Iteratively Re-weighted L1-PCA of Tensor Data

Konstantinos Tountas*, Dimitris G. Chachlakis†, Panos P. Markopoulos†, and Dimitris A. Pados*

*I-SENSE and Dept. of Computer and Electrical Engineering & Computer Science, Florida Atlantic University

E-mail: {ktountas2017, dpados}@fau.edu

†Department of Electrical and Microelectronic Engineering, Rochester Institute of Technology

Email: dimitris@mail.rit.edu, panos@rit.edu

***Abstract*—Modern big and multi-modal datasets often contain corrupted entries or outliers. Standard methods for multi-way/tensor data analysis, relying on L2-norm formulations, are often very sensitive to such dataset corruptions. A new paradigm in data analysis suggests the use of robust L1-norm formulations instead. In this work, we present a novel method for iterative tensor L1-PCA with data re-weighting. The proposed method (i) returns significantly robust tensor bases and (ii) is capable to identify outlying, non-conforming entries of the data tensor. The presented algorithmic developments are validated with numerical studies on real-world data.**

INTRODUCTION

Multi-way arrays, also known as tensors, have been attracting extended documented interest in the fields of signal processing, data analytics, and machine learning [1], [2]. In many applications, tensors offer a natural way to organize and process multi-modal measurements, or measurements across diverse sensor configurations. Accordingly, a broad selection of methods for principal-component analysis (PCA) of tensor data have been proposed in the literature, such as Tucker and PARAFAC decomposition [3]–[5], that enable superior inference and learning. Tucker decomposition is the extension of matrix PCA to tensors and it is typically computed by means of the Higher-Order Singular-Value Decomposition (HOSVD) algorithm or the Higher-Order Orthogonal Iterations (HOOI) algorithm [4]–[6].

Most of the state-of-the-art tensor decomposition algorithms, such as HOSVD and HOOI, rely on L2-norm formulations (minimization of the L2-norm of the residual-error or, equivalently, maximization of the L2-norm of the multi-way projection) and, accordingly, they have been shown to be sensitive against faulty entries. The same sensitivity has also been documented in matrix PCA, a special case of Tucker for 2-way tensors (matrices). As both sensing modalities and dataset sizes expand, data corruptions (*e.g.*, in the form of outliers) become increasingly common. Accordingly, over the past years, an array of algorithms have been proposed for robust, corruption-resistant analysis of tensors. In [7], the authors tackle the problem of robust low-rank tensor recovery by proposing two efficient iterative methods with global convergence guarantees. Authors in [8] consider the problem of low-rank tensor factorization in the presence of outlying slabs and propose an alternating optimization framework to handle the minimization-based low-rank tensor factorization problem. In addition, regularizations and constraints can be incorporated to make use of *a priori* information on the latent loading factors. In [7], a robust low-rank tensor recovery method that relies on principal component pursuit (PCP) was proposed. The method [7], called Higher Order Robust PCA (HORPCA), is a direct extension of Robust PCA (RPCA) and aims to decompose the tensor into a low-rank plus a sparse part. However, these methods are shown to be sensitive to parameter selection and non-sparse noise. A two-step approach for low-rank tensor decomposition was proposed in [9]. In the first stage, HORPCA is utilized to obtain a low-rank estimate from the noisy tensor, while in the second stage, the low-rank estimate is denoised by using truncated HOSVD.

For matrix decomposition, L1-norm PCA [10], formulated by simple substitution of the L2 norm in PCA with the L1 norm, has exhibited solid robustness against heavily corrupted data in an array of applications [11]–[14]. Similar outlier resistance has been recently attained by algorithms for L1-norm reformulation of Tucker2 decomposition of 3-way tensors (L1-Tucker2) [15], [16]. In [17], two new methods for robust L1-norm Tucker decomposition of general-order tensors were proposed, namely L1-HOSVD and L1-HOOI.

In this paper, we propose a novel method that generates a sequence of iteratively refined L1-norm tensor subspaces. In each iteration, for each mode of the tensor, the conformity of each tensor element is inferred by its distance relative to the L1-norm tensor subspaces calculated in the previous iteration. Highly conforming samples tend to be nominal entries, while entries with lower conformity are more likely to be outliers. Then, all the samples of the original tensor dataset are weighted according to their conformity values and the L1-norm tensor subspaces are re-calculated according to L1-HOOI [17]. This way the contribution of outlying entires is iteratively suppressed, resulting in improved tensor subspace estimates. Moreover, the resulting conformity values can indicate faulty/outlying entries of the data tensor.

The work of K. Tountas and D. A. Pados was supported by the National Science Foundation, under grant CNS-1753406. The work of D. A. Pados was also supported by the Schmidt Family Foundation. The work of D. G. Chachlakis and P. P. Markopoulos was supported in part by the National Science Foundation, under grant OAC-1808582, and in part by the U.S. Air Force Office of Scientific Research, under the Dynamic Data Driven Applications Systems program.

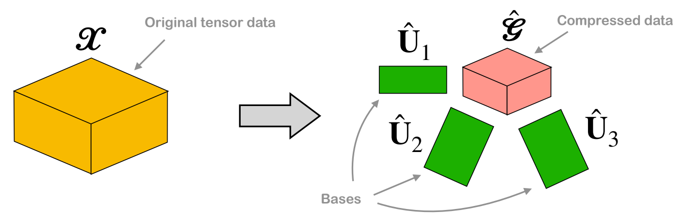

Fig. 1: L1-Tucker model for a 3-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$.

I. Technical Background

A. Definitions and Notation

The *order* of a tensor denotes the number of its dimensions, also known as *ways* or *modes*. A *fiber* is a vector extracted from a tensor by fixing all modes but one. *Matricization*, also known as *unfolding*, logically reorganizes tensors into other forms without changing the tensor values themselves. The mode-n matricization of a N-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is denoted by $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times I_1 I_2 \ldots I_{n-1} I_{n+1} \ldots I_N}$ and arranges the mode-n fibers of the tensor as columns of the resulting matrix.

B. L1-norm Tucker Decomposition

Given N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and low-dimensions $r_1, r_2, \ldots, r_N$ such that $r_n \leq \min\left(I_n, \tilde{I}_n\right)$ for all n, tensor L1-Tucker is formulated as [17], [18]

$$\underset{\{\mathbf{U}_n \in \mathbb{S}(I_n, r_n)\}_{n \in [N]}}{\text{maximize}} \left\| \boldsymbol{\mathcal{X}} \times_{n \in [N]} \mathbf{U}_n^T \right\|_1, \tag{1}$$

where $\tilde{I}_n = \prod_{k \in [N] \setminus n} I_k$, $\times_n$ denotes the mode-n tensor-to-matrix product, $[N]$ denotes the set $\{1, 2, \ldots, N\}$, $\mathbb{S}(I, r)$ is the set of $I \times r$ matrices with orthonormal columns, $\times_{n \in [N]} \mathbf{U}_n^T$ summarizes the mutli-mode product $\times_1 \mathbf{U}_1^T \cdots \times_N \mathbf{U}_N^T$, and $\|\cdot\|_1$ is the L1-norm, returning the summation of the absolute entries of its tensor argument. If the set of bases $\{\widehat{\mathbf{U}}_n\}_{n \in [N]}$ is a solution to (1), then the compressed data tensor (also known as Tucker core) is given by $\widehat{\boldsymbol{\mathcal{G}}} = \boldsymbol{\mathcal{X}} \times_{n \in [N]} \widehat{\mathbf{U}}_n^T \in \mathbb{R}^{r_1 \times r_2 \times \ldots \times r_N}$. A schematic illustration of this tensor analysis is offered in Fig. 1. The formulation in (1) constitutes a robust L1-norm-based analogous to standard Tucker decomposition, which employs instead the corruption-responsive L2-norm. Moreover, similar to Tucker decomposition, the exact solution to 1 remains to date unknown. For the special case of $N = 3$, $r_1 = r_2 = 1$, and $\hat{\mathbf{U}}_3$ fixed to $\mathbf{I}_{I_3}$ –i.e., no compression across mode 3– the exact solution to (1) was recently presented in [15]. For the general problem in (1) successful approximate algorithms have been proposed in the literature, such as L1-HOSVD and L1-HOOI [17], [18].

II. Proposed Iteratively Re-weighted L1-PCA

In the following, we describe our algorithmic developments for robust tensor decomposition. In the initialization step of the algorithm we set the tensor N bases $\{\hat{\mathbf{U}}_n^{(1)}\}_{n \in [N]}$ to the solution of L1-HOOI [17]. The resulting bases emphasize the subspaces spanned by the nominal (uncorrupted) entries of the original tensor $\boldsymbol{\mathcal{X}}$ and de-emphasizes entries that are contaminated with outlier data.

Next, for the n-th tensor unfolding for all $n \in [N]$, we project all columns of $\mathbf{X}_{(n)}$ onto the calculated bases $\{\hat{\mathbf{U}}_n^{(1)}\}_{n \in [N]}$ and measure

$$d_{n,i_n} = \left\| \hat{\mathbf{U}}_n^{(1)} \hat{\mathbf{U}}_n^{(1)T} \left[\mathbf{X}_{(n)}\right]_{:,i_n} \right\|_2, \tag{2}$$

for every $i_n \in [\tilde{I}_n]$, where for any $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\|\mathbf{A}\|_2$ is defined as $\sqrt{\sum_{i=1}^m \sum_{j=1}^n |\left[\mathbf{A}\right]_{i,j}|^2}$ and $|\cdot|$ denotes absolute value of a real number. Interestingly, we expect small d_{n,i_n} value if data vector $\left[\mathbf{X}_{(n)}\right]_{:,i_n}$ is an outlying and large d_{n,i_n} value if the data vector is nominal, or conforming.

After the calculation of the projections, we fold the d_{n,i_n} values into tensor form

$$\boldsymbol{\mathcal{W}}_n = \text{tensorization}\left(\mathbf{1}_{I_n}\left[d_{n,1}, d_{n,2}, \ldots, d_{n,\tilde{I}_n}\right], n\right), \tag{3}$$

where the operator $\text{tensorization}(\cdot, n)$ folds its argument along the n-th dimension (reversing the unfolding procedure). We say that the resulting tensor $\boldsymbol{\mathcal{W}}_n \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ contains the *conformity values* corresponding to each mode-n fiber of the original tensor $\boldsymbol{\mathcal{X}}$.

Having calculated the conformity tensors $\boldsymbol{\mathcal{W}}_n, \forall n \in [N]$, the individual entry conformity tensor is produced by additive weighting of the conformity tensors (according to assumed relative "importance") and max-min normalization to the range $[0, 1]$:

$$\boldsymbol{\mathcal{W}}^{(1)} = \frac{\sum_{n=1}^N \alpha_n \boldsymbol{\mathcal{W}}_n - \min\left(\sum_{n=1}^N \alpha_n \boldsymbol{\mathcal{W}}_n\right)}{\max\left(\sum_{n=1}^N \alpha_n \boldsymbol{\mathcal{W}}_n\right) - \min\left(\sum_{n=1}^N \alpha_n \boldsymbol{\mathcal{W}}_n\right)}, \tag{4}$$

where the weighting parameter α_n measures the importance of the n-th tensor mode, $\sum_{n \in [N]} a_n = 1$, $\min(\cdot)$ returns the minimum element of its tensor argument, and $\max(\cdot)$ returns the maximum element. The normalization in (4) leads to value 0 for the least conforming elements and value 1 for the most conforming ones.

In the next step of the algorithm, the original tensor dataset is globally weighted through the conformity tensor $\boldsymbol{\mathcal{W}}^{(1)}$ by element-by-element multiplication of $\boldsymbol{\mathcal{X}}$ with $\boldsymbol{\mathcal{W}}^{(1)}$. The new re-weighted dataset is denoted by

$$\boldsymbol{\mathcal{X}}^{(1)} = \boldsymbol{\mathcal{X}} \circ \boldsymbol{\mathcal{W}}^{(1)}, \tag{5}$$

where $\circ$ is the element-wise or Hadamard product. The refined L1-norm tensor bases of rank $\{r_n\}_{n \in [N]}$, $\{\hat{\mathbf{U}}_n^{(2)}\}_{n \in [N]}$, are calculated anew by means of L1-HOOI on $\boldsymbol{\mathcal{X}}^{(1)}$ initialized at $\{\hat{\mathbf{U}}_n^{(1)}\}_{n \in [N]}$.

Along this way, we continue the iterative generation of conformity weights $\boldsymbol{\mathcal{W}}^{(1)}, \boldsymbol{\mathcal{W}}^{(2)}, \ldots$, until numerical convergence to $\boldsymbol{\mathcal{W}}$ is observed at iteration $k \geq 2$; i.e.,

$$\boldsymbol{\mathcal{W}} = \boldsymbol{\mathcal{W}}^{(k)} \text{ such that } \left\| \boldsymbol{\mathcal{W}}^{(k)} - \boldsymbol{\mathcal{W}}^{(k-1)} \right\|_F < \epsilon \tag{6}$$

for some small $\epsilon > 0$. Algorithm 2 presents the complete pseudo-code for the calculation of $\boldsymbol{\mathcal{W}}$.

Algorithm 1 Proposed algorithm for iteratively re-weighted L1-PCA of tensor data

Input: $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$; $\{r_n\}_{n\in[N]}$; $\{\alpha_n\}_{n\in[N]}$; initialize $\{\hat{\mathbf{U}}_n^{(0)}\}_{n\in[N]}$ (*e.g.*, arbitrary or HOSVD); $\epsilon > 0$

```
Initialize l ← 1; W^(1) ← 1_{I1×I2×⋯×IN}; a ← ||W^(1)||_2; X^(1) ← X
while a > ε do
    {Û_n^(l)}_{n∈[N]} ← L1 − HOOI( X^(l), {Û_n^(l−1)}_{n∈[N]} )
    for n ∈ [N] do
        for i ∈ [Ĩ_n] do
            d_{n,i} ← || Û_n^(l) Û_n^(l)T [X_(n)]_{:,i} ||_2
        W_n ← tensorization( 1_{I_n} [d_{n,1}, …, d_{n,Ĩ_n}], n )
    W_temp ← Σ_{n∈[N]} α_n W_n; μ ← min(W_temp);
    ξ ← max(W_temp); l ← l + 1
    W^(l) ← (W_temp − μ)/(ξ − μ); X^(l) ← X ∘ W^(l)
    a ← || W^(l) − W^(l−1) ||_2
```

Output: $\boldsymbol{\mathcal{W}}^{(l)}$ and $\{\hat{\mathbf{U}}_n^{(l)}\}_{n\in[N]}$

Function: L1-HOOI$\left(\boldsymbol{\mathcal{X}}, \{\mathbf{U}_n^{(0)}\}_{n\in[N]}\right)$ [17]

```
l ← 1; a ← Σ_{n∈[N]} ||U_n^(0)||_2^2
while a > ε do
    for n ∈ [N] do
        B ← X ×_{m<n} U_m^(l)T ×_{k>n} U_k^(l−1)T;
        Q_0 ← U_n^(l−1); t ← 0
        for t ∈ [T] do
            [U, Σ, V^T] ← SVD( B_(n) sgn( B_(n)^T Q_{t−1} ) )
            Q_t ← [U]_{:,1:r_n} [V]^T_{:,1:r_n}
        U_n^(l) ← Q_T
    a ← Σ_{n∈[N]} || U_n^(l) U_n^(l)T − U_n^(l−1) U_n^(l−1)T ||_2^2
    l ← l + 1
```

III. Experimental Studies

We consider the problem of tensor reconstruction where our objective is to reconstruct the original (clean) tensor $\boldsymbol{\mathcal{X}}$ from a corrupted tensor $\boldsymbol{\mathcal{X}}^{\text{corr}}$ of the same dimensions. We utilize the Uber Pick Up dataset which contains timestamped coordinate data (GPS coordinates) from five Uber drivers in the greater New York area over a six month period. The dataset is inherently sparse, since rarely the drivers share the same pickup coordinates. In order to produce a more interpretable dataset we convert the GPS coordinates to their corresponding zip codes. We calculate the number of pickups per zip code per driver over each month. This transformation reduces the sparsity of the dataset, as well as the overall tensor dimension. The final data tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{150\times5\times6}$ contains the number of pickups of 150 different postal codes for 5 drivers over 6 months.

The dataset is then corrupted by replacing uniformly across each dimension the entries of the original tensor with integer values in the $[10, 100]$ range to create $\boldsymbol{\mathcal{X}}^{\text{corr}} \in \mathbb{R}^{150\times5\times6}$. The percentage of corrupted entries is controlled by a parameter ρ. The corruption simulates wrong entries or miscalculated position coordinates. Our goal is to decompose $\boldsymbol{\mathcal{X}}^{\text{corr}} \in \mathbb{R}^{150\times5\times6}$ and calculate the bases $\hat{\mathbf{U}}_1 \in \mathbb{R}^{150\times r_1}, \hat{\mathbf{U}}_2 \in \mathbb{R}^{5\times r_2}$, and $\hat{\mathbf{U}}_3 \in \mathbb{R}^{6\times r_3}$ and the core tensor $\hat{\boldsymbol{\mathcal{G}}} \in \mathbb{R}^{r_1\times r_2\times r_3}$, which are utilized to reconstruct the original uncorrupted tensor $\boldsymbol{\mathcal{X}}$ as

$$\hat{\boldsymbol{\mathcal{X}}} = \hat{\boldsymbol{\mathcal{G}}} \times_{n\in[N]} \hat{\mathbf{U}}_n. \tag{7}$$

We measure the squared reconstruction error (SRE) between the estimated $\hat{\boldsymbol{\mathcal{X}}}$ and the original clean $\boldsymbol{\mathcal{X}}$ as $\|\hat{\boldsymbol{\mathcal{X}}} - \boldsymbol{\mathcal{X}}\|_2^2$. We compare the performance of our proposed method versus standard L2-norm tensor decomposition techniques such as HOSVD and HOOI [4], [6], as well as HORPCA [7]. For HORPCA the parameters were set as in [7] to $r = 1/\sqrt{I_{\max}}$, where $I_{\max} = \max(I_1, I_2, I_3) = 150$ and $\mu = 10\text{std}\left(vec\left(\mathbf{X}_{(1)}\right)\right)$, where $\text{std}(\cdot)$ calculates the standard deviation of the input vector.

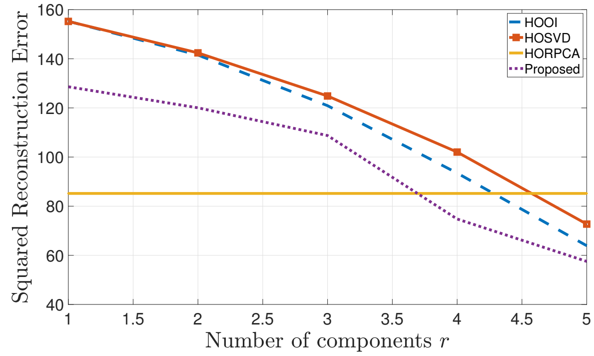

Fig. 3: Squared reconstruction error (SRE) versus number of components, r, for corruption percentage $\rho = 20\%$.

In Figure 3, we plot the SRE for corruption percentage $\rho = 20\%$, versus the number of components. For the L2-norm based methods (HOSVD and HOOI) and the proposed L1-norm based method the number of components is set to $r_1 = 5r, r_2 = r, r_3 = r$. This is done because the first dimension of out dataset is larger than the rest. It is clearly shown that the proposed method captures the low-rank characteristics of the nominal data better compared to the state-of-the-art and is able to outperform HORPCA for these tuning parameters.

Figure 4 depicts the SRE versus the corruption percentage parameter ρ. We vary the percentage of corrupted entries from 5% to 50% and observe that the proposed L1-norm based method significantly outperforms its counterparts, for every level of corruption.

IV. Conclusions

We presented a novel, iteratively re-weighted L1-norm tensor decomposition algorithm, which weighs each individual tensor entry according its conformity with respect to the whole tensor dataset. In each iteration, the L1-norm tensor bases

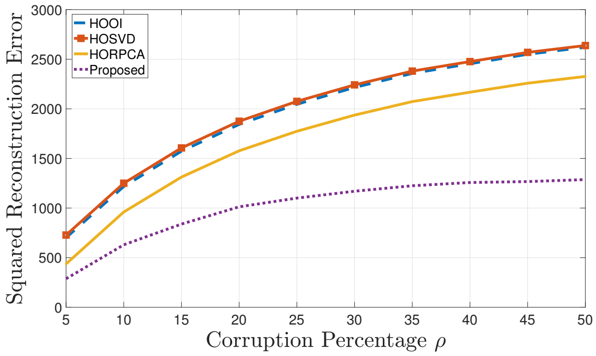

Fig. 4: Squared reconstruction error (SRE) versus corruption percentage, ρ, for fixed $r = 4$.

are computed computed by means of L1-HOOI in order to calculate the conformity of each data entry. Non-conforming entries are weighted down, resulting in bases that represent better the nominal data. The proposed method significantly enhances performance in tasks such as tensor reconstruction compared against Tucker-decomposition algorithms.

References

[1] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, 2017.

[2] Q. Song, H. Ge, J. Caverlee, and X. Hu, "Tensor completion algorithms in big data analytics," *ACM Trans. Knowledge Disc. from Data*, vol. 13, no. 1, p. 6, 2019.

[3] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Phychometrika*, vol. 31, pp. 279–311, 1966.

[4] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, pp. 455–500, 2009.

[5] K. Naskovska, Y. Cheng, A. L. de Almeida, and M. Haardt, "Efficient computation of the PARAFAC2 decomposition via generalized tensor contractions," in *Proc. IEEE Asilomar Conf. Signals Syst. and Comp.*, Pacific Grove, CA, Nov. 2018, pp. 323–327.

[6] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "On the best rank-1 and rank-(R1, R2, ... , Rn) approximation of higher-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 1324–1342, 2000.

[7] D. Goldfarb and Z. Qin, "Robust low-rank tensor recovery: Models and algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 1, pp. 225–253, Mar 2014.

[8] X. Fu, K. Huang, W. Ma, N. D. Sidiropoulos, and R. Bro, "Joint tensor factorization and outlying slab suppression with applications," *IEEE Trans. Signal Process.*, vol. 63, no. 23, pp. 6315–6328, Dec. 2015.

[9] S. E. Sofuoglu and S. Aviyente, "A two-stage approach to robust tensor decomposition," in *Proc. IEEE Stat. Signal Process. Workshop*, Freiburg, Germany, Jun. 2018, pp. 831–835.

[10] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for L1-subspace signal processing," *IEEE Trans. Signal Process.*, vol. 62, no. 19, pp. 5046–5058, Oct. 2014.

[11] Y. Liu and D. A. Pados, "Compressed-sensed-domain L1-PCA video surveillance," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 351–363, Mar. 2016.

[12] P. P. Markopoulos, M. Dhanaraj, and A. Savakis, "Adaptive L1-norm principal-component analysis with online outlier rejection," *IEEE J. Select Topics Signal Process.*, vol. 12, no. 6, pp. 1131–1143, Dec. 2018.

[13] P. P. Markopoulos, N. Tsagkarakis, D. A. Pados, and G. N. Karystinos, "Realified L1-PCA for direction-of-arrival estimation: Theory and algorithms," *EURASIP J. Adv. Signal Process.*, vol. 2019, Jun. 2019.

[14] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. A. Pados, "Efficient L1-norm principal-component analysis via bit flipping," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4252–4264, 2017.

[15] P. P. Markopoulos, D. G. Chachlakis, and E. E. Papalexakis, "The exact solution to rank-1 L1-norm Tucker2 decomposition," *IEEE Signal Process. Lett.*, vol. 25, no. 4, pp. 511–515, Apr. 2018.

[16] D. G. Chachlakis and P. P. Markopoulos, "Robust decomposition of 3-way tensors based on L1-norm," in *Proc. SPIE Defense and Commercial Sensing*, vol. 10658, Orlando, FL, Apr. 2018.

[17] D. G. Chachlakis, A. Prater-Bennette, and P. P. Markopoulos, "L1-norm tucker tensor decomposition," *IEEE Access*, 2019, [Early access online] Available DOI:10.1109/ACCESS.2019.2955134.

[18] P. P. Markopoulos, D. G. Chachlakis, and A. Prater-Bennette, "L_1-norm higher-order singular-value decomposition," in *Proc. IEEE Global Conf. Signal Info. Process.*, Anaheim, CA, Nov. 2018, pp. 1353–1357.