



HAL
open science

A semi-supervised rank tracking algorithm for on-line unmixing of hyperspectral images

Ludivine Nus, Sebastian Miron, Benoît Jaillais, Said Moussaoui, David Brie

► **To cite this version:**

Ludivine Nus, Sebastian Miron, Benoît Jaillais, Said Moussaoui, David Brie. A semi-supervised rank tracking algorithm for on-line unmixing of hyperspectral images. International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2020, May 2020, Barcelone, Spain. 10.1109/ICASSP40776.2020.9053931 . hal-02477639

HAL Id: hal-02477639

<https://hal.science/hal-02477639>

Submitted on 13 Feb 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A SEMI-SUPERVISED RANK TRACKING ALGORITHM FOR ON-LINE UNMIXING OF HYPERSPECTRAL IMAGES

Ludivine NUS¹, Sebastian MIRON¹, Benoît JAILLAIS², Said MOUSSAOUI³, David BRIE¹

¹ CRAN, Université de Lorraine, CNRS, Vandœuvre-lès-Nancy, France

² INRA-ONIRIS, StatSC Unité de Statistique, Sensométrie, Chimiométrie, Nantes, France

³ LS2N, École Centrale Nantes, CNRS, Nantes, France

¹ firstname.lastname@univ-lorraine.fr, ² benoit.jaillais@inra.fr, ³ said.moussaoui@ec-nantes.fr

ABSTRACT

This paper addresses the problem of rank tracking in real time hyperspectral image unmixing. Based on the On-line Alternating Direction Method of Multipliers (ADMM), we propose a new hyperspectral unmixing approach that integrates prior information as well as joint sparsity regularization, allowing to select only the active components on each sample of the image. This results in a semi-supervised algorithm, well adapted for on-line rank tracking for pushbroom imager. Experimental results on synthetic and real data sets demonstrate the effectiveness of our method for parameter estimation and rank change detection.

Index Terms— Hyperspectral imaging, Pushbroom imager, On-line semi-supervised unmixing, Alternating Direction Method of Multipliers, Rank tracking.

1. INTRODUCTION

In this paper, we address the problem of on-line spectral unmixing of hyperspectral images acquired by a *pushbroom* imager [1]. This problem is encountered, for example, in remote sensing applications but also industrial systems, particularly for product quality control. The pushbroom technique is a line scanning technique that can simultaneously acquire the whole stripe of the scene, resulting in a 2D slice of a hyperspectral image with both spatial and spectral dimensions.

Spectral unmixing is an important technique for hyperspectral data analysis as it decomposes an image observed at several wavelengths, into a collection of spectral signatures (also called *endmembers*) and their relative proportions (also called *abundances*). Due to the non-negativity of hyperspectral data, Non-negative Matrix Factorization (NMF) [2] has been widely used for this type of problem. However, classical NMF runs in batch mode and cannot be directly

applied to on-line processing and therefore, a family of *on-line* NMF algorithms has been developed (see *e.g.* [3–6]). They consist in recursively updating endmembers and abundances as the data increases, while ensuring a low and controlled computational complexity. We have recently proposed the OMDC-ADMM algorithm (for On-line Minimum Dispersion Constraint-ADMM) [7], which is an adaptation of [4] to pushbroom hyperspectral imaging system. Nevertheless, this method works well in practice if the decomposition rank (*i.e.* the number of endmembers) does not change between consecutive slices of the image; this is a strong assumption that is not always satisfied in practical applications where one or more sources can disappear / re-appear from one slice to another.

In batch mode (see *e.g.* [8–12]), several approaches have been proposed to select only variables that best describe the data. They model pixel observations as linear combinations of spectra from a library (potentially very large), built-up using a learning procedure or *a priori* knowledge on the observed phenomenon. The abundances are obtained by minimizing a cost function, containing a data fitting term and a sparsity-inducing regularizer, usually the $\ell_{2,0}$ norm, or its convex relaxation, the $\ell_{2,1}$ norm. This allows the zeroing of abundances corresponding to non-active endmembers of the image. This paper aims at investigating the interest of these approaches in an on-line setup.

In this article, we introduce a new on-line unmixing algorithm based on ADMM, devoted to track the evolution of the number of active endmembers (the rank) at each slice acquired by the pushbroom imager. The contributions of this algorithm compared to our previous work [7] and to the state-of-the-art on-line approaches are as follows: *i*) taking into account a slow evolution of endmembers, in a neighborhood defined by the vectors of the hyperspectral library; *ii*) the joint use of the $\ell_{2,1}$ and $\ell_{1,1}$ norms, applied to the abundance matrix which respectively promote row sparsity (low rank) and spatial sparsity on the active rows (for source support reconstruction with zero background). This algorithm is considered as semi-supervised since *a priori* knowledge, *i.e.* the hyper-

We benefit from the support of the ANR-OPTIFIN (Agence Nationale de la Recherche-OPTimisation des FINitions) and CNRS Imag'in ALOHA project.

spectral library, is added to the model. In the following, we will refer to the proposed algorithm as *OSS-ADMM*, for *On-line Semi-Supervised ADMM*.

The remainder of the paper is organized as follows: Section 2 is devoted to the derivation of the proposed algorithm; Section 3 presents several results obtained on simulated and real hyperspectral images. Some conclusions are drawn in Section 4.

2. PROPOSED APPROACH

2.1. Pushbroom data model

The proposed approach is an adaptive version of the classical NMF [2] that factorizes a non-negative data matrix \mathbf{X} as $\mathbf{X} \approx \mathbf{S}\mathbf{A}$, where \mathbf{S} and \mathbf{A} , of size $L \times R$ and $R \times P$, respectively, have also non-negative entries. In hyperspectral imaging, the P columns of \mathbf{X} represent the data samples recorded at L wavelengths. \mathbf{S} is a matrix containing the R normalized endmembers on its columns and \mathbf{A} is a matrix containing on its columns the abundances for the recorded samples. The principle of the proposed on-line method is to alternatively update the endmember and abundance matrices when a new sample arrives at time instant $k + 1$. One way to handle the pushbroom data is to unfold the hyperspectral image as shown in Figure 1, where $\tilde{\mathbf{X}}^{(1)}$ is the first slice of the hyperspectral image and $\tilde{\mathbf{X}}^{(k)}$ is the k^{th} slice. The entire data set at time instant $k + 1$, i.e., $\mathbf{X}^{(k+1)}$, can be represented as the concatenation of the first k samples with the new incoming sample i.e., $\mathbf{X}^{(k+1)} = [\mathbf{X}^{(k)} \tilde{\mathbf{X}}^{(k+1)}]$. Similarly, we define $\mathbf{S}^{(k+1)} = [\mathbf{S}^{(k)} \tilde{\mathbf{S}}^{(k+1)}]$ and $\mathbf{A}^{(k+1)} = [\mathbf{A}^{(k)} \tilde{\mathbf{A}}^{(k+1)}]$.

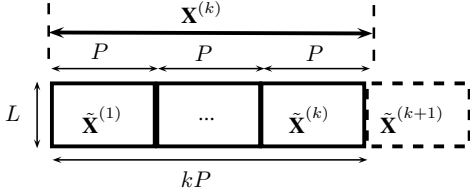


Fig. 1: Unfolded pushbroom hyperspectral image.

2.2. OSS-ADMM algorithm

In order to estimate $\tilde{\mathbf{S}}^{(k+1)}$ and $\tilde{\mathbf{A}}^{(k+1)}$, we consider the following cost function:

$$\begin{aligned} \mathcal{J}^{(k+1)}(\tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{A}}^{(k+1)}) &= \alpha \frac{1}{2} \sum_{\ell=1}^k \|\tilde{\mathbf{X}}^{(\ell)} - \tilde{\mathbf{S}}^{(k+1)} \tilde{\mathbf{A}}^{(\ell)}\|_F^2 \\ &+ (1 - \alpha) \frac{1}{2} \|\tilde{\mathbf{X}}^{(k+1)} - \tilde{\mathbf{S}}^{(k+1)} \tilde{\mathbf{A}}^{(k+1)}\|_F^2 + v \|\tilde{\mathbf{A}}^{(k+1)}\|_{2,1} \\ &+ \gamma \|\tilde{\mathbf{A}}^{(k+1)}\|_{1,1} + \frac{\omega}{2} \|\mathbf{B} - \tilde{\mathbf{S}}^{(k+1)}\|_F^2, \end{aligned} \quad (1)$$

where the coefficient α controls the trade-off between the contribution of the old and the new samples (tracking capability). The cost function (1) has five terms: the first two

terms allow to fit the model to the first k samples and to the new incoming sample respectively; a natural assumption is that endmembers vary only slightly between each slice i.e. $\tilde{\mathbf{S}}^{(k+1)} \approx \tilde{\mathbf{S}}^{(k)}$, $\forall k$ [3]. However, in the practical implementation of the algorithm, $\tilde{\mathbf{S}}$ is updated at each new slice, as explained in the sequel. The third term is a $\ell_{2,1}$ norm that promotes row sparsity on $\tilde{\mathbf{A}}^{(k+1)}$. The $\ell_{1,1}$ norm used in the fourth term reinforces sparsity on the active rows of $\tilde{\mathbf{A}}^{(k+1)}$, especially in the case of source support reconstruction with zero background. These norms are special cases of the $\ell_{q,p}$ norm, defined as $\|\tilde{\mathbf{A}}\|_{q,p} = \left(\sum_{r=1}^R \|\tilde{\mathbf{a}}^r\|_q^p \right)^{\frac{1}{p}}$, where $\tilde{\mathbf{a}}^r$ is the r^{th} row of $\tilde{\mathbf{A}}$. The last term forces the endmembers to be close to the hyperspectral library matrix \mathbf{B} , which gathers the R reference endmembers on its columns. Matrix \mathbf{B} also allows to remove the order indeterminacy between the endmembers, when their number evolves from one slice to another. In the case where the spectra of \mathbf{B} are not all present simultaneously in a slice, the sparsity term imposed on the abundance matrix will force to zero the rows corresponding to the non-active endmembers. v , γ and ω are hyperparameters corresponding to the different penalties. In order to minimize the cost function (1), we adopt an ADMM approach [13]; we introduce two auxiliary variables $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ and consider the following equivalent problem:

$$\begin{aligned} \underset{\tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{A}}^{(k+1)}, \tilde{\mathbf{V}}^{(k+1)}, \tilde{\mathbf{U}}^{(k+1)}}{\text{minimize}} \quad & \mathcal{J}^{(k+1)}(\tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{A}}^{(k+1)}) \\ & + \mathbb{I}_{\mathbb{R}_+}(\tilde{\mathbf{V}}^{(k+1)}) + \mathbb{I}_{\mathbb{R}_+}(\tilde{\mathbf{U}}^{(k+1)}), \\ \text{subject to} \quad & \tilde{\mathbf{S}}^{(k+1)} = \tilde{\mathbf{U}}^{(k+1)} \text{ and } \tilde{\mathbf{A}}^{(k+1)} = \tilde{\mathbf{V}}^{(k+1)}, \end{aligned} \quad (2)$$

where $\mathbb{I}_{\mathbb{R}_+}$ represents the indicator function of \mathbb{R}_+ ensuring the non-negativity of the endmembers and abundances. The augmented Lagrangian \mathcal{L} corresponding to the problem (2) is given by the expression (3), where $\rho > 0$ is a tuning parameter which controls the convergence speed of the method. $\tilde{\Pi}^{(k+1)}$ and $\tilde{\Lambda}^{(k+1)}$ are the scaled version of the dual variables. We first minimize the augmented Lagrangian (3) with respect to $(\tilde{\mathbf{A}}^{(k+1)}, \tilde{\mathbf{V}}^{(k+1)})$ and $(\tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{U}}^{(k+1)})$, and then update the dual variables $\tilde{\Pi}^{(k+1)}$ and $\tilde{\Lambda}^{(k+1)}$. **Algorithm 1** summarizes the proposed OSS-ADMM. It includes two main loops: the outer loop produces estimates of all parameters at each slice k . These estimates are iteratively refined in the inner loop using a fixed number of iterations N_{iter} . Note that the $\ell_{2,1}$ norm can be expressed in the following matrix form: $\|\tilde{\mathbf{A}}\|_{2,1} = 2 \text{trace}(\tilde{\mathbf{A}}^T \tilde{\mathbf{Q}} \tilde{\mathbf{A}})$ [14], where $\tilde{\mathbf{Q}}$ is a diagonal matrix with the r^{th} diagonal element defined by $\frac{1}{\|\tilde{\mathbf{a}}^r\|_2 + \delta}$. The parameter δ is a small value to avoid division by zero and $\|\cdot\|_2$ denotes the ℓ_2 norm. The identity matrix is denoted by \mathbf{I} and $\mathbf{1}$ is a matrix of ones.

$$\begin{aligned} \mathcal{L}(\tilde{\mathbf{A}}^{(k+1)}, \tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{V}}^{(k+1)}, \tilde{\mathbf{U}}^{(k+1)}, \tilde{\Pi}^{(k+1)}, \tilde{\Lambda}^{(k+1)}) &= \mathcal{J}^{(k+1)}(\tilde{\mathbf{S}}^{(k+1)}, \tilde{\mathbf{A}}^{(k+1)}) + \frac{\rho}{2} \|\tilde{\mathbf{A}}^{(k+1)} - \tilde{\mathbf{V}}^{(k+1)} + \tilde{\Pi}^{(k+1)}\|_F^2 - \frac{\rho}{2} \|\tilde{\Pi}^{(k+1)}\|_F^2 \\ &+ \frac{\rho}{2} \|\tilde{\mathbf{S}}^{(k+1)} - \tilde{\mathbf{U}}^{(k+1)} + \tilde{\Lambda}^{(k+1)}\|_F^2 - \frac{\rho}{2} \|\tilde{\Lambda}^{(k+1)}\|_F^2 + \mathbb{I}_{\mathbb{R}_+}(\tilde{\mathbf{V}}^{(k+1)}) + \mathbb{I}_{\mathbb{R}_+}(\tilde{\mathbf{U}}^{(k+1)}). \end{aligned} \quad (3)$$

Algorithm 1 OSS-ADMM

Inputs: $\mathbf{X}, \mathbf{B}, R, \alpha, v, \gamma, \omega, \rho, \delta, N_{iter}$;

Initialization: $k = 0, \mathbf{N} = \text{zeros}(L, R), \mathbf{M} = \text{zeros}(R, R), \tilde{\mathbf{S}} = \text{rand}(L, R), \tilde{\mathbf{V}} = \text{zeros}(R, P), \tilde{\mathbf{U}} = \text{zeros}(L, R), \tilde{\mathbf{\Pi}} = \text{zeros}(R, P), \tilde{\mathbf{\Lambda}} = \text{zeros}(L, R), \tilde{\mathbf{Q}} = \text{eye}(R, R), \mathbf{A} = [], \mathbf{S} = []$;

Outputs: \mathbf{A}, \mathbf{S} ;

while New sample $k + 1$ available **do**

$$\tilde{\mathbf{X}} = \tilde{\mathbf{X}}^{(k+1)};$$

 $t = 1$;

while $t < N_{iter}$ **do**

$$\tilde{\mathbf{A}} = \left((1 - \alpha) \tilde{\mathbf{S}}^T \tilde{\mathbf{S}} + \rho \mathbf{I} + 2v \tilde{\mathbf{Q}} \right)^{-1}$$

$$\left((1 - \alpha) \tilde{\mathbf{S}}^T \tilde{\mathbf{X}} + \rho (\tilde{\mathbf{V}} - \tilde{\mathbf{\Pi}}) - \gamma \mathbf{1} \right);$$

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \frac{1}{\|\tilde{\mathbf{a}}^1\|_2 + \delta} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \frac{1}{\|\tilde{\mathbf{a}}^R\|_2 + \delta} \end{bmatrix};$$

$$\tilde{\mathbf{V}} = \max(\mathbf{0}, \tilde{\mathbf{A}} + \tilde{\mathbf{\Pi}});$$

$$\tilde{\mathbf{\Pi}} \leftarrow \tilde{\mathbf{\Pi}} + \tilde{\mathbf{A}} - \tilde{\mathbf{V}};$$

$$\tilde{\mathbf{N}} = \alpha \mathbf{N} + (1 - \alpha) (\tilde{\mathbf{X}} \tilde{\mathbf{A}}^T);$$

$$\tilde{\mathbf{M}} = \alpha \mathbf{M} + (1 - \alpha) (\tilde{\mathbf{A}} \tilde{\mathbf{A}}^T);$$

$$\tilde{\mathbf{S}} = \left(\tilde{\mathbf{N}} + \rho (\tilde{\mathbf{U}} - \tilde{\mathbf{A}}) + \omega \mathbf{B} \right) (\tilde{\mathbf{M}} + \rho \mathbf{I} + \omega \mathbf{I})^{-1};$$

$$\tilde{\mathbf{U}} = \max(\mathbf{0}, \tilde{\mathbf{S}} + \tilde{\mathbf{A}});$$

$$\tilde{\mathbf{\Lambda}} \leftarrow \tilde{\mathbf{\Lambda}} + \tilde{\mathbf{S}} - \tilde{\mathbf{U}};$$

 $t \leftarrow t + 1$;

end while

$$\mathbf{N} = \tilde{\mathbf{N}}; \mathbf{M} = \tilde{\mathbf{M}};$$

$$\mathbf{A} \leftarrow [\mathbf{A}, \tilde{\mathbf{A}}]; \mathbf{S} \leftarrow [\mathbf{S}, \tilde{\mathbf{S}}];$$

end while

It can be shown, using a similar approach to [7, 15, 16], that any stationary point generated by a sequence of iterations of OSS-ADMM satisfies the Karush-Kuhn-Tucker conditions (for space reasons, the proof is not provided in this version of the paper). This statement is valid only in the case of our working assumption *i.e.* $\tilde{\mathbf{S}}^{(k+1)} \approx \tilde{\mathbf{S}}^{(k)}$, in other words, when the steady state is reached. It does not provide any insights into the rank tracking behavior. This point will be addressed through numerical simulations in the following section.

3. EXPERIMENTS

3.1. Synthetic data

In this section, a number of experiments are conducted on simulated hyperspectral data. The first experiment is designed to study the influence of parameters v, γ and ω on the performance of OSS-ADMM, in a simple scenario. We simulated a hyperspectral image of size $119 \times 40 \times 250$, consisting of a non-negative mixture of $R = 3$ endmembers that do not change over time (from one slice to another). The rank of the decomposition is identical for each slice ($= 3$). Here, 119 corresponds to the number of wavelengths and 40×250 to the

spatial dimensions. Each new sample is a 119×40 slice of the hyperspectral image. Data are corrupted by a low level noise. The first objective was to study the ability of sparsity constraints to deactivate endmembers not present in the analyzed image, in the case where the value of R is overestimated. The hyperspectral library \mathbf{B} gathers the three real spectra plus a fourth one that is not present in any sample of the image. The values of the different parameters were set to $R = 4, \alpha = 0.9, \omega = 1, \rho = 0.001$ and $N_{iter} = 50$. Figure 2(a) illustrates the rate of deactivation of the fourth endmember *vs.* the number of slices, for different values of the couple of hyperparameters (v, γ) . To assess the deactivation effect, we computed the following quantity: $\sum_{p=1}^P |\tilde{a}_{rp}|$, with $r = 4$ *i.e.*, the sum of all elements on the fourth row of the abundance matrix (the row corresponding to the non-existing endmember). We note that as γ increases, the endmember is deactivated much faster. However, the choice of the values for parameters v and γ must be done with care: from experience, a low value of v gives more accurate results than a strong value. We did not find yet a completely satisfactory explanation for this phenomenon, but we conjecture that this is because of the expression of the $\ell_{2,1}$ norm used in this paper, that yields a biased estimator (specifically the term δ , in the update of $\tilde{\mathbf{Q}}$). Figure 2(b) represents, as a bargraph, the Root Mean Square Error (RMSE), between true and estimated abundances, as a function of ω . v and γ were set to 0.0001 and 0.002, respectively. As the value of ω increases, the RMSE decreases, indicating that the abundances are better estimated by the algorithm. When $\omega = 0$, the value of RMSE is high because the sparsity constraints alone are not sufficient to regularize the solution, and may generate order permutation of the endmembers during the processing. A value of $\omega \geq 1$ seems to be adapted.

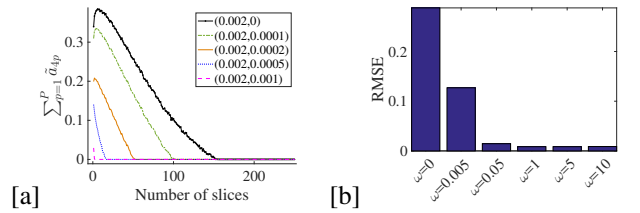


Fig. 2: Influence of the hyperparameters

In the second experiment, we compared the performances of OSS-ADMM to those of OMDC-ADMM [7], on a simulated hyperspectral image composed of three endmembers, where the rank of the decomposition evolves between slices. For OMDC-ADMM, the rank was set to $R = 3$. Figure 3 shows the abundance maps obtained by OSS-ADMM and OMDC-ADMM, compared to the ground truth. We notice that OMDC-ADMM totally fails to estimate abundances, whereas OSS-ADMM provides accurate estimates according to our expectations. Figure 4 illustrates the actual and estimated ranks of the decomposition *vs.* the number of slices.

The two plots are almost completely superposed meaning that OSS-ADMM is able to accurately track the evolution of the number of endmembers over time (from one slice to another).

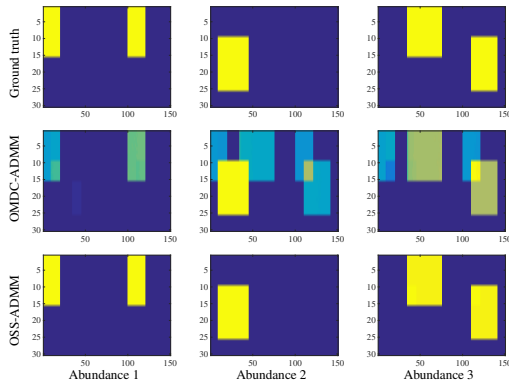


Fig. 3: Abundance maps

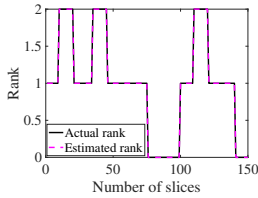


Fig. 4: Actual rank and estimated rank by OSS-ADMM

3.2. Real data

To validate the performance of OSS-ADMM, we applied it to two real hyperspectral images. The first real scenario is the tracking of the diffusion of water in cereal products during the drying stage. This is a problem of high practical interest in food industry because the mechanism of water transport is poorly known and can affect strongly the quality of products. In food products, the water presents two main states: Free Water (FW) and Bound Water (BW). The evacuation of the latter within a product (by drying, for example) must be meticulously carried out in order to avoid the alteration of the product. Experiments were conducted on a biscuit where two drops of water were placed on its surface. Hyperspectral cubes of this biscuit were acquired by a pushbroom imager (in spectral range 900-2500 nm) every minute, for 73 minutes. Each cube was processed by OSS-ADMM. The hyperspectral library \mathbf{B} was built-up using *a priori* knowledge on the spectra of the different components present in the analyzed samples; we used then the same \mathbf{B} for all 73 hyperspectral images. The OSS-ADMM parameters were set to $R = 4$, $\alpha = 0.9$, $\nu = 10^{-5}$, $\gamma = 0.045$, $\omega = 30$, $\rho = 0.001$ and $N_{iter} = 50$. The results obtained on the second data cube are shown in Figure 5 (left column). Because of the lack of space, we do not show the results of the other hyperspectral cubes. By expertise, the first two abundance maps are respectively FW and BW, and the last two characterize cellulose components of the biscuit. It is interesting to observe the evolution of the rank of the decomposition, computed from the estimates

provided by OSS-ADMM. For example, we find that the four endmembers are present simultaneously in slices 15 to 120 and 330 to 410, while the two states of water disappear in slices 130 to 320, which quite corresponds to the regions that can be observed on the abundance maps. For the considered dataset, the processing time by OSS-ADMM was of about 3 s, on a 2.7 GHz Macbook Pro with 4-core processor and 16 GB of RAM.

The second dataset is the hyperspectral image *Samson* with the ground truth available in [17]. This image is composed of three endmembers: water, tree and soil. For this scenario, the parameters of OSS-ADMM were set as follows: $R = 3$, $\alpha = 0.9$, $\nu = 10^{-5}$, $\gamma = 0.008$, $\omega = 50$, $\rho = 0.001$ and $N_{iter} = 50$. The matrix \mathbf{B} was constructed using *a priori* knowledge on the spectral signatures of the three components. The results generated by OSS-ADMM are shown in Figure 5 (right column). We observe that OSS-ADMM is able to closely follow the appearance and disappearance of endmembers during the processing, while generating very accurate abundance maps close to the ground truth (see [17]), for a processing time of about 0.5 s.

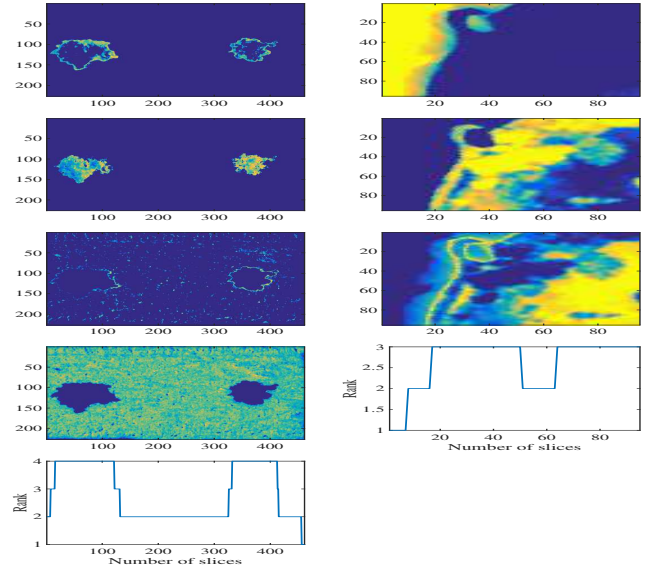


Fig. 5: Abundance maps and rank tracking by OSS-ADMM for the real datasets

4. CONCLUSION

We proposed a new algorithm (OSS-ADMM) that estimates and tracks the evolution of the number of endmembers for on-line pushbroom-like acquisition systems. By using a hyperspectral library and sparsity regularizations, OSS-ADMM is able to estimate for each new sample only the active endmembers and thus, to track the rank of the decomposition over time. Tests on simulated data have shown that this new algorithm outperforms state-of-the-art on-line processing methods, in terms of estimation quality. The performance of the proposed algorithm was also validated on two real data sets.

5. REFERENCES

- [1] Guolan Lu and Baowei Fei, “Medical hyperspectral imaging: a review,” *Journal of Biomedical Optics*, vol. 19, no. 1, pp. 010901, 2014.
- [2] Daniel D Lee and Hyunjune S Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, vol. 401, no. 6755, pp. 788, 1999.
- [3] Serhat S Bucak and Bilge Günsel, “Incremental subspace learning via non-negative matrix factorization,” *Pattern Recognition*, vol. 42, no. 5, pp. 788–797, 2009.
- [4] Guoxu Zhou, Zuyuan Yang, Shengli Xie, and Jun-Mei Yang, “On-line blind source separation using incremental non-negative matrix factorization with volume constraint,” *IEEE Transactions on Neural Networks*, vol. 22, no. 4, pp. 550–560, 2011.
- [5] Yi Wu, Bin Shen, and Haibin Ling, “Visual tracking via on-line non-negative matrix factorization,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 374–383, 2014.
- [6] Ludivine Nus, Sebastian Miron, and David Brie, “On-line blind unmixing for hyperspectral pushbroom imaging systems,” in *10th Workshop on Statistical Signal Processing*, 2018, pp. 418–422.
- [7] Ludivine Nus, Sebastian Miron, and David Brie, “Démélange d’images hyperspectrales à l’aide de la NMF en-ligne avec contrainte de dispersion minimale,” in *27ème Colloque GRETSI Traitement du Signal et des Images*, 2019.
- [8] Marian-Daniel Iordache, José M Bioucas-Dias, and Antonio Plaza, “Collaborative sparse regression for hyperspectral unmixing,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 341–354, 2013.
- [9] Wei Tang, Zhenwei Shi, Ying Wu, and Changshui Zhang, “Sparse unmixing of hyperspectral data using spectral a priori information,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 2, pp. 770–783, 2014.
- [10] Jun Li, José M Bioucas-Dias, Antonio Plaza, and Lin Liu, “Robust collaborative non-negative matrix factorization for hyperspectral unmixing,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6076–6090, 2016.
- [11] Dong Wang, Jin-Xing Liu, Ying-Lian Gao, Jiguo Yu, Chun-Hou Zheng, and Yong Xu, “An NMF- $\ell_{2,1}$ -norm constraint method for characteristic gene selection,” *PLOS One*, vol. 11, no. 7, pp. e0158494, 2016.
- [12] Shaoquan Zhang, Jun Li, Zebin Wu, and Antonio Plaza, “Spatial discontinuity-weighted sparse unmixing of hyperspectral images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 10, pp. 5767–5779, 2018.
- [13] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [14] Xu Han, Laurent Albera, Amar Kachenoura, Lotfi Senhadji, and Huazhong Shu, “Low rank canonical polyadic decomposition of tensors based on group sparsity,” in *25th IEEE European Signal Processing Conference*, 2017, pp. 668–672.
- [15] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge University Press, 2004.
- [16] Kejun Huang, Nicholas D Sidiropoulos, and Athanasios P Liavas, “A flexible and efficient algorithmic framework for constrained matrix and tensor factorization,” *IEEE Transaction Signal Processing*, vol. 64, no. 19, pp. 5052–5065, 2016.
- [17] Feiyun Zhu, “Hyperspectral unmixing: ground truth labeling, datasets, benchmark performances and survey,” *arXiv preprint arXiv:1708.05125*, 2017.