &= \left[\mathbf{a}_1^{(n)}, \ldots, \mathbf{a}_{i_n}^{(n)}, \ldots, \mathbf{a}_{I_n}^{(n)}\right]^T \\ &= \left[\mathbf{a}_{\cdot 1}^{(n)}, \ldots, \mathbf{a}_{\cdot r}^{(n)}, \ldots, \mathbf{a}_{\cdot R}^{(n)}\right].\end{aligned} \tag{2.2}$$

The likelihood of CP model can be factorized over tensor elements, which is given by (2.3)

$$\mathcal{Y}_{i_1 i_2 \ldots i_N} \,\Big|\, \{\mathbf{A}^{(n)}\}, \tau \sim \mathcal{N}\left(\left\langle \mathbf{a}_{i_1}^{(1)}, \mathbf{a}_{i_2}^{(2)}, \cdots, \mathbf{a}_{i_N}^{(N)} \right\rangle, \tau^{-1}\right). \tag{2.3}$$

where τ is the noise precision, and $\left\langle \mathbf{a}_{i_1}^{(1)}, \mathbf{a}_{i_2}^{(2)}, \cdots, \mathbf{a}_{i_N}^{(N)} \right\rangle$ is a generalized inner-product among N vectors. The observation model in (2.3) shows that $\mathcal{Y}_{i_1 \cdots i_N}$ is represented by a group of R-dimensional latent vectors $\left\{\mathbf{a}_{i_n}^{(n)} | n = 1, \ldots, N\right\}$, which results in that the multilinear interactions can be considered. As compared to matrix factorization, tensor factorization allows us to model the multilinear structure by the inner product of $N \geq 3$ vectors.

The number of latent components, i.e., $Rank_{CP}(\mathcal{X}) = R$, is a tuning parameter whose selection is very difficult in practical applications. To avoid manually adjusting this parameter, we aim to develop an automatic model selection, which can find the rank of the latent tensor $\mathcal{X}$ solely from the observed data $\mathcal{Y}$. Taking into account the low-rank property, the number of latent components is desired to be minimal. Therefore, we employ specific sparsity-inducing priors over latent components and control the variance of each component by individual hyperparameters. Through Bayesian inference, the variance of unnecessary components can be reduced to zero. This strategy is related to automatic relevance determination (ARD) or sparse Bayesian learning. The difference lie in that our method employs a group of sparsity-inducing priors over each mode-n factors and the hyperparameters are common among these priors. Hence, the low-rank constraint can be imposed jointly to the factor matrices.

For each mode-n factor matrix, we specify a prior distribution that is governed by hyperparameters $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_R]$, among which λ_r corresponds to rth component. The prior distribution over latent factors is thus given by

$$\begin{gathered}\mathbf{a}_{i_n}^{(n)} \mid \boldsymbol{\lambda} \sim \boldsymbol{\mathcal{N}}\left(\mathbf{a}_{i_n}^{(n)} \mid \mathbf{0}, \boldsymbol{\Lambda}^{-1}\right), \\ \forall n \in [1, N], \ \forall i_n \in [1, I_n].\end{gathered} \tag{2.4}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\boldsymbol{\lambda})$ is a diagonal matrix that is also called the precision matrix. This precision matrix is jointly shared by all latent factor matrices. Since the precision parameters $\boldsymbol{\lambda}$ is unknown, and need to be learned automatically, we employ the hyperprior over $\boldsymbol{\lambda}$, given by

$$\lambda_r \sim \mathrm{Ga}(\lambda_r | c_0^r, d_0^r), \quad \forall r \in [1, R]. \tag{2.5}$$

where the Gamma distribution is given by $\mathrm{Ga}(x|a, b) = \frac{b^a x^{a-1} e^{-bx}}{\Gamma(a)}$. The number of components (i.e., R) is usually initialized to be a maximum possible value. By employing a Bayesian inference framework, the effective number of components can be inferred automatically solely from observed data. Because the hyperparameters of sparsity priors over all factor matrices are common, the same number of components can be obtained for each factor matrix, resulting in that the minimum number of rank-one terms can be learned. Hence, the CP rank of the tensor can be effectively inferred while performing low-rank tensor factorization.

Since the noise variance is assumed to be unknown, we can also specify a hyperprior over the noise parameter τ, which is given by (2.6)

$$\tau \sim \mathrm{Ga}(\tau | a_0, b_0). \tag{2.6}$$

To simplify the notations, we collect and denote all unknown variables by $\Theta = \{\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(N)}, \boldsymbol{\lambda}, \tau\}$. Finally, the joint distribution of Bayesian low-rank tensor factorization model can be written as (2.7)

$$\begin{aligned} p(\boldsymbol{\mathcal{Y}}, \Theta) = & \rho\left(\boldsymbol{\mathcal{Y}} \,\middle|\, \{\mathbf{A}^{(n)}\}_{n=1}^{N}, \tau\right) \\ & \prod_{n=1}^{N} \rho\left(\mathbf{A}^{(n)} \,\middle|\, \boldsymbol{\lambda} p(\boldsymbol{\lambda}) p(\tau)\right). \end{aligned} \tag{2.7}$$

Generally, maximum a posteriori (MAP) estimation of Θ can be obtained by optimizing. In contrast to the MAP estimation, we aim to develop a Bayesian inference method to infer the full posterior distribution of unknown variables in Θ, which is computed by (2.8)

$$p(\Theta|\boldsymbol{\mathcal{Y}}) = \frac{p(\Theta, \boldsymbol{\mathcal{Y}})}{\int p(\Theta, \boldsymbol{\mathcal{Y}})\, d\Theta}. \tag{2.8}$$

2.3.2 Bayesian Model Inference

Since the exact Bayesian inference in is obviously analytically intractable, we must resort to the approximate inference framework. In this section, we employ the variational Bayesian (VB) inference strategy to perform model inference for tensor factorization model.

We assume that $q(\Theta)$ is an approximation of the true posterior distribution $p(\Theta|\boldsymbol{\mathcal{Y}})$, which is optimized by KL divergence between them, which can be shown to be

$$\begin{aligned} \mathrm{KL}\Big(q(\Theta) \,\Big|\Big|\, p(\Theta \mid \boldsymbol{\mathcal{Y}})\Big) = & \int q(\Theta) \ln \left\{ \frac{q(\Theta)}{p(\Theta|\boldsymbol{\mathcal{Y}})} \right\} d\Theta \\ = \ln p(\boldsymbol{\mathcal{Y}}) - & \int q(\Theta) \ln \left\{ \frac{p(\boldsymbol{\mathcal{Y}}, \Theta)}{q(\Theta)} \right\} d\Theta. \end{aligned} \tag{2.9}$$

where $\ln p(\boldsymbol{\mathcal{Y}})$ denotes the marginal likelihood, and $\boldsymbol{\mathcal{L}}(q) = \int q(\Theta) \ln \left\{ \frac{p(\boldsymbol{\mathcal{Y}}, \Theta)}{q(\Theta)} \right\} d\Theta$

can be defined as its *lower bound*. Therefore, instead of minimizing the KL divergence directly, we can maximize the lower bound alternatively due to the fact that the model evidence is a constant and not related to any unknown variables.

By employing the mean-field approximation, we assume that the variational distribution can be factorized as (2.10)

$$q(\Theta) = q_\lambda(\boldsymbol{\lambda}) q_\tau(\tau) \prod_{n=1}^{N} q_n \left(\mathbf{A}^{(n)}\right). \tag{2.10}$$

Therefore, it can been shown that the posterior distribution of factor matrices is also a Gaussian distribution and the distributions corresponding to each row are independent, which is written as

$$q_n(\mathbf{A}^{(n)}) = \prod_{i_n=1}^{I_n} \mathcal{N}\left(\mathbf{a}_{i_n}^{(n)} \,\middle|\, \tilde{\mathbf{a}}_{i_n}^{(n)} \mathbf{V}^{(n)},\ \forall n \in [1, N]\right). \tag{2.11}$$

where the variational parameters are computed by

$$\begin{aligned} \tilde{\mathbf{A}}^{(n)} &= \mathbb{E}_q[\tau] \boldsymbol{\mathcal{Y}}_{(n)} \mathbb{E}_q\left[\mathbf{A}^{(\backslash n)}\right] \mathbf{V}^{(n)}, \\ \mathbf{V}^{(n)} &= \left(\mathbb{E}_q[\tau] \mathbb{E}_q\left[\mathbf{A}^{(\backslash n)T} \mathbf{A}^{(\backslash n)}\right] + \mathbb{E}_q[\boldsymbol{\Lambda}]\right)^{-1}, \end{aligned} \tag{2.12}$$

where $\boldsymbol{\mathcal{Y}}_{(n)}$ denotes the mode-n matricization of $\boldsymbol{\mathcal{Y}}$ and

$$\mathbf{A}^{(\backslash n)} = \bigodot_{k \neq n} \mathbf{A}^{(k)}, \tag{2.13}$$

where the size of $\bigodot_{k \neq n} \mathbf{A}^{(k)}$ is $\prod_{k \neq n} I_k \times R$. Thus, $\mathbb{E}_q[\mathbf{A}^{(\backslash n)T} \mathbf{A}^{(\backslash n)}]$ denotes the expectation of covariance matrix, while the covariance matrix corresponds to the Khatri-Rao product of all factor matrices except the nth-mode.

Therefore, the parameters of posterior distribution over factor matrices can be approximated by, which can be also used to compute the posterior moments, such as $\forall n, \forall i_n$, $\mathbb{E}_q\left[\mathbf{A}^{(n)}\right]$, and $\mathbb{E}_q\left[\mathbf{A}^{(n)} \mathbf{A}^{(n)T}\right], \mathbb{E}_q\left[\mathbf{A}^{(n)T} \mathbf{A}^{(n)}\right]$.

For the inference of $\boldsymbol{\lambda}$, we can derive that the posterior distribution over $\lambda_r, \forall r \in [1, R]$ can be obtained by

$$q_{\lambda}(\boldsymbol{\lambda}) = \prod_{r=1}^{R} \text{Ga}(\lambda_r | c_M^r, d_M^r), \tag{2.14}$$

where the variational parameters are computed by

$$\begin{aligned} c_M^r &= c_0^r + \frac{1}{2} \sum_{n=1}^{N} I_n, \\ d_M^r &= d_0^r + \frac{1}{2} \sum_{n=1}^{N} \mathbb{E}_q \left[\mathbf{a}_{\cdot r}^{(n)T} \mathbf{a}_{\cdot r}^{(n)} \right]. \end{aligned} \tag{2.15}$$

The expectation term in above equations denotes the norm of the rth component from mode-n matrix, which can be easily computed by

$$\mathbb{E}_q \left[\mathbf{a}_{\cdot r}^{(n)T} \mathbf{a}_{\cdot r}^{(n)} \right] = \tilde{\mathbf{a}}_{\cdot r}^{(n)T} \tilde{\mathbf{a}}_{\cdot r}^{(n)} + I_n \left(\mathbf{V}^{(n)} \right)_{rr}. \tag{2.16}$$

For inference of hyperparameter τ, it can be derived that the variational posterior is a Gamma distribution, given by

$$q_{\tau}(\tau) = \text{Ga}(\tau | a_M, b_M), \tag{2.17}$$

the variational parameters of the posterior distribution are computed by

$$\begin{aligned} a_M &= a_0 + \frac{1}{2} \prod_n I_n, \\ b_M &= b_0 + \frac{1}{2} \mathbb{E}_q \left[\left|\left| \boldsymbol{\mathcal{Y}} - [\![\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(N)}]\!] \right|\right|_F^2 \right] \end{aligned} \tag{2.18}$$

2.3.3 Initialization of Model Parameters

In this probabilistic tensor decomposition model, it is important to initialize the hyperparameters. Specifically, $\mathbf{c}_0, \mathbf{d}_0, a_0, b_0$ are set to 10^{-6} yielding a noninformative prior. The mode-n factor matrices $\{\mathbf{A}^{(n)}\}_{n=1}^{N}$ can be either randomly drawn from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ or initialized by SVD method, i.e., $\mathbf{A}^{(n)} = \mathbf{U}^{(n)} \mathbf{\Sigma}^{(n)^{\frac{1}{2}}}$, where $\mathbf{U}^{(n)}$ is the singular vectors and $\mathbf{\Sigma}^{(n)}$ is the singular values matrix.

2.4 Experimental Results

2.4.1 Image Denoising

We use color image (Lena, Peppers, Barbara) denoising to evaluate our method BCPF. For the noise model $\boldsymbol{\mathcal{N}}(0, \sigma)$, we select $\sigma \in \{0.4, 0.8, 1.2\}$. In the experiment, we use images of size $256 \times 256 \times 3$ consisting of R, G, B channels. The method performance is evaluated by PSNR which is defined by $10 \log_{10}(Max_I^2 / MSE)$ (Max_I is maximum possible pixel value of the image, and MSE denotes the mean squared errors).

The result are shown in Table 2.1 and the noisy and denoised images are shown in Fig. 2.1. The size of sub-tensors is selected to be $4 \times 4 \times 3$ and the maximum number of similar sub-tensors is set to 30. We observe that our method can obtain high quality of denoised images when noise level is low, especially, it can obtain a relatively good quality even when the noise level is extremely high.

2.4.2 MRI Denoising

Magnetic resonance imaging (MRI) is a medical imaging which is widely employed in the clinical diagnosis. Because of the movements of the subject or electronic interference, MRI data always contain noise, the denoising of MRI data is thus important for the diagnosis quality. In this experiment, we use the public MRI data (http://brainweb.bic.mni.mcgill.ca/brainweb/), the size of MRI data is $181 \times 217 \times 165$, and we use the sub-tensor size in $4 \times 4 \times 4$

TABLE 2.1: The denoising performances evaluated by PSNR for Lena, Peppers, Barbara images under three different noise levels.

Lena		Noise standard deviation		
	Methods	0.4	0.8	1.2
	BCPF	32.90	32.39	31.72
	HOSVD	30.70	30.54	30.46
Peppers		Noise standard deviation		
	Methods	0.4	0.8	1.2
	BCPF	31.37	30.89	30.77
	HOSVD	30.73	30.49	29.84
Barbara		Noise standard deviation		
	Methods	0.4	0.8	1.2
	BCPF	30.32	30.89	30.78
	HOSVD	30.55	30.54	30.36

and the maximum number of similar sub-tensors is set to 30. The result are shown in Table 2.2 and the noisy and denoised images are shown in Fig. 2.2.

TABLE 2.2: The denoising performances evaluated by PSNR for MRI denoising under three different noise levels (0.05, 0.1, 0.15).

	Noise standard deviation		
Methods	0.05	0.10	0.15
BCPF	35.97	33.81	33.00
HOSVD	36.64	33.71	32.86

2.5 Summary

In this Chapter, we propose a Bayesian tensor factorization based denoising framework and apply it to image and MRI denoising tasks. In contrast to

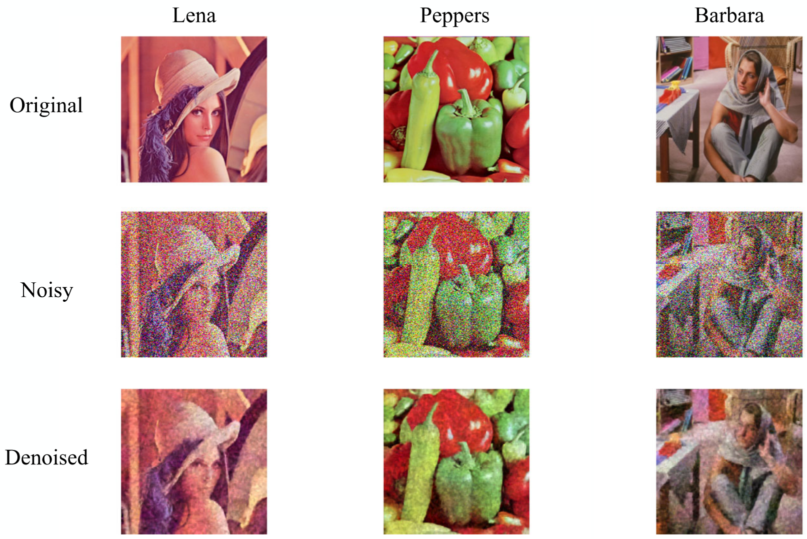

FIGURE 2.1: Visualization of Image data. From top to bottom rows, the original, noisy and denoised images are shown under the condition of $\sigma = 0.4$.

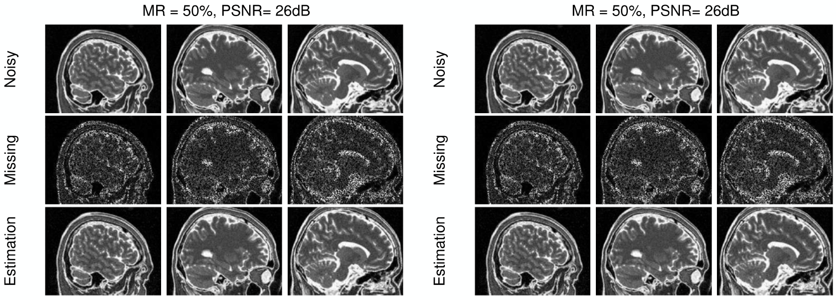

FIGURE 2.2: Visualization of MRI denoising results. From top to bottom rows, the original, noisy and denoised slice of MRI data are shown under the condition of $\sigma = \{0.05, 0.1, 0.15\}$.

most existing denoising methods, we use sub-tensors instead of 2D patches. Moreover, the transform bases of a group of sub-tensors can be learned by the probabilistic CP factorization with a low-rank assumption. As compared to other methods, our method enables us to infer automatically the noise variance, which indicates that our method is more practical. Experimental results show that our method can outperform HOSVD based denoising method.

Chapter 3

Image and Video Completion by Bayesian Tensor Decomposition

Reconstruction of image and video from sparse observations attract a great deal of interest in the filed of image/video compression, feature extraction and denoising. Since the color image and video data can be naturally expressed as a tensor structure, many methods based on tensor algebra have been studied together with promising predictive performance. However, one challenging problem in those methods is tuning parameters empirically which usually requires computational demanding cross validation or intuitive selection. In this Chapter, we introduce Bayesian Tucker decomposition to reconstruct image and video data from incomplete observation. By specifying the sparsity priors over factor matrices and core tensor, the tensor rank can be automatically inferred via variational bayesian, which greatly reduce the computational cost for model selection. We conduct several experiments on image and video data, which shows that our method outperforms the other tensor methods in terms of completion performance.

3.1 Background

Image or video completion, which is to reconstruct a full image/video from only sparsely observed information, plays an important role in image processing field. Image data can be naturally expressed as a 3rd-order tensor of size *height* × *width* × *color channel*, while the video data can be represented as 4th-order tensor of size *height* × *width* × *color* × *time*. The most popular models of tensor decomposition are Tucker decomposition (Tucker, 1966) and CANDECOMP/PARAFAC (CP) decomposition (Carroll and Chang, 1970; Harshman, 1970; Kiers, 2000). Moreover, tensor method has been applied in various research field such as: image completion (Gui, Zhao, and Cao, 2017; Yuan, Zhao, and Cao, 2017b; Liu et al., 2013; Yuan, Zhao, and Cao, 2017a; Yuan et al., 2018b; Zhao, Zhang, and Cichocki, 2015; Filipović and Jukić, 2015; Yuan et al., 2018c; Yuan et al., 2018a), signal processing (De Lathauwer and Castaing, 2008; Gui, Zhao, and Cao, 2016; Cichocki et al., 2015; Gui et al., 2017; Muti and Bourennane, 2005; De Lathauwer and De Moor, 1998), brain machine interface (BMI) (Liu et al., 2014; Mocks, 1988; Zhang et al., 2016), image classification (Shashua and Levin, 2001), face recognition (Geng et al., 2011), machine learning (Zhao et al., 2016), etc. Basically, there are two type of methods for tensor completion. One type is based on minimization of the convex relaxation function of tensor rank by using nuclear norm of tensor. The nuclear norm can be defined in several different ways related to the different tensor decomposition models. By applying the appropriate optimization algorithm, we can find the optimal low-rank tensor as the approximation of full tensor. Another type is based on tensor decomposition of incomplete tensor. The specific algorithm must be developed to find latent factors under the specific tensor decomposition model by using partially observed entries. It is necessary to predefine the tensor rank, which is considered as a model selection problem. Although cross-validation can be used to determine an optimal tensor rank, it is quite computational demanding. Especially, when the Tucker decomposition is considered, the number of possibilities of tensor rank increases exponentially to the order of tensor.

To overcome these limitations, we introduce a Bayesian tensor decomposition method to perform image and video completion. Our methods can automatically adapt model complexity and infer an optimal multilinear rank by the principle of maximum lower bound of model evidence. Experimental results and comparisons on image and video data demonstrate remarkable performance of our models for recovering the groundtruth of multilinear rank and missing pixels.

3.2 Bayesian Tucker Decomposition

3.2.1 Model Specification

In this section, we introduce Bayesian Tucker decomposition for tensor completion. Let $\boldsymbol{\mathcal{Y}}$ be an incomplete tensor with missing entries, and $\boldsymbol{\mathcal{O}}$ is a binary tensor which indicates the observation positions. Ω denotes a set of N-tuple indices of observed entries. The value of $\boldsymbol{\mathcal{O}}$ is defined by

$$\begin{cases} \mathcal{O}_{i_1 \cdots i_N} = 1 & \text{if } (i_1, \ldots, i_N) \in \Omega, \\ \mathcal{O}_{i_1 \cdots i_N} = 0 & \text{if } (i_1, \ldots, i_N) \notin \Omega. \end{cases} \tag{3.1}$$

$\boldsymbol{\mathcal{Y}}_\Omega$ is a tensor which only include observed entries. The generative model is assumed as

$$\boldsymbol{\mathcal{Y}}_\Omega = \boldsymbol{\mathcal{X}}_\Omega + \varepsilon, \tag{3.2}$$

where the latent tensor $\boldsymbol{\mathcal{X}}$ is represented exactly by a Tucker model with a low multilinear rank and ε denotes i.i.d. Gaussian noise.

Given an incomplete image tensor, Bayesian Tucker model only considers the observed data, thus the likelihood function can be represented by

$$p\Big(\mathcal{Y}_\Omega\Big) = \prod_{(i_1,i_2,i_3)\in\Omega} \mathcal{N}\Big(\mathcal{Y}_{i_1 i_2 i_3} \,\Big|\, \mathcal{X}_{i_1 i_2 i_3},\, \tau^{-1}\Big). \tag{3.3}$$

Since the latent tensor $\boldsymbol{\mathcal{X}}$ can be decomposed exactly by a Tucker model, we can thus represent the observation model as that $\forall (i_1, i_2, i_3)$,

$$\begin{aligned} \mathcal{Y}_{i_1 i_2 i_3} \,\Big|\, \Big\{\mathbf{u}_{i_n}^{(n)}\Big\}, \boldsymbol{\mathcal{G}}, \tau \sim \\ \boldsymbol{\mathcal{N}}\Big(\Big(\bigotimes_n \mathbf{u}_{i_n}^{(n)T}\Big)\mathrm{vec}(\boldsymbol{\mathcal{G}}),\, \tau^{-1}\Big)^{\mathcal{O}_{i_1 i_2 i_3}}. \end{aligned} \tag{3.4}$$

where $n = 1,2,3$. $\mathbf{u}_{i_n}^{(n)}$ is the i_n-th row of the factor matrix $\mathbf{U}^{(n)}$, $\boldsymbol{\mathcal{O}}$ is the indicator of missing points. τ is the precision of Gaussian noise.

To employ sparsity priors, we can specify the hierarchical prior distributions by

$$\begin{aligned} \tau &\sim Ga\Big(a_0^\tau,\, b_0^\tau\Big), \\ \mathrm{vec}(\boldsymbol{\mathcal{G}}) \,\Big|\, \Big\{\boldsymbol{\lambda}^{(n)}\Big\}, \beta &\sim \boldsymbol{\mathcal{N}}\Big\{\mathbf{0},\, \Big(\beta \bigotimes_n \boldsymbol{\Lambda}^{(n)}\Big)^{-1}\Big), \\ \beta &\sim Ga\Big(a_0^\beta,\, b_0^\beta\Big), \\ \mathbf{u}_{i_n}^{(n)} \,\Big|\, \boldsymbol{\lambda}^{(n)} &\sim \boldsymbol{\mathcal{N}}\Big(\mathbf{0},\, \boldsymbol{\Lambda}^{(n)^{-1}}\Big), \quad \forall n, \forall i_n. \\ \boldsymbol{\Lambda}_{r_n}^{(n)} &\sim Ga\Big(a_0^\lambda,\, b_0^\lambda\Big), \quad \forall n, \forall r_n, \end{aligned} \tag{3.5}$$

where β is a scale parameter related to the magnitude of $\boldsymbol{\mathcal{G}}$, on which a hyperprior can be placed. The hyperprior for $\boldsymbol{\lambda}^{(n)}$ play a key role for different sparsity inducing priors. We propose the hierarchical prior corresponding to the Student-t distribution for group sparsity. Note that $\boldsymbol{\Lambda}^{(n)} = \mathrm{diag}(\boldsymbol{\lambda}^{(n)})$.

For Tucker decomposition of an incomplete tensor, the problem is ill-conditioned and has infinite solutions. The low-rank assumption play an key role for successful tensor completion, which implies that the determination of multilinear rank significantly affects the predictive performance. However, standard model selection strategies, such as cross-validation, cannot be applied for finding the optimal multilinear rank because it varies dramatically with missing ratios. Therefore, the inference of multilinear rank is more challenging when missing values occur.

As shown in (3.5), we employ a hierarchical group sparsity prior over the factor matrices and core tensor with aim to seek the minimum multilinear rank automatically, which is more efficient and elegant than the standard model selections by repeating many times and selecting one optimum model. By combining likelihood model in (3.4), we propose a Bayesian Tucker Completion (BTC) method, which enables us to infer the minimum multilinear rank as well as the noise level solely from partially observed data without requiring the tuning parameters.

3.2.2 Model Inference Algorithm

To learn the BTC model, we employ the VB inference framework under a fully Bayesian treatment. In this section, we present only the main solutions. As can be derived, the variational posterior distribution over the core tensor $\mathcal{G}$ is given by

$$q(\mathcal{G}) = \mathcal{N}\left(\mathrm{vec}(\mathcal{G}) \,\middle|\, \mathrm{vec}(\widetilde{\mathcal{G}}),\, \Sigma_G\right), \tag{3.6}$$

where the posterior parameters can be updated by

$$\mathrm{vec}(\widetilde{\mathcal{G}}) = \mathbb{E}[\tau]\Sigma_G \sum_{(i_1,i_2,i_3)\in\Omega} \left(\mathcal{Y}_{i_1 i_2 i_3} \bigotimes_{n=1}^{3} \mathbb{E}\left[\mathbf{u}_{i_n}^{(n)}\right]\right). \tag{3.7}$$

$$\begin{aligned}\Sigma_G = \Big\{ & \mathbb{E}[\beta] \bigotimes_n \mathbb{E}\left[\boldsymbol{\Lambda}^{(n)}\right] + \\ & \mathbb{E}[\tau] \sum_{(i_1,i_2,i_3)\in\Omega} \bigotimes_{n=1}^{3} \mathbb{E}\left[\mathbf{u}_{i_n}^{(n)} \mathbf{u}_{i_n}^{(n)T}\right] \Big\}^{-1}.\end{aligned} \tag{3.8}$$

Since the variational posterior distribution over $\left\{\mathbf{U}^{(n)}\right\}$ can be factorized as

$$q(\mathbf{U}^{(n)}) = \prod_{i_n} \mathcal{N}\left(\mathbf{u}_{i_n}^{(n)} \,\middle|\, \widetilde{\mathbf{u}}_{i_n}^{(n)}, \boldsymbol{\Psi}_{i_n}^{(n)}\right), \ \ n = 1, \dots, 3. \tag{3.9}$$

the posterior parameters are updated by

$$\begin{aligned}\widetilde{\mathbf{u}}_{i_n}^{(n)} = & \mathbb{E}[\tau] \boldsymbol{\Psi}_{i_n}^{(n)} \mathbb{E}\left[\mathbf{G}_{(n)}\right] \\ & \sum_{(i_1,i_2,i_3)\in\Omega} \left(\mathcal{Y}_{i_1 i_2 i_3} \bigotimes_{k\neq n} \mathbb{E}\left[\mathbf{u}_{i_k}^{(k)}\right] \right).\end{aligned} \tag{3.10}$$

$$\boldsymbol{\Psi}_{i_n}^{(n)} = \left\{ \mathbb{E}[\boldsymbol{\Lambda}^{(n)}] + \mathbb{E}[\tau] \mathbb{E}\left[\mathbf{G}_{(n)} \boldsymbol{\Phi}_{i_n}^{(n)} \mathbf{G}_{(n)}^T\right] \right\}^{-1}. \tag{3.11}$$

$$\boldsymbol{\Phi}_{i_n}^{(n)} = \sum_{(i_1,\dots,i_N)\in\Omega} \bigotimes_{k\neq n} \mathbf{u}_{i_k}^{(k)} \mathbf{u}_{i_k}^{(k)T}. \tag{3.12}$$

The summation is performed over the observed data locations whose mode-n index is fixed to i_n. In other words, $\boldsymbol{\Phi}_{i_n}^{(n)}$ represents the statistical information of mode-k ($k \neq n$) latent factors that interact with $\mathbf{u}_{i_n}^{(n)}$. In (3.11), the complex posterior expectation can be computed efficiently by

$$\mathrm{vec}\left\{ \mathbb{E}\left[\mathbf{G}_{(n)} \boldsymbol{\Phi}_{i_n}^{(n)} \mathbf{G}_{(n)}^T\right] \right\} = \mathbb{E}\left[\mathbf{G}_{(n)} \otimes \mathbf{G}_{(n)}\right] \mathrm{vec}\left(\boldsymbol{\Phi}_{i_n}^{(n)}\right). \tag{3.13}$$

The variation posterior distribution over $\{\boldsymbol{\lambda}^{(n)}\}$ is i.i.d. Gamma distributions due to the conjugate priors, which is $\forall n = 1, \ldots, 3$,

$$q(\boldsymbol{\lambda}^{(n)}) = \prod_{r_n=1}^{R_n} Ga\Big(\lambda_{r_n}^{(n)} \mid \tilde{a}_{r_n}^{(n)},\, \tilde{b}_{r_n}^{(n)}\Big). \tag{3.14}$$

where the posterior parameters can be updated by

$$\begin{aligned} \tilde{a}_{r_n}^{(n)} &= a_0^\lambda + \frac{1}{2}\Big(I_n + \prod_{k\neq n} R_k\Big), \\ \tilde{b}_{r_n}^{(n)} &= b_0^\lambda + \frac{1}{2}\mathbb{E}\Big[\mathbf{u}_{\cdot r_n}^{(n)T}\mathbf{u}_{\cdot r_n}^{(n)}\Big] \\ &\quad + \frac{1}{2}\mathbb{E}[\beta]\mathbb{E}\Big[\mathrm{vec}(\boldsymbol{\mathcal{G}}^2_{\cdots r_n \cdots})^T\Big] \bigotimes_{k\neq n} \mathbb{E}\big[\boldsymbol{\lambda}^{(k)}\big]. \end{aligned} \tag{3.15}$$

Finally, the predictive distributions over missing entries, given observed entries, can be approximated by using variational posterior distributions $q(\Theta)$ as follows

$$\begin{aligned} p(\mathcal{Y}_{i_1 i_2 i_3} \mid \boldsymbol{\mathcal{Y}}_\Omega) &= \int p(\mathcal{Y}_{i_1 i_2 i_3} \mid \Theta) p(\Theta \mid \boldsymbol{\mathcal{Y}}_\Omega)\, \mathrm{d}\Theta \\ &\approx \boldsymbol{\mathcal{N}}\Big(\mathcal{Y}_{i_1 i_2 i_3} \,\Big|\, \tilde{\mathcal{Y}}_{i_1 i_2 i_3},\, \mathbb{E}[\tau]^{-1} + \sigma^2_{i_1 i_2 i_3}\Big). \end{aligned} \tag{3.16}$$

where the posterior parameters can be obtained by

$$\begin{aligned} \tilde{\mathcal{Y}}_{i_1 i_2 i_3} &= \Big(\bigotimes_n \mathbb{E}\Big[\mathbf{u}_{i_n}^{(n)T}\Big]\Big)\mathbb{E}\big[\mathrm{vec}(\boldsymbol{\mathcal{G}})\big], \\ \sigma^2_{i_1 i_2 i_3} &= \mathrm{Tr}\Big(\mathbb{E}\Big[\mathrm{vec}(\boldsymbol{\mathcal{G}})\mathrm{vec}(\boldsymbol{\mathcal{G}})^T\Big] \bigotimes_n \mathbb{E}\Big[\mathbf{u}_{i_n}^{(n)}\mathbf{u}_{i_n}^{(n)T}\Big]\Big) \\ &\quad - \mathbb{E}\big[\mathrm{vec}(\boldsymbol{\mathcal{G}})\big]^T \Big(\bigotimes_n \mathbb{E}\Big[\mathbf{u}_{i_n}^{(n)}\Big]\mathbb{E}\Big[\mathbf{u}_{i_n}^{(n)T}\Big]\Big)\mathbb{E}\big[\mathrm{vec}(\boldsymbol{\mathcal{G}})\big]. \end{aligned} \tag{3.17}$$

Therefore, our model can provide not only predictions over missing entries, but also the uncertainty of predictions, which is quite important for some specific applications.

3.3 Experimental Results

We verified the proposed method experimentally and compared it with related methods, i.e., high accuracy low rank tensor completion (HaLRTC) (Liu et al., 2013). Alternating Direction Method of Multipliers (ADMM) (Lin, Chen, and Ma, 2010) algorithm, developed in the 1970s, was employed by HaLRTC to solve the nuclear norm optimization problems with multiple non-smooth terms. HaLRTC algorithm using ADMM framework is based on simple low rank tensor completion (SiLRTC) algorithm (Liu et al., 2013). By replacing the dummy matrices M_is by their tensor versions, the algorithm is shown in Algorithm (1).

Algorithm 1 HaLRTC Algorithm

```
Input: $\mathcal{X}$ with $\mathcal{X}_\Omega = \mathcal{T}_{\Omega,p}$ and $K$
Output: $\mathcal{X}$
Set $\mathcal{X}_\Omega = \mathcal{T}_\Omega$ and $\mathcal{X}_{\overline{\Omega}} = 0$
for $k = 0$ to $K$ do
    for $i = 1$ to $n$ do
        $\mathcal{M}_i = fold_i \left[ D_{\frac{\alpha_i}{\rho}} \left( \mathcal{X}_{(i)} + \frac{1}{\rho} \mathcal{Y}_{i(i)} \right) \right]$
    end for
    $\mathcal{X}_\Omega = \frac{1}{n} \left( \sum_{i=1}^{n} \mathcal{M}_i - \frac{1}{\rho} \mathcal{Y}_i \right)_{\overline{\Omega}}$
    $\mathcal{Y}_i = \mathcal{Y}_i - \rho(\mathcal{M}_i - \mathcal{X})$
end for
```

3.3.1 MRI Completion

We evaluate our method by using MRI data [1], this dateset contains a set of realistic MRI data volumes produced by an MRI simulator. Because MRI data is high-dimensional, the completion from sparse observations becomes

[1] http://brainweb.bic.mni.mcgill.ca/brainweb

TABLE 3.1: The Performance of MRI Completion Evaluated by PSNR and RRSE

Methods	50%				60%			
	Original		Noisy		Original		Noisy	
	PSNR	RRSE	PSNR	RRSE	PSNR	RRSE	PSNR	RRSE
BTC-T	27.27	0.11	26.42	0.12	27.84	0.10	27.12	0.11
HaLRTC	24.19	0.16	23.17	0.18	26.73	0.12	25.00	0.14

TABLE 3.2: The Performance of MRI Completion Evaluated by PSNR and RRSE

Methods	70%				80%			
	Original		Noisy		Original		Noisy	
	PSNR	RRSE	PSNR	RRSE	PSNR	RRSE	PSNR	RRSE
BTC-T	28.12	0.10	27.55	0.11	28.38	0.10	27.83	0.10
HaLRTC	29.57	0.085	26.69	0.12	32.93	0.057	28.22	0.099

very challenging. So we separate the high-dimensional tensor data into low-dimensional small tensors. Hence, our method can be applied to small tensors completion. In experiment, we use the size of small tensors in $50 \times 50 \times 50$.

We use missing ratio (20% - 50%) and consider the noises in MRI data, and evaluation the algorithms using Peak Signal to Noise Ratio (PRSN) and RRSE. The result are shown in Table 3.1 and 3.2, and the visual quality is shown in Fig. 3.1. As we can see that the proposed method can effectively recover the missing values with high performance.

3.3.2 Video Completion

The video data is natural representation by a tensor as shown in Fig. 3.2. We evaluate the performance of the proposed method on a video sequence corrupted by additive Gaussian noise. The video sequence is downloaded from the benchmark data in (Dabov et al., 2007). We consider the noise standard deviation of 0.03, 0.15, 0.27 and missing ratio of 20% - 50%. The results are shown in Table 3.3.

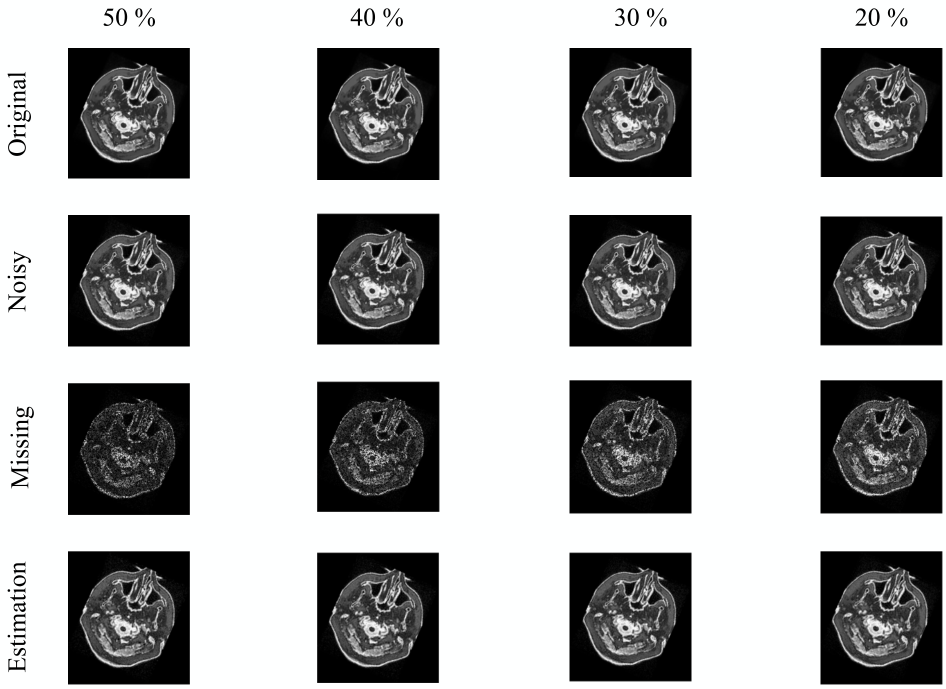

FIGURE 3.1: Visualization of MRI data completion obtained by BTC

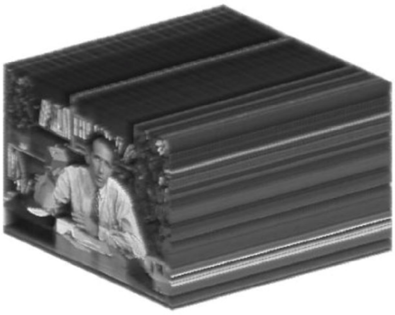

FIGURE 3.2: Tensor representation of a video sequence

TABLE 3.3: The Performance of Video Completion Evaluated by RRSE

	Missing				
	60%	50%	40%	30%	20%
Noise	RRSE	RRSE	RRSE	RRSE	RRSE
0.03	0.645	0.559	0.476	0.397	0.325
0.15	0.646	0.561	0.480	0.402	0.336
0.27	0.650	0.564	0.483	0.408	0.344

3.4 Summary

In this Chapter, we proposed an image completion method based on Bayesian Tucker decomposition. By using variational bayesian inference, we can avoids the computational demanding rank selection procedure. We apply the proposed method to image and video with 20-50 % missing voxels, the experimental results demonstrate that our method can effectively recover the whole data with a high predictive performance.

Chapter 4

Summary and Prospectives

4.1 Summary

In this thesis, we studied the basic tensor decomposition models, i.e., CPD and Tucker. The we developed the tensor denoising method, which uses higher-order tensor patches. To solve the problem of unknown noise variance, we proposed to apply Bayesian CP factorization for low-rank approximation of similar patches. The formulation and the inference algorithm of Bayesian CP factorization is presented in details. Another challenging problem is tensor completion by using fewer observed entries. To solve the rank selection problem in Tucker decomposition, we employ Bayesian setting of Tucker decomposition. By using the specially designed sparsity prior on factor matrices and core tensor, our method is able to learn the Tucker rank automatically from the given observed data entries.

Based on our proposed methods, we apply them to several real-world applications, which includes image, video and MRI denoising; image, video and MRI completion. These two applications are very important to obtain the high quality data, to predict some missing values. The extensive experimental results show that our method is very effective and perform better than the other related methods.

4.2 Prospectives

Tensor decomposition has already been applied for feature extraction, dimension reduction, and clustering problems. This thesis mainly focus on the denoising and completion problems. Besides, tensor decomposition can be also applied to improve the computation efficiency or achieve high compression of model parameters. The more applications to machine learning field seems to be a potential research direction, which is very important to show the advantages of tensor methods.

On the other hand, tensor network is an emerging topic in recent few years. It has shown to be very flexible and provide extremely high representation ability for very high-order tensor. There are some tensor network models such as tensor train decomposition and tensor ring decomposition. However, this field is relatively new and many fundamental problem and underlying principle is still not clear. We can expect that tensor network will be a next generation of tensor decomposition methods, which will be an attractive research topic in machine learning field in the future.

Bibliography

Appellof, Carl J and Ernest R Davidson (1981). “Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents”. In: *Analytical Chemistry* 53.13, pp. 2053–2056.

Buades, Antoni, Bartomeu Coll, and Jean-Michel Morel (2005). “A review of image denoising algorithms, with a new one”. In: *Multiscale Modeling & Simulation* 4.2, pp. 490–530.

Carroll, J Douglas and JihJie Chang (1970). “Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition”. In: *Psychometrika* 35.3, pp. 283–319.

Cichocki, Andrzej et al. (2015). “Tensor decompositions for signal processing applications: From two-way to multiway component analysis”. In: *IEEE Signal Processing Magazine* 32.2, pp. 145–163.

Dabov, Kostadin et al. (2007). “Image denoising by sparse 3-D transform-domain collaborative filtering”. In: *IEEE Transactions on image processing* 16.8, pp. 2080–2095.

De Lathauwer, Lieven and Joséphine Castaing (2008). “Blind identification of underdetermined mixtures by simultaneous matrix diagonalization”. In: *IEEE Transactions on Signal Processing* 56.3, pp. 1096–1105.

De Lathauwer, Lieven and Bart De Moor (1998). “From matrix to tensor: Multilinear algebra and signal processing”. In: *Institute of Mathematics and Its Applications Conference Series*. Vol. 67. Citeseer, pp. 1–16.

De Lathauwer, Lieven, Bart De Moor, and Joos Vandewalle (2000). "A multilinear singular value decomposition". In: *SIAM journal on Matrix Analysis and Applications* 21.4, pp. 1253–1278.

Filipović, Marko and Ante Jukić (2015). "Tucker factorization with missing data with application to low-n-rank tensor completion". In: *Multidimensional systems and signal processing* 26.3, pp. 677–692.

Geng, Xin et al. (2011). "Face image modeling by multilinear subspace analysis with missing values". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41.3, pp. 881–892.

Gui, Lihua, Qibin Zhao, and Jianting Cao (2016). "Tensor denoising using Bayesian CP factorization". In: *Information Science and Technology (ICIST), 2016 Sixth International Conference on*. IEEE, pp. 49–54.

— (2017). "Brain image completion by Bayesian tensor decomposition". In: *Digital Signal Processing (DSP), 2017 22nd International Conference on*. IEEE, pp. 1–4.

Gui, Lihua et al. (2017). "Video denoising using low rank tensor decomposition". In: *Ninth International Conference on Machine Vision (ICMV 2016)*. Vol. 10341. International Society for Optics and Photonics, p. 103410V.

Harshman, Richard A (1970). "Foundations of the PARAFAC procedure: Models and conditions for an explanatory multimodal factor analysis". In: *Foundations of the PARAFAC procedure*.

Hitchcock, Frank L (1927). "The expression of a tensor or a polyadic as a sum of products". In: *Journal of Mathematics and Physics* 6.1-4, pp. 164–189.

— (1928). "Multiple invariants and generalized rank of a p-way matrix or tensor". In: *Journal of Mathematics and Physics* 7.1-4, pp. 39–79.

Kiers, Henk AL (2000). "Towards a standardized notation and terminology in multiway analysis". In: *Journal of Chemometrics: A Journal of the Chemometrics Society* 14.3, pp. 105–122.

Kolda, Tamara G and Brett W Bader (2009). “Tensor decompositions and applications”. In: *SIAM review* 51.3, pp. 455–500.

Lin, Zhouchen, Minming Chen, and Yi Ma (2010). “The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices”. In: *arXiv preprint arXiv:1009.5055*.

Liu, Ji et al. (2013). “Tensor completion for estimating missing values in visual data”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.1, pp. 208–220.

Liu, Ye et al. (2014). “A tensor-based scheme for stroke patients? motor imagery EEG analysis in BCI-FES rehabilitation training”. In: *Journal of neuroscience methods* 222, pp. 238–249.

Mocks, J (1988). “Topographic components model for event-related potentials and some biophysical considerations”. In: *IEEE transactions on biomedical engineering* 35.6, pp. 482–484.

Muti, Damien and Salah Bourennane (2005). “Multidimensional filtering based on a tensor approach”. In: *Signal Processing* 85.12, pp. 2338–2353.

Rajwade, Ajit, Anand Rangarajan, and Arunava Banerjee (2011). “Using the higher order singular value decomposition for video denoising”. In: *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer, pp. 344–354.

— (2013). “Image denoising using the higher order singular value decomposition”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.4, pp. 849–862.

Shashua, Amnon and Anat Levin (2001). “Linear image coding for regression and classification using the tensor-rank principle”. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE, pp. I–I.

Tucker, L. R. (1963a). “Implications of factor analysis of three-way matrices for measurement of change”. In: *Problems in measuring change.* Ed. by C. W. Harris. Madison WI: University of Wisconsin Press, pp. 122–137.

Tucker, Ledyard R (1963b). “Implications of factor analysis of three-way matrices for measurement of change”. In: *Problems in measuring change* 15, pp. 122–137.

— (1964). “The extension of factor analysis to three-dimensional matrices”. In: *Contributions to mathematical psychology* 110119.

— (1966). “Some mathematical notes on three-mode factor analysis”. In: *Psychometrika* 31.3, pp. 279–311.

Wang, Zhou and David Zhang (1999). “Progressive switching median filter for the removal of impulse noise from highly corrupted images”. In: *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 46.1, pp. 78–80.

Yuan, Longhao, Qibin Zhao, and Jianting Cao (2017a). “Completion of high order tensor data with missing entries via tensor-train decomposition”. In: *International Conference on Neural Information Processing*. Springer, pp. 222–229.

— (2017b). “High-order Tensor Completion for Data Recovery via Sparse Tensor-train Optimization”. In: *arXiv preprint arXiv:1711.02271*.

Yuan, Longhao et al. (2018a). “High-dimension Tensor Completion via Gradient-based Optimization Under Tensor-train Format”. In: *arXiv preprint arXiv:1804.01983*.

Yuan, Longhao et al. (2018b). “Higher-dimension Tensor Completion via Low-rank Tensor Ring Decomposition”. In: *arXiv preprint arXiv:1807.01589*.

Yuan, Longhao et al. (2018c). “Rank Minimization on Tensor Ring: A New Paradigm in Scalable Tensor Decomposition and Completion”. In: *arXiv preprint arXiv:1805.08468*.

Zhang, Xinyuan et al. (2015). “Denoising of 3D magnetic resonance images by using higher-order singular value decomposition”. In: *Medical image analysis* 19.1, pp. 75–86.

Zhang, Yu et al. (2016). “Removal of EEG artifacts for BCI applications using fully Bayesian tensor completion”. In: *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, pp. 819–823.

Zhao, Qibin, Liqing Zhang, and Andrzej Cichocki (2015). “Bayesian CP factorization of incomplete tensors with automatic rank determination”. In: *IEEE transactions on pattern analysis and machine intelligence* 37.9, pp. 1751–1763.

Zhao, Qibin et al. (2016). “Bayesian robust tensor factorization for incomplete multiway data”. In: *IEEE transactions on neural networks and learning systems* 27.4, pp. 736–748.