Online deconvolution for industrial hyperspectral imaging systems

Yingying Song, El-Hadi Djermoune, Jie Chen, Cédric Richard, David Brie

ONLINE DECONVOLUTION FOR INDUSTRIAL HYPERSPECTRAL IMAGING SYSTEMS*

YINGYING SONG†, EL-HADI DJERMOUNE†, JIE CHEN‡, CÉDRIC RICHARD§, AND DAVID BRIE†

Abstract. This paper proposes a hyperspectral image deconvolution algorithm for the online restoration of hyperspectral images as provided by wiskbroom and pushbroom scanning systems. We introduce a least-mean-squares (LMS)-based framework accounting for the convolution kernel non-causality and including non-quadratic (zero attracting and piece-wise constant) regularization terms. This results in the so-called sliding block regularized LMS (SBR-LMS) which maintains a linear complexity compatible with real-time processing in industrial applications. A model for the algorithm mean and mean-squares transient behavior is derived and the stability condition is studied. Experiments are conducted to assess the role of each hyper-parameter. A key feature of the proposed SBR-LMS is that it outperforms standard approaches in low SNR scenarios such as ultra-fast scanning.

Key words. hyperspectral image, online deconvolution, LMS, ZA-LMS

AMS subject classifications. 68U10

1. Introduction.

Hyperspectral imaging has received considerable attention in the last decade as it combines the power of digital imaging and spectroscopy. Every pixel in a hyperspectral image provides local spectral information about a scene of interest across a large number of contiguous bands. This information can be used to characterize objects with great precision and details in a number of areas, including agricultural monitoring, industrial inspection, and defense. The core characteristics of hyperspectral images raise new data processing issues ranging from image restoration to pattern recognition [27, 10, 2, 38]. Several sensing techniques have been devised for hyperspectral imaging. They can be categorized into four main groups [46, 33]: whiskbroom (point scan), pushbroom (line scan), tunable filter (wavelength scan), and snapshot. Pushbroom systems are used in many areas such as food safety [13, 28], geo-referencing [7] and material sorting [42, 35]. The main feature of pushbroom imaging systems is that the hyperspectral data cubes are acquired slice by slice, sequentially in time (which in fact represents the so-called along track spatial dimension). This paper is a first step towards the development of advanced online (real-time) hyperspectral image processing methods required in industrial processes that aim at controlling and sorting input materials right after each line scanning. For example, this is the case for paper and chipboard plants that use recycled materials in the manufacturing process. The aim of this paper is to address the fast online (sequential) deconvolution of hyperspectral images captured by pushbroom imaging systems. However, it is worth noting that the proposed sequential deconvolution algorithm can be easily extended

*A short and preliminary version of this work appears in the conference publication [39].

Funding: This work has been supported by the FUI AAP 2015 Trispirabois Project, the Conseil Régional de Lorraine, the GDR ISIS CNRS, and the CNRS Imag'in ALOhA Project. The work of J. Chen was supported in part by National Natural Science Foundation of China (NSFC) under grant 61671382 and in part by NSF of Shenzhen under grant JCYJ20170301553158 73.

†Centre de Recherche en Automatique de Nancy (CRAN), Université de Lorraine, CNRS, Boulevard des Aiguillettes B.P. 239 F-54506 Vandœuvre-lès-Nancy, France (first-name.lastname@univ-lorraine.fr).

‡School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China (dr.jie.chen@ieee.org).

§Université Côte d'Azur, CNRS, OCA, Nice, France (cedric.richard@unice.fr).

to whiskbroom (point-by-point) systems.

Multichannel image restoration was carried out with Wiener methods in [29, 22]. Other strategies such as those in [23, 25, 27, 50] were also introduced, but only in an offline setting. Let us mention the works of Jemec *et al.* [30, 31] which specifically address the calibration and (offline) deconvolution of pushbroom hyperspectral images. The idea of restoring images in a sequential way can be traced back to the end of the 70's and has suscited extensive works up to the mid 90's. Early works on 2D-image restoration are due to Woods and Radewan, who developed a Kalman-filter (KF)-based approach allowing a pixel by pixel image restoration [48]. This approach was then modified in [47] to lower the computational cost, resulting in the so-called reduced update KF which was then extended to the restoration of color images [43]. An alternative KF-based approach was proposed by Biemond *et al.* [4] to perform a line-by-line image restoration and which, due to Toeplitz matrix structures in the image and observation models, can be efficiently implemented in the Fourier domain. This approach was then extended to multichannel image restoration [21] allowing a slice by slice restoration. The interest of KF-based image restoration started to decrease in the mid 90's with the growth of the computational capabilities. This led to the development of new iterative off-line image restoration approaches handling efficiently non-quadratic terms (including edge-preserving regularization) as in [8, 3, 5]. There is currently a renewed interest in extending these approaches to online estimation as [40, 24, 15, 14] that comes from the need for processing tools that are able to handle very large data sets. However, extending such approaches to online hyperspectral image deconvolution can be tricky. For example, the online ADMM algorithm proposed in [40] addresses the problem of estimating fixed parameters and derives online algorithms that converge to the same solution as the batch ADMM. Basically they consist of (sub)-gradient-based algorithms with a step size that decreases to zero as the number of time samples increases. This may be irrelevant for online hyperspectral image deconvolution since, in that case, the key feature is the tracking capability of the algorithms.

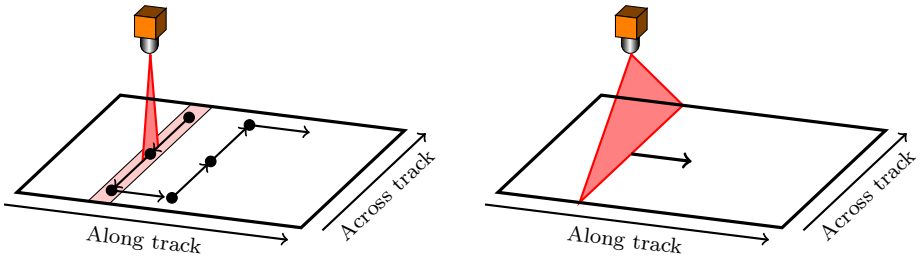
The approach proposed in this paper has connections with [21] since it addresses the slice-by-slice hyperspectral image restoration problem. It also uses the same approach to transform the non-causal blurring model into a (semi-)causal model. However, the KF equations need to update the state vector, the associated Kalman gain and covariance matrices. This complexity may not be compatible with real-time processing in industrial applications of hyperspectral imaging systems. The main motivation is to derive algorithms that allow to incorporate additional constraints enforcing some prior properties in the restored images while maintaining a linear complexity with respect to the number of unknown parameters. This work introduces an least mean-squares (LMS) framework for sequential deconvolution of hyperspectral images in which the subgradient is substituted by an instantaneous stochastic approximation and the step size is constant. Our algorithm operates in the spirit of LMS-based algorithms used, for instance, for adaptive system identification [45, 12, 9] and super-resolution restoration of sequences of images [19, 16]. However, LMS-based sequential deconvolution of (hyperspectral) images has never been reported in the literature. Accounting for the specificities of the online image deconvolution (non-causal blurring, low signal-to-noise ratio), we propose a sliding-block regularized least mean-squares (SBR-LMS) algorithm that allows the restoration of images within a preset delay Q.

The paper is organized as follows. In Section 2, the image-blurring model is stated. Causality of the convolution kernel and of the estimator are discussed. In Section 3, the hyperspectral image deconvolution problem is introduced. The SBR-LMS algo-

rithm is proposed to address this problem. The criterion includes a spatial regularizer promoting the restoration of piecewise constant objects, a zero-attracting regularizer promoting the removal of the conveyor background, and a spectral regularizer promoting the smoothness of the spectral response. In Section 4, the transient behavior of the algorithm is analyzed in the mean and mean-squares sense. Its stability is also discussed. In Section 5, simulation results are presented to assess the accuracy of the convergence behavior model in both stationary and non-stationary scenarios. The role of each hyper-parameter is discussed. Numerical results are reported to assess the performance of the proposed approach and to compare it with state-of-the-art methods. Finally, the algorithm is applied to real hyperspectral images acquired under different experimental conditions.

2. Blurring and causality issues.

2.1. Scanning technologies and data structure. Recording a 3D hyperspectral data cube can be performed in several ways, depending on how the 3D signal is acquired, as a sequence of either 1D or 2D signals. Each configuration results in a specific imaging system. Here, we restrict our attention to whiskbroom and pushbroom scanners which are the most usual configurations in industrial applications. Figure 1 depicts schematically how a hyperspectral image is captured by whiskbroom and pushbroom imagers. In both cases, the scene is scanned by moving the imager or its field-of-view (FOV) across the scene. For example, for material analysis, the samples to be imaged are carried by a conveyor moving at constant speed, while in remote sensing applications, the imaging system is carried on an airborne platform moves over the scene. In a whiskbroom imaging system, an 1D sensor (*i.e.*, a spectrometer) is used to form the 3D data cube. Thus, the whole spatial scene is observed point-by-point yielding a stream of spectra which are stacked into a 3D data cube. A mechanized angular movement using a scanning mirror sweeping allows to cover a spatial line from one edge to the other. Alternatively, pushbroom imaging systems make use of 2D sensors allowing to observe the scene line-by-line at each time instant. The stream of spatial-spectral arrays is stacked to form the 3D-data cube.



(a) Whiskbroom imaging system (b) Pushbroom imaging system

Fig. 1: Data acquisition by hyperspectral imaging systems

Following [36], respective pros and cons of whiskbroom and pushbroom imaging systems are as follows. Because whiskbroom systems rely on 1D detector, the calibration is much simpler than with pushbroom systems. The disadvantages of this design include the presence of a mechanical scanning system, the shorter integration time than in pushbroom systems, and the image forming geometry which is dependent on the scanning speed, the scan mirror arrangement and the conveyor movement.

As compared to whiskbroom scan, pushbroom scanners allow for longer integration times for individual detector elements; the pixel dwell time is approximatively the inverse of the line frequency. In addition, there are distinct but fixed geometric relations between the pixels within a scan line. Since 2D sensors are used as focal planes in these systems, the uniform calibration of the detector response is critical. Note that in a combined analysis of signal-to-noise ratio (SNR), uniformity, and stability, pushbroom scanners might not necessarily outperform whiskbroom systems even though they have a longer integration time. For example, [35] developed a whiskbroom hyperspectral imaging system for ultra fast material sorting application allowing to reach up to 3 meters per second scanning velocity. However, pushbroom scanners are increasingly used in industrial applications.

A hyperspectral image is a 3D data cube whose dimensions are generally referred to as across-track, along-track and spectral dimensions. However, in the sequel, to avoid any confusion, they will be designated as spatial (across-track), time (along-track) and spectral dimensions. We will denote a hyperspectral image by $\mathbf{Y} \in \mathbb{R}^{N\times P\times K}$, where N, P, and K are the number of spatial, spectral and time measurements, respectively (see Figure 2). The hyperspectral image is then obtained slice by slice, each slice being denoted by $\mathbf{Y}_k \in \mathbb{R}^{N\times P}$, $k = 1, \ldots, K$. The size of $\mathbf{Y}$ increases with k, which can possibly grow to infinity.

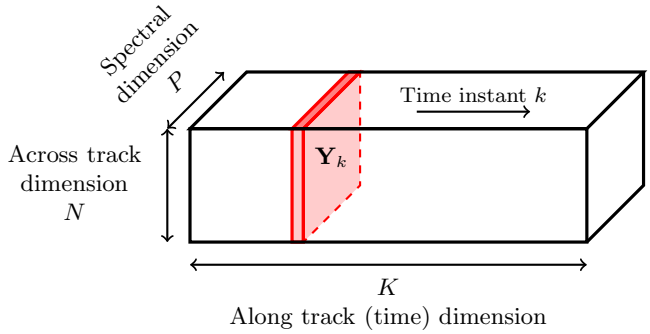


Fig. 2: Structure of a hyperspectral data cube $\mathbf{Y}$

2.2. Blurring and noise. A hyperspectral image $\mathbf{Y}$ may suffer from distortions caused by the measuring devices and acquisition process such as keystone effect, smile effect, non-uniform illumination, and sensor response. These degradations may be corrected using standard calibration techniques [36, 20]. In this paper, we are primarily concerned with another type of distortion, namely, spatial distortions resulting in a loss of spatial resolution.

In a whiskbroom imaging system, the spatial resolution is mainly controlled by the size of the beam footprint on the scanned object. Additional distortion can be caused by the conveyor motion but, in this paper, we shall assume that the integration time is small enough so that this distortion can be neglected. In general, the spatial sampling period ι_s is chosen equal to the beam footprint size yielding a low resolution image. To increase the spatial resolution, it is necessary to reduce ι_s, but this results in a spatial blurring of the hyperspectral image as sketched in Figure 3, where pixels are represented by the dots, and the beam footprint is represented by the circles.

The integration time T mainly acts on the noise level: a low value for T results in a low SNR. Consequently, assuming a constant acquisition velocity ι_s/T, any resolution improvement results in an increase of both blurring and noise level. Assuming that the scanning angle is small, the beam footprint size can be considered as invariant with respect to the pixel position. We also assume that the footprint size is independent of the wavelength. Thus, the corresponding distortion can be modeled by linear invariant convolution.

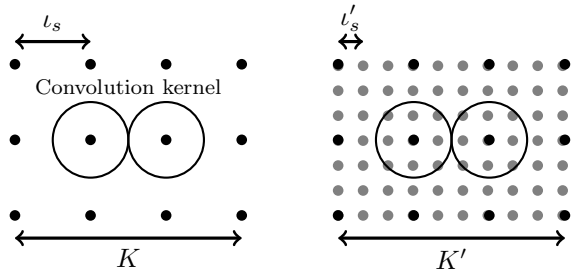


Fig. 3: Data acquisition with (a) low and (b) high spatial sampling rate. high spatial sampling rate results in a blurring of the image.

For pushbroom imaging systems, the quality of the acquired images can be significantly degraded by the blurring effect resulting from the miscalibration of the optical system [30, 31, 20]. According to [41], there are many reasons that can cause spatial degradation of the pushbroom imaging system such as optical aberrations, slit size, and detector element size. In [31], the observed image corresponds to the 3D convolution of the original image with a spatial-spectral blurring filter varying with the spatial position. Here, the dependence of the spatial blurring with respect to the spatial position is neglected. We consider situations where the spectral response is smooth enough to choose a spectral sampling ι_λ (controlled by the spectral binning) greater than the support of the spectral blurring and a spatial sampling ι_s smaller than the support of the point spread function. This results in a degradation model involving only spatial blurring (possibly depending on the wavelength) of the hyperspectral image which is given by Equation (2.1) in the next section. The integration time is supposed to be small for fast scanning yielding a low SNR.

2.3. Causality. We shall now discuss issues related to the causality of the convolution kernel and associated estimates. Following [27], hyperspectral image blurring can be seen as P simultaneous spatial convolutions. For each wavelength λ_p, the blurred spatial image $\mathbf{Y}^p \in \mathbb{R}^{N\times K}$ is given by the 2D convolution:

$$\mathbf{Y}^p = \mathbf{H}^{\star p} * \mathbf{X}^p + \mathbf{Z}^p \tag{2.1}$$

where $\mathbf{Y}^p \in \mathbb{R}^{N\times K}$ is the blurred spatial image for each wavelength λ_p, $\mathbf{X}^p \in \mathbb{R}^{N\times K}$ is the image to restore, $\mathbf{H}^{\star p} \in \mathbb{R}^{M\times L}$ is a convolution kernel (filter), and $\mathbf{Z}^p \in \mathbb{R}^{N\times K}$ is an additive i.i.d. noise. The symbol $*$ stands for the 2D convolution operator. First we derive a sequential causal formulation of model (2.1). Without loss of generality, we focus on the sequential model for 2D images. The image $\mathbf{Y}^p$, collected online, can be represented as a sequence of vectors $\mathbf{y}^p_k := [y^p_{1,k}, \ldots, y^p_{N,k}]^\top$, with $k = 1, \ldots, K$,

where $^\top$ denotes the transpose of a matrix. We use the same notation for the columns of $\mathbf{X}^p$. We assume a finite length blurring kernel of size L along the time dimension, centered around 0 which means that both past and future values of $\mathbf{x}_k^p$ contribute to the observation $\mathbf{y}_k^p$. In order to make the blurring kernel causal[1], it has to be shifted by $(L-1)/2$. It is thus necessary to delay the observation by $(L-1)/2$ samples, that is, $\tilde{\mathbf{y}}_k^p = \mathbf{y}_{k-(L-1)/2}^p$. Writing $\mathbf{H}^{\star p} = [\mathbf{h}_L^p, \ldots, \mathbf{h}_1^p]$ with $\mathbf{h}_\ell^p = \left[h_{M,\ell}^p, \ldots, h_{1,\ell}^p\right]^\top$, model (2.1) can be expressed as:

$$\tilde{\mathbf{y}}_k^p = \mathbf{y}_{k-(L-1)/2}^p = \sum_{\ell=1}^{L} \mathbf{H}_\ell^p \mathbf{x}_{k-\ell+1}^p + \mathbf{z}_{k-(L-1)/2}^p \tag{2.2}$$

where the measurement noise $\mathbf{z}_k^p$ is zero mean and statistically independent of the other signals. $\mathbf{H}_\ell^p$ is the $N \times N$ Toeplitz matrix with first column and first row given by $[h_{(M+1)/2,\ell}^p, \ldots, h_{M,\ell}^p, 0, \ldots, 0]$ and $[h_{(M+1)/2,\ell}^p, \ldots, h_{1,\ell}^p, 0, \ldots, 0]$, respectively[2]. Relation (2.2) introduces a delay in time dimension because the filter is made causal along this dimension. Another consequence of causality issues concerns the estimation process of $\mathbf{x}_k^p$. First, $\mathbf{x}_k^p$ is involved in past and future observations $(\mathbf{y}_{k-(L-1)/2}^p, \ldots, \mathbf{y}_k^p, \ldots, \mathbf{y}_{k+(L-1)/2}^p)$. Secondly, let us consider the problem of estimating only $\mathbf{x}_k^p$ from the dataset:

$$\tilde{\mathbf{y}}^p \triangleq \mathrm{col}\{\tilde{\mathbf{y}}_k^p\}_{k=1}^K. \tag{2.3}$$

where $\mathrm{col}\{\cdot\}$ stacks its vector arguments on top of each other. The least squares criterion can be written as:

$$||\tilde{\mathbf{y}}^p - \mathbf{F}^p \mathbf{x}^p||^2 \tag{2.4}$$

where $\mathbf{x}^p$ is built similarly to $\tilde{\mathbf{y}}^p$ and $\mathbf{F}^p$ is a Toeplitz-block-Toeplitz matrix of proper dimensions. To make the dependence of the criterion on $\mathbf{x}_k^p$ explicit, we introduce the following partitions:

$$\mathbf{x}^p = \left[\mathbf{x}_{1:k-1}^{p\top}, \mathbf{x}_k^{p\top}, \mathbf{x}_{k+1:K}^{p\top}\right]^\top, \tag{2.5}$$

$$\mathbf{F}^p = \left[\mathbf{F}_{1:k-1}^p, \mathbf{F}_k^p, \mathbf{F}_{k+1:K}^p\right] \tag{2.6}$$

where $\mathbf{x}_{i:j}^p \triangleq \mathrm{col}\{\mathbf{x}_k^p\}_{k=i}^j$ and $\mathbf{F}_{i:j}^p$ is the the submatrix formed by the columns $(i-1)N+1$ through jN of $\mathbf{F}^p$. This results in:

$$||\tilde{\mathbf{y}}^p - \mathbf{F}^p \mathbf{x}^p||^2 = ||\tilde{\mathbf{y}}^p - \mathbf{F}_{1:k-1}^p \mathbf{x}_{1:k-1}^p - \mathbf{F}_{k+1:K} \mathbf{x}_{k+1:K}^p - \mathbf{F}_k^p \mathbf{x}_k^p||^2. \tag{2.7}$$

It is now clear that optimally estimating $\mathbf{x}_k^p$ requires all the past estimations $\hat{\mathbf{x}}_{1:k-1}^p$ and future estimations $\hat{\mathbf{x}}_{k+1:K}^p$, which precludes the derivation of a sequential estimator for $\mathbf{x}_k^p$. To address these issues, we propose to produce the estimates $\hat{\mathbf{x}}_k^p$ with a delay Q. More precisely, at time instant k, we shall estimate $\mathbf{x}_{k-Q+1}^p$ from Q observations $(\tilde{\mathbf{y}}_k^p, \tilde{\mathbf{y}}_{k-1}^p \ldots, \tilde{\mathbf{y}}_{k-Q+1}^p)$, coarse posterior estimates $(\hat{\mathbf{x}}_k^p, \hat{\mathbf{x}}_{k-1}^p, \ldots, \hat{\mathbf{x}}_{k-Q+2}^p)$ refined as k increases, and anterior estimates $(\hat{\mathbf{x}}_{k-Q}^p, \hat{\mathbf{x}}_{k-Q-1}^p, \ldots, \hat{\mathbf{x}}_{k-Q-L+2}^p)$ which are no longer updated. This is the key idea of the sliding-block LMS deconvolution algorithm developed in the next section.

[1] For simplicity, L is assumed to be odd.

[2] For simplicity, M is assumed to be odd.

3. Online image deconvolution.

3.1. Block Tikhonov. In this section, we introduce the Tikhonov-like approach proposed in [27] and a direct extension for online hyperspectral image restoration. They will serve as reference methods to assess the performance of the proposed LMS-based method. This Tikhonov-like approach, originally developed in an offline setting, consists of seeking the minimum of a criterion composed of three terms: a data fitting term, a spatial regularizer, and a spectral regularizer:

$$\min_{\mathbf{x}} \mathcal{J}(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{\eta_s}{2}\|\mathbf{\Delta}_s\mathbf{x}\|^2 + \frac{\eta_\lambda}{2}\|\mathbf{\Delta}_\lambda\mathbf{x}\|^2. \tag{3.1}$$

where $\mathbf{x}$ and $\mathbf{y}$ denote, respectively, the original and the observed vectorized hyperspectral images. Operator $\mathbf{\Delta}_s$ corresponds to a Laplacian filter promoting the smoothness along both spatial and time dimensions, and $\mathbf{\Delta}_\lambda$ corresponds to a first-order derivative filter along the spectral dimension. Parameters η_s and η_λ are respectively the spatio-temporal and spectral regularization parameters. Matrix $\mathbf{H}$ is block-diagonal and corresponds to the convolution kernel. Because the criterion is quadratic, solution of problem (3.1) results in the following linear estimator:

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{H} + \eta_s\mathbf{\Delta}_s^T\mathbf{\Delta}_s + \eta_\lambda\mathbf{\Delta}_\lambda^T\mathbf{\Delta}_\lambda)^{-1}\mathbf{H}^T\mathbf{y}. \tag{3.2}$$

Due to the block diagonal structure of $\mathbf{H}$, the estimator (3.2) can be efficiently implemented in the frequency domain (see [27] for details).

The Block Tikhonov (BT) deconvolution approach addresses problem (3.1) in an online way by sequentially restoring spatio-spectral arrays $\mathbf{X}_k \in \mathbb{R}^{N\times P}$. Let $\mathbf{x}_k$ and $\mathbf{y}_k$ be the vectorized matrices $\mathbf{X}_k \in \mathbb{R}^{N\times P}$ and $\mathbf{Y}_k \in \mathbb{R}^{N\times P}$, respectively:

$$\mathbf{x}_k \triangleq \operatorname{col}\{\mathbf{x}_k^p\}_{p=1}^P, \quad \mathbf{y}_k \triangleq \operatorname{col}\{\mathbf{y}_k^p\}_{p=1}^P. \tag{3.3}$$

where superscript p refers to the spectral band. This BT algorithm implements estimator (3.2) with slidding blocks of Q samples $[\mathbf{x}_{k-Q+1}, \ldots, \mathbf{x}_k], \forall k = Q, \ldots, K$ and outputs the estimated vector $\hat{\mathbf{x}}_{k-(Q-1)/2}$ as the final result[3]. This algorithm works fast since only quadratic regularization terms are considered. However, when other constraints are needed, such as ℓ_1-norm-based ones, no explicit solutions are available and time consuming iterative algorithms are required.

In the next section, we focus on extending the zero-attracting LMS (ZA-LMS) algorithm proposed in [12] to online deconvolution, while accounting for some image specificities (non-causal blurring, presence of smooth regions separated by abrupt edges, and low SNR) resulting in the so-called Sliding-block regularized LMS (SBR-LMS).

3.2. Sliding-block regularized LMS (SBR-LMS). Consider the problem of sequentially estimating $\mathbf{x}_{k-Q+1}$ based on the delayed observations $(\tilde{\mathbf{y}}_k, \ldots, \tilde{\mathbf{y}}_{k-Q+1})$ and past estimates $\hat{\mathbf{x}}_{k-Q}, \ldots, \hat{\mathbf{x}}_{k-Q-L+2}$. Due to the non-causality of the estimation process, we propose to determine the future estimates with respect to time $k-Q+1$, namely, $\mathbf{x}_k, \ldots, \mathbf{x}_{k-Q+2}$. In what follows, to simplify notations, $\mathbf{y}_k$ refers to the

[3] For simplicity, Q is chosen to be odd.

delayed observation $\tilde{\mathbf{y}}_k$ and $\hat{}$ symbol is omitted. This results in the following criterion:

$$\mathcal{J}(\overbrace{\mathbf{x}_k,\ldots,\mathbf{x}_{k-Q+1}}^{\text{updated}}|\overbrace{\mathbf{x}_{k-Q},\ldots,\mathbf{x}_{k-Q-L+2}}^{\text{past estimates}}) = \sum_{q=1}^{Q}\mathbb{E}\left\|\mathbf{y}_{k-q+1}-\sum_{\ell=1}^{L}\mathbf{H}_\ell\mathbf{x}_{k-q-\ell+2}\right\|^2$$

$$+\eta_z\sum_{q=1}^{Q}\|\mathbf{x}_{k-q+1}\|_1+\eta_s\sum_{q=1}^{Q}\|\mathbf{D}_s\mathbf{x}_{k-q+1}\|_1+\eta_\lambda\sum_{q=1}^{Q}\|\mathbf{D}_\lambda\mathbf{x}_{k-q+1}\|^2 \tag{3.4}$$

where $\mathbb{E}\{\cdot\}$ stands for the expectation operator, $\|\cdot\|_1=\sum_n|\{\cdot\}_n|$ denotes the ℓ_1-norm of its argument, and $\{\cdot\}_n$ stands for the n-th entry of a vector. Matrix $\mathbf{H}_\ell \triangleq \text{blkdiag}\{\mathbf{H}_\ell^p\}_{p=1}^P$ is a block-diagonal matrix.

The regularizer $\|\mathbf{D}_s\mathbf{x}_{k-q+1}\|_1$ promotes the restoration of piecewise constant patterns along the spatial dimension. The first-order filtering operator $\mathbf{D}_s$ is defined as:

$$\mathbf{D}_s \triangleq \mathbf{I}_P\otimes\mathbf{T}_N \tag{3.5}$$

where $\otimes$ stands for the Kronecker product, matrix $\mathbf{I}_J$ denotes the $J\times J$ identity matrix and $\mathbf{T}_J$ is the Toeplitz matrix of size $(J-1)\times J$ with first column $[1,0,\ldots,0]$ and first row $[1,-1,0,\ldots,0]$. The zero-attracting regularizer $\|\mathbf{x}_{k-q+1}^p\|_1$ aims at removing the conveyor background. The choice of these regularization terms is thus mainly motivated by the targeted application, namely, the inspection of objects put on a conveyor belt. At a given wavelength, the response of the conveyor after background removal is close to zero while that of the objects is supposed to be piecewise constant. The strength of the first derivative regularizers along spatial dimension is controlled by $\eta_s\geq 0$. The strength of the zero-attracting regularizer is controlled by $\eta_z\geq 0$.

The spectral regularization term $\|\mathbf{D}_\lambda\mathbf{x}_{k-q+1}\|^2$ promotes spectral smoothness which corresponds to the very nature of NIR hyperspectral images of biological material such as wood [44]. Matrix $\mathbf{D}_\lambda$ is a first-order filtering operator along the spectral dimension weighted by the coefficients $\{c_p\}_{p=1}^{P-1}$. It is defined as:

$$\mathbf{D}_\lambda \triangleq (\text{diag}(c_1,\ldots,c_{P-1})\,\mathbf{T}_P)\otimes\mathbf{I}_N. \tag{3.6}$$

The parameter η_λ controls the strength of the spectral smoothness penalty term. The spectral weights c_p were introduced to provide some additional flexibility to the algorithm. Basically, the strength of the weight c_p promotes the smoothness around the p-th spectral band. In our case, the spectral data are sampled on a regular grid. hence, $c_1,\ldots,c_{P-1}$ are all set to 1. however, in the case where the spectral data would be recorded on two non adjacent spectral bands, smoothness between these bands should not be promoted and the corresponding weight should be set to a small value (zero).

The criterion (3.1) developed in [27] introduces a regularization using second order derivatives along spatial and time dimensions. An inspection of criterion (3.4) may lead to the assertion that no regularization term is acting along the time dimension (time-regularization). However, we prove in Appendix B that, for the denoising case, LMS-based algorithm implicitly introduces a time regularization and presents a time delay in the estimation. A formal link between μ and the regularization strength is given. A small value μ results in strong regularization strength and large delay along the time dimension. Experimental results (Subsection 5.1) confirm that the interpretation is valid for the deconvolution case.

A subgradient of (3.4) is given by:

$$\nabla\mathcal{J}(\mathbf{x}_k,\ldots,\mathbf{x}_{k-Q-L+2}) \triangleq \left(\frac{\partial\mathcal{J}}{\partial\mathbf{x}_k},\ldots,\frac{\partial\mathcal{J}}{\partial\mathbf{x}_{k-Q+1}},\mathbf{0}_{N\times 1},\ldots,\mathbf{0}_{N\times 1}\right) \tag{3.7}$$

where $\mathbf{0}_{I\times J}$ denotes the $I\times J$ zero matrix. Zero terms in the subgradient indicate that past estimates $\hat{\mathbf{x}}_{k-Q},\ldots,\hat{\mathbf{x}}_{k-Q-L+2}$ are no longer updated. We now derive the sliding-block regularized LMS algorithm. Consider the vectorized data:

$$\mathbf{x}'_k \triangleq \mathrm{col}\{\mathbf{x}_{k-q+1}\}_{q=1}^{Q+L-1}, \quad \mathbf{y}'_k \triangleq \mathrm{col}\{\mathbf{y}_{k-q+1}\}_{q=1}^{Q+L-1}. \tag{3.8}$$

A valid subgradient for $|x|$ is $\mathrm{sign}(x)$ [18] where the sign function is defined as $\mathrm{sign}(x) = 0$ for $x=0$, and $\mathrm{sign}(x) = x/|x|$ otherwise. Approximating the subgradient in (3.7) by its instantaneous value yields:

$$\nabla\mathcal{J}(\mathbf{x}'_k) = -2\boldsymbol{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\mathbf{x}'_k\right) + \eta_z\boldsymbol{\Gamma}\mathrm{sign}(\mathbf{x}'_k) + \eta_s\boldsymbol{\Lambda}_s^\top\mathrm{sign}(\boldsymbol{\Lambda}_s\mathbf{x}'_k) + 2\eta_\lambda\boldsymbol{\Lambda}_\lambda^\top\boldsymbol{\Lambda}_\lambda\mathbf{x}'_k \tag{3.9}$$

Matrices $\boldsymbol{\Phi}$, $\mathbf{G}$ and $\boldsymbol{\Gamma}$ are of size $(Q+L-1)PN\times(Q+L-1)PN$. Matrix $\boldsymbol{\Phi}$ is defined by:

$$\boldsymbol{\Phi} \triangleq \left[\begin{array}{c|c} \boldsymbol{\Phi}_{11}^\top & \mathbf{0}_{QPN\times(L-1)PN} \\ \hline \mathbf{0}_{(L-1)PN\times(Q+L-1)PN} & \end{array}\right] \tag{3.10}$$

where $\boldsymbol{\Phi}_{11}$ is the matrix of size $QPN\times QPN$ defined as:

$$\boldsymbol{\Phi}_{11} \triangleq \begin{bmatrix} \mathbf{H}_1 & \cdots & \mathbf{H}_Q \\ & \ddots & \vdots \\ \mathbf{0} & & \mathbf{H}_1 \end{bmatrix} \tag{3.11}$$

in which $\mathbf{H}_\ell = \mathbf{0}_{PN\times PN}$ for $\ell > L$. It is necessary to define the matrix $\boldsymbol{\Phi}_{11}$ as above to properly account for the cases $Q<L$ and $Q\geq L$. Matrix $\mathbf{G}$ has the form:

$$\mathbf{G} \triangleq \left[\begin{array}{ccccc} \mathbf{H}_1 & \cdots & \mathbf{H}_L & & \mathbf{0} \\ & \ddots & & \ddots & \\ \mathbf{0} & & \mathbf{H}_1 & \cdots & \mathbf{H}_L \\ \hline \multicolumn{5}{c}{\mathbf{0}_{(L-1)PN\times(Q+L-1)PN}} \end{array}\right] \triangleq \left[\begin{array}{c|c} \boldsymbol{\Phi}_{11} & \mathbf{G}_{12} \\ \hline \mathbf{0}_{(L-1)PN\times(Q+L-1)PN} & \end{array}\right], \tag{3.12}$$

(see Appendix A for more details). Matrix $\boldsymbol{\Gamma}$ is defined by:

$$\boldsymbol{\Gamma} \triangleq \left[\begin{array}{c|c} \mathbf{I}_{QPN} & \mathbf{0}_{QPN\times(L-1)PN} \\ \hline \mathbf{0}_{(L-1)PN\times(Q+L-1)PN} & \end{array}\right]. \tag{3.13}$$

The first-order derivative filters for spatial and spectral dimensions are:

$$\boldsymbol{\Lambda}_s \triangleq [\mathbf{I}_Q\otimes\mathbf{D}_s \quad \mathbf{0}_{QP(N-1)\times(L-1)PN}] \triangleq [\boldsymbol{\Lambda}_{s,1} \quad \mathbf{0}_{QP(N-1)\times(L-1)PN}], \tag{3.14}$$

$$\boldsymbol{\Lambda}_\lambda \triangleq [\mathbf{I}_Q\otimes\mathbf{D}_\lambda \quad \mathbf{0}_{Q(P-1)N\times(L-1)PN}] \triangleq [\boldsymbol{\Lambda}_{\lambda,1} \quad \mathbf{0}_{Q(P-1)N\times(L-1)PN}]. \tag{3.15}$$

Finally, the SBR-LMS algorithm for hyperspectral image deconvolution is given by:

$$\hat{\mathbf{x}}'_{k+1} = \boldsymbol{\Omega}\hat{\mathbf{x}}'_k + \mu\boldsymbol{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right) - \rho_z\boldsymbol{\Gamma}\mathrm{sign}(\hat{\mathbf{x}}'_k) - \rho_s\boldsymbol{\Lambda}_s^\top\mathrm{sign}(\boldsymbol{\Lambda}_s\hat{\mathbf{x}}'_k) - \mu\eta_\lambda\boldsymbol{\Lambda}_\lambda^\top\boldsymbol{\Lambda}_\lambda\hat{\mathbf{x}}'_k \tag{3.16}$$

where $\rho_z = \mu\eta_z/2$, $\rho_s = \mu\eta_s/2$ and μ is a step size parameter that controls the trade off between convergence rate and algorithm stability. Matrix $\mathbf{\Omega}$ is given by:

(3.17)
$$\mathbf{\Omega} \triangleq \left[\begin{array}{cc|cc} \multicolumn{2}{c|}{\mathbf{I}_{QPN}} & \multicolumn{2}{c}{\mathbf{0}_{QPN\times(L-1)PN}} \\ \hline \mathbf{0} & \mathbf{I}_{PN} & \mathbf{0} & \mathbf{0}_{PN\times PN} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{(L-2)PN} & \mathbf{0}_{(L-2)PN\times PN} \end{array}\right] \triangleq \left[\begin{array}{c|c} \mathbf{I}_{QPN} & \mathbf{0}_{QPN\times(L-1)PN} \\ \hline \mathbf{\Omega}_{21} & \mathbf{\Omega}_{22} \end{array}\right].$$

The upper part of matrix $\mathbf{\Omega}$ corresponds to the set of updated variables, and the lower part allows to shift the past estimates. The final result $\hat{\mathbf{x}}_{k-Q+2}$ is obtained by selecting the Q-th block of vector $\hat{\mathbf{x}}'_{k+1}$, that is,

$$\hat{\mathbf{x}}_{k-Q+2} = \mathbf{S}\hat{\mathbf{x}}'_{k+1} \tag{3.18}$$

where $\mathbf{S} \triangleq [\mathbf{0}_{PN\times(Q-1)PN}, \mathbf{I}_{PN}, \mathbf{0}_{PN\times(L-1)PN}]$. It is worth to mention that the proposed algorithm is different from the standard block-LMS algorithm for which the output $\mathbf{x}_k$ is updated only once for every block of size Q. On the contrary, in the proposed algorithm, to account for the causality issues discussed in Subsection 2.3, $\mathbf{x}_k$ is updated Q times.

Finally, depending on the hyperparameter values, different LMS-like algorithms can be defined as presented in Table 1.

Table 1: The different LMS-like algorithms and corresponding hyperparameters

	μ	Q	η_z	η_s	η_λ
ZA-LMS	>0	1	>0	0	0
SB-LMS	>0	>1	0	0	0
SBZA-LMS	>0	>1	>0	0	0
SBR-LMS	>0	>1	>0	>0	>0

3.3. Algorithm implementation and computational cost. Algorithm 3.1 details the SBR-LMS implementation. It should be noted that the storage of matrix $\mathbf{H}_\ell^p$ is avoided and that only 1D convolutions are required. Two different implementations of the convolution are possible: direct (spatial) domain or frequency domain. We also specify the particular case of using separable convolution kernel in form of $\mathbf{H}^{\star p} = \mathbf{g}^p\mathbf{f}^{p\top}$ where $\mathbf{f}^p = [f_L^p, \ldots, f_1^p]^\top$ and $\mathbf{g}^p = [g_M^p, \ldots, g_1^p]^\top$ which yields lower computational burden. The validity of the separable kernel assumption is mainly controlled by the calibration of the optical systems. If the optical elements are misaligned, then the kernel will be non separable. Otherwise, a separable Gaussian kernel is a good and widely used model (see [26] for details).

The computational complexity per iteration of SBR-LMS is given in Table 2, where the costs of both direct and frequency domain implementations of the convolution are given. The approximate costs in Table 3 give the number of multiplications obtained by setting $Q = L = M$ and retaining the dominant complexity. In the separable kernel case, both direct and frequency domain implementations yield lower complexity than for a non-separable kernel. In the direct domain, the complexity is linear w.r.t. the number of unknowns PN. For $Q \gg \log_2 N$, we can reduce the complexity for the non-separable kernel case by applying the algorithm in the frequency domain while for separable kernel case, both direct domain and frequency domain yield a computational cost of the same order. As expected, when $Q \ll \log_2 N$, the frequency domain implementation provides no benefit.

Table 2: Computational cost per iteration of recursion (3.16)

Operation	Multiplications	Additions
$\mathbf{\Omega}\hat{\mathbf{x}}'_k + \mu\mathbf{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right)$ direct domain, non-separable kernel	$PQ\left(L+\frac{Q+1}{2}\right)M\left(N-\frac{M-1}{2}\right)$	$PQL\left[(M-1)(N-\frac{M-1}{2})+N\right]$ $+PQ\left[\frac{Q+1}{2}(M-1)(N-\frac{M-1}{2})+\frac{Q-1}{2}N\right]$
$\mathbf{\Omega}\hat{\mathbf{x}}'_k + \mu\mathbf{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right)$ direct domain, separable kernel	$PQ\left(L+\frac{Q+1}{2}\right)N+2PQM\left(N-\frac{M-1}{2}\right)$	$PQ\left(L+\frac{Q-1}{2}\right)N$ $+2PQ(M-1)(N-\frac{M-1}{2})$
$\mathbf{\Omega}\hat{\mathbf{x}}'_k + \mu\mathbf{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right)$ frequency domain, non-separable kernel	$4PQ\left(L+\frac{Q+1}{2}\right)N+2PQ(L+1)N\log_2 N$	$PQ(2L+Q-1)N+2PQ(L+1)N\log_2 N$
$\mathbf{\Omega}\hat{\mathbf{x}}'_k + \mu\mathbf{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right)$ frequency domain, separable kernel	$PQ(L+Q+1)N+2PQN(2\log_2 N+4)$	$PQ(L+Q)N+4PQN\log_2 N$
$\rho_z\mathbf{\Gamma}\text{sign}(\hat{\mathbf{x}}'_k)$	0	0
$\rho_s\mathbf{\Lambda}_s^\top\text{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)$	0	$2(N-1)PQ$
$\mu\eta_\lambda\mathbf{\Lambda}_\lambda^\top\mathbf{\Lambda}_\lambda\hat{\mathbf{x}}'_k$	$4N(P-1)(Q+L-1)$	$2NP(P-1)(Q+L-1)$

Algorithm 3.1 SBR-LMS algorithm for hyperspectral image deconvolution

Data: $\mathbf{Y} \in \mathbb{R}^{N\times K\times P}$, $\mathbf{H}^{\star p} = [\mathbf{h}_L^p, \ldots, \mathbf{h}_1^p]$, $\forall p = 1, \ldots, P$
Result: $\mathbf{X} \in \mathbb{R}^{N\times K\times P}$
Initialization Q, μ, ρ_z, ρ_s, η_λ, $c_1, \ldots, c_{P-1}$ and $\mathbf{T}_N$

$$\mathbf{h}_\ell^p = \mathbf{0}_{N\times 1}, \quad \forall \ell = L+1, \ldots, Q, \quad \forall p = 1, \ldots, P\,;$$

Selection of parameters $\mathbf{X} = \mathbf{0}_{N\times K\times P}$, $c_0 = c_P = 0$ **for** $k = L+Q-1 : K-1$ **do**
  for $q = 1 : Q$ **do**
    for $p = 1 : P$ **do**
      Non-separable convolution kernel:

$$\mathbf{err}_{k-q+1}^p = \mathbf{y}_{k-q+1}^p - \sum_{\ell=1}^{L} \mathbf{h}_\ell^p * \mathbf{x}_{k-q-\ell+2}^p$$

$$\mathbf{grad} = \sum_{i=1}^{q} \mathbf{h}_{q-i+1}^p * \mathbf{err}_{k-i+1}^p$$

      Separable convolution kernel:

$$\mathbf{err}_{k-q+1}^p = \mathbf{y}_{k-q+1}^p - \mathbf{g}^p * \sum_{\ell=1}^{L} f_\ell^p \mathbf{x}_{k-q-\ell+2}^p$$

$$\mathbf{grad} = \mathbf{g}^p * \sum_{i=1}^{q} f_{q-i+1}^p \mathbf{err}_{k-i+1}^p$$

      where $*$ is the 1D convolution operator.

$$\begin{aligned}\mathbf{x}_{k-q+2}^p =& \mathbf{x}_{k-q+1}^p + \mu\,\mathbf{grad} - \rho_z \mathrm{sign}(\mathbf{x}_{k-q+1}^p) - \rho_s \mathbf{T}_N^T \mathrm{sign}(\mathbf{T}_N \mathbf{x}_{k-q+1}^p) \\ & - \mu\eta_\lambda \left[(c_{p-1}^2 + c_p^2)\mathbf{x}_{k-q+1}^p \; - c_{p-1}^2 \mathbf{x}_{k-q+1}^{p-1} - c_p^2 \mathbf{x}_{k-q+1}^{p+1}\right];\end{aligned}$$

    end
  end
end

Table 3: Approximate computational cost per iteration for the convolution in (3.16)

	Direct domain	Frequency domain
Non-separable kernel	$\mathcal{O}(Q^3PN)$	$\mathcal{O}(Q^2PN\log_2 N)$
Separable kernel	$\mathcal{O}(Q^2PN)$	$\mathcal{O}(QPN(Q+\log_2 N))$

4. Transient behavior analysis. In this section, we derive the transient behavior model of the deconvolution algorithm (3.16) both in the mean and mean-squares sense. The analysis provides us with an important guidance for selecting the step-size parameter, and also allows us to study the influence of the other parameters (Subsection 5.2). The stability condition of the algorithm is also studied.

4.1. Mean and mean-squares transient behavior model. Using the notations of Subsection 3.2, Equation (2.2) can be written as: $\mathbf{y}'_k = \mathbf{G}\mathbf{x}'_k + \mathbf{z}'_k$ with $\mathbf{z}'_k$ a zero-mean measurement noise of covariance $\sigma_z^2\mathbf{I}_{(Q+L-1)NP}$. Taking the expectation of both sides of (3.16), the mean sliding block vector evolves according to the following recursion:

$$\mathbb{E}\{\hat{\mathbf{x}}'_{k+1}\} = \mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}'_k\} + \mu\mathbf{\Phi}\mathbf{G}\mathbf{x}'_k - \rho_z\mathbf{\Gamma}\,\mathbb{E}\{\mathrm{sign}(\hat{\mathbf{x}}'_k)\} - \rho_s\mathbf{\Lambda}_s^\top\,\mathbb{E}\{\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)\} \tag{4.1}$$

where $\mathbf{A} \triangleq \mathbf{\Omega} - \mu\mathbf{\Phi}\mathbf{G} - \mu\eta_\lambda\mathbf{\Lambda}_\lambda^\top\mathbf{\Lambda}_\lambda$. The mean vector can be then obtained by using:

$$\mathbb{E}\{\hat{\mathbf{x}}_{k-Q+2}\} = \mathbf{S}\mathbb{E}\{\hat{\mathbf{x}}'_{k+1}\}, \tag{4.2}$$

The main difficulty in (4.1) lies in evaluating the expectation of the sign function. This point will be discussed later.

Consider the error:

$$\boldsymbol{\epsilon}_k = \mathbf{S}(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k) = \mathbf{S}\left(\mathbf{z}'_k - \mathbf{G}(\hat{\mathbf{x}}'_k - \mathbf{x}'_k)\right), \tag{4.3}$$

as $\mathbf{z}'_k$ is assumed independent of other variables, the mean-squared error (MSE) can be expressed as:

$$\begin{aligned}\mathbb{E}\|\boldsymbol{\epsilon}_k\|^2 =& NP\sigma_z^2 + \mathrm{trace}(\mathbf{G}^\top\mathbf{S}^\top\mathbf{S}\mathbf{G}\mathbb{E}\{\hat{\mathbf{x}}'_k\hat{\mathbf{x}}'^\top_k\}) \\ &+ \mathbf{x}'^\top_k\mathbf{G}^\top\mathbf{S}^\top\mathbf{S}\mathbf{G}\mathbf{x}'_k - 2\mathbf{x}'^\top_k\mathbf{G}^\top\mathbf{S}^\top\mathbf{S}\mathbf{G}\mathbb{E}\{\hat{\mathbf{x}}'_k\}.\end{aligned}$$

It follows that $\mathbb{E}\{\hat{\mathbf{x}}'_k\hat{\mathbf{x}}'^\top_k\}$ can be updated as:

$$\begin{aligned}\mathbb{E}\{\hat{\mathbf{x}}'_{k+1}\hat{\mathbf{x}}'^\top_{k+1}\} =& \mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}'_k\hat{\mathbf{x}}'^\top_k\}\mathbf{A}^\top + \mu^2\sigma_z^2\mathbf{\Phi}\mathbf{\Phi}^\top + \mu\mathbf{\Phi}\mathbf{G}\mathbf{x}'_k\mathbb{E}\{\hat{\mathbf{x}}'^\top_k\}\mathbf{A}^\top \\ &+ \mu\mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}'_k\}\mathbf{x}'^\top_k\mathbf{G}^\top\mathbf{\Phi}^\top + \mu^2\mathbf{\Phi}\mathbf{G}\mathbf{x}'_k\mathbf{x}'^\top_k\mathbf{G}^\top\mathbf{\Phi}^\top \\ &- \rho_z(\mathbf{U}_1 + \mathbf{U}_1^\top + \mathbf{U}_2 + \mathbf{U}_2^\top) + \rho_z^2\mathbf{U}_3 \\ &- \rho_s(\mathbf{U}_4 + \mathbf{U}_4^\top + \mathbf{U}_5 + \mathbf{U}_5^\top) + \rho_s^2\mathbf{U}_6 + \rho_s\rho_z(\mathbf{U}_7 + \mathbf{U}_7^\top)\end{aligned}$$

where

$$\begin{aligned}\mathbf{U}_1 &= \mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}'_k\mathrm{sign}(\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Gamma}^\top, \\ \mathbf{U}_2 &= \mu\mathbf{\Phi}\mathbf{G}\mathbf{x}'_k\mathbb{E}\{\mathrm{sign}(\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Gamma}^\top, \\ \mathbf{U}_3 &= \mathbf{\Gamma}\mathbb{E}\{\mathrm{sign}(\hat{\mathbf{x}}'_k)\mathrm{sign}(\hat{\mathbf{x}}'_k)^\top\}\,\mathbf{\Gamma}^\top, \\ \mathbf{U}_4 &= \mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}'_k\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Lambda}_s, \\ \mathbf{U}_5 &= \mu\mathbf{\Phi}\mathbf{G}\mathbf{x}'_k\mathbb{E}\{\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Lambda}_s, \\ \mathbf{U}_6 &= \mathbf{\Lambda}_s^\top\mathbb{E}\{\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Lambda}_s, \\ \mathbf{U}_7 &= \mathbf{\Gamma}\mathbb{E}\{\mathrm{sign}(\hat{\mathbf{x}}'_k)\mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)^\top\}\mathbf{\Lambda}_s.\end{aligned}$$

Again, the main difficulty lies in the evaluation of the expectations in form of $\mathbb{E}\{\mathrm{sign}(u)\,\mathrm{sign}(v)\}$ and $\mathbb{E}\{u\,\mathrm{sign}(v)\}$. An analysis of the stochastic behavior of ZA-LMS is carried out in [11] where exact expressions of these expectations are derived under a Gaussian assumption on the error vectors. however, due to the size of the covariance matrix (4.4), we decided to resort to use the following approximations [49]:

$$\mathbb{E}\{\mathrm{sign}(u)\} \approx \mathrm{sign}(\mathbb{E}\{u\}), \tag{4.4}$$

$$\mathbb{E}\{u\,\mathrm{sign}(v)^\top\} \approx \mathbb{E}\{u\}\mathrm{sign}(\mathbb{E}\{v\})^\top, \tag{4.5}$$

$$\mathbb{E}\{\mathrm{sign}(u)\,\mathrm{sign}(v)^\top\} \approx \mathrm{sign}(\mathbb{E}\{u\})\mathrm{sign}\left(\mathbb{E}\{v\}\right)^\top \tag{4.6}$$

which lead to sufficiently accurate models. When $|u|$ and $|v|$ are large, the approximations are quite good meaning that the mean behavior is well approximated when the dynamic of the image to restore is high enough. When they are close to zero, the approximations are of lower quality. However, following [12], these approximations yield a worst case transient behavior model. In addition, this approximation does not assume any statistical model on u and v. Only their statistical independence is required.

The mean and mean-squares transient models will be used to study the SBR-LMS behavior in both steady-state and non-stationary cases (Subsection 5.1). In the next section, to study the stability condition of the algorithm, we focus on the steady-state case.

4.2. Stability condition. We shall now discuss the stability of the proposed algorithm in the mean and mean-squares sense. Consider a steady-state solution for which:

$$\mathbf{x}_k^p = \bar{\mathbf{x}}^p, \quad \forall\, k. \tag{4.7}$$

The error-vector $\mathbf{v}_k^p$ is defined as:

$$\mathbf{v}_k^p = \bar{\mathbf{x}}^p - \hat{\mathbf{x}}_k^p \tag{4.8}$$

with its vectorized version

$$\mathbf{v}_k \triangleq \mathrm{col}\{\mathbf{v}_k^p\}_{p=1}^P \tag{4.9}$$

$$\mathbf{v}'_k \triangleq \mathrm{col}\{\mathbf{v}_{k-q+1}\}_{q=1}^{Q+L-1}. \tag{4.10}$$

By combining (3.16), (4.8) and (4.10), $\mathbf{v}'_k$ can be updated as:

$$\begin{aligned}\mathbf{v}'_{k+1} &= \mathbf{A}\mathbf{v}'_k - \mu\mathbf{\Phi}\mathbf{z}'_k + (\mathbf{I} - \mathbf{\Omega} + \mu\eta_\lambda {\mathbf{\Lambda}_\lambda}^\top \mathbf{\Lambda}_\lambda)\bar{\mathbf{x}}' + \rho_z\mathbf{\Gamma}\mathrm{sign}(\hat{\mathbf{x}}'_k) + \rho_s {\mathbf{\Lambda}_\mathbf{s}}^\top \mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k) \\ &= \mathbf{A}\mathbf{v}'_k - \mu\mathbf{\Phi}\mathbf{z}'_k + \mu\eta_\lambda {\mathbf{\Lambda}_\lambda}^\top \mathbf{\Lambda}_\lambda\bar{\mathbf{x}}' + \rho_z\mathbf{\Gamma}\mathrm{sign}(\hat{\mathbf{x}}'_k) + \rho_s {\mathbf{\Lambda}_\mathbf{s}}^\top \mathrm{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)\end{aligned} \tag{4.11}$$

where $\bar{\mathbf{x}} \triangleq \mathrm{col}\{\bar{\mathbf{x}}^p\}_{p=1}^P$ and $\bar{\mathbf{x}}' \triangleq [\bar{\mathbf{x}}^\top, \ldots, \bar{\mathbf{x}}^\top]^\top$ of size $(Q+L-1)PN \times 1$. We consider the following partition of the matrix $\mathbf{A}$:

$$\begin{aligned}\mathbf{A} &= \left[\begin{array}{c|c} \mathbf{I} & \mathbf{0} \\ \hline \mathbf{\Omega}_{21} & \mathbf{\Omega}_{22} \end{array}\right] - \mu \left[\begin{array}{c|c} \mathbf{\Phi}_{11}^\top & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \left[\begin{array}{c|c} \mathbf{\Phi}_{11} & \mathbf{G}_{12} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] - \mu\eta_\lambda \left[\begin{array}{c|c} \mathbf{\Lambda}_{\lambda,1}^\top \mathbf{\Lambda}_{\lambda,1} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \\ &\triangleq \left[\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{\Omega}_{21} & \mathbf{\Omega}_{22} \end{array}\right]\end{aligned} \tag{4.12}$$

where $\mathbf{\Phi}_{11}$ is defined by (3.11), $\mathbf{\Omega}_{21}$ and $\mathbf{\Omega}_{22}$ are defined by (3.17), $\mathbf{A}_{11} = \mathbf{I} - \mu(\mathbf{\Phi}_{11}^\top \mathbf{\Phi}_{11} + \eta_\lambda \mathbf{\Lambda}_{\lambda,1}^\top \mathbf{\Lambda}_{\lambda,1})$, $\mathbf{A}_{12} = -\mu\mathbf{\Phi}_{11}^\top \mathbf{G}_{12}$.

Mean stability. We now consider the mean stability of (4.11). Taking the expectation of both sides, the mean-error vector can be expressed as:

$$\mathbb{E}\{\mathbf{v}'_{k+1}\} = \mathbf{A}\mathbb{E}\{\mathbf{v}'_k\} + \mathbf{d}_k \tag{4.13}$$

where

$$\mathbf{d}_k \triangleq \mu\eta_\lambda \mathbf{\Lambda}_\lambda^\top \mathbf{\Lambda}_\lambda \bar{\mathbf{x}}' + \rho_z \mathbb{E}\{\mathrm{sign}(\hat{\mathbf{x}}'_k)\} + \rho_s \mathbf{\Lambda}_s^\top \mathbb{E}\{\mathrm{sign}(\mathbf{\Lambda}_s \hat{\mathbf{x}}'_k)\}. \tag{4.14}$$

Let $\mathbb{E}\{\mathbf{v}'_1\}$ be the initial condition. Then we obtain:

$$\mathbb{E}\{\mathbf{v}'_{k+1}\} = (\mathbf{A})^k \mathbb{E}\{\mathbf{v}'_1\} + \sum_{n=0}^{k-1} (\mathbf{A})^n \mathbf{d}_{k-n} \tag{4.15}$$

The Euclidean norm of $\mathbf{d}_k$ is bounded. Indeed, we have:

$$||\mathbf{d}_k|| \leq \mu\eta_\lambda ||\mathbf{\Lambda}_\lambda^\top \mathbf{\Lambda}_\lambda \bar{\mathbf{x}}'|| + \rho_z ||\mathbb{E}\{\text{sign}(\hat{\mathbf{x}}'_k)\}|| + \rho_s ||\mathbf{\Lambda}_s^\top \mathbb{E}\{\text{sign}(\mathbf{\Lambda}_s \hat{\mathbf{x}}'_k)\}||. \tag{4.16}$$

The first term $||\mathbf{\Lambda}_\lambda^\top \mathbf{\Lambda}_\lambda \bar{\mathbf{x}}'||$ corresponds to the initial condition and is bounded if $\bar{\mathbf{x}}'$ is bounded. The last two terms are also bounded:

$$||\mathbb{E}\{\text{sign}(\hat{\mathbf{x}}'_k)\}|| \leq \sqrt{QPN} \tag{4.17}$$

$$||\mathbf{\Lambda}_s^\top \mathbb{E}\{\text{sign}(\mathbf{\Lambda}_s \hat{\mathbf{x}}'_k)\}|| \leq \sqrt{QP(4N-6)}. \tag{4.18}$$

Following the arguments reported in [17], if the spectral norm of $\mathbf{A}$ is less than 1, the series $\sum_{n=0}^{\infty} (\mathbf{A})^n \mathbf{d}_{k-n}$ is convergent. Thus, the convergence of (4.15) is entirely controlled by the spectral norm of $\mathbf{A}$. Because matrix $\mathbf{A}$ is not symmetric, standard stability results cannot be applied straightforwardly. Let $\mathbb{D}$ be the closed unit disc in the complex plane. We say that the matrix $\mathbf{A}$ is discrete stable (see [6]) if $\delta(z) = \det(\mathbf{I} - z\mathbf{A}) \neq 0$, $\forall z \in \mathbb{D}$. Using the Schur determinant formula, it can be shown that:

$$\delta(z) = \det(\mathbf{I} - z\mathbf{A}_{11}) \cdot \det(\mathbf{I} - z\mathbf{\Omega}_{22} - z^2 \mathbf{\Omega}_{21}(\mathbf{I} - z\mathbf{A}_{11})^{-1}\mathbf{A}_{12}). \tag{4.19}$$

Then, $\delta(z) \neq 0$ if and only if the following two conditions hold:

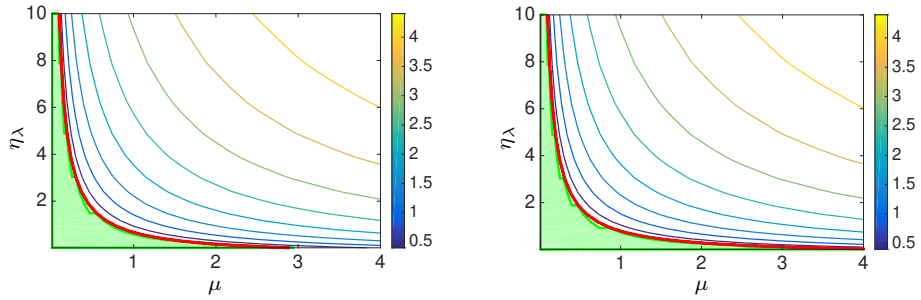
$$\det(\mathbf{I} - z\mathbf{A}_{11}) \neq 0, \quad \forall z \in \mathbb{D} \tag{4.20}$$

$$\det(\mathbf{I} - z\mathbf{\Omega}_{22} - z^2 \mathbf{\Omega}_{21}(\mathbf{I} - z\mathbf{A}_{11})^{-1}\mathbf{A}_{12}) \neq 0, \quad \forall z \in \mathbb{D} \tag{4.21}$$

The condition in (4.20) is fulfilled if all the eigenvalues of the symmetric matrix $\mathbf{A}_{11}$ lie inside the unit disc, i.e., its spectral radius $r(\mathbf{A}_{11}) < 1$. The matrix $\mathbf{\Phi}_{11}^\top \mathbf{\Phi}_{11} + \eta_\lambda \mathbf{\Lambda}_{\lambda,1}^\top \mathbf{\Lambda}_{\lambda,1}$ being the sum of a positive semi-definite matrix and a positive definite matrix, it is positive definite; thus, standard stability result applies from which we conclude that $\mathbf{A}_{11}$ is stable iff:

$$0 < \mu < \frac{2}{r(\mathbf{\Phi}_{11}^\top \mathbf{\Phi}_{11} + \eta_\lambda \mathbf{\Lambda}_{\lambda,1}^\top \mathbf{\Lambda}_{\lambda,1})}. \tag{4.22}$$

In practice, it is not easy to check the second condition in (4.21) for all values of z in the unit disc. However, in the case of Gaussian convolution kernels, we checked experimentally that the stability of $\mathbf{A}$ depends only on that of $\mathbf{A}_{11}$. For example, Figure 4 displays the spectral radius of $\mathbf{A}$ as a function of the tuning parameters μ and η_λ for two Gaussian kernels. The red curve is obtained using (4.22) which corresponds to the stability limit of $\mathbf{A}_{11}$. The region where the spectral radius of $\mathbf{A}$ is less than 1 (i.e., $\mathbf{A}$ is stable) is highlighted in green. One can observe that for different Gaussian kernels, except for some numerical errors, the stability region of $\mathbf{A}$ is the same as that of $\mathbf{A}_{11}$. In conclusion, we conjecture that for Gaussian filters, the stability condition of $\mathbf{A}$ is also given by (4.22). Numerical experiments seem to indicate that the stability of $\mathbf{A}$ is also controlled by the stability of $\mathbf{A}_{11}$ for a larger class of low-pass filters, including hanning, hamming and rectangular windows. We also found counter examples for which the stability of $\mathbf{A}_{11}$ does not guarantee the stability of $\mathbf{A}$. It is worth mentioning that the stability region depends on both μ and η_λ and that increasing η_λ necessitates to lower μ to guarantee the stability of the algorithm.



(a) $L = 3$, $Q = 4$ (b) $L = 11$, $Q = 12$

Fig. 4: Spectral radius of $\mathbf{A}$ (in log scale) as a function of μ and η_λ. The green area corresponds to the values of μ and η_λ for which $\mathbf{A}$ is stable. The red curve corresponds to the stability limit of $\mathbf{A}_{11}$.

Mean-squares stability. We now address the stability in the mean-squares sense. First, define the zero-mean misalignment vector $\mathbf{w}_k$ as:

$$\mathbf{w}_k = \mathbf{v}'_k - \mathbb{E}\{\mathbf{v}'_k\}, \tag{4.23}$$

then using (4.11) and (4.13), $\mathbf{w}_k$ can be updated as:

$$\mathbf{w}_{k+1} = \mathbf{A}\mathbf{w}_k - \mu\mathbf{\Phi}\mathbf{z}'_k + \rho_z\mathbf{e}_{\mathbf{z}k} + \rho_s\mathbf{e}_{\mathbf{s}k} \tag{4.24}$$

where

$$\mathbf{e}_{\mathbf{z}k} = \text{sign}(\hat{\mathbf{x}}'_k) - \mathbb{E}\{\text{sign}(\hat{\mathbf{x}}'_k)\}, \tag{4.25}$$

$$\mathbf{e}_{\mathbf{s}k} = \mathbf{\Lambda}_s^\top\,(\text{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k) - \mathbb{E}\{\text{sign}(\mathbf{\Lambda}_s\hat{\mathbf{x}}'_k)\}). \tag{4.26}$$

The covariance matrix of $\mathbf{w}_k$ is defined as:

$$\mathbf{W}_k = \mathbb{E}\{\mathbf{w}_k\mathbf{w}_k^\top\} \tag{4.27}$$

Since $\mathbb{E}\{\mathbf{w}_k\} = \mathbf{0}$, $\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}\} = \mathbf{0}$ and $\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}\} = \mathbf{0}$, $\mathbf{W}_k$ can be updated as follows:

$$\begin{aligned}\mathbf{W}_{k+1} =& \mathbf{A}\mathbf{W}_k\mathbf{A}^\top + \mu^2\sigma_z^2\mathbf{\Phi}\mathbf{\Phi}^\top + \rho_z\mathbf{A}\mathbb{E}\{\mathbf{w}_k\mathbf{e}_{\mathbf{z}k}^\top\} + \rho_z\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}\mathbf{w}_k^\top\}\mathbf{A}^\top + \rho_z^2\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}\mathbf{e}_{\mathbf{z}k}^\top\} \\ &+ \rho_s\mathbf{A}\mathbb{E}\{\mathbf{w}_k\mathbf{e}_{\mathbf{s}k}^\top\} + \rho_s\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}\mathbf{w}_k^\top\}\mathbf{A}^\top + \rho_s^2\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}\mathbf{e}_{\mathbf{s}k}^\top\} \\ &+ \rho_s\rho_z\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}\mathbf{e}_{\mathbf{z}k}^\top\} + \rho_s\rho_z\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}\mathbf{e}_{\mathbf{s}k}^\top\}\end{aligned} \tag{4.28}$$

From (4.8) and (4.23) we have:

$$\mathbb{E}\{\mathbf{w}_k\mathbf{e}_{\mathbf{z}k}^\top\} = \mathbb{E}\{\hat{\mathbf{x}}'_k\}\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}^\top\} - \mathbb{E}\{\hat{\mathbf{x}}'_k\mathbf{e}_{\mathbf{z}k}^\top\} = -\mathbb{E}\{\hat{\mathbf{x}}'_k\mathbf{e}_{\mathbf{z}k}^\top\}, \tag{4.29}$$

$$\mathbb{E}\{\mathbf{w}_k\mathbf{e}_{\mathbf{s}k}^\top\} = \mathbb{E}\{\hat{\mathbf{x}}'_k\}\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}^\top\} - \mathbb{E}\{\hat{\mathbf{x}}'_k\mathbf{e}_{\mathbf{s}k}^\top\} = -\mathbb{E}\{\hat{\mathbf{x}}'_k\mathbf{e}_{\mathbf{s}k}^\top\}. \tag{4.30}$$

Thus, we observe:

$$\begin{aligned}\text{trace}(\mathbf{W}_{k+1}) =& \text{trace}(\mathbf{A}^\top\mathbf{A}\mathbf{W}_k) + \mu^2\sigma_z^2\text{trace}(\mathbf{\Phi}\mathbf{\Phi}^\top) + 2\rho_s\rho_z\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}^\top\mathbf{e}_{\mathbf{z}k}\} \\ &- 2\rho_z\text{trace}(\mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}_k^{'\top}\mathbf{e}_{\mathbf{z}k}\}) + \rho_z^2\mathbb{E}\{\mathbf{e}_{\mathbf{z}k}^\top\mathbf{e}_{\mathbf{z}k}\} \\ &- 2\rho_s\text{trace}(\mathbf{A}\mathbb{E}\{\hat{\mathbf{x}}_k^{'\top}\mathbf{e}_{\mathbf{s}k}\}) + \rho_s^2\mathbb{E}\{\mathbf{e}_{\mathbf{s}k}^\top\mathbf{e}_{\mathbf{s}k}\}.\end{aligned} \tag{4.31}$$

It is straightforward to verify that $\mathbf{e}_{\mathbf{z}k}$ and $\mathbf{e}_{\mathbf{s}k}$ are bounded. With $\mathbb{E}\{\hat{\mathbf{x}}'_k\} = \bar{\mathbf{x}}' - \mathbb{E}\{\mathbf{v}'_k\}$, the terms $\mathbb{E}\{\hat{\mathbf{x}}_k'^{\top}\mathbf{e}_{\mathbf{z}k}\}$ and $\mathbb{E}\{\hat{\mathbf{x}}_k'^{\top}\mathbf{e}_{\mathbf{s}k}\}$ converge if $\mathbb{E}\{\mathbf{v}'_k\}$ converges which is true if $\mathbf{A}$ is stable and $\bar{\mathbf{x}}'$ is bounded. Thus to have the stability of (4.28), only the stability of its first term on the right hand side has to be studied (see for example [37]). By using the same argument as in [17], the mean-squares stability of (4.28) is guaranteed if the mean stability holds.

5. Experimental results.

5.1. Validation of the transient behavior model. The experiment described below aims at validating the transient behavior model of the SBR-LMS algorithm described in Subsection 4.1. We first address the steady-state case. The simulated image is a 2D image whose lines are constant over time k. Its columns are set to: $\mathbf{x}^{stat.} = [\mathbf{0}_{1\times 3}, 1, 0.9, \ldots, 0.1, \mathbf{0}_{1\times 3}]^{\top}$. The convolution kernel is a Gaussian filter of size 3×3. The zero-mean i.i.d. Gaussian noise $\mathbf{z}_k$ is added to reach a 16 dB SNR. The SNR is defined as follows:

$$\text{SNR} = 10\log_{10}\frac{||\mathbf{H}^{\star} * \mathbf{X}||_F^2}{||\mathbf{Z}||_F^2}. \tag{5.1}$$

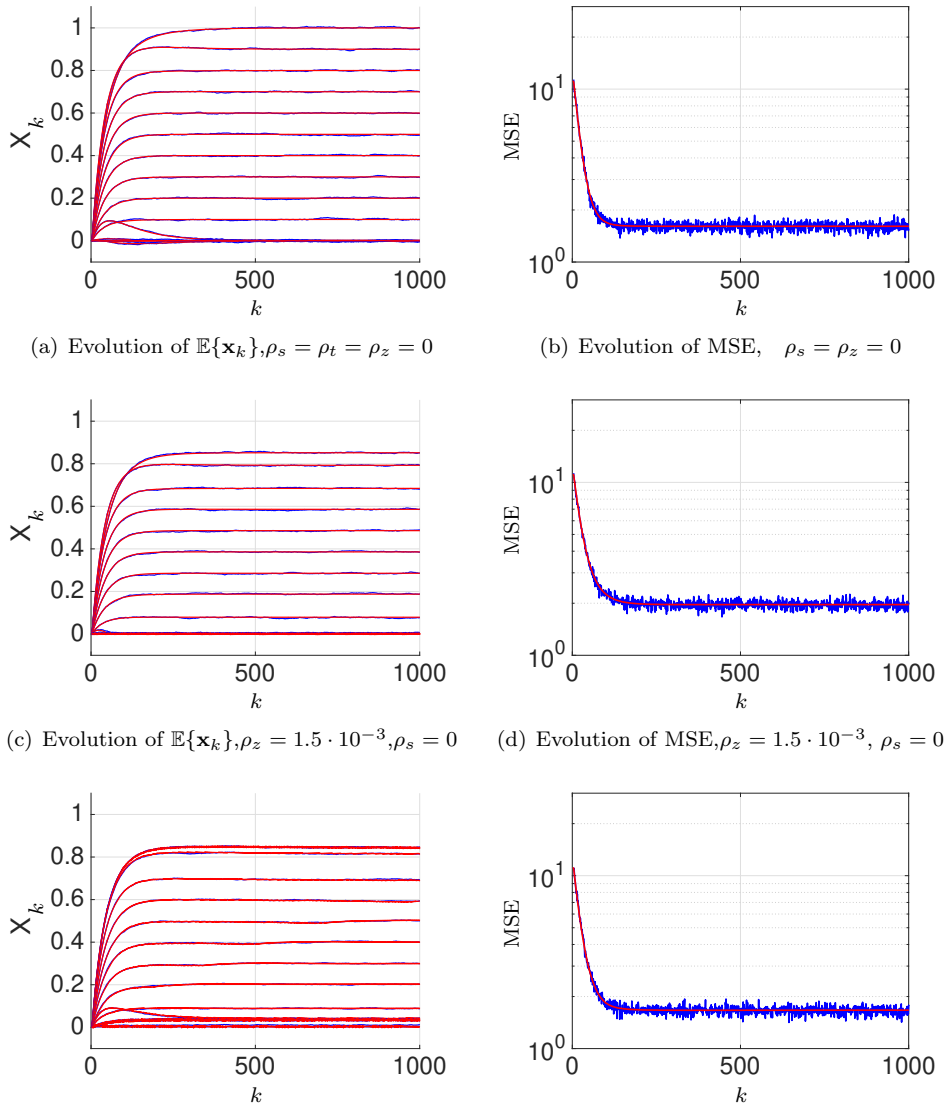
The block size Q is set to 5, the initial value $\mathbf{x}_0$ to zero and the step size μ to 0.01. Experimental results were obtained by averaging the estimations over 50 runs. The convergence behavior in the mean and mean-squares sense of SBR-LMS with different regularization parameter values are presented in Figure 5. The experimental curves (in blue) and the theoretical curves (4.1)–(4.4) (in red) are superimposed thus confirming the accuracy of the model. It appears that the zero-attracting property results in a faster convergence to zero (Figure 5(c)) than that of the algorithm without any regularization (Figure 5(a)). The first order derivative regularizer (Figure 5(e)) is favoring the reconstruction of piecewise constant objects along the spatial dimension by decreasing the difference between two adjacent rows. However, both zero-attracting and the first order derivative properties introduce a bias on the amplitudes explaining why the MSEs shown in Figure 5(d) and Figure 5(f) are larger than that shown in Figure 5(b).

The next experiment intends to show that the proposed model is still valid in the non-stationary case. The results are reported in Figure 6. The n-th pixel at time k of the 2D simulated image is defined as:

$$x_{nk}^{(nonstat.)} = x_n^{(stat.)} + \frac{|x_n^{(stat.)}|}{10}\sin\left(\frac{2\pi}{T_o}k + 2\pi\frac{n-1}{K}\right) \tag{5.2}$$

with $T_o = 100$. The convolution filter and noise level are the same as before. Q is set to 5, ρ_z and ρ_s are set to $0.2 \cdot 10^{-3}$. The simulated images (in black dots) have to be compared with the superimposed experimental (blue) and theoretical (red) curves. Figure 6(a), Figure 6(b) and Figure 6(c), Figure 6(d) confirm that the step size influences the convergence speed and the estimation variance. A large value for μ results in an increase of the convergence speed and estimation variance. Also, the delay mentioned in Subsection 3.2 and Appendix B can be observed on these figures. Increasing the step size (Figure 6(c)) results in a decreasing of the delay.

5.2. Effects of the parameters. To assess the influence of the parameters Q, μ, ρ_s and ρ_z, numerical simulations were conducted on a 2D image. The original image shown in Figure 7(a) was blurred by a Gaussian kernel of size 15×15 with



(a) Evolution of $\mathbb{E}\{\mathbf{x}_k\}$,$\rho_s = \rho_t = \rho_z = 0$

(b) Evolution of MSE, $\rho_s = \rho_z = 0$

(c) Evolution of $\mathbb{E}\{\mathbf{x}_k\}$,$\rho_z = 1.5 \cdot 10^{-3}$,$\rho_s = 0$

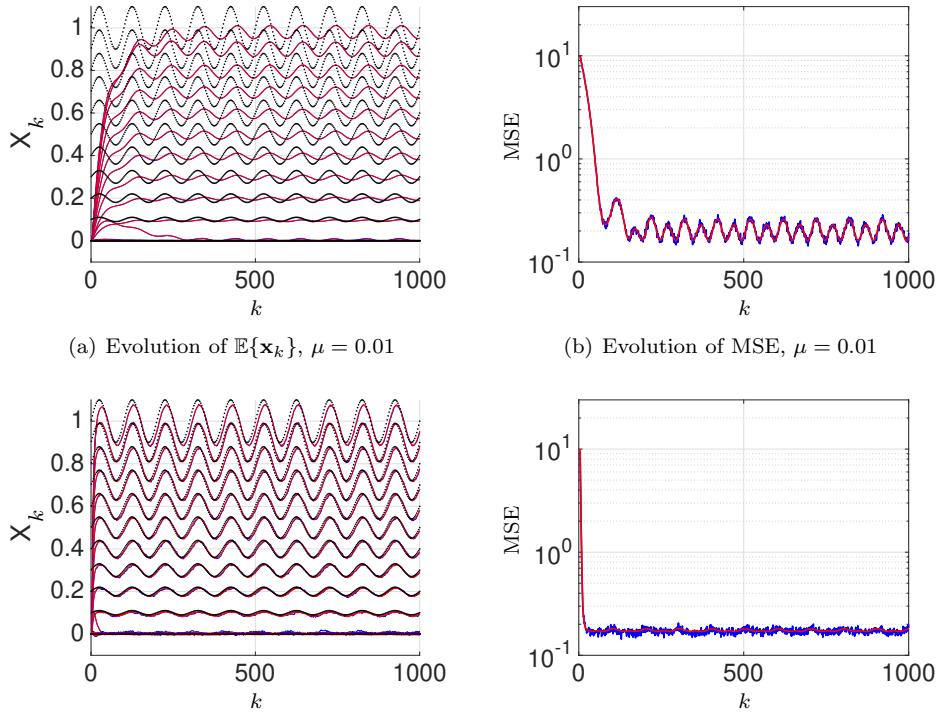
(d) Evolution of MSE,$\rho_z = 1.5 \cdot 10^{-3}$, $\rho_s = 0$

(e) Evolution of $\mathbb{E}\{\mathbf{x}_k\}$,$\rho_z = 0$, $\rho_s = 1.5 \cdot 10^{-3}$

(f) Evolution of MSE,$\rho_z = 0$, $\rho_s = 1.5 \cdot 10^{-3}$

Fig. 5: Transient behavior model in stationary environment

full width at half-maximum (FWHM) set to 7 pixels. A Gaussian noise was added to reach a 5 dB SNR. This blurred and noisy image is shown in Figure 7(b). Figure 7(c) to Figure 7(f) show respectively the results corresponding to the ZA-LMS algorithm obtained by setting $Q = 1$, $\mu = 0.06$ and $\rho_z = 0.02$, the SB-LMS algorithm with hyperparameters $Q = 15$, $\mu = 0.006$, the SBZA-LMS (sliding block LMS with zero-attracting term only) with hyperparameters $Q = 15$, $\mu = 0.006$ and $\rho_z = 0.05$ and the SBR-LMS with hyperparameters $Q = 15$, $\mu = 0.006$, $\rho_z = 0.05$ and $\rho_s = 0.01$. The image restored with ZA-LMS in Figure 7(c) exhibits lower noise level than the



(a) Evolution of $\mathbb{E}\{\mathbf{x}_k\}$, $\mu = 0.01$

(b) Evolution of MSE, $\mu = 0.01$

(c) Evolution of $\mathbb{E}\{\mathbf{x}_k\}$, $\mu = 0.1$

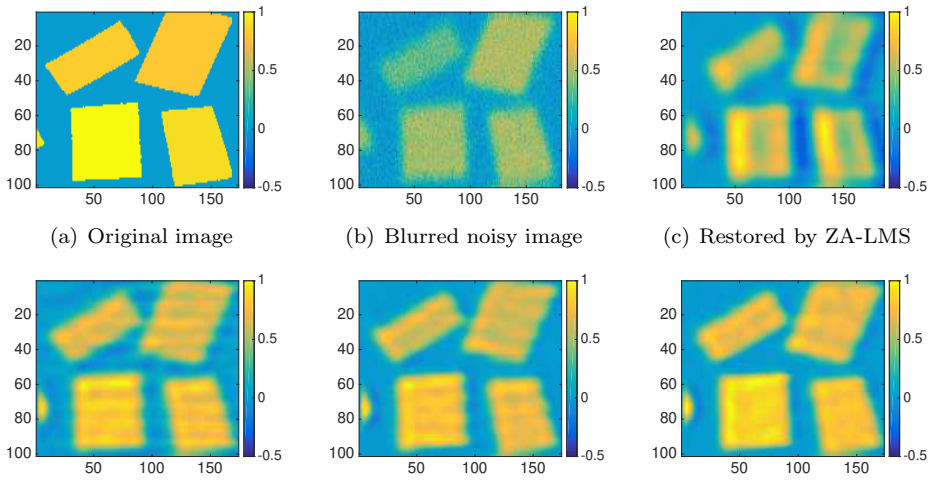
(d) Evolution of MSE, $\mu = 0.1$

Fig. 6: Transient behavior model validation in non-stationary environment

original image but deblurring is very limited. Better results are obtained when the block size increases as shown in Figure 7(d). The image restored by SBZA-LMS has a better resolution and a lower noise level as illustrated in Figure 7(e). As mentioned in Subsection 3.2 and Appendix B, the SB-LMS and SBZA-LMS induce a time (and no spatial) regularization resulting in horizontal structures visible on the restored images. This effect is attenuated by introducing a spatial regularization as shown in Figure 7(f).

Figure 8 shows the influence of the spectral regularization parameter η_λ on a simulated hyperspectral image. Original and blurred noisy images corresponding to three different wavelengths are shown in Figure 8(a) and Figure 8(b), respectively. The blurring kernel and the noise level are the same as those in Figure 7(b). By comparing he estimated results shown in Figure 8(c) (without spectral regularization) and Figure 8(d) (with spectral regularization), it appears that spectral regularization can help to recover objects whose amplitude is very small (at a given wavelength) as pattern 1 in the second band. However, the spectral regularization may also induce artifacts: for example, pattern 2 appears in the second band when the spectral regularization is used while it is not present in the original image.

We can assess the effect of the parameter Q on the convergence of the SBR-LMS by examining the transient behavior of the MSE. The simulated image is the same as the one in Subsection 5.1 (2D image whose lines are constant over time k). It was blurred by Gaussian filter of size 5×5 with FWHM equal to 3 and the SNR of the



(a) Original image (b) Blurred noisy image (c) Restored by ZA-LMS

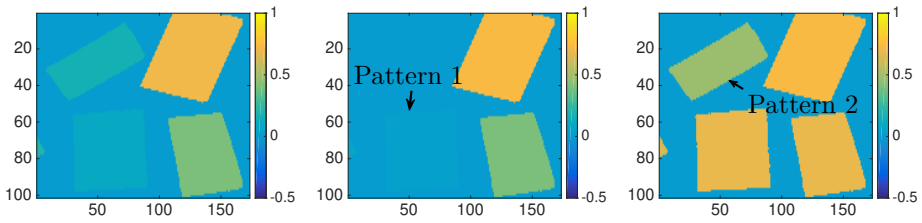
(d) Restored by SB-LMS (e) Restored by SBZA-LMS (f) Restored by SBR-LMS

Fig. 7: Influence of different hyperparameters

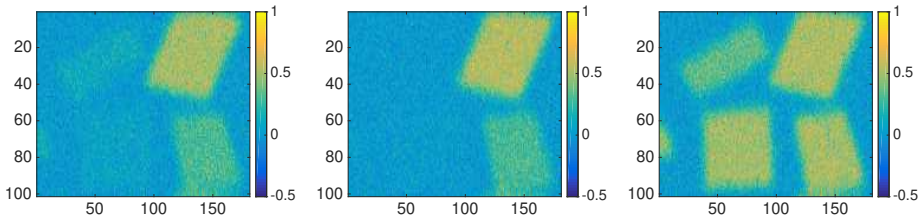
image was set to 40 dB. The initial value $\mathbf{x}_0$ was set to zero. Experimental MSE were obtained by averaging over 100 runs. The result comparing the transient behavior of MSE for different values of Q is shown in Figure 9. Convergence speed using the SBR-LMS increases when Q increases. Extensive simulations shows that Q should be chosen slightly larger than L (convolution kernel length along the time dimension). No improvements are observed when Q is much larger than L.

The next experiment aims at showing that the evolution of each parameter (the others being fixed) w.r.t. the SNR is complies with intuition. Figure 10 shows the evolution of the MSE as a function of μ, ρ_z, ρ_s and η_λ respectively for different values of the SNR. For a given SNR, the MSE curve (horizontal lines on Figure 10) decreases and then increases: the MSE minimum value (red point) corresponds to the best tradeoff between bias and variance. The values of ρ_z, ρ_s and η_λ minimizing the MSE increase as the noise level increases. The value of μ minimizing the MSE decreases as the noise level increases: this is in accordance with the time regularization implicitly enforced by the LMS (presented in Appendix B) whose strength is controlled by $1/\mu$.

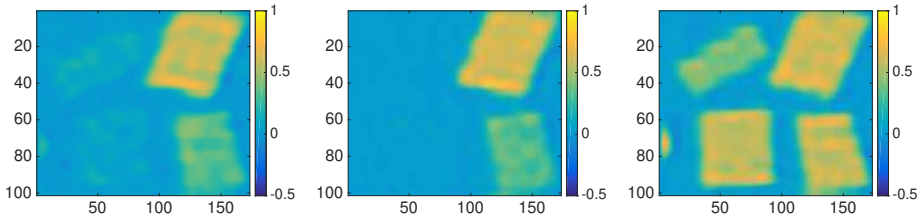
Let us now concentrate on the parameter ρ_z which controls the strength of the ZA terms. In Subsection 5.1, it was mentioned that ZA was favoring a faster convergence to zero. At the same time it can also slow down the convergence to non-zero values. To illustrate this point, we consider the following experiment. A simulated image of size $27 \times 500 \times 4$, shown in Figure 11(a) includes many zero values and some non-zero values. It was blurred by a Gaussian filter of size 3×3. Q was set to 5, $\mu = 0.01$, $\sigma_z^2 = .001$, $\rho_s = \rho_t = \eta_\lambda = 0$, $c_p = 1$. The initial value $\mathbf{x}_0$ was set to one. The practical MSE (blue) was obtained by averaging over 10 runs while the theoretical MSE (red) was evaluated according to (4.4). They are shown in Figure 11(b) for three different values of ρ_z. The result contains three parts. At first, the algorithm converges from one to smaller values, and larger ρ_z results in a faster convergence. However, as the first part of the image contains some non-zero values, the large value of ρ_z introduces



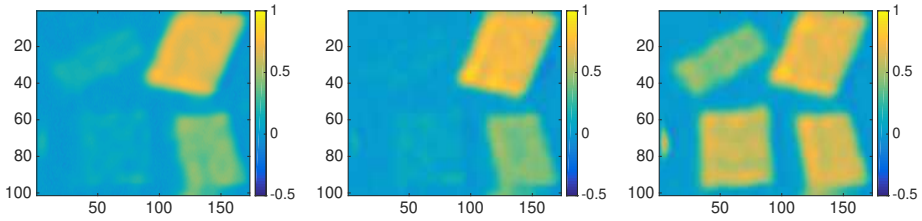
(a) Original simulated hyperspectral image



(b) Blurred noisy image



(c) Estimated image ($\eta_\lambda = 0$)



(d) Estimated image ($\eta_\lambda = 20$)

Fig. 8: Influence of spectral regularization parameter

a large bias resulting in a large MSE after convergence (around $k = 500$). In the second part of the image, pixels were all set to zero. The convergence rate of the algorithm increases with ρ_z. In that case, the zero-attracting does not introduce any bias and the MSEs after convergence (around $k = 1000$) are the same for all values of ρ_z. In the third part of the image, the algorithm starts from zero values and has to converge to non-zero values. Since the zero-attracting algorithm is pulling down the values to zero, large ρ_z will slow down the convergence rate. We note that the MSE after convergence (around $k = 1700$) are the same as that of the first part of the image. This experiment shows that the ZA term introduces an asymmetric transient behavior: it results in a faster convergence when starting from a non-zero

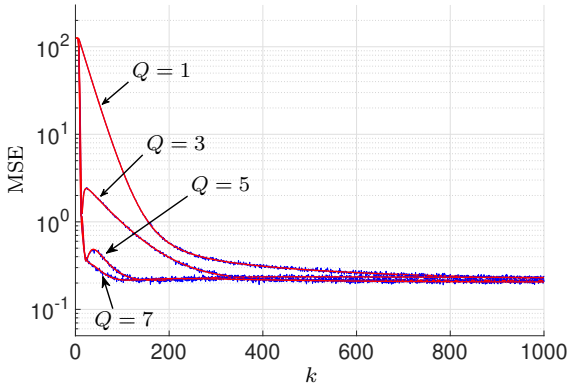
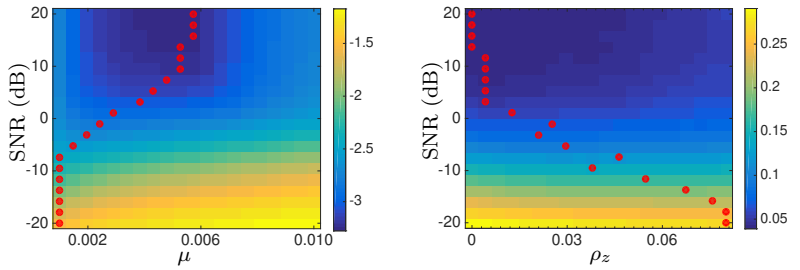


Fig. 9: Evolution fo MSE with different values of Q



(a) MSE as a function of μ ($\rho_s = \rho_z = 0$, $Q = 15$)

(b) MSE as a function ρ_z ($\mu = 0.006$, $\rho_s = 0$, $Q = 15$)

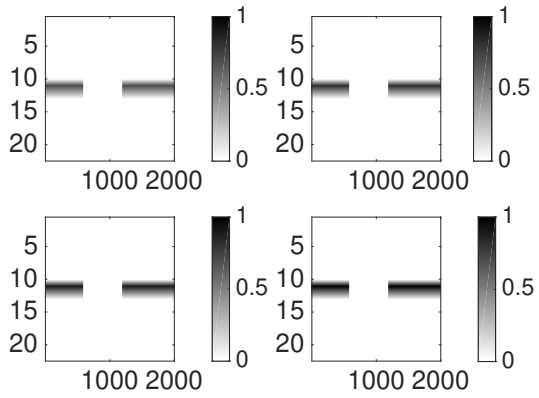
(c) MSE as a function ρ_s ($\mu = 0.006$, $\rho_z = 0$, $Q = 15$)

(d) MSE as a function η_λ ($\mu = 0.006$, $\rho_z = \rho_s = 0$, $Q = 15$)

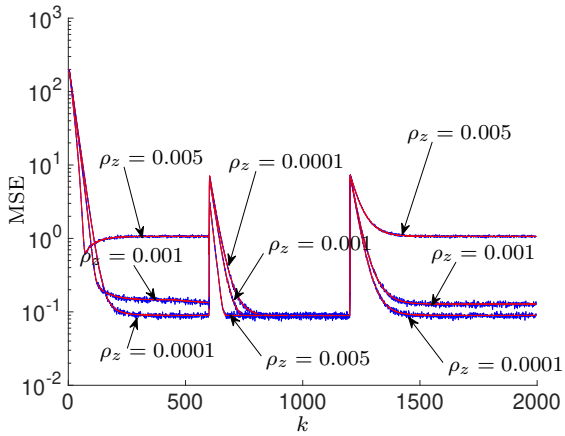
Fig. 10: Evaluation of MSE as a function of parameters for different values of the SNR

value toward a zero value and a slower convergence when starting from a zero value toward a non-zero value.

5.3. Performances. To illustrate the performance of our algorithm, we generated an unblurred hyperspectral image of 261 slices with spatial size 171 and 16 spectral bands. The chosen spatial image corresponds to 5 piecewise constant patterns put on the background whose response is zero. The spectra of the different



(a) Simulated image (4 wavelengths)



(b) Evolution of MSE with 3 values of ρ_z

Fig. 11: MSE as function of k for different values of ρ_z

objects were chosen to be smooth making the spectral smoothness penalty effective. The convolution filter $\mathbf{H}^\star$ was assumed to be a low-pass Gaussian filter of size (9×9). It is invariant with respect to spectral band p. A Gaussian noise was then added to the blurred image. The noise level was the same for all bands.

The SBR-LMS results are compared to an off-line deconvolution algorithm proposed in [27] which corresponds to the Tikhonov approach with ℓ_2 spatial and spectral regularizers. It is also compared to the BT approach presented in Subsection 3.1 whose block size is the same as the block size of the SBR-LMS. Note that better performance would certainly be achieved by using much more refined methods enforcing additional constraints (non-negativity, edge preserving) at the price of an increased computational burden. That is why we decided to restrict the comparison to deconvolution approaches having a computational burden similar to that of the SBR-LMS. Three versions of SBR-LMS algorithms are considered for performance evaluation: SBR-LMS without spatial and zero-attracting regularizations, SBR-LMS with spatial

regularization but without zero-attracting regularization, SBR-LMS with spatial and zero-attracting regularizations.

As a performance measure, we consider the MSE as a function of the SNR and the goal is to assess the performance of the proposed SBR-LMS and to compare them to those of alternative approaches. As shown in the previous section, the optimal hyperparameter values are depending on the noise level. Thus, comparing the performances of different methods would require to determine for each noise level the set of optimal parameters by an exhaustive grid search which, due to the large number of hyperparameters (2 for Tikhonov, 3 for Block Tikonov and up to 5 for SBR-LMS), is out of reach in reasonable time. To handle this problem, while keeping a fair comparison between the different methods, we propose the following performance evaluation: the strength of the spectral regularization is fixed to a small value (the same for all methods) and the block size Q is fixed to 11 (BT, SBR-LMS). All the other parameters are fixed by a grid search yielding the minimum MSE for a fixed SNR= 12 dB. These parameter values are then used for all other noise levels. Figure 12 shows that SBR-LMS without spatial and zero-attracting regularization terms yields the highest MSE for all noise level. This can be attributed to the fact that, in that case, only a time regularization is used. For high SNR, Tikhonov and BT give the lowest MSEs. however, when SNR is lower than 12 dB, SBR-LMS with spatial regularization has almost the same performance as the Tikhonov approach and better performance than BT algorithm. Adding a zero-attracting regularization improves the performance at low SNR even more. In conclusion, if we focus on sequential deconvolution, BT has to be chosen if the SNR is high while SBR-LMS should be preferred for low SNR.

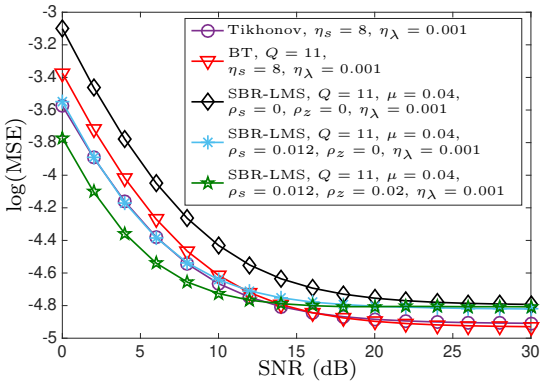


Fig. 12: Performances of the hyperspectral image deconvolution algorithms

5.4. Real hyperspectral image deblurring. The last experiment aims at illustrating the performance of the SBR-LMS algorithm on real blurred hyperspectral images of size $481 \times 656 \times 28$ (spatial, time and spectral sizes respectively) with wavelengths varying from 947.71 nm to 1707.7 nm with an increment of 27.68 nm. The spectral response of the conveyor (background) was estimated from data in an area of size $140 \times 140 \times 28$. It was then subtracted from each pixel of the hyperspectral image. The imaged objects put on the conveyor are pieces of woods (wood wastes). The convolution filter was estimated from data (using a calibration target) to be a Gaussian kernel of size 7×7 with FWHM of 3 points. No dependence of the

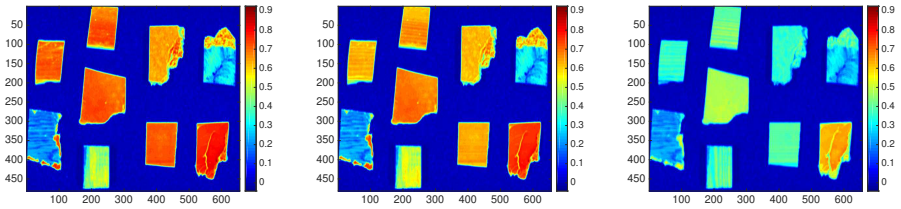
convolution kernel w.r.t. the wavelength was observed.

First, we compare different algorithms in the case of an image acquired using a large integration time (2.146 ms) resulting in a high SNR. This corresponds to the situation where Tikhonov and BT are expected to yield the best results. Figure 13 presents the deconvolution result obtained on the whole real hyperspectral image (only 3 slices corresponding to wavelengths 1058.64 nm, 1198.12 nm and 1479.78 nm are shown). Figure 14 is a part selected from the whole image. The original images are shown in Figure 13(a) and Figure 14(a). The images restored with SBR-LMS ($\mu = 2.3$, $Q = 9$, $\rho_z = 0.005$, $\rho_s = 0.01$, $\eta_\lambda = 0.001$) are shown in Figure 13(b) and Figure 14(b). Figure 13(c) and Figure 14(c) correspond to the Tikhonov approach with ℓ_2 spatial and spectral regularizers. Figure 13(d) and Figure 14(d) correspond to the BT approach. The hyperparameters of the Tikhonov approaches were estimated by the minimum distance criterion (MDC) proposed in [38]. The FWHB of the kernel being equal to 3 points, the blurring of the real images is quite limited. However, an increased resolution allowing to recover the fine texture of the objects can be observed on the zoomed view restored images in Figure 14(b)-Figure 14(d). More importantly, Figures Figure 13 and Figure 14 show that at high SNR, the SBR-LMS yields similar performances as those of the reference methods (Tikhonov and BT).

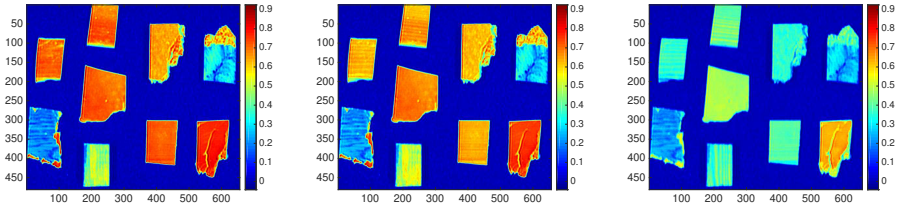
The second image in Figure 15(a) corresponds to the same observed scene with an integration time of 0.013 ms which results in a low SNR. The image restored with SBR-LMS ($\mu = 0.5$, $Q = 9$, $\rho_z = 0.1$, $\rho_s = 0.05$, $\eta_\lambda = 0.001$) is shown in Figure 15(b). Results obtained by Tikhonov and BT methods, whose hyperparameters are estimated by the MDC approach, are shown in Figure 15(b) and Figure 15(c), respectively. The bias introduced by the regularization terms can be observed in Figure 15(b) since the dynamic range is lower than that of Tikhonov methods. It clearly appears that the noise level on the background estimated by SBR-LMS is much lower than that of the Tikhonov approaches. Due to the noise level, all approaches cannot reveal fine structures accurately, but the edges are well preserved.

6. Conclusions. In this work, the online deconvolution problem of hyperspectral images collected by industrial pushbroom imaging systems was addressed. The contribution of this work is the derivation of the SBR-LMS algorithm that allows the fast slice-by-slice hyperspectral image restoration, accounting for convolution kernel non-causality and low SNR issues. The key feature of the proposed approach relies on the possibility to include non-quadratic regularization terms while maintaining a computational complexity linear w.r.t. the number of unknowns. The transient behavior model of the algorithm was analyzed; it allows to assess the influence of each regularization parameter. Experimental results on both simulated and real hyperspectral images showed that the SBR-LMS algorithm has good noise removal and deblurring capabilities, especially at low SNR which is the relevant case for industrial imaging systems.

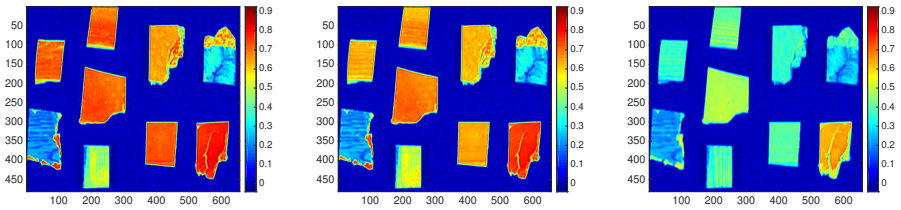
The effective implementation of such method is hampered by the choice of the regularization parameters. In general, this choice is made by successive trials which can be highly time consuming. Future works will focus on the automatic learning of hyperparameters. A first idea could be to learn the optimal parameters in a way similar to what is done in [38] on a typical sample and to use these fixed parameters for the online processing. A more promising approach would be to perform the online learning of these parameters: for example, [32], a variable-parameter algorithm is proposed to adjust optimally the algorithm parameters of ZA-LMS. This will avoid the necessity of having representative image samples and will meet the requirements



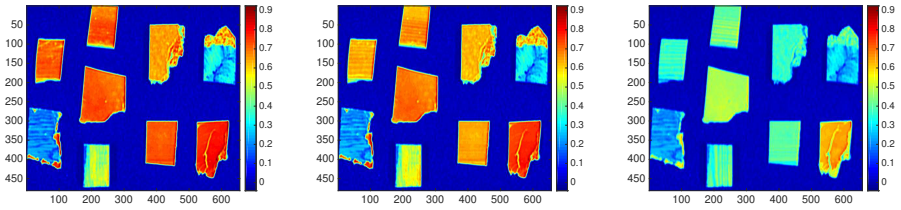
(a) Original hyperspectral image



(b) Estimated image (SBR-LMS)



(c) Estimated image (Tikhonov)

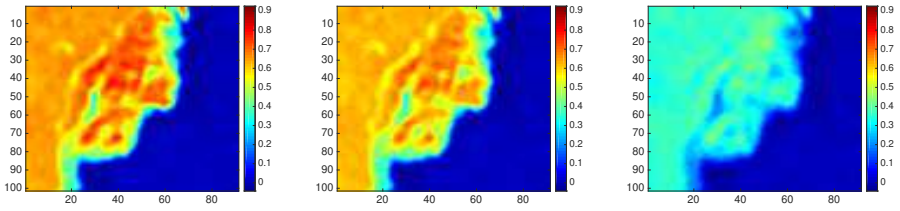


(d) Estimated image (Block Tikhonov)

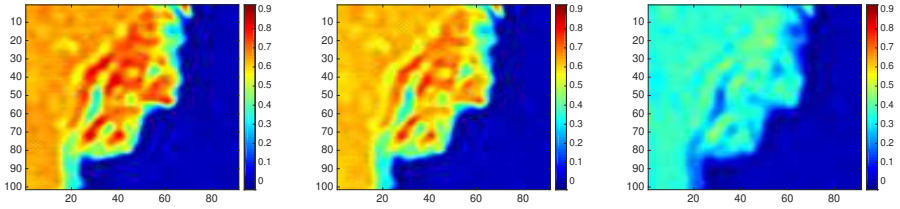
Fig. 13: hyperspectral image restoration at 3 wavelengths.

of a full online processing. To circumvent the asymmetric transient behavior induced by the ℓ_1 norm zero-attracting regularization, an ℓ_0-like regularizer such as introduced in [17] could be considered. For different types of materials and/or spectroscopies, other spectral regularizations could be used such as TV-like spectral regularization proposed in [1] for promoting piecewise constant spectra reconstruction. Finally, accounting for the low-rank structure of the data to restore (which results from a non-negative linear mixing model of the data to restore), a joint online deconvolution and unmixing algorithm is worth being studied. This is expected to yield a very low computational burden and accurate image restoration approach.

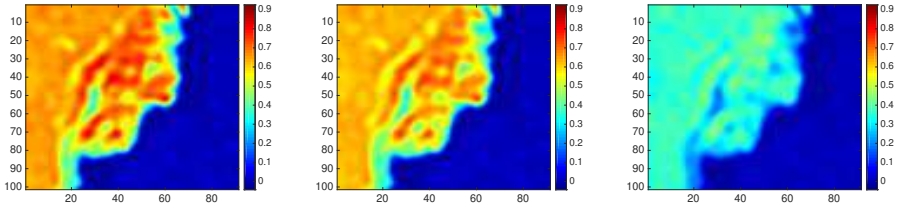
Appendix A. Structure of matrix G for both cases $Q \geq L$ and $Q < L$.



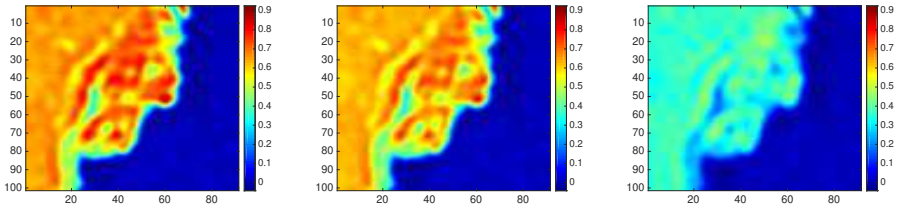
(a) Original hyperspectral image



(b) Estimated image (SBR-LMS)



(c) Estimated image (Tikhonov)

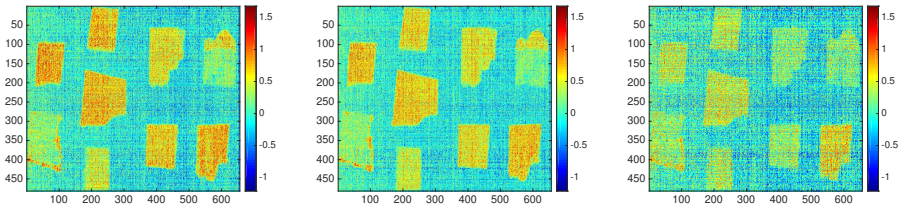


(d) Estimated image (Block Tikhonov)

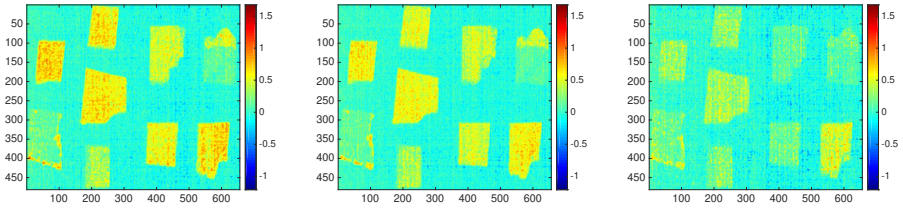
Fig. 14: hyperspectral image restoration at 3 wavelengths.

For $Q \geq L$, matrix $\mathbf{G}$ can be partitioned as follows:

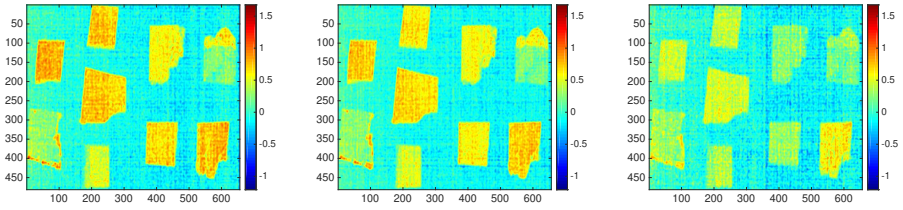
$$\mathbf{G} \triangleq \left[\begin{array}{cccc|ccc} \mathbf{H}_1 & \cdots & \mathbf{H}_L & & \mathbf{0} & & \\ & \ddots & & \ddots & & & \\ & & \mathbf{H}_1 & \cdots & \mathbf{H}_L & & \\ & & & \ddots & & \ddots & \\ & & & & \mathbf{H}_1 & \cdots & \mathbf{H}_L \\ \hline \multicolumn{7}{c}{\mathbf{0}_{(L-1)PN\times(Q+L-1)PN}} \end{array}\right], \tag{A.1}$$



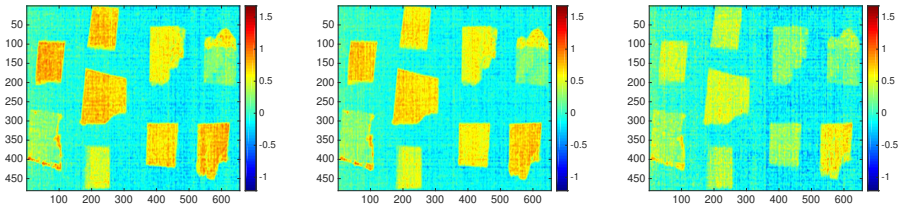
(a) Original hyperspectral image



(b) Estimated image (SBR-LMS)



(c) Estimated image (Tikhonov)



(d) Estimated image (Block Tikhonov)

Fig. 15: hyperspectral image restoration at 3 wavelengths.

and for $Q < L$, we have the form:

$$\mathbf{G} \triangleq \left[\begin{array}{ccc|ccc} \mathbf{H}_1 & \cdots & \mathbf{H}_Q & \cdots & \mathbf{H}_L & \mathbf{0} \\ & \ddots & & & \ddots & \\ & & \mathbf{H}_1 & \cdots & \mathbf{H}_Q & \cdots \quad \mathbf{H}_L \\ \hline \multicolumn{6}{c}{\mathbf{0}_{(L-1)PN\times(Q+L-1)PN}} \end{array}\right]. \tag{A.2}$$

Thus, we can conclude that for both cases $Q \geq L$ and $Q < L$, matrix $\mathbf{G}$ can be

written as:

$$\mathbf{G} \triangleq \left[\begin{array}{c|c} \mathbf{\Phi}_{11} & \mathbf{G}_{12} \\ \hline \mathbf{0}_{(L-1)PN\times(Q+L-1)PN} & \end{array}\right]. \tag{A.3}$$

Appendix B. SB-LMS interpreted as a time-regularized Tikhonov-like estimator. This appendix reveals that there is a hidden time-regularization introduced by the SB-LMS (that is the SBR-LMS where $\rho_s = \rho_z = \eta_\lambda = 0$) whose strength is controlled by the step size μ. Setting the spatial and spectral regularization terms to zero in (3.16), the SB-LMS is defined as:

$$\hat{\mathbf{x}}'_{k+1} = \mathbf{\Omega}\hat{\mathbf{x}}'_k + \mu\mathbf{\Phi}\left(\mathbf{y}'_k - \mathbf{G}\hat{\mathbf{x}}'_k\right) \tag{B.1}$$

Denoting $\hat{\mathbf{X}}'(z)$ as the z-transform of $\hat{\mathbf{x}}'_k$, Equation (B.1) can be expressed as:

$$\hat{\mathbf{X}}'(z) = \mu\left(z\mathbf{I} - \mathbf{\Omega} + \mu\mathbf{\Phi}\mathbf{G}\right)^{-1}\mathbf{\Phi}\mathbf{Y}'(z) \tag{B.2}$$

In another word, $\hat{\mathbf{X}}'(z)$ is the output of a linear filter:

$$\mathbf{F}(z) = \mu\left(z\mathbf{I} - \mathbf{\Omega} + \mu\mathbf{\Phi}\mathbf{G}\right)^{-1}\mathbf{\Phi} \tag{B.3}$$

with input $\mathbf{Y}'(z)$. It is well known that Tikhonov estimators result in a linear filtering of the data. Thus, the question we are addressing is to link these two linear filters. To simplify the analysis, we consider the denoising problem for which the convolution kernel is set to be identity. In that case, the block length Q is set to 1 (which results in $\hat{\mathbf{x}}'_k = \hat{\mathbf{x}}_k$) and (B.2) can be simplified as:

$$\hat{\mathbf{X}}(z) = \frac{\mu}{z - 1 + \mu}\mathbf{Y}(z) \tag{B.4}$$

Assuming a unit sampling step size, the forward transform is given by $s = z - 1$ where s is the Laplace parameter. The Fourier transform on the filter $F(z) = \dfrac{\mu}{z - 1 + \mu}$ is obtained by setting $s = j\omega$ where ω is the angular velocity:

$$F(\omega) = \frac{1}{1 + j\omega/\mu} = \frac{1}{(1 + \omega^2/\mu^2)^{0.5}}\exp^{-j\ \arctan(\omega/\mu)} \tag{B.5}$$

Combining the first-order approximation of $\arctan(\omega/\mu) \approx \omega/\mu$ together with the series expansion of $\sqrt{1 + x^2}$ can be used to give a low-pass approximation of the filter:

$$F(\omega) \approx \frac{1}{1 + \sum_{i=1}^{\infty}\beta_i(\mu)\omega^{2i}}\exp^{-j\ \omega/\mu} \tag{B.6}$$

where $\beta_i(\mu) = \dfrac{(2i)!}{(2\mu)^{2i}(i!)^2}$. Following [34], any filter of the form (B.6) results from to the minimization of a criterion:

$$\mathcal{J}(\mathbf{x}_{k+k_0}, \mathbf{y}_k) = ||\mathbf{y}_k - \mathbf{x}_{k+k_0}||^2 + \sum_{i=1}^{\infty}\beta_i(\mu)||\mathbf{D}_i\mathbf{x}_{k+k_0}||^2 \tag{B.7}$$

where $\mathbf{D}_i$ is the ith-order (discrete) derivative matrix along the time dimension and $k_0 = 1/\mu$. This means the LMS-based algorithms can be interpreted as delayed

Tikhonov-like algorithms. The first point to mention is that LMS-based estimators present a time delay in the estimation which is (approximately) proportional to $1/\mu$. The smaller μ is, the larger delay is. This has to be opposed to Tikhonov estimators which are null-phase filters and do not introduce any delay. The second point is related to the regularization parameters $\beta_i(\mu)$ which is proportional to $1/(2\mu)^{2i}$. A small value μ results in strong regularization strength along the time dimension.

REFERENCES

[1] H. K. Aggarwal and A. Majumdar, *Hyperspectral image denoising using spatio-spectral total variation*, IEEE Geoscience and Remote Sensing Letters, 13 (2016), pp. 442–446.

[2] R. Ammanouil, A. Ferrari, C. Richard, and D. Mary, *Blind and fully constrained unmixing of hyperspectral images*, IEEE Transactions on Image Processing, 23 (2014), pp. 5510–5518.

[3] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM journal on imaging sciences, 2 (2009), pp. 183–202.

[4] J. Biemond, J. Rieske, and J. Gerbrands, *A fast Kalman filter for images degraded by both blur and noise*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 31 (1983), pp. 1248–1256.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Foundations and Trends® in Machine learning, 3 (2011), pp. 1–122.

[6] Z. M. Buchevats and L. T. Gruyitch, *Linear Discrete-Time Systems*, CRC Press, 2017.

[7] C. Cariou and K. Chehdi, *Automatic georeferencing of airborne pushbroom scanner images with missing ancillary data using mutual information*, IEEE Transactions on Geoscience and Remote Sensing, 46 (2008), pp. 1290–1300.

[8] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, *Deterministic edge-preserving regularization in computed imaging*, IEEE Transactions on image processing, 6 (1997), pp. 298–311.

[9] J. Chen, C. Richard, J.-C. M. Bermudez, and P. Honeine, *Nonnegative least-mean-square algorithm*, IEEE Transactions on Signal Processing, 59 (2011), pp. 5225–5235.

[10] J. Chen, C. Richard, and P. Honeine, *Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model*, IEEE Transactions on Signal Processing, 61 (2013), pp. 480–492.

[11] J. Chen, C. Richard, Y. Song, and D. Brie, *Transient performance analysis of zero-attracting lms*, IEEE Signal Processing Letters, 23 (2016), pp. 1786–1790.

[12] Y. Chen, Y. Gu, and A. O. Hero, *Regularized least-mean-square algorithms*, arXiv preprint arXiv:1012.5066, (2010).

[13] Y. R. Chen, R. W. Huffman, B. Park, and M. Nguyen, *Transportable spectrophotometer system for on-line classification of poultry carcasses*, Applied Spectroscopy, 50 (1996), pp. 910–916.

[14] E. Chouzenoux and J.-C. Pesquet, *A stochastic majorize-minimize subspace algorithm for online penalized least squares estimation*, IEEE Transactions on Signal Processing, 65 (2017), pp. 4770 –4783.

[15] P. L. Combettes and J.-C. Pesquet, *Stochastic forward-backward and primal-dual approximation algorithms with application to online image restoration*, in 24th European Signal Processing Conference (EUSIPCO), 2016, pp. 1813–1817.

[16] G. H. Costa and J.-C. M. Bermudez, *Statistical analysis of the LMS algorithm applied to super-resolution image reconstruction*, IEEE Transactions on Signal Processing, 55 (2007), pp. 2084–2095.

[17] P. Di Lorenzo and A. H. Sayed, *Sparse distributed learning based on diffusion adaptation*, IEEE Transactions on signal processing, 61 (2013), pp. 1419–1433.

[18] E. M. Eksioglu, *Sparsity regularised recursive least squares adaptive filtering*, IET signal processing, 5 (2011), pp. 480–487.

[19] M. Elad and A. Feuer, *Superresolution restoration of an image sequence: adaptive filtering approach*, IEEE Transactions on Image Processing, 8 (1999), pp. 387–395.

[20] F. W. Esmonde-White, K. A. Esmonde-White, and M. D. Morris, *Minor distortions with major consequences: correcting distortions in imaging spectrographs*, Applied spectroscopy, 65 (2011), pp. 85–98.

[21] N. P. Galatsanos and R. T. Chin, *Restoration of color images by multichannel Kalman filtering*, IEEE Transactions on signal processing, 39 (1991), pp. 2237–2252.

[22] N. P. Galatsanos, A. K. Katsaggelos, R. T. Chin, and A. D. Hillery, *Digital restoration of multichannel images*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 37 (1989), pp. 415–421.
[23] N. P. Galatsanos, A. K. Katsaggelos, R. T. Chin, and A. D. Hillery, *Least squares restoration of multichannel images*, IEEE Transactions on Signal Processing, 39 (1991), pp. 2222–2236.
[24] W. Gao, J. Chen, C. Richard, and J. Huang, *Online dictionary learning for kernel LMS*, IEEE Transactions on Signal Processing, 62 (2014), pp. 2765–2777.
[25] J. F. Giovannelli and A. Coulais, *Positive deconvolution for superimposed extended source and point sources*, Astronomy & Astrophysics, 439 (2005), pp. 401–412.
[26] J. Goodman, *Introduction to Fourier optics*, McGraw-hill, 2008.
[27] S. Henrot, C. Soussen, and D. Brie, *Fast positive deconvolution of hyperspectral images*, IEEE Transactions on Image Processing, 22 (2013), pp. 828–833.
[28] H. Huang, L. Liu, and M. O. Ngadi, *Recent developments in hyperspectral imaging for assessment of food quality and safety*, Sensors, 14 (2014), pp. 7248–7276.
[29] B. R. Hunt and O. Kubler, *Karhunen-Loeve multispectral image restoration, part I: Theory*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 32 (1984), pp. 592–600.
[30] J. Jemec, F. Pernuš, B. Likar, and M. Bürmen, *Push-broom hyperspectral image calibration and enhancement by 2D deconvolution with a variant response function estimate*, Optics Express, 22 (2014), pp. 27655–27668.
[31] J. Jemec, F. Pernuš, B. Likar, and M. Bürmen, *Deconvolution-based restoration of SWIR pushbroom imaging spectrometer images*, Optics Express, 24 (2016), pp. 24704–24718.
[32] D. Jin, J. Chen, C. Richard, and J. Chen, *Adaptive parameters adjustment for group reweighted zero-attracting*, in 43th International Conference on Acoustics, Speech, and Signal Processing, (ICASSP), 2018.
[33] Q. Li, X. He, Y. Wang, H. Liu, D. Xu, and F. Guo, *Review of spectral imaging technology in biomedical engineering: achievements and challenges*, Journal of biomedical optics, 18 (2013), pp. 100901–100901.
[34] S. Moussaoui, D. Brie, and A. Richard, *Regularization aspects in continuous-time model identification*, Automatica, 41 (2005), pp. 197–208.
[35] Pellenc Selective Technology, *Mistral product*. `http://www.pellencst.com/products`. Accessed: 2016-09-01.
[36] M. E. Schaepman, *Imaging spectrometers*, The SAGE handbook of remote sensing, (2009), pp. 166–178.
[37] K. Shi and P. Shi, *Convergence analysis of sparse lms algorithms with l1-norm penalty based on white input signal*, Signal Processing, 90 (2010), pp. 3289–3293.
[38] Y. Song, D. Brie, E.-H. Djermoune, and S. Henrot, *Regularization parameter estimation for non-negative hyperspectral image deconvolution*, IEEE Transactions on Image Processing, 25 (2016), pp. 5316–5330.
[39] Y. Song, E.-H. Djermoune, J. Chen, C. Richard, and D. Brie, *Online deconvolution for pushbroom hyperspectral imaging systems*, in 7th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, CAMSAP 2017, 2017.
[40] T. Suzuki, *Dual averaging and proximal gradient descent for online alternating direction multiplier method*, in International Conference on Machine Learning, 2013, pp. 392–400.
[41] D. Tao, G. Jia, Y. Yuan, and H. Zhao, *A digital sensor simulator of the pushbroom offner hyperspectral imaging spectrometer*, Sensors, 14 (2014), pp. 23822–23842.
[42] P. Tatzer, M. Wolf, and T. Panner, *Industrial application for inline material sorting using hyperspectral imaging in the NIR range*, Real-Time Imaging, 11 (2005), pp. 99–107.
[43] A. M. Tekalp and G. Pavlović, *Multichannel image modeling and kalman filtering for multispectral image restoration*, Signal Processing, 19 (1990), pp. 221–232.
[44] S. Tsuchikawa and H. Kobori, *A review of recent application of near infrared spectroscopy to wood science and technology*, Journal of Wood Science, 61 (2015), pp. 213–220.
[45] B. Widrow and S. D. Stearns, *Adaptive signal processing*, Englewood Cliffs, NJ, Prentice-Hall, Inc., 1985.
[46] R. M. Willett, M. F. Duarte, M. A. Davenport, and R. G. Baraniuk, *Sparsity and structure in hyperspectral imaging: sensing, reconstruction, and target detection*, IEEE Signal Processing Magazine, 31 (2014), pp. 116–126.
[47] J. Woods and V. Ingle, *Kalman filtering in two dimensions: Further results*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 29 (1981), pp. 188–197.
[48] J. Woods and C. Radewan, *Kalman filtering in two dimensions*, IEEE Transactions on Information Theory, 23 (1977), pp. 473–482.
[49] S. Zhang and J. Zhang, *Transient analysis of zero attracting NLMS algorithm without gaus-*

sian inputs assumption, Signal processing, 97 (2014), pp. 100–109.
[50] X. Zhao, F. Wang, T. Huang, M. K. Ng, and R. J. Plemmons, *Deblurring and sparse unmixing for hyperspectral images*, IEEE Transactions on Geoscience and Remote Sensing, 51 (2013), pp. 4045–4058.