NONCONVEX COMPRESSIVE SENSING RECONSTRUCTION FOR TENSOR USING STRUCTURES IN MODES

Xin Ding*, Wei Chen+*, Ian Wassell*

* Computer Laboratory, University of Cambridge, UK

+ State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, China

- This work focuses on the reconstruction of a tensor captured using Compressive Sensing (CS).
- Compared to CS, Tensor CS (TCS) does not involve any vectorization, hence has the benefits of easing hardware implementation, reducing the amount of storage and preserving multidimensional structures of signals.
- We propose to exploit diverse structures along each dimension (i.e., mode) of a tensor during the reconstruction.
- The proposed multi-structure optimization problem is solved by ADMM, in which nonconvex reconstruction is employed.
- We derive to reduce the memory requirements and computation loads encountered for the recovery of large scale tensors.
- The proposed approach improves the reconstruction accuracy by providing the flexibility of involving various structures.

1. Problem Formulation

Extending the sensing model in CS, a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{N_1 \times \ldots \times N_n}$ is sampled by:*

$$\underline{\mathbf{Y}} = \underline{\mathbf{X}} \times_1 \mathbf{\Phi}_1 \times_2 \mathbf{\Phi}_2 \ldots \times_n \mathbf{\Phi}_n, \quad (1)$$

where $\underline{\mathbf{Y}} \in \mathbb{R}^{M_1 \times \ldots M_n}$ is the measurement, $\mathbf{\Phi}_i \in \mathbb{R}^{M_i \times N_i}$ $(i = 1, \ldots, n)$ are sensing matrices and $M_i < N_i$. A tensor is regarded as K sparse when it can be represented as:

$$\underline{\mathbf{X}} = \underline{\mathbf{S}} \times_1 \mathbf{\Psi}_1 \times_2 \mathbf{\Psi}_2 \ldots \times_n \mathbf{\Psi}_n, \quad (2)$$

where $\mathbf{\Psi}_i \in \mathbb{R}^{N_i \times N_i}$ $(i = 1, \ldots, n)$ are the sparsifying basis, e.g., a Discrete Wavelet Transform (DWT), $\underline{\mathbf{S}} \in \mathbb{R}^{N_1 \times \ldots N_n}$ is the sparse representation which has only K non-zero coefficients. Then the sensing procedure is equivalent to:

$$\underline{\mathbf{Y}} = \underline{\mathbf{S}} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \ldots \times_n \mathbf{A}_n, \quad (3)$$

where $\mathbf{A}_i = \mathbf{\Phi}_i \mathbf{\Psi}_i$ $(i = 1, \ldots, n)$ are the equivalent sensing matrices.

2. TCS Recovery Using Structures in Modes

- The reconstruction problem that provides the flexibility of exploiting various structures is formulated as:

$$\min_{\underline{\mathbf{X}}} \sum_{i=1}^{n} \alpha_i ||\mathbf{\Omega}_i \mathbf{X}_{(i)}||_{(i-th\ norm)}$$
$$s.t.\ \underline{\mathbf{Y}} = \underline{\mathbf{X}} \times_1 \mathbf{\Phi}_1 \ldots \times_n \mathbf{\Phi}_n. \quad (4)$$

This problem can be rewritten as:

$$\min_{\underline{\mathbf{X}}, \underline{\mathbf{Z}_1}, \ldots, \underline{\mathbf{Z}_n}} \sum_{i=1}^{n} \alpha_i ||\mathbf{\Omega}_i \mathbf{Z}_{i(i)}||_{(i-th\ norm)}$$
$$s.t.\ \underline{\mathbf{Y}} = \underline{\mathbf{X}} \times_1 \mathbf{\Phi}_1 \ldots \times_n \mathbf{\Phi}_n,$$
$$\underline{\mathbf{X}} = \underline{\mathbf{Z}_i},\ i = 1, \ldots, n, \quad (5)$$

where $\underline{\mathbf{Z}_i}$ $(i = 1, \ldots, n)$ are intermediate variables.

- We derive efficient calculations in the ADMM steps to enable implementation. Updating each $\underline{\mathbf{Z}_i}$ for $i = 1, \ldots, n$ yields:

$$\underline{\mathbf{Z}_i}^{t+1} = \mathbf{\Omega}_i^T prox_{\frac{\alpha_i}{\rho},\ (i-th\ norm)}[\mathbf{\Omega}_i(\frac{1}{\rho}\underline{\mathbf{P}_i}^t + \underline{\mathbf{X}}^t)], \quad (6)$$

where $prox(\cdot)$ denotes the proximity operator and is defined for function f as: $prox_{\frac{1}{\rho},\ f}(x) = arg\ min_y f(y) + \frac{\rho}{2}||x-y||_2^2$.

- With $\underline{\mathbf{Z}_i}$ $(i = 1, \ldots, n)$ determined, we then update $\underline{\mathbf{X}}$ by

$$\underline{\mathbf{X}}^{t+1} = [ten(\mathbf{v}) \circ (\underline{\mathbf{G}} \times_1 \mathbf{U}_1^T \ldots \times_n \mathbf{U}_n^T)] \times_1 \mathbf{U}_1 \ldots \times_n \mathbf{U}_n, \quad (7)$$

where $\mathbf{v} = diag[(n\mathbf{I} + \mathbf{V}_n \otimes \ldots \otimes \mathbf{V}_1)^{-1}]$, $\underline{\mathbf{G}} = \sum_{i=1}^{n}(\underline{\mathbf{Z}_i} - \frac{1}{\rho}\underline{\mathbf{P}_i}) + (\frac{1}{\rho}\underline{\mathbf{Q}} + \underline{\mathbf{Y}}) \times_1 \mathbf{\Phi}_1^T \ldots \times_n \mathbf{\Phi}_n^T$.

- In the final steps, the dual variables are updated by:

$$\underline{\mathbf{P}_i}^{t+1} = \underline{\mathbf{P}_i}^t - \rho(\underline{\mathbf{Z}_i}^{t+1} - \underline{\mathbf{X}}^{t+1}), \quad (8)$$
$$\underline{\mathbf{Q}}^{t+1} = \underline{\mathbf{Q}}^t - \rho(\underline{\mathbf{X}}^{t+1} \times_1 \mathbf{\Phi}_1 \ldots \times_n \mathbf{\Phi}_n - \underline{\mathbf{Y}}). \quad (9)$$

3. Nonconvex Recovery for Tensors with Low rank and Sparse Modes

We adopt the proximal l_p $(p \leq 1)$ norm (denoted by $G_{\rho,p}$) as defined in [1], of which the proximity has been derived as a p-shrinkage operator that functions element-wise as: $shrink_p(t,\ \mu) = max\{0,\ |t| - \mu|t|^{p-1}\}\frac{t}{|t|}$.

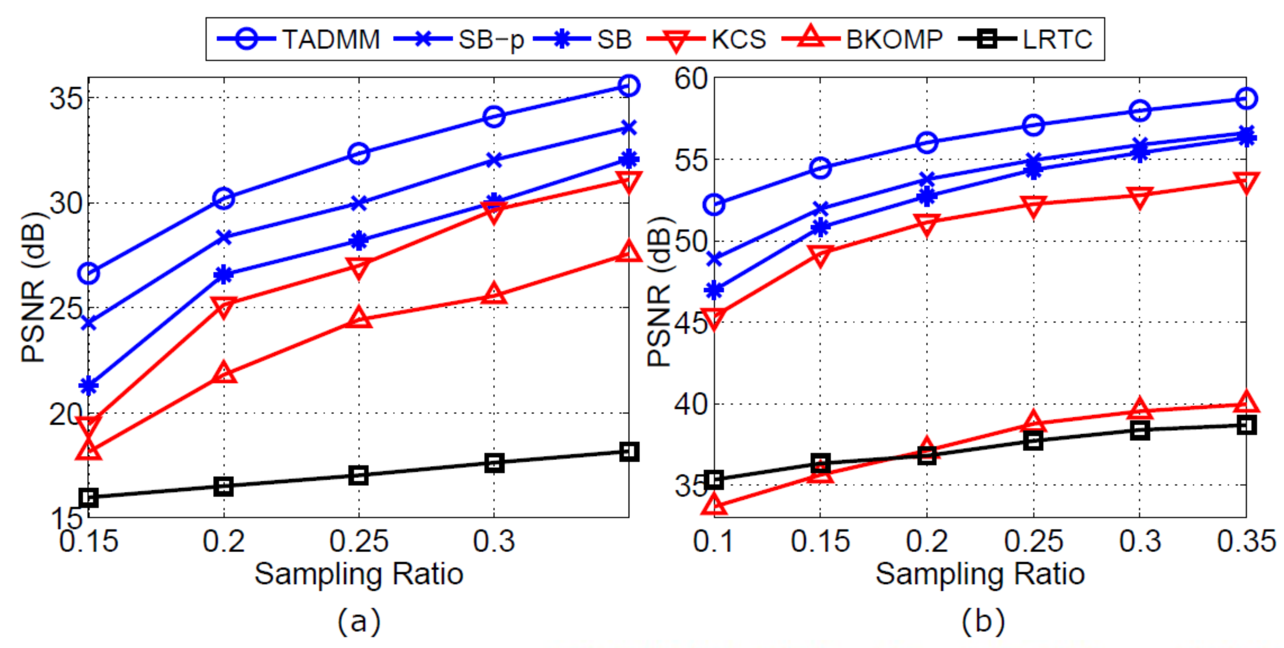
Algorithm 1 TADMM

Input: $\underline{\mathbf{Y}}$, $\mathbf{\Phi}_i$, $\mathbf{\Omega}_i$, $\underline{\mathbf{X}}^0$, $\underline{\mathbf{P}_i}^0$, $\underline{\mathbf{Q}}^0$, $\underline{\mathbf{Z}_i}^0$, $\mathcal{L}$, $\mathcal{S}$, p, ρ, α_i, $(i = 1, \ldots n)$.
Output: $\underline{\mathbf{X}}$.
Repeat
  For $i = 1$ to n **do**
    if $i \in \mathcal{L}$
      $\mathbf{\Gamma}\mathbf{\Lambda}\boldsymbol{W}^T = svd(\frac{1}{\rho}\mathbf{P}_{i(i)}^t + \mathbf{X}_{(i)}^t)$;
      $\underline{\mathbf{Z}_i}^{t+1} = fold_i\{\mathbf{\Gamma} shrink_p[diag(\mathbf{\Lambda}),\ \frac{\alpha_i}{\rho}]\boldsymbol{W}^T\}$;
    else
      $\underline{\mathbf{Z}_i}^{t+1} = shrink_p(\frac{1}{\rho}\underline{\mathbf{P}_i}^t + \underline{\mathbf{X}}^t,\ \frac{\alpha_i}{\rho})$;
    end
  end for
  Calculate $\underline{\mathbf{X}}^{t+1}$ using (7);
  Update $\underline{\mathbf{P}_i}^{t+1}$ $(i = 1, \ldots, n)$ using (8);
  Update $\underline{\mathbf{Q}}^{t+1}$ using (9).
Until a stopping criteria is met at iteration t.

$$\min_{\underline{\mathbf{X}}} \sum_{i \in \mathcal{S}} \alpha_i [G_{\rho,p}(\mathbf{\Omega}_i \mathbf{X}_{(i)})] + \sum_{i \in \mathcal{L}} \alpha_i \{G_{\rho,p}[\sigma(\mathbf{X}_{(i)})]\}$$
$$s.t.\ \underline{\mathbf{Y}} = \underline{\mathbf{X}} \times_1 \mathbf{\Phi}_1 \ldots \times_n \mathbf{\Phi}_n,$$

The update equation (7) involves derivation of efficient calculations. See the article for details.

4. Simulation Results



(a) MRI; (b) HSI

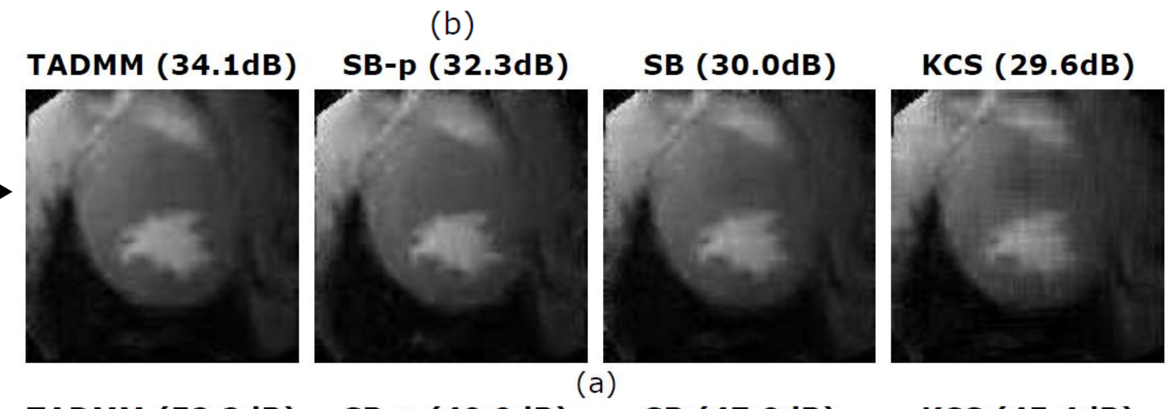
p=0.5.

Blue: Sparse & Low rank

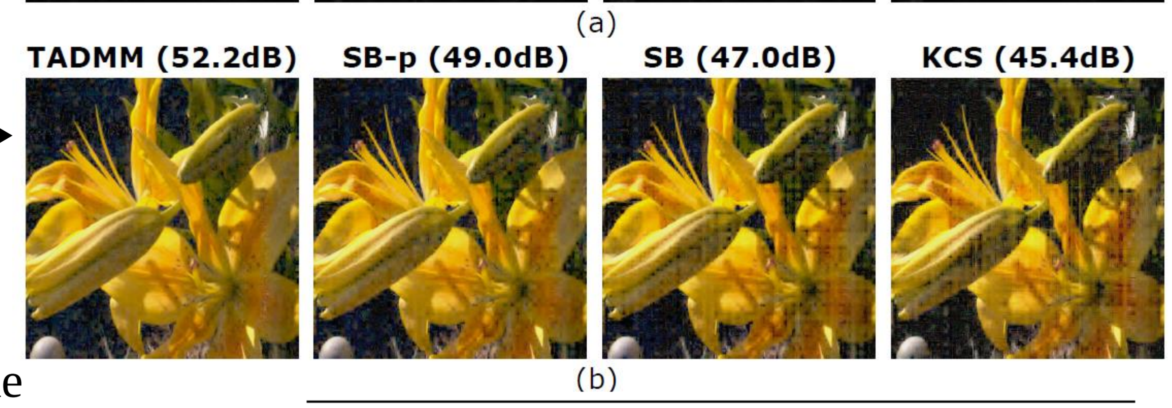
Red: Sparse

Black: Low rank

(a) MRI (128x128x128); sampling ratio: 0.3; p=0.5.



(b) HSI (1024x1024x32); sampling ratio: 0.1; p=0.5.



Accuracy and running time comparison for HSI; sampling ratio: 0.2.

Algorithm	PSNR (dB)	Time (h)
TADMM ($p = 0.2$)	56.22	1.73
TADMM ($p = 0.7$)	55.64	1.69
TADMM ($p = 1$)	53.95	1.39
SB ($p = 0.2$)	53.51	1.74
SB ($p = 0.7$)	53.84	1.70
SB ($p = 1$)	52.73	1.35
KCS	51.13	25.1
BKOMP	37.13	0.77
LRTC	36.82	0.54

To take advantage of the various structures in tensor modes, one can define various dictionaries and norms for different modes, provided that the proximity for the norms can be calculated.

* Given a matrix $\mathbf{A} \in \mathbb{R}^{J \times N_k}$, the mode-$k$ tensor by matrix product is defined as $\underline{\mathbf{Z}} = \underline{\mathbf{X}} \times_k \mathbf{A}$, where $\underline{\mathbf{Z}} \in \mathbb{R}^{N_1 \times \ldots \times N_{k-1} \times J \times N_{k+1} \times \ldots \times N_n}$ and it is calculated by: $\underline{\mathbf{Z}} = fold_i(\mathbf{A}\mathbf{X}_{(i)})$, where $fold_i(\cdot)$ is an operator that folds up a matrix along mode i to a tensor.

[1] R.Chartrand, "Nonconvex splitting for regularized low rank + sparse decomposition", IEEE Trans. Signal Process., vol. 60, no. 11, pp. 5810-5819, 2012.

UNIVERSITY OF CAMBRIDGE
Computer Laboratory