Statistical Destriping of Pushbroom-Type Images Based on an Affine Detector Response

Mehdi Amrouche, Hervé Carfantan, Jérôme Idier, Vincent Martin

Statistical Destriping of Pushbroom-Type Images Based on an Affine Detector Response

Mehdi Amrouche, Hervé Carfantan, Jérôme Idier, *Member, IEEE*, and Vincent Martin

***Abstract*—Remote sensing pushbroom-type imaging systems acquire entire columns of an image with a single detector. As a consequence, the miss-calibration of the detectors produces stripes on the image. In this context, this paper introduces a new self-calibration destriping method based on an affine response model for the detectors, called Statistical Affine Destriping (SAD). In contrast, some previous contributions were limited to a purely linear model, while many others only considered an additive structured noise model. It is based on the maximum a posteriori estimation of the gain and offset parameters attached to each detector given the observed image. Simple statistical prior assumptions are adopted: respectively, a Gaussian white noise model for the gains and offsets, and a first-order, homogeneous Markov model for the observed scene. Based on a simplification of the posterior likelihood, we propose a very efficient optimization scheme based on a constrained Majorize-Minimize principle, allowing us to process large dimension images. Moreover, simple empirical rules are given to tune the hyperparameters of the destriping method for high-resolution PLEIADES-type images. Compared to the performance of a destriping method limited to gain correction, we observe that the new version provides reliable results in a wider range of situations. We also extend the method in two directions. On the one hand, we consider that some detectors may be atypical, with very high or very low gains or offsets. On the other hand, we extend the method to multispectral image destriping.**

***Index Terms*—Image destriping; Affine detector model; Statistical self-calibration; Constrained Majorize-Minimize algorithm.**

I. Introduction

In remote sensing imaging, pushbroom-type images suffer from a well-known phenomenon, called striping effect, which is caused by a difference in the response of the sensors. Indeed, pushbroom-type imaging instruments acquire images line by line with a linear array of sensors placed across-track. The other dimension is obtained thanks to the motion of the satellite along its orbit. Therefore, differences in the response of the sensors produce some stripes along the column of the image. During the prelaunch calibration of the satellite, the response of the detectors are identified and partially corrected. Nevertheless, the evolution of the detectors over time may require on-flight corrections. Correcting for the striping effect is crucial, as stripes may perturb the analysis of the images, by introducing errors in the outcome of post-processing methods, for the purpose of geometric structure detection, image segmentation, or image cross correlation, for instance.

Many destriping methods have been proposed in the literature. [1]. We invite the readers to refer to [1] for a recent and thorough review of destriping methods and of their properties. According to our own classification, most destriping methods adopt a methodology that falls into one of the three categories.
(i) Stripe filtering methods are based on filtering to get rid of the stripes in a filtering domain (Fourier, Wavelets...) [2]–[7]. Such methods are particularly efficient for whiskbroom acquisition systems, where the stripes are generally periodic as each sensor acquire several rows across-track. Nonetheless, the filtering domain also contains information about the observed scene, which are lost in the filtering process, leading to the possible generation of artefacts such as blurring or ringing effects.
(ii) Image denoising: A lot of methods simply consider the stripe effect as additive noise. Most of these methods are based on an optimization principle where a cost function is built using prior information on the stripe noise and on the observed scene in a decomposition principle. Various kinds of prior have been taken into account for the stripe noise and for the scene, such as sparsity (with ℓ_0 [8] or ℓ_1 [8] norms or mixed norms [9]), low-rank assumption [10]–[12], total variation [11], [13], [14] or other variational properties [8], [12], [15]. More recently, models of images [16] or stripes [1], [17] have been learned from data using deep convolutional neural networks trained on a large database of small simulated images. Finally, some of these methods have been extended to denoise multispectral or hyperspectral images, accounting for the inter-band correlation of the images with a 3D prior such as a low-rank tensor assumption [18], 3D-TV [11] or group sparsity [18]. Finally, note that some recent image denoising methods [1], [15] aim at proposing a universal destriping tool, whatever the kind of stripes in the image (vertical, oblique, local, global, periodic, banding, ...).
(iii) Statistical calibration methods exploit statistical properties of the image. Typically, some statistics of the observed scene are assumed invariant along the columns, and an estimation of the detector responses is built on this statistical basis [19]–[24]. The histogram matching techniques [20] assume that the inputs of all sensors share the same probability density functions. This assumption allows one to estimate a shape of response function for the detectors. The moment matching [22] techniques only make the hypothesis that the inputs of all sensors share the same mean and standard deviation. Based on these assumptions, an affine response function of the detectors can be estimated. Unfortunately, such statistical assumptions are quite strong. As a consequence, moment

M. Amrouche and H. Carfantan are with the Institut de Recherche en Astrophysique et Planétologie, Université de Toulouse, CNRS/UPS/CNES, Toulouse, France. (e-mail: firstname.name@irap.omp.eu)

J. Idier is with the CNRS at the Laboratoire des Sciences du Numérique de Nantes (LS2N, CNRS UMR 6004), Nantes, France. (e-mail: jerome.idier@ls2n.fr).

V. Martin is with French National Space Agency (CNES), Toulouse, France. (e-mail: Vincent.Martin@cnes.fr).

matching methods lack robustness, in the sense that they are sensitive to variations in the observed image. More recently, [25] proposed a Statistical Linear Destriping (SLD) method to estimate either the offset or the gain of the sensors, modeling the pixel intensities or their logarithm, respectively, with a homogeneous Markov Model, which is a weaker assumption compared to the previous ones. According to the comparative study proposed in [1], SLD is one of the very best existing methods of all categories. Moreover, the only methods with (slightly) better performance (such as the one proposed in [1]) are dependent on a training phase involving labelled data, in contrast with SLD.

During the commissioning phase of the high-resolution imaging satellites PLEIADES-HR, it has been pointed out that an affine model of the detector responses is relevant to describe the striping effect [26]. Unfortunately, the only existing destriping method handling an affine model is the moment matching technique, which lacks robustness. This paper originates from the need for a reliable destriping method in the affine response case. Surprisingly, while the stripes in the image clearly originate from a miscalibration of the detector responses, only the statistical calibration methods explicitly rely on a physical model for the detector responses. For this reason, our new contribution to destriping in the affine response case belongs to the statistical calibration category. More precisely, given that SLD is a very efficient destriping method based on a linear gain model, we have designed our new destriping method as an extension of SLD to the affine case. The proposed method will be referred to as *Statistical Affine Destriping* (SAD).

The detector response being considered as an affine function, the relation between the perfect scene $\boldsymbol{z} = \{z_{r,c}\}_\Gamma$ and the observed data $\boldsymbol{w} = \{w_{r,c}\}_\Gamma$ depends on two kinds of calibration parameters: gains $\boldsymbol{g} = \{g_c\}_{\Gamma_C}$ and offsets $\boldsymbol{o} = \{o_c\}_{\Gamma_C}$ and it can be written as:

$$z_{r,c} = g_c w_{r,c} - o_c \tag{1}$$

where $\Gamma = \Gamma_R \times \Gamma_C$ is a grid of R rows $\Gamma_R = \{1, ..., R\}$ and C columns $\Gamma_C = \{1, ..., C\}$. In [25], it was assumed that the offsets $\boldsymbol{o}$ are known, so the simpler model $z_{r,c} = g_c w_{r,c}$ was considered, where the offsets are already corrected. Furthermore, the estimation of gains $\boldsymbol{g}$ was tackled by considering the equivalent logarithmic relation$\log z_{r,c} = \log g_c + \log w_{r,c}$. Such a simplification is no more possible when the offsets are unknown. Here, a new penalized fidelity-to-data criterion is proposed, and a fast optimization scheme is deduced from a simplification of the cost function without altering the performance of calibration. The proposed method is compatible with the calibration of large dimension/high-resolution images.

Throughout this paper, we will illustrate the results of the proposed method on PLEIADES-type high resolution images, but the proposed method can be used on images from any pushbroom instrument as long as the sensor responses can be considered affine.

The organization of this paper is as follows. In Section II, the proposed estimator is presented in a statistical framework. Section III is devoted to computational issues: a simplification of the cost function allows us to derive an efficient constrained Majorize-Minimize algorithm. The resulting numerical scheme is tested on both simulated and real images, and an empirical method is proposed to tune the hyper-parameters. In Section IV, we propose a useful adaptation of the method to the frequent cases where a small number of detectors have been identified as behaving in an atypical way. Finally, we extend the method to multispectral images in Section V. We show that the correlation between the different spectral bands can be taken into account in order to improve the calibration, with no additional computational cost. Section VI concludes the paper.

II. General Bayesian framework

Our objective is to estimate the C pairs of gains $\boldsymbol{g}$ and offsets $\boldsymbol{o}$ given the $R \times C$ pixels of the observed image $\boldsymbol{w}$, where the relation between the perfect scene $\boldsymbol{z}$, the observed image $\boldsymbol{w}$ and the calibration parameters $(\boldsymbol{g}, \boldsymbol{o})$ is given by (1).

In the Bayesian framework, taking statistical prior models on the calibration parameters and on the perfect scene $\boldsymbol{z}$ into account, one can define the Maximum a Posteriori (MAP) estimator of the calibration parameters that maximizes the posterior neg-log-likelihood function $J(\boldsymbol{g}, \boldsymbol{o})$. This is the goal of the present section.

A. A priori image model

Akin to [25], we adopt a basic first-order Markov field model to model the spatial correlation between pixels of the perfect scene. Its probability density function is given as:

$$f_Z(\{z_{r,c}\}_\Gamma) \propto \exp\left\{ -\frac{1}{T} \sum_{(r,c)\sim(r',c')} \phi_s(z_{r,c} - z_{r',c'}) \right\}, \tag{2}$$

where $(r,c) \sim (r',c')$ means summation over all distinct pairs of horizontal and vertical neighboring pixels. The potential function ϕ_s is an edge preserving function, which models the strong correlation between neighboring pixels except at the interface between distinct regions [27]. We have chosen to focus our study on two functions that behave quadratically near zero, while they are asymptotically linear or constant, respectively:

$$\phi_s^{2-1}(x) = \sqrt{x^2 + s^2} - s$$
$$\phi_s^{2-0}(x) = \frac{x^2}{x^2 + s^2}$$

From a modeling viewpoint, ϕ_s^{2-0} has optimal edge preserving characteristics thanks to its constant asymptotic behavior. Therefore, on either side of an interface between two regions, the difference between the pixel values can be arbitrarily large. However, ϕ_s^{2-0} is not a convex function. As a consequence, the corresponding neg-log-likelihood function may be multimodal, and local descent algorithms may converge in local mimina. On the other hand, ϕ_s^{2-1} is convex, but its edge preserving capacity is weaker than that of ϕ_s^{2-0}. Nonetheless, ϕ_s^{2-1} is better suited than a simple quadratic function, since the latter has no edge preserving property at all.

Finally, T and s are hyper-parameters of the Markov model related to the real scene $\boldsymbol{z}$. Parameter s is a threshold between

the quadratic behavior of the potential functions near zero, and the asymptotic one, whereas T is a scale parameter. Hyper-parameter tuning is discussed in Section III-D.

B. A priori model for the parameters

In an ideal case (*i.e.*, image acquisition with no striping effect), the observed image $\boldsymbol{w}$ should be equal to the real scene $\boldsymbol{z}$, which means that:

$$(g_c, o_c) = (1, 0), \quad \forall\, c \in \Gamma_C. \tag{3}$$

Considering an effective absolute radiometric calibration of the instrument, we can assume that the global radiometry of the image is preserved. Therefore, as $\mathrm{E}[z_{r,c}] = \mathrm{E}[g_c]\,\mathrm{E}[w_{r,c}] + \mathrm{E}[o_c]$, it seems natural to consider that the gains have a unitary mean, and the offsets are zero-mean:

$$\frac{1}{C}\sum_{c=1}^{C} g_c = 1, \quad \frac{1}{C}\sum_{c=1}^{C} o_c = 0. \tag{4}$$

In addition, let us assume that the parameters $\boldsymbol{g}$ and $\boldsymbol{o}$ are mutually independent and identically distributed according to Gaussian prior distributions:

$$\forall c, g_c \sim \mathcal{N}(1, \sigma_g^2), \quad o_c \sim \mathcal{N}(0, \sigma_o^2) \tag{5}$$

where σ_g^2 and σ_o^2 are the variances of g_c and o_c, respectively.

In the rest of the paper, we will assume that both standard deviations σ_g and σ_o are known. In practice, it is reasonable to consider that at least their order of magnitude can be deduced from informations resulting from the prelaunch phase or from previous on-flight calibrations.

C. MAP estimator

Using the probabilistic models presented above, and according to the Bayes rule, one can calculate the posterior distribution of the parameters given the data:

$$f(\boldsymbol{g}, \boldsymbol{o}|\boldsymbol{w}) = f(\boldsymbol{w}|\boldsymbol{g}, \boldsymbol{o}) f(\boldsymbol{g}) f(\boldsymbol{o}) / f(\boldsymbol{w}), \tag{6}$$

where the likelihood term can be deduced from (2), with a simple change of variable:

$$f(\boldsymbol{w}|\boldsymbol{g}, \boldsymbol{o}) \propto \prod_{r=1}^{R}\prod_{c=1}^{C} |g_c|\, f_Z(g_c w_{r,c} - o_c). \tag{7}$$

The proposed MAP estimator minimizes the anti-log posterior density (under the constraints (4)), which is equal to (up to an additional constant):

$$J(\boldsymbol{g}, \boldsymbol{o}) = Q(\boldsymbol{g}, \boldsymbol{o}) + J_0(\boldsymbol{g}, \boldsymbol{o}) + J_1(\boldsymbol{g}) \tag{8}$$

where

$$Q(\boldsymbol{g}, \boldsymbol{o}) = \lambda_g \sum_{c=1}^{C} (g_c - 1)^2 + \lambda_o \sum_{c=1}^{C} o_c^2$$

with $\lambda_g = \frac{1}{2\sigma_g^2}$ and $\lambda_o = \frac{1}{2\sigma_o^2}$, and

$$J_0(\boldsymbol{g}, \boldsymbol{o}) = \frac{1}{T}\sum_{r=1}^{R}\sum_{c=1}^{C-1} \phi_s(g_c w_{r,c} - g_{c+1} w_{r,c+1} - o_c + o_{c+1}),$$

$$J_1(\boldsymbol{g}) = \frac{1}{T}\sum_{r=1}^{R-1}\sum_{c=1}^{C} \phi_s(g_c(w_{r,c} - w_{r+1,c})) - R\sum_{c=1}^{C} \log|g_c|.$$

Moreover, it is easy to show that the solution to this optimization problem always satisfies $\sum_c o_c = 0$. Thus, regarding the constraints (4), only the constraint on the gains $\sum_c g_c = C$ is relevant in the optimization process.

III. Optimization

It is possible to recast the linearly constrained problem as an unconstrained one [28, Chap. 11]. As a consequence, a gradient based method such as the preconditioned conjugate gradient (PCG) algorithm can be used to minimize the cost function (8) under the constraint $\sum_c g_c = C$. However, a specificity of (8) is the presence of a logarithmic barrier, which makes it difficult to design an efficient preconditioning scheme. In practice, we selected a diagonal preconditioner, among other choices. However, the resulting algorithm has a slow convergence compared to the calibration method proposed in [25], by at least one order of magnitude (see the numerical validation in § III-C).

As an alternative, the next two subsections introduce a simplified version of the cost function where the barrier term is removed, and an associated algorithm that is substantially faster.

A. Cost function simplification

Several tests on a PLEIADES-type simulated image database have shown that the contribution of term J_1 (and more specifically, on the barrier term) has a negligible influence on the constrained minimizer of J (see the numerical validation in § III-C). Such an empirical observation can be explained by two elements:

- The gain parameter is assumed constant along each column, so the first term of J_1 has small variations, and brings a limited amount of information on the gain parameter itself.
- As far as the gain parameters have limited variations around 1, the logarithmic shape of the second term of J_1 can be ignored.

As a consequence, let us define a simplified minimization problem as:

$$\min_{\boldsymbol{g},\boldsymbol{o}} K(\boldsymbol{g}, \boldsymbol{o}) \quad \text{s.t.} \sum_c g_c = C \tag{9}$$

where

$$K(\boldsymbol{g}, \boldsymbol{o}) = Q(\boldsymbol{g}, \boldsymbol{o}) + J_0(\boldsymbol{g}, \boldsymbol{o}). \tag{10}$$

Simpler algorithms can now be considered to solve problem (9). A natural choice is to rely on an Iterative Re-weighted Least Squares (IRLS) scheme. Compared with other descent algorithms, IRLS algorithms are easy to implement since only a linear system is solved at each iteration, and no step-size tuning is required. Furthermore, as shown below, the linear constraint can be taken into account in a simple and direct way.

B. A constrained IRLS algorithm

IRLS is a *Majorize-Minimize* (MM) algorithm, *i.e.*, an iterative minimization scheme where at each iteration, the minimization of the original objective function is replaced by that of a majorizing approximation [29]. MM iterations ensure that the objective function is always descending towards a (possibly local) minimum. Within the wide family of MM algorithms, IRLS have the specificity to rely on a quadratic majorizer.

Let us denote $\boldsymbol{x} = [\boldsymbol{g}^{\mathrm{t}}, \boldsymbol{o}^{\mathrm{t}}]^{\mathrm{t}}$, so that the objective function (10) can be written in a more compact form:

$$K(\boldsymbol{x}) = Q(\boldsymbol{x}) + \frac{1}{T}\sum_{r,c} \phi_s([\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(k)}]_c), \tag{11}$$

where

$$\begin{aligned} Q(\boldsymbol{x}) &= \boldsymbol{x}^{\mathrm{t}}\mathbf{Q}\boldsymbol{x} - 2\lambda_g \boldsymbol{e}^{\mathrm{t}}\boldsymbol{x}, \\ \mathbf{Q} &= \mathrm{diag}\left\{\lambda_g, \ldots, \lambda_g, \lambda_o, \ldots, \lambda_o\right\}, \\ \mathbf{V}_r &= \begin{bmatrix}\mathbf{V}_r^1 \\ \mathbf{V}_r^2\end{bmatrix}, \quad \boldsymbol{e} = \begin{bmatrix}\mathbb{1}_{[C\times 1]} \\ \mathbb{0}_{[C\times 1]}\end{bmatrix}, \end{aligned} \tag{12}$$

$\mathbf{V}_r^1$ and $\mathbf{V}_r^2$ being $C \times C-1$ matrices[1]:

$$\begin{aligned} \mathbf{V}_r^1 &= \mathrm{diag}_0\{w_{r,1}, \ldots, w_{r,C-1}\} - \mathrm{diag}_{-1}\{w_{r,2}, \ldots, w_{r,C}\} \\ \mathbf{V}_r^2 &= -\mathrm{diag}_0\{1, \ldots, 1\} + \mathrm{diag}_{-1}\{1, \ldots, 1\}. \end{aligned}$$

Moreover, the gain normalization constraint reads $\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x} = C$.

In Appendix A, we derive the following constrained IRLS algorithm to solve the minimization problem (9). The algorithm alternately updates auxiliary variables $t_{r,c}^{(k)}$, and solves a linear system to get the next iterate $\boldsymbol{x}^{(k+1)}$:

$$t_{r,c}^{(k)} = \frac{\phi_s'([\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(k)}]_c)}{2[\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(k)}]_c}, \tag{13}$$

$$\boldsymbol{x}^{(k+1)} = \frac{C(\mathbf{B}^{(k)})^{-1}\boldsymbol{e}}{\boldsymbol{e}^{\mathrm{t}}(\mathbf{B}^{(k)})^{-1}\boldsymbol{e}}. \tag{14}$$

with

$$\mathbf{B}^{(k)} = \mathbf{Q} + \frac{1}{T}\sum_r \mathbf{V}_r\mathbf{L}_r^{(k)}\mathbf{V}_r^{\mathrm{t}}, \tag{15}$$

$$\mathbf{L}_r^{(k)} = \mathrm{diag}\left\{t_{r,c}^{(k)}\right\}_{c\in\Gamma_{C-1}}. \tag{16}$$

Furthermore, matrices $\mathbf{B}^{(k)}$ are block-tridiagonal, which makes the linear system easier to solve. For more details on the construction of matrices $\mathbf{B}^{(k)}$, the reader may refer to Appendix B. The proposed SAD method identifies with the constrained IRLS algorithm (13)-(14).

Let us mention that (9) is a convex programming problem if function ϕ_s is convex. In such a case, the proposed constrained IRLS is ensured to converge to the global solution. As stated before, function ϕ_s^{2-1} is convex.

In contrast, function ϕ_s^{2-0} allows to model the high dynamic variations more accurately, but at the price of loosing convexity. As a consequence, local solutions may exist, and then the constrained IRLS (as any other local descent algorithm) may be trapped in one of them. Nevertheless, we have empirically observed that the algorithm always converges to the same solution if initialized near the point $(g_c, o_c) = (1, 0), \forall c \in \Gamma_C$.

[1]For any positive integer n, diag_n (respectively, diag_{-n}) defines a matrix by its nth diagonal above (resp., below) the main diagonal, all other entries being zero. In the diagonal case, $\mathrm{diag}_0 = \mathrm{diag}$.

C. Numerical validation

Here, we illustrate the behavior of the proposed SAD method with an experiment using two large-size PLEIADES-type test images, corresponding to two different types of observed scenes (of size 4000×2000). In particular, test image (A) has more intensity variations and texture compared to test image (B), which has mainly smooth and uniform regions. Examples are given in Fig. 1 under the form of two snapshots of dimensions reduced to 200×200, for the sake of display quality. Then, from a generated set of gains and offset we have artificially simulated the striping effect according to model (1).

The purpose of this experiment is twofold. On the one hand, is to assess the effect of the cost function simplification on the quality of the calibration of the image and the computational time (CT). On the other hand, the proposed SAD method is compared to the minimization of (8) by PCG, and to the SLD method presented in [25]. In order to quantify the quality of the calibrated images we rely on several indexes including the *Structural Similarity Index* (SSIM) [30], which measures the degradation of image structures, and the *Peak Signal to Noise Ratio* (PSNR):

$$\mathrm{PSNR}_{\mathrm{dB}} = 20\log_{10}\left(\frac{I_{\max}}{\mathrm{RMSE}}\right),$$

which is related to the ratio between the maximum intensity of the image and the root mean squared error (RMSE) between the original image and the calibrated one. For both potential functions ϕ_s^{2-0} and ϕ_s^{2-1}, Fig. 2 gives the PSNR of the calibrated image and the computational cost of PCG and of SAD, compared to the performance of the gain correction with SLD, for several values of σ_o. Note that plotting the SSIM instead of the PSNR would lead to the very same conclusions. Such an additional figure is omitted here, for the sake of brevity. For PCG and SAD, hyper-parameters were set in an unsupervised way (see § III-D for a discussion on unsupervised estimation), while for the gain correction with SLD, we selected the optimal hyper-parameters using a grid-search method. Indeed, the selection of empirical values discussed in [25] for SLD does not apply to PLEIADES-type images.

For function ϕ_s^{2-1}, the calibration quality is almost identical for PCG and SLD (with a deviation of at most 0.04 dB in terms of PSNR), regardless of the value of σ_o. The same observation holds for ϕ_s^{2-0} when the variance of the offsets is small ($\sigma_o \leq 1$). However, for large values of σ_o, minimizing J with PCG becomes troublesome and provides poor calibration results, as it tends to get stuck in local minima. On the contrary, SAD remains robust in all cases. Moreover, SAD is approximately four times faster than PCG in the tested cases, while being approximately two to three times slower than SLD.

Additionally, for large values of σ_o, affine correction using SAD provides better results than gain correction using SLD. Obviously, this result is expected since the latter does not

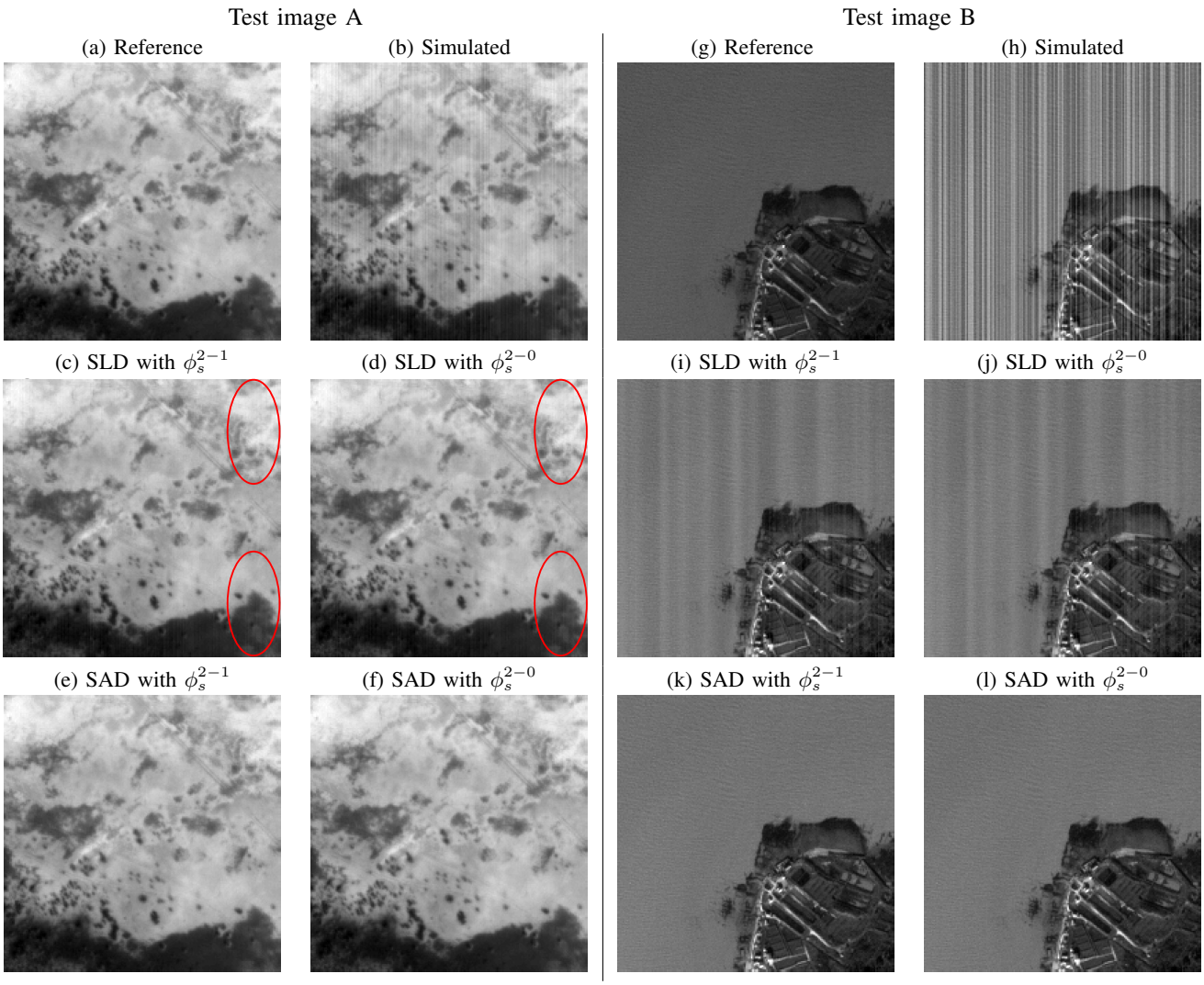

Fig. 1: Calibration results on PLEIADES-type simulated images of size 4000×2000. 200×200 snapshots of: (a), (g) the reference images, (b), (h) simulated images with the striping effect (with $\sigma_g = 2 \times 10^{-3}$ and $\sigma_o = 29$), (c), (i) and (d), (j): calibrated images using SLD for ϕ_s^{2-1} and ϕ_s^{2-0}, respectively, (e), (k) and (f), (l): same with SAD.

account for offsets. However, for $\sigma_o \ll 1$ (*i.e.*, when the offsets can be considered negligible), the new SAD method still provides as reliable calibrations as SLD [25], which stresses the efficiency of the proposed hyper-parameter proposed below.

Fig. 1 provides a visual comparison of the calibration results between SLD and SAD (second and third rows, respectively) for both test images A and B and potential functions ϕ_s^{2-1} and ϕ_s^{2-0}. SLD clearly fails to remove the stripes on test image B, whatever the potential function. The correction with SLD is more effective on test image A, but residual stripes can still be observed when zooming in on the circled areas. On the other hand, SAD provides very satisfactory results since the striping effect is no more visible, regardless of the potential function.

D. Hyper-parameter tuning

In order to achieve unsupervised calibration of the image, the hyper-parameters T and s of the SAD method need to be tuned. In a simulated context, where one has access to the ground truth, it is possible to acquire some empirical knowledge about the optimal parameter values given some real data $\boldsymbol{w}$. The training strategy adopted here consisted of making use of a database of 60 PLEIADES-type images to perform an empirical analysis of the optimal hyper-parameters $\hat{T}$ and $\hat{s}$ in the sense of the reached PSNR, over a predefined grid of (T, s) values. From the obtained empirical results, we could deduce the following empirical rules:

- For ϕ_s^{2-1}, set $s^2 = 0.1$, which is a very low value, relatively to the amplitude of the images coded in $[0, 4095]$, which amounts to select a version of ϕ_s^{2-1} that is closed to $\phi^1(x) = |x|$. On the other hand, set $T = (c_{\delta w} s)^{-1}$, where $c_{\delta w}$ is the curvature at the origin of the log-histogram of the column-wise gradient image:

$$\boldsymbol{\delta w} = \{\delta w_{r,c} = w_{r,c} - w_{r,c+1}, (r,c) \in \Gamma_R \times \Gamma_{C-1}\}.$$

- For ϕ_s^{2-0}, set $s^2 = \sigma_{\delta w}$ and

$$T = \log\left(\frac{2}{c_{\delta w}\sigma_{\delta w}}\right),$$

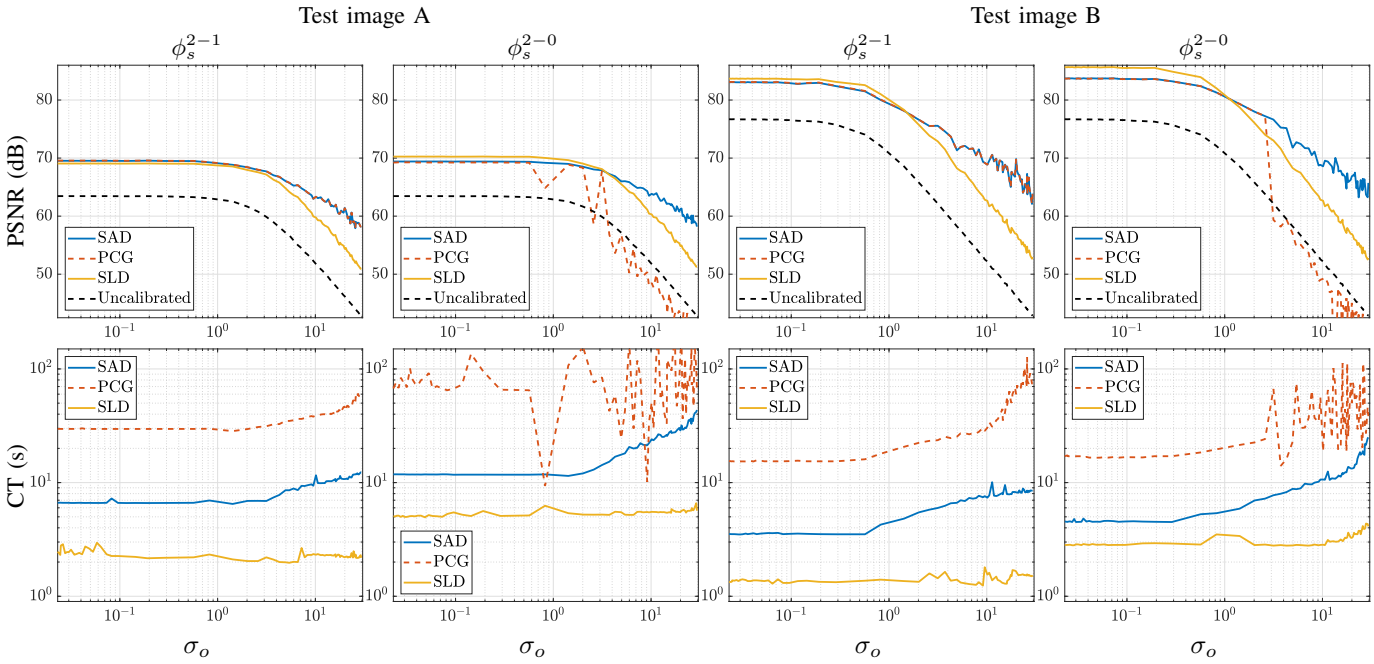

Fig. 2: Calibration results on PLEIADES-type simulated images of size 4000×2000. First row: PSNR of the calibrated image w.r.t. σ_o. Second row: time in seconds for each algorithm to converge.

where $\sigma_{\delta w}$ is the empirical standard deviation of $\boldsymbol{\delta w}$.

Fig. 3 summarizes the results of the experiment for both potential functions. For each image, the PSNR evaluated after an unsupervised calibration (using the empirical rules above) is plotted against the PSNR evaluated after a supervised calibration (best result obtained using grid-search). For ϕ_s^{2-0}, all the points lie around the diagonal line, meaning that the calibration results are almost identical with a degradation of approximately 1 dB, at most. The same observations predominantly hold for ϕ_s^{2-1}, with a few exceptions for which the mismatch is larger. Note that in some cases, supervision provided poorer results, which is due to the fact a relatively coarse grid of (T, s) values was used.

E. Noise robustness analysis

In practice, in addition to the striping effect, the observed images can be affected by noise. Hereafter, we provide a robustness analysis of SAD, assuming that a standard denoising procedure can be achieved posterior to calibration.

As long as the noise is considered additive, we do not expect it to strongly degrade the precision of statistical calibration. In particular, for high resolution images with a large number of rows, one can still get robust estimation of the calibration parameter from averaged data over subsets of rows, as the affine model (1) remains valid:

$$\frac{1}{L}\sum_{\ell=0}^{L-1} z_{r+\ell,c} = g_c \frac{1}{L}\sum_{\ell=0}^{L-1} w_{r+\ell,c} - o_c \qquad (17)$$

Our objective here is to evaluate the effect of noise on the estimation of gain and offset calibration parameters, without considering any denoising process, apart from the averaging step (17). Thus, we propose to compare the results after a calibration using parameters estimated from noisy images with an optimal calibration of the noisy images using ground-truth parameter values. We conducted an experiment over a database of 30 PLEIADES-type simulated images, with additive, zero-mean, independent noise distributed according to a Gaussian distribution:

$$n_{r,c} \sim \mathcal{N}(0, a w_{r,c} + b),$$

where $a = 0.007$ and $b = 0.2$, are instrument dependent values, that were estimated during the prelaunch calibration phase. Fig. 4 summarizes the comparison in terms of PSNR values. When calibration parameters are deduced from the noisy images, we have empirically set $L = 20$. We notice that almost all the points lie slightly underneath the diagonal line, with a deviation around 1.5 dB at most, which shows the robustness of the calibration in a noisy setting.

F. Real data experiment

In a complementary way to the previous numerical validations, we have performed a calibration experiment on a real data set. For this purpose, we have used a 3000×3000 SPOT4 (*Satellite Pour l'Observation de la Terre*) image. Fig. 5 provides a 200×200 snapshot of a real image along with the calibration results provided by SAD and SLD. It is worth mentioning that for SAD, hyper-parameters (T, s) were set in an unsupervised way using the proposed empirical rules in § III-D. Whereas for SLD, the parameters were chosen based on the empirical discussion provided in [25, §VI-A]. Furthermore, parameters σ_g and σ_o were set empirically by an expert. The ground truth is not know, but the improvement given by SAD over SLD is obvious for ϕ_s^{2-1}. Such an improvement can also be observed for ϕ_s^{2-0} by zooming on the circled area as it appears that there are still stripes for SLD.

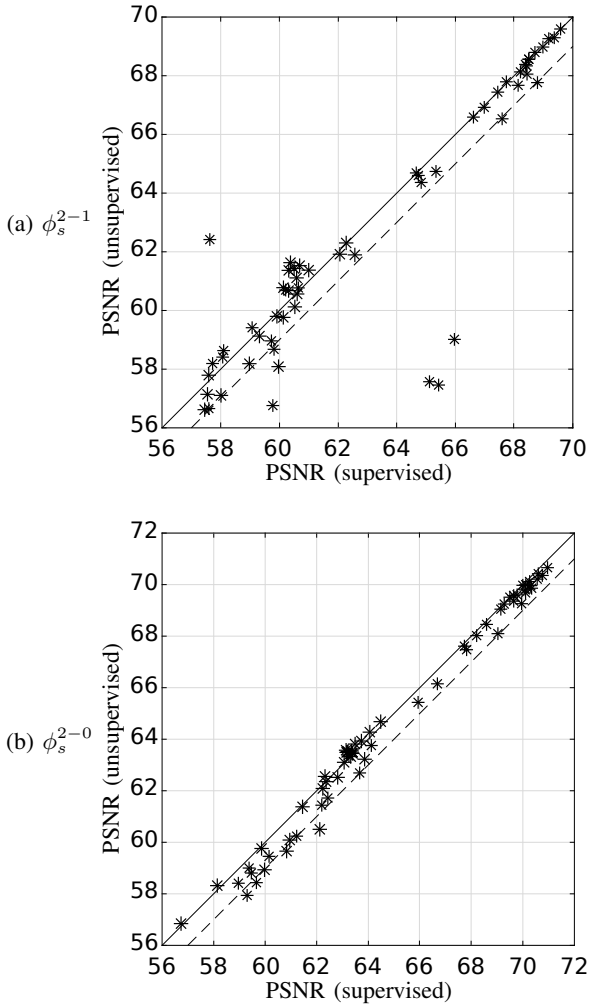

Fig. 3: Hyper-parameter tuning experiment for (a) ϕ^{2-1} and (b) ϕ^{2-0}. Comparison of the PSNR evaluated after unsupervised calibration using empirical rules (y-axis), with PSNR evaluated after supervised calibration using grid-search (x-axis). The dashed line represents a degradation of 1 dB.

Such results stress both the efficiency of the proposed method and the robustness of our hyper-parameter tuning rules.

IV. Accounting for atypical columns

In cases where the behavior of some detectors has become pathological, images must be calibrated despite the fact that they present some atypical columns, corresponding to detectors with gain and offset values much further away from the reference than the standard deviation. The proposed SAD method can be easily adapted to account for such atypical columns. Fig. 6(b) depicts a simulated example of an image incorporating atypical columns. Even if atypical columns represent a small proportion of the image, they can strongly perturb the whole calibration process, if not taken care of. Hereafter, we show that the proposed method allows us to handle such outliers in a simple way.

A first possibility could be to simply remove these columns. However, this is a suboptimal solution since it would distort the distance between pixels, with an uncontrolled impact on

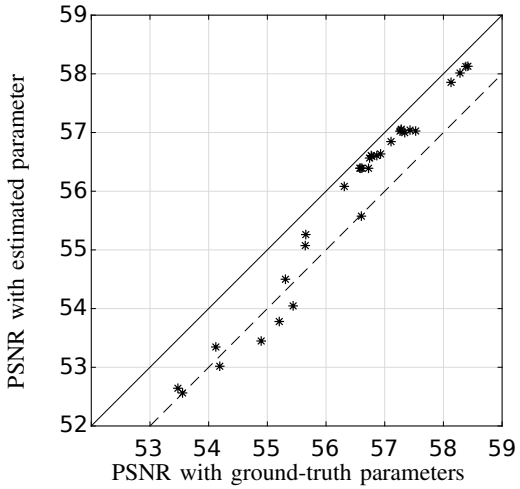

Fig. 4: PSNR obtained after calibrating the noisy images using the proposed method, compared to the PSNR obtained after perfect calibration. The dashed line represents a degradation of 1 dB.

the Markov probability density (2). In order to take into account that atypical columns have been produced by detectors for which the gain and the offset take arbitrarily values, we simply consider that the standard deviations of the calibration parameters are very large for the set E of atypical columns:

$$\forall c \in E,\ g_c \sim \mathcal{N}(1, \eta_g^2), \quad o_c \sim \mathcal{N}(0, \eta_o^2),$$

with $\eta_g \gg \sigma_g$ and $\eta_o \gg \sigma_o$, while the prior given by (5) remains valid for regular columns. Note that this only impacts the penalization term Q of the objective function, the atypical parameters being less regularized than the others. For the sake of simplicity, we will consider the limit case where $\eta_g^2 \to \infty$ and $\eta_o^2 \to \infty$, *i.e.*, atypical parameters will not be penalized at all. Finally, we consider that the gain normalization constraint must be restricted to the set of regular detectors.

It is straightforward to adapt the IRLS algorithm of Section III-B to the new situation. Indeed, Eqs (13)-(16) remain valid, provided that quantities $\mathbf{Q}$, $\boldsymbol{e}$, and C are replaced by $\widetilde{\mathbf{Q}} = \mathbf{U}\mathbf{Q}$, $\widetilde{\boldsymbol{e}} = \mathbf{U}\mathbf{e}$, and $\widetilde{C}$, respectively, with $\mathbf{U} = \mathrm{diag}\{\boldsymbol{u}\}$, $\boldsymbol{u} = (u_c)_{\Gamma_C}$,

$$u_c = \begin{cases} 0 & \text{if } c \in E, \\ 1 & \text{if } c \notin E, \end{cases}$$

$\widetilde{C}$ being the number of regular columns. Note that such modifications have a negligible impact on the computation cost. We will refer to the resulting version of SAD as SAD-AC.

Fig. 6 illustrates the calibration results for a test image simulated with two adjacent atypical columns. Let us stress that the striping effect affects the whole image, which is hardly visible because of the atypical columns. Regardless of the potential function, the simulation results of Table I show that SAD-AC substantially improves the performance of calibration both in terms of PSNR and SSIM. However, Table I also shows that SAD-AC requires more iterations than SAD. According to our tests, this is due to the under regularization of the

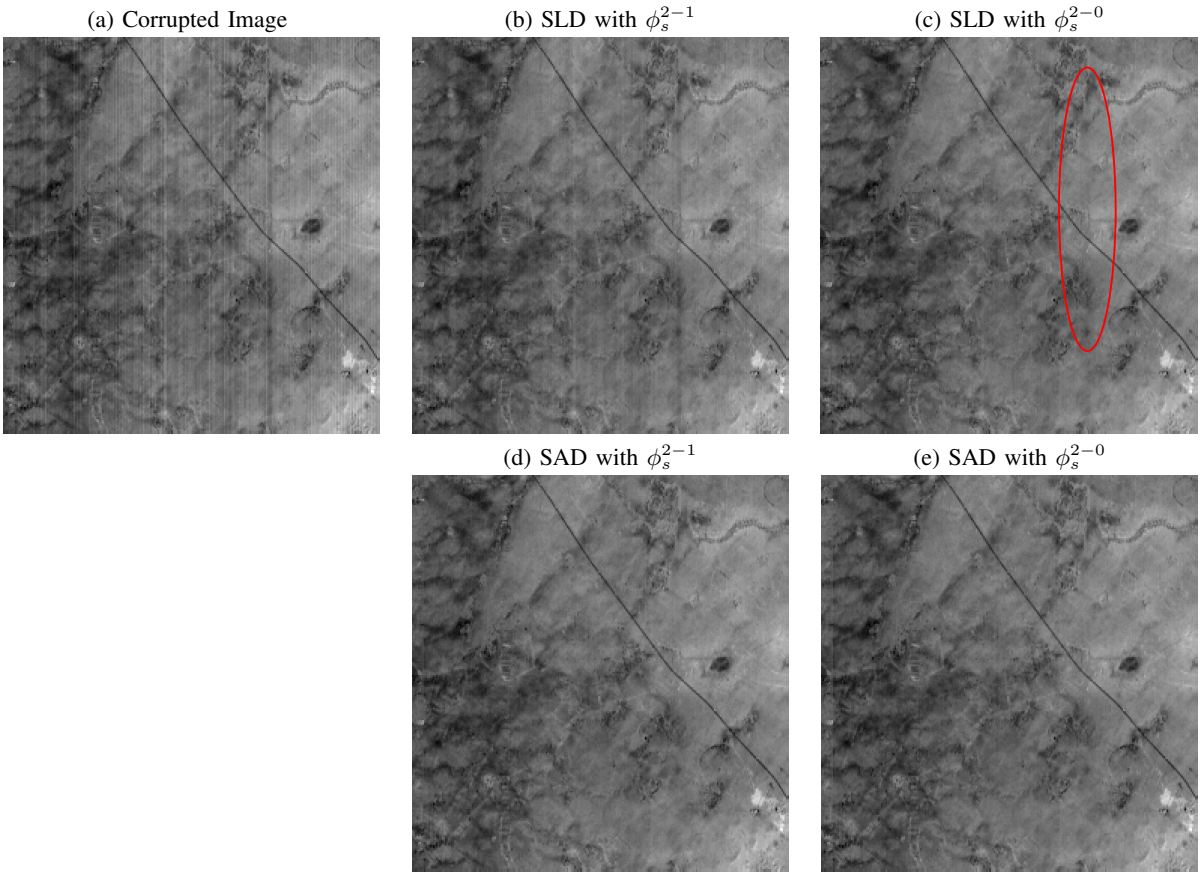

Fig. 5: Calibration results on a SPOT4 real image of size 3000×3000. 200×200 snapshots of: (a) SPOT4 image with vertical striping, (b) and (c): calibrated images using SLD for ϕ_s^{2-1} and ϕ_s^{2-0}, respectively, (d) and (e): same with SAD.

atypical calibration parameters. Indeed, after the 20th iteration, the remaining ones only apply to atypical columns and the neighboring ones. Therefore, it would be possible to restrict the additional updates to a reduced set of colums, and thus to strongly limit the extra computing time.

TABLE I: Calibration results for the simulated PLEIADES images of Fig. 6 with atypical columns using SAD, and the adapted method for atypical columns SAD-AC.

ϕ_s	Method	PSNR_{dB}	SSIM	N_{it}	CT (s)
ϕ_s^{2-1}	SAD	69.7054	0.996	12	6.9700
	SAD-AC	**82.2139**	**0.9994**	21	12.5500
ϕ_s^{2-0}	SAD	64.0101	0.9948	18	9.1200
	SAD-AC	**83.4399**	**0.9996**	82	41.7900

V. Multispectral image calibration

While dealing with a multispectral image of P spectral bands, a naive approach would be to consider a calibration of each band independently. Indeed, it is more efficient to calibrate these bands simultaneously, considering that they are observing the same scene, and to exploit the correlation between the bands. In the following, we show how the framework based on the simplified cost function (9) can be easily extended to the multispectral case, with almost no extra computing cost compared to a band-wise calibration.

A. Multispectral image model and prior

Considering a multispectral image with P bands $\boldsymbol{z} = \{\boldsymbol{z}_{r,c}\}_\Gamma$, $\boldsymbol{z}_{r,c} = (z^p_{r,c})_{p\in\Gamma_P}$, $\Gamma_P = \{1, \dots, P\}$, we propose to extend the Markov field model (2), assuming that the homogeneous areas of the image are homogeneous in each spectral band and that the edges share the same locations[2]. To this end, we propose to the following joint probability density:

$$f(\boldsymbol{z}) \propto \exp\left(-\frac{1}{T}\sum_{(r,c)\sim(r',c')} \phi_s(\|\boldsymbol{z}_{r,c} - \boldsymbol{z}_{r',c'}\|)\right)$$

where $\|\boldsymbol{\zeta}\| = \sqrt{(\zeta^1)^2 + (\zeta^2)^2 + \cdots + (\zeta^P)^2}$ is the Euclidean norm along the spectral dimension.

We keep the same priors for the parameters of each spectral band, and we assume that they are mutually independent from one band to the other:

$$g_c^p \sim \mathcal{N}\left(1, (\sigma_g^p)^2\right), \quad o_c^p \sim \mathcal{N}\left(0, (\sigma_o^p)^2\right),$$

[2]Of course, if a geometric registration step is necessary, it should be done in a way preserving the striping effect. In particular, linear interpolation must be avoided.

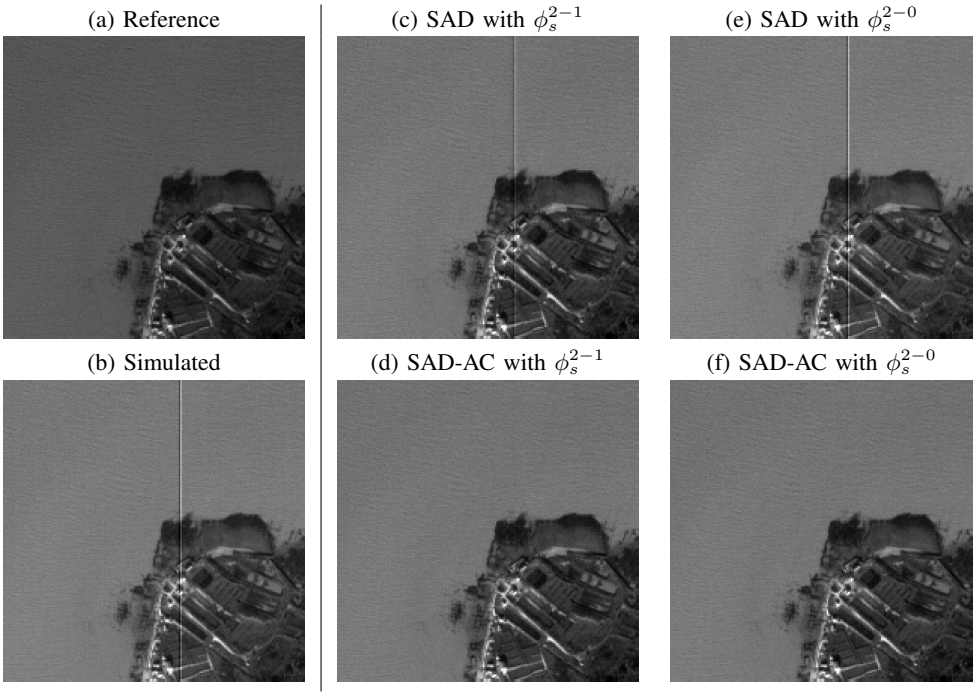

Fig. 6: Simulated PLEIADES-type test image B of size 4000×2000 with atypical columns. A 200×200 Snapshot of: (a) real image, (b) simulated image with two outliers columns, (c) & (e) calibration with the standard SAD method, (d) & (f) calibration with SAD-AC.

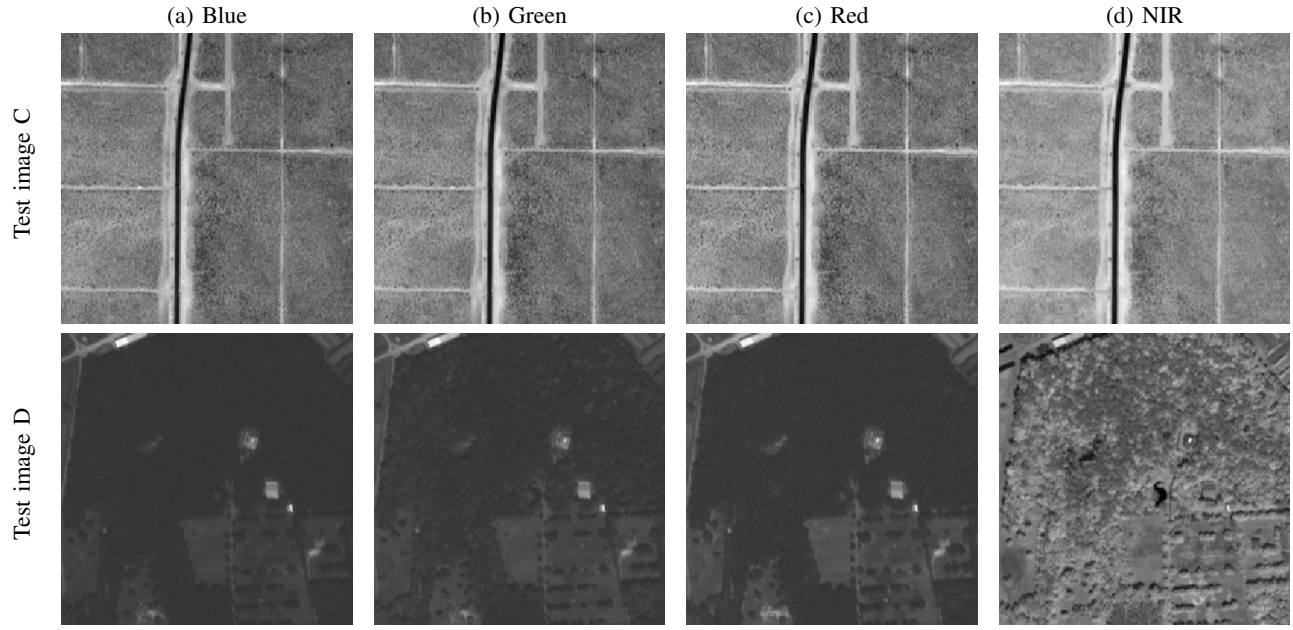

Fig. 7: Multispectral PLEIADES-type images of size 10000×2500: 200×200 snapshots of the four spectral bands of test images C and D

where g_c^p (resp. o_c^p) is a gain (resp., offset) parameter associated to column c and spectral band p. This also implies that the constraints are defined on each spectral band separately: $\forall p,\ \frac{1}{C}\sum_c g_c^p = 1$.

B. Optimization

Let $\boldsymbol{g} = (g_c^p)_{\Gamma_C \times \Gamma_P}$ and $\boldsymbol{o} = (o_c^p)_{\Gamma_C \times \Gamma_P}$. The MAP estimator of $(\boldsymbol{g}, \boldsymbol{o})$ is the minimizer of the following cost

function:

$$K(\boldsymbol{g},\boldsymbol{o}) = \frac{1}{T}\sum_{r,c}\phi_s\left(\|\boldsymbol{\delta}_{r,c}\|\right) + \sum_{c,p}\left(\lambda_g^p(g_c^p-1)^2 + \lambda_o^p(o_c^p)^2\right) \tag{18}$$

with $\lambda_g^p = \frac{1}{2(\sigma_g^p)^2}$, $\lambda_o^p = \frac{1}{2(\sigma_o^p)^2}$, and

$$\|\boldsymbol{\delta}_{r,c}\|^2 = \sum_p \left(g_c^p w_{r,c}^p - g_{c+1}^p w_{r,c+1}^p - o_c^p + o_{c+1}^p\right)^2,$$

under the P constraints:

$$\forall p \in \Gamma_P,\ \frac{1}{C}\sum_c g_c^p = 1.$$

Introducing vectors $\boldsymbol{x}_p^{\mathrm{t}} = \left[g_1^p, \ldots, g_C^p, o_1^p, \ldots, o_C^p\right]^{\mathrm{t}}$ and $\overline{\boldsymbol{x}} = \left[\boldsymbol{x}_1^{\mathrm{t}}, \ldots, \boldsymbol{x}_P^{\mathrm{t}}\right]^{\mathrm{t}}$, the cost function (18) can be written:

$$K(\overline{\boldsymbol{x}}) = \frac{1}{T}\sum_{r,c}\phi_s(\|\boldsymbol{\delta}_{r,c}\|) + \overline{\boldsymbol{x}}^{\mathrm{t}}\overline{\mathbf{Q}}\overline{\boldsymbol{x}} - 2\overline{\boldsymbol{q}}^{\mathrm{t}}\overline{\boldsymbol{x}} \tag{19}$$

where $\overline{\mathbf{Q}}$ is a $2PC \times 2PC$ diagonal matrix, and $\overline{\boldsymbol{q}}$ a column vector of length $2PC$:

$$\overline{\mathbf{Q}} = \mathrm{diag}\left\{\mathbf{Q}_1, \ldots, \mathbf{Q}_P\right\}, \tag{20}$$

$$\mathbf{Q}_p = \mathrm{diag}\left\{\lambda_g^p, \ldots, \lambda_g^p, \lambda_o^p, \ldots, \lambda_o^p\right\} \tag{21}$$

$$\overline{\boldsymbol{q}} = \left[\lambda_g^1\boldsymbol{e}^{\mathrm{t}}, \ldots, \lambda_g^P\boldsymbol{e}^{\mathrm{t}}\right]^{\mathrm{t}}. \tag{22}$$

On the other hand, each linear constraint reads $\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x}_p = C$. As shown in Appendix C, constrained minimization of (19) can be obtained using a constrained IRLS algorithm according to the following scheme:

$$\boldsymbol{x}_p^{(k+1)} = \frac{C\left(\mathbf{B}_p^{(k)}\right)^{-1}\boldsymbol{e}}{\boldsymbol{e}^{\mathrm{t}}\left(\mathbf{B}_p^{(k)}\right)^{-1}\boldsymbol{e}},\ \forall p \in \Gamma_P,$$

where

$$\mathbf{B}_p^{(k)} = \mathbf{Q}_p + \frac{1}{T}\sum_{r=1}^{R}\mathbf{V}_r^p\mathbf{L}_r^{(k)}(\mathbf{V}_r^p)^{\mathrm{t}} \tag{23}$$

is the matrix associated to the pth spectral band, with

$$\mathbf{L}_r^{(k)} = \mathrm{diag}\left\{\frac{\phi_s'\left(\left\|\boldsymbol{\delta}_{r,c}^{(k)}\right\|\right)}{2\left\|\boldsymbol{\delta}_{r,c}^{(k)}\right\|}\right\}_{c\in\Gamma_{C-1}}.$$

From a computational perspective, the difference between the multispectral version of SAD (SAD-MS) and standard SAD lies only in the *Majorize* step of the algorithm (construction of matrix $\mathbf{B}$), while the *Minimize* step is done independently on each spectral band in both algorithms, using the same analytical expression. Incidentally, let us remark that a multispectral version of SLD would be easy to develop on the very same basis.

C. Simulation results

The calibration results based on the multispectral model are illustrated on the two PLEIADES-type simulated images (test images C and D) of Fig. 7 with four spectral bands each (Blue, Green, Red and NIR).

The results, summarized in Table II, can be compared to a simple independent calibration of the four different bands using the standard SAD method. On the one hand, they show that SAD-MS leads to a better calibration of test image C, both in terms of PSNR and SSIM, regardless of the potential function ϕ_s, and that the computing cost is almost identical for SAD and SAD-MS. On the other hand, for test image D, SAD-MS improves the calibration of the three first bands in terms of PSNR only in the case of ϕ_s^{2-1}, while standard SAD does uniformly better in the case of ϕ_s^{2-0}.

TABLE II: Calibration results of test images C and D using SAD and SAD-MS. To reduce the computation cost, an averaging of the data over subsets of $L = 20$ rows has been performed, as in Subsection. III-E.

Image	ϕ_s	Method	Band	$\mathrm{PSNR}_{\mathrm{dB}}$	SSIM	N_{it}	CT (s)
C	ϕ_s^{2-1}	SAD-MS	Blue	**64.74**	**0.99940**	14	5.57
			Green	**63.12**	**0.99939**		
			Red	**61.98**	**0.99944**		
			NIR	**59.41**	**0.99951**		
		SAD	Blue	62.72	0.99871	28	1.98
			Green	61.43	0.99863	30	1.99
			Red	60.28	0.99878	25	1.71
			NIR	57.80	0.99829	28	1.86
	ϕ_s^{2-0}	SAD-MS	Blue	**65.88**	**0.99955**	24	8.52
			Green	**64.76**	**0.99949**		
			Red	**63.18**	**0.99950**		
			NIR	**62.14**	**0.99952**		
		SAD	Blue	65.07	0.99945	22	1.41
			Green	63.47	0.99929	28	1.58
			Red	61.41	0.99918	19	1.21
			NIR	60.42	0.99932	21	1.16
D	ϕ_s^{2-1}	SAD-MS	Blue	**67.94**	**0.99927**	17	28.49
			Green	**67.67**	**0.99920**		
			Red	**68.35**	**0.99894**		
			NIR	51.47	**0.99398**		
		SAD	Blue	66.10	0.99824	22	7.48
			Green	65.72	0.99801	23	7.87
			Red	66.82	0.99775	23	7.80
			NIR	**57.09**	0.99236	35	13.93
	ϕ_s^{2-0}	SAD-MS	Blue	68.53	**0.99955**	38	57.04
			Green	68.15	**0.99944**		
			Red	69.85	0.99918		
			NIR	56.23	**0.99509**		
		SAD	Blue	**69.51**	0.99949	43	15.92
			Green	**68.26**	0.99934	49	18.27
			Red	**69.87**	**0.99923**	21	6.82
			NIR	**57.95**	0.99434	146	46.33

In order to analyze these results further, let us evaluate the correlation between the spectral bands using their empirical

correlation. More precisely, let us introduce the normalized correlation coefficient between column-wise gradient images:

$$C_{\delta z}(p,q) = \frac{\sum_{r,c}\left(\boldsymbol{\delta z}^p_{r,c} - \overline{\boldsymbol{\delta z}}^p\right)\left(\boldsymbol{\delta z}^q_{r,c} - \overline{\boldsymbol{\delta z}}^q\right)}{\sqrt{\sum_{r,c}\left(\boldsymbol{\delta z}^p_{r,c} - \overline{\boldsymbol{\delta z}}^p\right)^2 \sum_{r,c}\left(\boldsymbol{\delta z}^q_{r,c} - \overline{\boldsymbol{\delta z}}^q\right)^2}}$$

where $\forall p \in \Gamma_P$

$$\boldsymbol{\delta z}^p = \left\{\delta z^p_{r,c} = z^p_{r,c} - z^p_{r,c+1}, (r,c) \in \Gamma_R \times \Gamma_{C-1}\right\},$$
$$\overline{\boldsymbol{\delta z}}^p = \frac{1}{R(C-1)}\sum_{r,c}\boldsymbol{\delta z}^p_{r,c}$$

In practice we cannot compute coefficients $C_{\delta z}(p,q)$, but only their equivalent $C_{\delta w}(p,q)$ computed from observed images. However, it can be shown that if the calibration parameters are uncorrelated from one spectral band to the other, and are of the same mean, then $C_{\delta z}(p,q) \approx C_{\delta w}(p,q)$.

The correlation coefficients of the two tested images are given in Table III. For test image C, the four spectral bands are highly correlated. For test image D, bands Blue, Green and Red are highly correlated, while band NIR has low correlation coefficients with the other bands.

TABLE III: Normalized correlation coefficients of the spectral bands of test images C and D, computed with the real images $\boldsymbol{z}$ and the observed ones $\boldsymbol{w}$.

Test image C

$C_{\delta z}$

	B	G	R	NIR
B	1	0.954	0.883	0.665
G		1	0.911	0.718
R			1	0.736
NIR				1

$C_{\delta w}$

	B	G	R	NIR
B	1	0.862	0.807	0.606
G		1	0.848	0.657
R			1	0.686
NIR				1

Test image D

$C_{\delta z}$

	B	G	R	NIR
B	1	0.908	0.882	0.318
G		1	0.890	0.474
R			1	0.298
NIR				1

$C_{\delta w}$

	B	G	R	NIR
B	1	0.752	0.759	0.277
G		1	0.810	0.434
R			1	0.283
NIR				1

Consequently, we recommend to compute correlation coefficients between bands $C_{\delta w}(p,q)$, prior to calibration, in order to determine which groups of bands could benefit from a joint calibration using SAD-MS.

Finally, let us mention that it is very difficult to evaluate the improvement brought by SAD-MS over SAD by visual inspection of the corrected image. This is because the intrinsic efficiency of SAD being already high, stripes are neither visible in the images corrected by SAD or by SAD-MS.

VI. Conclusion

This paper introduced a new unsupervised calibration method, based on an affine detector response, for statistical destriping of pushbroom-type images. An efficient constrained IRLS algorithm was derived, which allows one to perform fast calibration of high resolution, large size images. In the case where the detector offsets are significant, the new method based on an affine detector response is shown to provide significantly better calibration results than a previous method based on a linear model. Furthermore, when the detector offsets can be considered negligible, both methods provide comparable calibration results. Moreover, the additional computing load of the new version remains moderate. In addition, we have proposed heuristic formulas for hyper-parameter tuning from the observed image only, which allows us to propose an unsupervised calibration method. The robustness to noise was empirically demonstrated over a database of simulated PLEIADES-type images.

Two types of extension were further considered. The first one handles the possible presence of pathological detectors producing atypical columns. Another extension to the multispectral case was introduced, so that joint calibration of multiple spectral bands can be obtained. In practice, the application of the latter is only beneficial to groups of spectral bands with a significant level of structural correlation. We introduced a simple way to determine such groups prior to the calibration step.

At least two types of perspectives remain to be investigated. The first one would consist in adapting the proposed calibration method to acquisition structures that differ from pushbroom linear arrays of detectors. For instance, whiskbroom instruments make use of a reduced set of detectors, and they generate stripes with a periodic assignment between the detectors and the acquired lines. The proposed method could be modified to take such a geometry into account.

On the other hand, an extension from the multispectral case to hyperspectral instruments is an important perspective. A first possibility would be to still measure the correlation between bands and to consider groups of bands accordingly. Another possibility would be to correlate neighboring bands in a systematic way thanks to a Markov chain model, up to suited modifications of the proposed method. In both cases, special attention will be needed to the computing effort, given the potentially huge size of hyperspectral cubes. In this regard, let us stress that the proposed method already exploits the structured sparse character of the calibration problem through the block-tridiagonal structure of matrices, which naturally offers good possibilities of parallel implementations. The latter should be further examined to limit the computing time of an extension to the calibration of hyperspectral instruments.

Appendix

A. Constrained IRLS Algorithm

To construct a quadratic majorizing approximation of the cost function (11), we adopt the so-called *half-quadratic* framework, akin to [25], [27], [31]. In particular, under specific conditions on $\phi_s(x)$, there exists a function $\psi(t)$ such that

$$\phi_s(x) = \inf_t \phi^*_s(x,t) \tag{24}$$

where

$$\phi^*_s(x,t) = x^2 t + \psi(t). \tag{25}$$

Moreover, as a function of t, $\phi^*_s(x,t)$ is convex for any value of x, and its minimizer t^* reads

$$\arg\min_t \phi^*_s(x,t) = \frac{\phi'_s(x)}{2x}, \tag{26}$$

where ϕ'_s stands for the first derivative of ϕ_s.

As a consequence of (24)-(26), we can build a quadratic majorizing approximation of (11) at any point $\boldsymbol{x}^{(0)}$ as

$$L\big(\boldsymbol{x},\boldsymbol{x}^{(0)}\big) = Q(\boldsymbol{x}) + \frac{1}{T}\sum_{r,c}\Big(t_{r,c}^{(0)}([\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}]_c)^2 + \psi(t_{r,c})\Big) \quad (27)$$

where

$$t_{r,c}^{(0)} = \arg\min_t \phi_s^*\Big(\big[\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(0)}\big]_c, t\Big) = \frac{\phi'_s(\big[\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(0)}\big]_c)}{2\big[\mathbf{V}_r^{\mathrm{t}}\boldsymbol{x}^{(0)}\big]_c}.$$

Furthermore, it is straightforward to check that $L(\boldsymbol{x},\boldsymbol{x}^{(0)})$ also reads:

$$L(\boldsymbol{x},\boldsymbol{x}^{(0)}) = \boldsymbol{x}^{\mathrm{t}}\mathbf{B}^{(0)}\boldsymbol{x} - 2\lambda_g\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x} + \frac{1}{T}\sum_{r,c}\psi(t_{r,c}^{(0)})$$

where

$$\mathbf{B}^{(0)} = \mathbf{Q} + \frac{1}{T}\sum_r \mathbf{V}_r\mathbf{L}_r^{(0)}\mathbf{V}_r^{\mathrm{t}},$$
$$\mathbf{L}_r^{(0)} = \mathrm{diag}\big\{t_{r,c}^{(0)}\big\}_{c\in\Gamma_{C-1}}.$$

As a consequence, the unconstrained minimizer of $L(\boldsymbol{x},\boldsymbol{x}^{(0)})$ w.r.t. $\boldsymbol{x}$ reads $\lambda_g\big(\mathbf{B}^{(0)}\big)^{-1}\boldsymbol{e}$. Unconstrained IRLS algorithms make use of the latter expression to update the current point $\boldsymbol{x}^{(0)}$.

Here, our goal is to take the constraint $\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x} = C$ into account in the IRLS framework. Our proposition is simply to leave the quadratic majorizer (27) unchanged, and to replace its unconstrained minimization by a constrained version. Minimization of a quadratic objective function under a linear constraint is a mathematically simple problem. Let the Lagrangian be defined as

$$\mathcal{L}(\boldsymbol{x},\boldsymbol{x}^{(0)},\mu) = L(\boldsymbol{x},\boldsymbol{x}^{(0)}) - \mu(\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x} - C)$$

where μ is the Lagrange multiplier. Cancelling the derivative w.r.t. $\boldsymbol{x}$ yields:

$$\boldsymbol{x}(\mu) = \Big(\lambda_g + \frac{\mu}{2}\Big)\big(\mathbf{B}^{(0)}\big)^{-1}\boldsymbol{e}. \quad (28)$$

Then, it is possible to determine the Lagrange parameter value μ^* so that $\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x}(\mu^*) = C$ holds:

$$\mu^* = \frac{2C}{\boldsymbol{e}^{\mathrm{t}}(\mathbf{B}^{(0)})^{-1}\boldsymbol{e}} - 2\lambda_g,$$

so we get the following expression for the constrained IRLS update:

$$\boldsymbol{x}^{(1)} = \arg\min_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x},\mu^*) = \frac{C(\mathbf{B}^{(0)})^{-1}\boldsymbol{e}}{\boldsymbol{e}^{\mathrm{t}}(\mathbf{B}^{(0)})^{-1}\boldsymbol{e}}. \quad (29)$$

B. Sparse implementation

Given the sparse structure of matrices $\mathbf{V}_r$, matrix

$$\mathbf{B} = \mathbf{Q} + \frac{1}{T}\sum_r \mathbf{V}_r\mathbf{L}_r\mathbf{V}_r^{\mathrm{t}}$$

can be efficiently encoded as follows. Indeed, matrix $\mathbf{B}$ is block-tridiagonal, and we can define each block separately:

$$\mathbf{B} = \begin{bmatrix}\mathbf{M}_1 & \mathbf{M}_2\\ \mathbf{M}_2^{\mathrm{t}} & \mathbf{M}_3\end{bmatrix},$$

where $\mathbf{M}_1$, $\mathbf{M}_2$ and $\mathbf{M}_3$ are tridiagonal matrices, $\mathbf{M}_1$ and $\mathbf{M}_3$ being also symmetric.

Let us denote $\odot$ the element-wise vector multiplication, and

$$\begin{aligned}
\boldsymbol{t}_r &= [t_{r,1}, t_{r,2}, \ldots, t_{r,C-1}],\\
\boldsymbol{t}_r^+ &= [t_{r,1}, t_{r,2}, \ldots, t_{r,C-1}, 0],\\
\boldsymbol{t}_{r+} &= [0, t_{r,1}, \ldots, t_{r,C-2}, t_{r,C-1}],\\
\boldsymbol{w}_r &= [w_{r,1}, w_{r,2}, \ldots, w_{r,C-1}, w_{r,C}],\\
\boldsymbol{w}_r^- &= [w_{r,2}, w_{r,3}, w_{r,4}, \ldots, w_{r,C}],\\
\boldsymbol{w}_{r-} &= [w_{r,1}, w_{r,2}, w_{r,3}, \ldots, w_{r,C-1}].
\end{aligned}$$

Then we have

$$\begin{aligned}
\mathbf{M}_1 = &\frac{1}{T}\mathrm{diag}_0\Big\{\sum_r(\boldsymbol{w}_r\odot\boldsymbol{w}_r)\odot(\boldsymbol{t}_r^+ + \boldsymbol{t}_{r-}) + \lambda_g\Big\}\\
&- \frac{1}{T}\mathrm{diag}_1\Big\{\sum_r(\boldsymbol{w}_{r-}\odot\boldsymbol{w}_r^-\odot\boldsymbol{t}_r)\Big\},\\
&- \frac{1}{T}\mathrm{diag}_{-1}\Big\{\sum_r(\boldsymbol{w}_{r-}\odot\boldsymbol{w}_r^-\odot\boldsymbol{t}_r)\Big\},\\
\mathbf{M}_3 = &\frac{1}{T}\mathrm{diag}_0\Big\{\sum_r(\boldsymbol{t}_r^+ + \boldsymbol{t}_{r+}) + \lambda_o\Big\}\\
&- \frac{1}{T}\mathrm{diag}_1\Big\{\sum_r\boldsymbol{t}_r\Big\},\\
&- \frac{1}{T}\mathrm{diag}_{-1}\Big\{\sum_r\boldsymbol{t}_r\Big\},\\
\mathbf{M}_2 = &-\frac{1}{T}\mathrm{diag}_0\Big\{\sum_r\boldsymbol{w}_r\odot(\boldsymbol{t}_r^+ + \boldsymbol{t}_{r+})\Big\},\\
&+ \frac{1}{T}\mathrm{diag}_1\Big\{\sum_r\boldsymbol{w}_{r-}\odot\boldsymbol{t}_r\Big\}\\
&+ \frac{1}{T}\mathrm{diag}_{-1}\Big\{\sum_r\boldsymbol{w}_r^-\odot\boldsymbol{t}_r\Big\}.
\end{aligned}$$

C. Constrained multispectral algorithm

In the multispectral case, (27) becomes

$$\begin{aligned}
L\big(\overline{\boldsymbol{x}},\overline{\boldsymbol{x}}^{(0)}\big) &= \overline{\boldsymbol{x}}^{\mathrm{t}}\overline{\mathbf{Q}}\overline{\boldsymbol{x}} - 2\overline{\boldsymbol{q}}^{\mathrm{t}}\overline{\boldsymbol{x}} + \sum_{r,c}\Big(t_{r,c}^{(0)}\left\|\boldsymbol{\delta}_{r,c}\right\|^2 + \psi\big(t_{r,c}^{(0)}\big)\Big)\\
&= \overline{\boldsymbol{x}}^{\mathrm{t}}\overline{\mathbf{B}}^{(0)}\overline{\boldsymbol{x}} - 2\overline{\boldsymbol{q}}^{\mathrm{t}}\overline{\boldsymbol{x}} + \sum_{r,c}\psi\big(t_{r,c}^{(0)}\big)
\end{aligned}$$

where

$$t_{r,c}^{(0)} = \frac{\phi'_s\big(\big\|\boldsymbol{\delta}_{r,c}^{(0)}\big\|\big)}{2\big\|\boldsymbol{\delta}_{r,c}^{(0)}\big\|}$$

and $\overline{\mathbf{B}}^{(0)}$ is a $2PC\times 2PC$ block-diagonal matrix defined as:

$$\overline{\mathbf{B}}^{(0)} = \mathrm{diag}\Big\{\mathbf{B}_1^{(0)},\ldots,\mathbf{B}_P^{(0)}\Big\},$$

each $\mathbf{B}_p^{(0)}$ corresponding to the p-th spectral band defined in (23).

As mentioned before, each of the P spectral channels induces an equality constraint

$$\boldsymbol{e}^{\mathrm{t}}\boldsymbol{x}_p = \overline{\boldsymbol{e}}_p^{\mathrm{t}}\overline{\boldsymbol{x}} = C,$$

where $\overline{\boldsymbol{e}}_p$ is a $2PC \times 1$ vector composed of P vectors of size $2C \times 1$, all of which being zero vectors except the p-th, which is equal to $\boldsymbol{e}$. Again, let us write the augmented Lagrangian:

$$\mathcal{L}(\overline{\boldsymbol{x}}, \overline{\boldsymbol{x}}^{(0)}, \boldsymbol{\mu}) = L\big(\overline{\boldsymbol{x}}, \overline{\boldsymbol{x}}^{(0)}\big) - \sum_{p}^{P} \mu_p \big(\overline{\boldsymbol{e}}_p^{\mathrm{t}} \overline{\boldsymbol{x}} - C\big) \tag{30}$$

with μ_p the Lagrange parameters. By cancelling the derivative of the Lagrangian w.r.t. $\boldsymbol{x}$, we get:

$$2\overline{\mathbf{B}}^{(0)} \overline{\boldsymbol{x}} - 2\overline{\boldsymbol{q}} - \sum_{p=1}^{P} \mu_p \overline{\boldsymbol{e}}_p = \mathbf{0} \tag{31}$$

where

$$\overline{\boldsymbol{q}} = \sum_{p=1}^{P} \lambda_g^p \overline{\boldsymbol{e}}_p$$

according to (22). Finally, (31) is yields:

$$\forall p \in \Gamma_P, \quad \boldsymbol{x}_p(\mu_p) = \left(\lambda_g^p + \frac{\mu_p}{2}\right) \left(\mathbf{B}_p^{(0)}\right)^{-1} \boldsymbol{e}$$

which generalizes (28). Once again, it is possible to determine the Lagrange parameter values μ_p^* so that $\boldsymbol{e}^{\mathrm{t}} \boldsymbol{x}_p(\mu^*) = C$, and we get the following expression for the constrained IRLS update in the multispectral case:

$$\forall p \in \Gamma_P, \quad \boldsymbol{x}_p^{(1)} = \frac{C \left(\mathbf{B}_p^{(0)}\right)^{-1} \boldsymbol{e}}{\boldsymbol{e}^{\mathrm{t}} \left(\mathbf{B}_p^{(0)}\right)^{-1} \boldsymbol{e}}.$$

REFERENCES

[1] Y. Chang, M. Chen, L. Yan, X.-L. Zhao, Y. Li, and S. Zhong, "Toward universal stripe removal via wavelet-based deep convolutional neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 4, pp. 2880–2897, Dec. 2020.

[2] J.-J. Pan and C.-I. Chang, "Destriping of landsat MSS images by filtering techniques," *Photogrammetric engineering and remote sensing*, vol. 58, pp. 1417–1417, Oct. 1992.

[3] B. Zhao, B. He, and Y. Cong, "Destriping method using lifting wavelet transform of remote sensing image," in *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering*, vol. 6. IEEE, Aug. 2010, pp. 110–113.

[4] B. Munch, P. Trtik, F. Marone, and M. Stampanoni, "Stripe and ring artifact removal with combined wavelet-Fourier filtering," *Optics Express*, vol. 17, no. 10, p. 8567, May 2009.

[5] R. E. Crippen, "A simple spatial filtering routine for the cosmetic removal of scan-line noise from Landsat TM P-tape imagery," *Photogrammetric Engineering*, p. 5, Mar. 1989.

[6] D. L. Helder, "A technique for the reduction of banding in Landsat thematic mapper images," *Photogrammetric Engineering*, vol. 58, no. 10, pp. 1425–1431, Oct. 1992.

[7] J. Simpson, "Improved destriping of GOES images using finite impulse response filters," *Remote Sensing of Environment*, vol. 52, no. 1, pp. 15–35, Apr. 1995.

[8] X. Liu, X. Lu, H. Shen, Q. Yuan, Y. Jiao, and L. Zhang, "Stripe noise separation and removal in remote sensing images by consideration of the global sparsity and local variational properties," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 5, pp. 3049–3060, Jan. 2016.

[9] Y. Chen, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, and J. Huang, "Stripe noise removal of remote sensing images by total variation regularization and group sparsity constraint," *Remote Sensing*, vol. 9, no. 6, p. 559, Jun. 2017.

[10] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, "Hyperspectral image restoration using low-rank matrix recovery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 8, pp. 4729–4743, Aug. 2014.

[11] Y. Chang, L. Yan, T. Wu, and S. Zhong, "Remote sensing image stripe noise removal: From image decomposition perspective," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7018–7031, Dec. 2016.

[12] W. Cao, Y. Chang, G. Han, and J. Li, "Destriping remote sensing image via low-rank approximation and nonlocal total variation," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 6, pp. 848–852, Mar. 2018.

[13] M. Bouali and S. Ladjal, "Toward optimal destriping of modis data using a unidirectional variational model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 8, pp. 2924–2935, Apr. 2011.

[14] Y. Chang, H. Fang, L. Yan, and H. Liu, "Robust destriping method with unidirectional total variation and framelet regularization," *Optics Express*, vol. 21, no. 20, pp. 23 307–23 323, Oct. 2013.

[15] X. Liu, H. Shen, Q. Yuan, X. Lu, and C. Zhou, "A universal destriping framework combining 1-D and 2-D variational optimization methods," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 808–822, Feb. 2018.

[16] X. Kuang, X. Sui, Q. Chen, and G. Gu, "Single infrared image stripe noise removal using deep convolutional networks," *IEEE Photonics Journal*, vol. 9, no. 4, pp. 1–13, Aug. 2017.

[17] Y. Chang, L. Yan, H. Fang, S. Zhong, and W. Liao, "HSI-DeNet: Hyperspectral image restoration via convolutional neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 2, pp. 667–682, Feb. 2019.

[18] Y. Chen, T.-Z. Huang, and X.-L. Zhao, "Destriping of multispectral remote sensing image using low-rank tensor decomposition," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 12, pp. 4950–4967, Dec. 2018.

[19] V. Algazi and G. E. Ford, "Radiometric equalization of nonperiodic striping in satellite data," *Computer Graphics and Image Processing*, vol. 16, no. 3, pp. 287–295, Jul. 1981.

[20] B. Cao, Y. Du, D. Xu, H. Li, and Q. Liu, "An improved histogram matching algorithm for the removal of striping noise in optical remote sensing imagery," *Optik*, vol. 126, no. 23, pp. 4723–4730, Dec. 2015.

[21] G. Corsini, M. Diani, and T. Walzel, "Striping removal in MOS-B data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 3, pp. 1439–1446, May 2000.

[22] F. L. Gadallah, F. Csillag, and E. J. M. Smith, "Destriping multisensor imagery with moment matching," *International Journal of Remote Sensing*, vol. 21, no. 12, pp. 2505–2511, Jan. 2000.

[23] B. K. Horn and R. J. Woodham, "Destriping LANDSAT MSS images by histogram modification," *Computer Graphics and Image Processing*, vol. 10, no. 1, pp. 69–83, May 1979.

[24] P. Rakwatin, W. Takeuchi, and Y. Yasuoka, "Stripe noise reduction in MODIS data by combining histogram matching with facet filter," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 6, pp. 1844–1856, Jun. 2007.

[25] H. Carfantan and J. Idier, "Statistical linear destriping of satellite-based pushbroom-type images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 4, pp. 1860–1871, Apr. 2010.

[26] V. Martin, G. Blanchet, P. Kubik, S. Lacherade, C. Latry, L. Lebegue, F. Lenoir, and F. Porez-Nadal, "PLEIADES-HR 1A&1B image quality commissioning: innovative radiometric calibration methods and results," in *Earth Observing Systems XVIII*, vol. 886610. San Diego, CA, USA: SPIE, Sep. 2013, pp. 304 – 314.

[27] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 2, pp. 298–311, Feb. 1997.

[28] R. Fletcher, *Practical Methods of Optimization*, 2nd ed. New York, NY, USA: John Wiley & Sons, 1987.

[29] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, Feb. 2004.

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[31] J. Idier, "Convex half-quadratic criteria and interacting auxiliary variables for image restoration," *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1001–1009, Jul. 2001.