An Algorithm for Parallel Reconstruction of Jointly Sparse Tensors with Applications to Hyperspectral Imaging

Qun Li
Microsoft Corporation
11025 NE 8th St., Bellevue, WA 98004
qul@microsoft.com

Edgar A. Bernal
United Technologies Research Center
411 Silver Ln., East Hartford, CT 06118
ber nalea@utrc.utc.com

Abstract

A wide range of Compressive Sensing (CS) frameworks have been proposed to address the task of color and hyperspectral image sampling and reconstruction. Methods for reconstruction of jointly sparse vectors that leverage joint sparsity constraints such as the Multiple Measurement Vector (MMV) approach have been shown to outperform Single Measurement Vector (SMV) frameworks which don’t exploit such constraints. Recent work has shown that exploiting joint sparsity while simultaneously preserving the high-dimensional structure of the data results in further performance improvements. In this paper, we introduce a parallelizable extension of a previously proposed serial tensorial MMV approach which, like its predecessor, exploits joint sparsity constraints multiple data dimensions simultaneously, but that, in contrast, is parallelizable in nature. We demonstrate empirically that the proposed method provides better reconstruction fidelity of hyperspectral imagery given a fixed number of measurements, and that it is also more computationally efficient than the current state of the art.

1. Introduction

The acquisition and storage of hyperspectral imagery impose stringent hardware constraints which result in increased cost and complexity of hyperspectral imagers. Hyperspectral imaging is the process of using specialized sensors to collect image information across the electromagnetic spectrum, often beyond the visible electromagnetic wavelength range. A hyperspectral image can be represented as a three-dimensional data cube where the first and second dimensions correspond to spatial data and the third dimension corresponds to the spectral bands. Objects have their own respective fingerprints known as spectral signatures; consequently, there is a wide range of applications, e.g., remote sensing, astronomy, mineralogy, agriculture, healthcare and surveillance, that rely on decomposing a two-dimensional image of a scene into its spectral bands in order to enable object identification within the scene. The main disadvantages of hyperspectral imaging are related to the intrinsic high dimensionality of the data, which imposes storage, computational, and sensitivity constraints. These factors drive up the complexity and cost of traditional hyperspectral imagers. Fortunately, there is significant redundancy in hyperspectral images along both the spatial and the spectral dimensions which can be exploited by judicious sampling and reconstruction techniques, which in turn gives rise to sensing equipment with reduced complexity and cost.

Compressive Sensing (CS) techniques have been proposed which exploit redundancies in hyperspectral data in order to improve the efficiency of the sampling and reconstruction processes. CS techniques aim at reconstructing sparse signals from a small number of linear measurements, relative to the original signal dimensionality. CS is often posed as an optimization problem whose objective consists in finding the solution of an underdetermined system of equations under a set of sparsity constraints. A vector \( x \in \mathbb{R}^N \) is called \( s \)-sparse if it has at most \( s \) non-zero entries. In mathematical terms, the linear sampling process of \( x \) results in a set of samples \( y \in \mathbb{R}^m \) obtained by computing \( y = Ax \). In this expression, \( A \in \mathbb{R}^{m \times N} \), where \( m < N \), denotes the sampling matrix. The reconstruction process attempts to recover \( x \) by finding a \( \hat{z} \in \mathbb{R}^N \) that satisfies

\[
\hat{z} = \arg\min_z \|z\|_0 \quad \text{s.t.} \quad y = Az. \tag{1}
\]

These sampling and recovery processes constitute what is traditionally termed the Single Measurement Vector (SMV) framework. The \( \ell_0 \) minimization problem in Eq.1 is nonpolynomial-deterministic- (NP-) hard, and thus, computationally intractable, so it is often approximated by optimizing its \( \ell_1 \) equivalent or by implementing greedy algorithms.

When the signal to be reconstructed consists of a set of vectors with common support, simultaneous sampling of
of multi-dimensional signals and its inherent modality. In order to alleviate this issue, we have proposed to jointly address MMV recovery with joint-tensor sparsity constraints. Under the premise of the Multiple Measurement Vector (MMV) framework, we derive a method for tensorial CS recovery, which is shown to surpass state-of-the-art techniques in high-dimensional cases.

The method we propose is based on the MMV framework, which is well-suited for multi-dimensional signal recovery. The MMV framework addresses two problems simultaneously: providing an accurate reconstruction of one-dimensional signals, and naive extension of tensorial CS to multidimensional problems. This approach allows for improved reconstruction performance and reduced computational complexity compared to existing methods.

1.2. Contributions

We propose a parallelizable version of the approach proposed in [3] which is, to our knowledge, the state-of-the-art with regards to MMV recovery of a multi-dimensional signal with jointly sparse components. The method in [3] is tensorial in nature and, as such, achieves better performance than competing methods because it is able to preserve the multi-dimensional nature of the data being reconstructed. One of its limitations lies in its sequential multi-stage implementation, given that it requires one reconstruction stage for each of the tensorial modes.

This *modus operandi* results in decreased fidelity since the reconstruction error from early stages propagates into subsequent stages. In contrast, the algorithm proposed in this paper decouples the reconstruction stages for the multiple tensor modes, which results in improved reconstruction quality and reduced computational complexity: since the reconstruction stages are effectively independent, they can be implemented in a parallel manner. The proposed method, like its predecessor, is capable of leveraging the tensorial nature of the data in the sampling and reconstruction stages; its parallel nature, on the other hand, enables it to provide improved reconstruction performance and achieve increased computational efficiency.

2. Notation and Terminology

As per the convention used in [13][2], we use Italic characters to represent scalar values (*e.g.*, *a, B*), bold-face characters to represent vectors (*e.g.*, *a, b*), capital italic boldface characters to represent matrices (*e.g.*, *A, B*) and capital calligraphic characters to represent tensors (*e.g.*, *X, F*). A tensor is a multi-dimensional array which may be used to represent high-dimensional signals such as color and hyperspectral images and videos. Tensor *F* ∈ ℜ³×³ has order *d* (or is *d*-dimensional), and the dimensionality of its *i*-th mode (also denoted mode *i*) is *N*ᵢ. In short, the order or number of dimensions of a tensor corresponds to its number of modes, and each mode has its own associated dimensionality. The remainder of this section provides a brief overview of relevant multilinear algebra concepts.

Kronecker product The Kronecker product of matrices $A ∈ ℜ^{I×J}$ and $B ∈ ℜ^{K×L}$, denoted by $A ⊗ B$, is a matrix of size $(I·K) × (J·L)$ which is computed as $A ⊗ B = \begin{pmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}B & a_{I2}B & \cdots & a_{IJ}B \end{pmatrix}$.

Mode-*i* fiber and mode-*i* unfolding The mode-*i* fibers of tensor $F = [x_{α_1,..,α_i,..,α_d}] ∈ ℜ^{N_1×\cdots×N_i×\cdots×N_d}$ are obtained
by fixing every index but $\alpha_i$. The mode-$i$ unfolding of $\mathcal{X}$, denoted $X_{(i)}$, is a matrix of size $N_i \times (N_1 \cdot N_2 \cdot \ldots \cdot N_{i-1} \cdot N_{i+1} \cdot \ldots \cdot N_d)$ whose columns are the mode-$i$ fibers of $\mathcal{X}$. We use $R_{i}(\mathcal{X})$ to denote the space formed by the mode-$i$ fibers of $\mathcal{X}$.

**Mode-$i$ product** The mode-$i$ product between tensor $\mathcal{X} = [x_{a_1,\ldots,a_n}] \in \mathbb{R}^{N_1 \times \cdots \times N_d}$ and matrix $U = [u_j,\alpha_i] \in \mathbb{R}^{N_d \times N_i}$ is a tensor $\mathcal{X} \times_i U \in \mathbb{R}^{N_1 \times \cdots \times N_{i-1} \times N_{i+1} \times \cdots \times N_d}$, and is defined element-wise as

$$(\mathcal{X} \times_i U)_{a_1,\ldots,a_{i-1},j,a_{i+1},\ldots,a_d} = \sum_{\alpha_i=1}^{N_i} x_{a_1,\ldots,a_{i},\ldots,a_d} u_j,\alpha_i.$$ 

### 3. Tensorial compressive sensing of hyperspectral images

In the context of this paper, a hyperspectral image is represented as a third-order tensorial data cube, the first two tensorial modes of which correspond to the spatial dimensions, and the third mode of which is associated with the spectral dimension of the image. More specifically, the sampled hyperspectral cube representing the image is a tensor of order three $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times N_s}$, where $N_1$ and $N_2$ are the number of rows and columns of the image, respectively, and $N_s$ is the number of spectral bands. As in most CS frameworks, the proposed algorithm comprises a sampling and a reconstruction stage. The **sampling stage** is identical to that of [3], and involves the acquisition of a set of linear measurements from the hyperspectral image of a scene of interest. In the **reconstruction stage**, the hyperspectral representation of the scene is estimated by independently recovering each tensorial mode from the measurements acquired in the sampling stage. As in [3], the additional reconstruction problem is formulated in terms of a tensorial MMV framework; in contrast, the reconstruction stage of the proposed algorithm is parallelizable, and differs from that in [3] which is sequential in nature since it requires the reconstruction of certain modes to be completed before the reconstruction of other modes can take place.

#### 3.1. Sampling stage

As in [3], the sampling of $\mathcal{X}$ involves the computation of a set of mode-$i$ products between $\mathcal{X}$ and sampling matrices $U_i$, for $i = 1, 2$, one for each spatial tensor mode. The entries of the sampling matrices are randomly generated. The outcome of the sampling stage is a tensor $\mathcal{Y}$ that can be computed as

$$\mathcal{Y} = \mathcal{X} \times_1 U_1 \times_2 U_2,$$ 

where $U_1 \in \mathbb{R}^{m_1 \times N_1}$ and $U_2 \in \mathbb{R}^{m_2 \times N_2}$, where $m_1$ and $m_2$ are the number of measurements along the mode 1 and mode 2 (e.g., columns and rows) respectively, and $\mathcal{Y} \in \mathbb{R}^{m_1 \times m_2 \times N_s}$.

It should be appreciated that the outcome of the sampling stage is still a tensor of order 3, in contrast with vectorial methods (see, e.g., [15]) where the dimensions of the measurement matrix are effectively $(m_c \times N_c) \times N_b$.

#### 3.2. Reconstruction stage

The reconstruction stage obtains $\mathcal{X}$, an estimate of the hyperspectral image cube $\mathcal{X}$, by solving a set of $\ell_1$ minimization tasks that rely on knowledge of the set of samples $\mathcal{Y}$ and the set of sampling matrices. Recovering the original data $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times N_s}$ from samples $\mathcal{Y} \in \mathbb{R}^{m_1 \times m_2 \times N_s}$ involves reconstruction of the modes that are compressed in the sampling stage. We introduce two alternative approaches to the reconstruction stage, which can be seen as parallelizable MMV extensions of the sequential tensorial CS recovery stages proposed in [3]. The following discussion applies equally to both approaches and is equivalent to the parallelizable tensorial reconstruction framework first proposed in [9]. Fig. 1 visually illustrates the steps involved in the reconstruction stage.

Let matrix $A_k \in \mathbb{R}^{m_c \times m_c}$ denote the $k$-th slice of $\mathcal{Y}$ corresponding to the $k$-th spectral band, for $k = 1, 2, \ldots, N_s$ (see ‘Slicing’ stage in Fig. 1.) Decompose $A_k$ such that

$$A_k = \sum_{i=1}^{r_k} b_{ik}^{(r)} b_{ik}^{(c)^\top},$$ 

where $r_k$ is the rank of $A_k$, $b_{ik}^{(r)} \in R_1(A_k) \subseteq U_1 R_1(\mathcal{X}) \subseteq \mathbb{R}^{m_c}$, and $b_{ik}^{(c)} \in R_2(A_k) \subseteq U_2 R_2(\mathcal{X}) \subseteq \mathbb{R}^{m_c}$ (see ‘Decomposition’ stage in Fig. 1.) Here $R_1(A_k)$ and $R_2(A_k)$ denote the row and column spaces of $A_k$ respectively, and $R_1(\mathcal{X})$ and $R_2(\mathcal{X})$ denote the spaces formed by mode-1 and mode-2 fibers of $\mathcal{X}$ respectively. The decomposition from Eq. 4 can be achieved via techniques such as matrix singular value decomposition (SVD) or QR decomposition [1]. Theoretically, all exact matrix decomposition methods will yield identical reconstruction results. This rank decomposition of $A_k$ is expressed as the multiplication of two matrices, as illustrated in Fig. 1. These matrices are denoted as left and right matrices, and their columns consist of $b_{ik}^{(r)}$ and $b_{ik}^{(c)}$ for $i = 1, \ldots, r_k$, respectively. In Fig. 1, single columns with index $i$, namely $b_{ik}^{(r)}$ in the left matrix and $b_{ik}^{(c)}$ in the right matrix, are highlighted in red.

Let $f_{ik}^{(r)*} \in R_1(\mathcal{X}) \subseteq \mathbb{R}^{N_1}$ be a solution to

$$f_{ik}^{(r)*} = \arg \min ||f_{ik}^{(r)}||_1, \text{ s.t. } U_1 f_{ik}^{(r)} = b_{ik}^{(r)}, \text{ for } i = 1, \ldots, r_k,$$

and $f_{ik}^{(c)*} \in R_2(\mathcal{X}) \subseteq \mathbb{R}^{N_2}$ be a solution to

$$f_{ik}^{(c)*} = \arg \min ||f_{ik}^{(c)}||_1, \text{ s.t. } U_2 f_{ik}^{(c)} = b_{ik}^{(c)}, \text{ for } i = 1, \ldots, r_k.$$
A matrix $G_k \in \mathbb{R}^{N_b \times N_c}$ is then formed by

$$G_k = \sum_{i=1}^{r_k} f_i^{(r)_k} f_i^{(r)_k T}, \quad k = 1, \ldots, N_b.$$  

Then $\hat{\mathcal{F}} \in \mathbb{R}^{N_b \times N_c \times N_b}$ is the tensor whose $k$-th slice corresponding to the $k$-th spectral band is $G_k$, for $k = 1, 2, \ldots, N_b$. The arrow labeled ‘Reconstruction’ in Fig. 1 involves solving Eq. 5 and Eq. 6 and recovering the previously sampled/compressed dimensions from the left and right matrices of $A_k$. Single columns corresponding to index $i$, namely $f_i^{(r)_k}$ in the matrix recovered from the left matrix and $f_i^{(c)_k}$ in the matrix recovered from the right matrix, are highlighted in green. Multiplication of the two reconstructed matrices results in matrix $G_k$ (see ‘Composition’ stage in Fig. 1.) which corresponds to the $k$-th spectral band of the final reconstructed tensor $\hat{\mathcal{F}}$, obtained by stacking $G_k$ (see ‘Assembly’ stage in Fig. 1.)

Then $\mathcal{X}^{12}\mathcal{B}$ represents the joint recovery across the spectral dimension. The fundamental difference between the reconstruction methods proposed in this paper and those first introduced in [9] lies in the way the optimization tasks from Eqs. 5 and 6 are solved: the algorithms introduced herein exploit the joint sparsity between the different components in the hyperspectral data, while the algorithm from [9] reconstructs each vector independently. The main difference between the reconstruction methods introduced herein and those proposed in [3], which also exploit the joint sparsity across components in the data, lies in the parallelizability of the newly introduced algorithms, which is not present in the algorithm from [3].

3.2.1 Joint recovery across the spectral dimension

The first alternative to performing the reconstruction step from Eqs. 5 and 6 exploits the correlation of the data across the spectral dimension. To that end, the solution to the set of $\ell_1$-minimization tasks can be framed as an MMV joint recovery problem. For simplicity, consider first the case where $r_1 = r_2 = \ldots = r_k$ for all $k$, namely, each matrix $A_k$ in Eq. 4 has the same rank; the more general scenario will be discussed later. Instead of solving the $2 \times 2 \times 1$ $r_k$ minimization tasks from Eq. 5 and Eq. 6 individually (as is done in [9]), which would fail to leverage the joint sparsity property across the spectral dimension, $r_k$ jointly sparse matrices of dimensions $N_f \times N_b$ can be recovered from $r_k$ measurement matrices of dimensions $m_c \times N_b$, and $r_k$ jointly sparse matrices of dimensions $N_c \times N_b$ can be recovered from $r_k$ measurement matrices of dimensions $m_c \times N_f$. Any recovery method for jointly sparse matrices can be employed to achieve this goal. In this paper, we pose this problem as an optimization task which is effectively the tensorial extension of the method in [5], or, alternatively, the parallelizable version of the method in [3], namely:

$$F_i^{(c)_k} = \{ f_i^{(c)_1}, f_i^{(c)_2}, \ldots, f_i^{(c)_b} \} = \text{ arg min } \| F_i^{(c)_k} \|_{2,0} \quad \text{s.t.}$$

$$B_i^{(c)_k} = \{ b_i^{(c)_1}, b_i^{(c)_2}, \ldots, b_i^{(c)_b} \} = U_1 F_i^{(c)_k}, \quad \text{for } i = 1, 2, \ldots, N_c,$$

$$(8)$$

$$F_i^{(c)_k} = \{ f_i^{(c)_1}, f_i^{(c)_2}, \ldots, f_i^{(c)_b} \} = \text{ arg min } \| F_i^{(c)_k} \|_{2,0} \quad \text{s.t.}$$

$$B_i^{(c)_k} = \{ b_i^{(c)_1}, b_i^{(c)_2}, \ldots, b_i^{(c)_b} \} = U_2 F_i^{(c)_k}, \quad \text{for } i = 1, 2, \ldots, r_k$$

$$(9)$$

where each column $f_i^{(c)}$ of $F_i^{(c)_k}$ and $f_i^{(c)}$ of $F_i^{(c)_k}$ correspond to the $i$-th largest singular value of $A_k$ for $k = 1, 2, \ldots, N_b$. Fig. 2 visually illustrates the formation of the MMV matrices in the reconstruction stage just described.

Each matrix $G_k \in \mathbb{R}^{N_b \times N_c}$ can then be obtained by first multiplying the $k$-th column of $F_i^{(c)_k}$ and the transpose of the $k$-th column of $F_i^{(c)_k}$ and then summing up the result over all $r_k$ such multiplications. More specifically, $G_k$ can be computed as follows:
jointly sparse matrix of size \( N_c \times (\sum_{k=1}^{N_b} r_k) \) from a single measurement matrix of size \( m_c \times (\sum_{k=1}^{N_b} r_k) \). For simplicity, we again assume \( r_1 = r_2 = \ldots = r_k \) for all \( k \). The discussion that follows is easily extendable to the more general case.

\[
\hat{F}^{(r)*} = \arg\min \| \hat{F}^{(r)} \|_2 \quad \text{s.t.} \quad \hat{F}^{(r)} = \begin{bmatrix} f_{11}^{(r)} & f_{12}^{(r)} & \cdots & f_{1N_b}^{(r)} \\ f_{21}^{(r)} & f_{22}^{(r)} & \cdots & f_{2N_b}^{(r)} \\ f_{k1}^{(r)} & f_{k2}^{(r)} & \cdots & f_{kN_b}^{(r)} \\ \vdots & \vdots & \ddots & \vdots \\ f_{r_k1}^{(r)} & f_{r_k2}^{(r)} & \cdots & f_{r_kN_b}^{(r)} \end{bmatrix}
\]

\[
\hat{B}^{(r)*} = \begin{bmatrix} b_{11}^{(r)} & b_{12}^{(r)} & \cdots & b_{1N_b}^{(r)} \\ b_{21}^{(r)} & b_{22}^{(r)} & \cdots & b_{2N_b}^{(r)} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r_k1}^{(r)} & b_{r_k2}^{(r)} & \cdots & b_{r_kN_b}^{(r)} \end{bmatrix} = U_1 \hat{F}^{(r)*}.
\]

\[
\hat{F}^{(c)*} = \arg\min \| \hat{F}^{(c)} \|_2 \quad \text{s.t.} \quad \hat{F}^{(c)} = \begin{bmatrix} f_{11}^{(c)} & f_{12}^{(c)} & \cdots & f_{1N_b}^{(c)} \\ f_{21}^{(c)} & f_{22}^{(c)} & \cdots & f_{2N_b}^{(c)} \\ f_{k1}^{(c)} & f_{k2}^{(c)} & \cdots & f_{kN_b}^{(c)} \\ \vdots & \vdots & \ddots & \vdots \\ f_{r_k1}^{(c)} & f_{r_k2}^{(c)} & \cdots & f_{r_kN_b}^{(c)} \end{bmatrix}
\]

\[
\hat{B}^{(c)*} = \begin{bmatrix} b_{11}^{(c)} & b_{12}^{(c)} & \cdots & b_{1N_b}^{(c)} \\ b_{21}^{(c)} & b_{22}^{(c)} & \cdots & b_{2N_b}^{(c)} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r_k1}^{(c)} & b_{r_k2}^{(c)} & \cdots & b_{r_kN_b}^{(c)} \end{bmatrix} = U_1 \hat{F}^{(c)*}.
\]

Then \( \mathcal{X}^r = \mathcal{G} \in \mathbb{R}^{N_c \times N_r \times N_b} \) can be obtained by stacking all \( G_k \) for \( k = 1, \ldots, N_b \) along the spectral dimension. The parallelizable nature of this approach stems from the fact that Eqs. 9 and 9 can be solved independently.

In the more general case where \( r_k \) is not the same for all \( k \), at most \( r_{k,\text{max}} \) where \( k = \arg\max r_k \) for \( k = 1, 2, \ldots, N_b \) jointly sparse matrices of dimensions \( N_r \times N_i \), where \( N_i \leq N_b \), can be recovered from \( r_{k,\text{max}} \) measurement matrices of dimensions \( m_c \times N_i \). In other words, the number of MMV problems is determined by the matrix \( A_k \) with the largest rank, and the number of columns in \( F_k^{(r)*} \) and \( F_k^{(c)*} \) can range from 1 to \( N_b \).

### 3.2.2 Joint recovery across the spectral and spatial dimensions

The second procedure to perform the reconstruction step from Eq. 5 and Eq. 6 exploits the correlation across both the spectral and the spatial dimensions. To that end, the solution to Eq. 5 is reduced to a single recovery step of a jointly sparse matrix of size \( N_r \times (\sum_{k=1}^{N_b} r_k) \) from a single measurement matrix of size \( m_r \times (\sum_{k=1}^{N_b} r_k) \); similarly, the solution to Eq. 6 is reduced to a single recovery step of a jointly sparse matrix of size \( N_c \times (\sum_{k=1}^{N_b} r_k) \) from a single measurement matrix of size \( m_c \times (\sum_{k=1}^{N_b} r_k) \).
4. Experimental results

We tested the performance of the proposed algorithms both in terms of reconstruction accuracy and execution time. We compare our results to those achieved by the method introduced in [15], which we denote SL2.0, and believe to be state-of-the-art in terms of vectorial only MMV reconstruction when only imposing the sparsity constraint on the data to be sensed. In addition, we evaluate the proposed parallelizable methods against the previously proposed serial versions from [3], which we believe to be state-of-the-art with regards to tensorial MMV reconstruction. The legends in the figures in this section refer to the proposed first (Sec. 3.2.1) and second (Sec. 3.2.2) variants of the proposed algorithm as TMMV1P and TMMV2P respectively, which stand for parallelizable Tensorial MMV exploiting correlation across the spectral dimension, and across the spectral and spatial dimensions, respectively. We refer to the serial counterparts from [3] as TMMV1S and TMMV2S respectively, which stand for serial Tensorial MMV exploiting correlation across the spectral dimension, and across the spectral and spatial dimensions, respectively.

In order to test the performance of the algorithms, acquisition and reconstruction of hyperspectral images from the AVIRIS Yellowstone dataset introduced in [12] was simulated. The AVIRIS data consists of five calibrated and corresponding 16-bit raw images acquired over Yellowstone, WY in 2006, as well as two additional 12-bit uncalibrated images. Each image is a 512-line scene containing 224 spectral bands. Reconstruction accuracy was measured in terms of Normalized Mean Squared Error (NMSE) between the reconstructed and original image, and execution time was measured in seconds on a Windows 10 machine with 32GB of RAM and an Intel i7 2.60GHz processor. The implementation of both algorithms was done in Matlab R2016b. Each data point in every figure corresponds to an average of ten runs.

In order to keep the computational complexity of the problem manageable for the SL2.0 algorithm, the hyperspectral images were cropped to $32 \times 32$ pixels and 32 bands. Let $n = m_r = m_c$ denote the number of measurements along both spatial dimensions of the hyperspectral image; the entries of the sampling matrices are drawn from a Gaussian distribution with zero-mean and standard deviation $\sqrt{\frac{1}{n}}$. For simplicity, we set the number of measurements for both spatial dimensions to be equal and use the same sampling matrix for both dimensions ($U_1 = U_2 = U$); that is, the randomly constructed Gaussian matrix $U$ is of size $n \times 32$ for each mode. Therefore, the sampling matrix for the method in [15] is equivalent to $U \otimes U$ of size $n^2 \times 32^2$, which results in a total number of measurements of $32n^2$ across the spectral bands. We refer to $\frac{n^2}{32^2}$ as the normalized number of measurements. We vary the normalized number of measurements from 0.025 to 0.5 in steps of 0.025.
believe the reasons for this performance improvement are twofold. In terms of reconstruction error, the serial version reconstructs two tensorial modes sequentially, which transfers the reconstruction error in the first tensor mode to the next stage, causing the errors to accumulate and possibly magnify between stages. In contrast, the parallelizable TMMV methods decouple the reconstruction of the two tensor modes, so there is no error accumulation across the reconstructed modes. In terms of computational complexity, the proposed approach benefits from the reduced number of reconstruction problems that are solved, which is determined by the rank of the measurement matrices. We note that, as its name indicates, the parallelizable approach can benefit from multi-threading; however, for comparison purposes, we implemented and tested a non-parallelized version of the algorithm. The results show that despite its superiority, there is significant room for improvement for the parallelizable version of the algorithm when multiple processing cores are available, regardless of the rank of the original signal.

The fact that TMMV2P is generally superior to TMMV1P shows that exploiting spatial and spectral correlations together leads to a slight gain in both computational efficiency and accuracy. Both TMMV1P and TMMV2P are faster than SL2,0 and their sequential counterparts since the measurement matrix for the tensorial approaches has lower dimensionality than its vectorial counterpart. As expected, as the number of measurements being acquired increases, reconstruction error improves at a slight cost of computational complexity. We note that, in contrast with the results reported in [3], SL2,0 is more computationally efficient than both TMMV1S and TMMV2S. We believe this is because the experimental validation in [3] evaluated the performance of the algorithms on the reconstruction of 3-channel color images, which seems to indicate that SL2,0 scales better as the number of channels grows.

As stated, the NMSE reported in Fig. 4 for each data point is an average across 10 trials; however, some tasks impose constraints on the largest NMSE that can be allowed. Consequently, additional insight into the reconstruction performance of the algorithms can be gained from evaluating and comparing reconstruction success rates. A reconstruction is deemed successful when the NMSE is smaller than some predetermined threshold; in this case, we set that threshold to $1 \times 10^{-3}$, which seems to be standard in previous work in the CS literature. Success rate is the ratio between the number of successful reconstructions and the total number of trials for each data point. Fig. 5 shows the success rate comparison. The plots show that the proposed methods are more likely to achieve a successful reconstruction that satisfies the NSP constraint according to CS methodology with the same number of measurements.

Fig. 6 includes reconstruction errors (left column) and execution times (right column) as a function of different values of column- and row-sparsity for different numbers of normalized measurements (0.125, 0.25, 0.375 and 0.5 for first, second, third and fourth rows, respectively). The results are indicative of similar trends as those observed in Fig. 4. It can also be seen that the reconstruction time of the parallelizable methods, while always being the smallest among all methods being compared, increases with the degree of sparsity; this is because the matrix rank of each spectral slice grows as the degree of sparsity increases. In contrast, the degree of sparsity doesn’t have a significant effect on the execution times for the method in [15] and the serial methods [3]. As expected, the reconstruction error grows as the sparsity level increases for all methods.

5. Conclusion

We proposed two different variants of a parallelizable reconstruction algorithm that exploits joint sparsity across different modes of a hyperspectral image while at the same time preserving its high-dimensional nature throughout the sampling and reconstruction stages of a novel CS framework. We demonstrated empirically that the newly introduced methods outperform previously proposed approaches that vectorize the data [15], as well as methods that, while still respecting the tensorial nature of the data, reconstruct the different modes sequentially [3]. The advantages of the proposed methods in terms of both computational efficiency and reconstruction accuracy were observed to be consistent across a wide range of experimental settings.
Figure 6. Performance comparison between proposed reconstruction approach and the approach in [15] with the numbers of normalized measurements being fixed to 0.125 (row 1), 0.25 (row 2), 0.375 (row 3) and 0.5 (row 4) respectively.

References


