BLOCK-COORDINATE PROXIMAL ALGORITHMS
FOR SCALE-FREE TEXTURE SEGMENTATION

Barbara Pascal[1], Nelly Pustelnik[1], Patrice Abry[1], Jean-Christophe Pesquet[2]

[1] Univ Lyon, ENS de Lyon, Univ Claude Bernard Lyon 1, CNRS,
Laboratoire de Physique, F-69342 Lyon, France, `firstname.lastname@ens-lyon.fr`
[2] Center for Visual Computing, INRIA, CentraleSupélec, Univ. Paris-Saclay,
9 rue Joliot Curie, 91190 Gif sur Yvette, France, `jean-christophe.pesquet@centralesupelec.fr`

ABSTRACT

Texture segmentation still constitutes an on-going challenge, especially when processing large-size images. Recently, procedures integrating a scale-free (or fractal) wavelet-leader model allowed the problem to be reformulated in a convex optimization framework by including a TV penalization. In this case, the TV penalty plays a prominent role with respect to the data fidelity term, which makes the approach costly in terms of memory and computation cost. The present contribution aims to investigate the potential of recent block-coordinate dual and primal-dual proximal algorithms for overcoming this numerical issue. Our study shows that a key ingredient in the success of the proposed block-coordinate approaches lies in the design of the blocks of variables which are updated at each iteration. Numerical experiments conducted over synthetic textures having piece-wise constant fractal properties confirm our theoretical analysis. The proposed lattice block design strategy is shown to yield significantly lower memory and computational requirements.

Index Terms— Total variation, block-coordinate methods, nonsmooth optimization, primal dual algorithms, texture segmentation, wavelets.

1. INTRODUCTION

Scale-free and total variation based texture segmentation. Texture segmentation constitutes a major challenge in image processing, medical imaging being a prominent example, e.g., for tumor localization [1] or for macroscopic brain functional connectivity assessment [2]. Biological tissues being complicated organic structures, medical images often consists of complex textures, the segmentation of which requires the use of advanced image processing tools.

Beyond classical features, such as local variance or spectral histograms [3], it has recently been shown that scale-free dynamics features yield an efficient segmentation of textures (cf. e.g., [4]). Interestingly, a joint estimation and segmentation procedure combining scale-free features with functional optimization has been proposed and shown to provide state-of-the-art performance [5]. To perform segmentation based on piece-wise constant scale-free characteristics, the objective function is constructed as a data fidelity term applied to scale-free features balanced by a total variation (TV) term. Because such formulation ends up in sums of possibly nonsmooth functions, the minimization procedure relies on the use of proximal algorithms [6, 7, 8, 9].

While promising, such approaches suffer from practical limitations.

Scale-free features are often computed from multiscale representations, such as wavelets, or wavelet-leaders [10], thus requiring a large continuum of scales and hence large size or high-resolution images. Furthermore, in the functional formulation, the balance between data fidelity and TV penalization is tuned by a regularization parameter. The optimal parameter selection thus implies that the minimization is performed several times. Finally, although efficient algorithms were designed for classical TV based image denoising [11, 12], these algorithms converge far slower in the scale-free texture segmentation problem. The main limitations in the use of TV-based scale-free texture segmentation procedures consist of the large memory capacities needed to store images, features and additional variables involved in the minimization process, and of prohibitive computational costs. There is thus a strong need to overcome these limitations to permit an actual use of TV-based scale-free texture segmentation in real-world applications.

Related works. TV-based functional minimization is often achieved by using either a dual formulation of the forward-backward algorithms [11, 13] or primal-dual proximal algorithms [14, 15]. Numerous proposals were made to accelerate proximal algorithms (e.g. [16, 17]), which remains a hot topic. In the same time, some block-coordinate strategies were devised for solving generic convex problems, both for forward-backward iterations [12, 18, 19] and primal-dual algorithms [18, 20, 21, 22, 23, 2]. In [19], an accelerated forward-backward algorithm based on deterministic block activation is proposed and applied to the joint deconvolution and deinterlacing of video sequences. In [23], an application in tomography illustrates the acceleration of primal-dual algorithms by means of a block-coordinate formulation. In [24], an approximation of isotropic total variation permits to propose an alternating algorithm for TV-denoising.

Goals, contributions and outline. The present contribution aims to explore efficient designs of block-coordinate proximal algorithms for anisotropic TV denoising of textures characterized by piecewise constant scale-free properties. Elaborating on [5], scale-free properties characterized by a local regularity measure, referred to as the Hölder exponent [10] are described in Section 2.1. Both Dual Forward-Backward (DFB) and Primal-Dual schemes (PD) are considered, which are recalled in Section 2.2. Block-coordinate formulations are further detailed in Section 2.3. Section 3 investigates formally the impact of block-design both in terms of memory and computational costs. Section 4 further assesses the numerically achieved performance, both for block-DFB and block-PD, on synthetic textures. The algorithm convergence quantified by means of the duality gap shows significant gains resulting from the proposed approach.

2. SCALE-FREE AND TOTAL VARIATION BASED TEXTURE SEGMENTATION

2.1. Local regularity as scale-free feature

Textures can be characterized by several different scale-free features [10]. Elaborating on [5], our approach is grounded on the use of local regularity, measured by the Hölder exponent h. Let $f\colon \mathbb{R}^2 \to \mathbb{R}$ denote a 2D real field. Local regularity at location $\underline{z}_0 \in \mathbb{R}^2$ is quantified by the Hölder exponent $h(\underline{z}_0)$ defined as the largest $\alpha > 0$ such that there exists a constant $\chi > 0$ and a polynomial $\mathcal{P}_{\underline{z}_0}$ of degree lower than α, satisfying $\left| f(\underline{z}) - \mathcal{P}_{\underline{z}_0}(\underline{z}) \right| \leq \chi \|\underline{z} - \underline{z}_0\|^{\alpha}$ for every $\underline{z}$ in a neighborhood of $\underline{z}_0$. Beyond this formal definition the practical assessment of $h(\underline{z}_0)$ is conducted using multiresolution analysis. Let $X \in \mathbb{R}^{N \times N}$ correspond to discretization of f on a finite grid and let $w_j = \mathbf{W}_j X$ be the wavelet coefficients of image X, at resolution j, with $\{\mathbf{W}_j \colon \mathbb{R}^{N \times N} \to \mathbb{R}^{M_j \times M_j}, j \in \{1, \ldots, J\}\}$ performing an orthonormal wavelet transform. It has been proven [10] that the estimation $h_{\underline{n}}$ of h at pixel $\underline{n}$ can be measured from a non-linear transform $\mathcal{L}(w_{j,\underline{k}}) = \mathcal{L}_{j,\underline{k}}$ of the wavelet coefficients at resolution level j and location $\underline{k}$ (either consisting of its absolute value or of a local supremum of absolute values taken over a spatial neighborhood across all finer scales, referred to as wavelet-leaders [10]):

$$\mathcal{L}_{j,\underline{k}} \simeq \eta_{\underline{n}} 2^{j h_{\underline{n}}}, \qquad 2^j \to 0, \quad \underline{n} = 2^j \underline{k}. \tag{1}$$

An estimate $\widehat{h}_{\underline{n}}$ is thus obtained from a linear regression across scales of $\log_2 \mathcal{L}_{j,\underline{k}}$.

2.2. TV denoising

It is assumed here that textures of interest are well-modeled as a collection of piecewise constant h patches Ω_q forming a partition of the spatial domain of interest Ω, i.e. $h_{\underline{n}} \equiv v_q \in \mathbb{R}$ for $\underline{n} \in \Omega_q$, with $\bigcup_q \Omega_q = \Omega$ and $\forall q \neq q'$, $\Omega_q \bigcap \Omega_{q'} = \varnothing$. Because local in nature, the estimate $\widehat{h}$ suffers from a large variance that may preclude to detect actual changes in h along the texture, hence motivating the use of a posterior TV regularization aiming at smoothing piecewise constant h regions while preserving actual changes. An efficient estimation strategy is then based on the Total Variation (TV) penalization:

$$\widehat{\widehat{h}} = \arg\min_h \frac{1}{2}\|h - \widehat{h}\|_{\mathrm{F}}^2 + \lambda\|\mathbf{D}h\|_{2,1} \tag{2}$$

where $\lambda > 0$, $\|\cdot\|_{\mathrm{F}}$ refers to the Frobenius norm, $\mathbf{D} : \mathbb{R}^{N \times N} \to \mathbb{R}^{2 \times (N \times N)}$ computes the horizontal and vertical variations, for $(n_1, n_2) \in \{1, \ldots, N\}^2$:

$$(\mathbf{D}h)_{n_1,n_2} = \begin{pmatrix} h_{n_1,n_2+1} - h_{n_1,n_2} \\ h_{n_1+1,n_2} - h_{n_1,n_2} \end{pmatrix} = \begin{pmatrix} (\mathbf{H}h)_{n_1,n_2} \\ (\mathbf{V}h)_{n_1,n_2} \end{pmatrix} \tag{3}$$

and the isotropic total variation is defined as

$$\|\mathbf{D}h\|_{2,1} = \sum_{n_1=1}^{N-1} \sum_{n_2=1}^{N-1} \sqrt{(\mathbf{H}h)_{n_1,n_2}^2 + (\mathbf{V}h)_{n_1,n_2}^2}. \tag{4}$$

Problem (2) is convex but non-smooth and it can be solved by using proximal algorithms [9]. Two algorithms are considered: the forward-backward algorithm [13] applied on the dual formulation of (2), and the primal-dual algorithm (cf. [15], [23], [2]). In this context, such algorithms are efficient for medium size images and for small values of the regularization parameter λ. However, their computational cost increases significantly for large value of λ, needed when exponents $\widehat{h}$ are correlated, and for large sample size $N \times N$. Block-coordinate approaches are then worth being explored, as recently introduced for both the DFB and PD algorithms.

2.3. Block-coordinate algorithms

Splitting The TV (global) operator $\mathbf{D}$ serving to compute horizontal and vertical gradients at *each* pixel $(n_1, n_2) \in \{1, \ldots, N\}^2$, can be split into sub-operators $\mathbf{D}_\ell$ with $\ell \in \{1, \ldots, L\}$, that compute these gradients at pixels with spatial coordinates belonging to a restricted area $\mathbb{I}_\ell \subset \{1, \ldots, N\}^2$. We have then

$$\mathrm{TV}(h) = \|\mathbf{D}h\|_{2,1} = \sum_{\ell=1}^{L} \|\mathbf{D}_\ell h\|_{2,1}. \tag{5}$$

Block dual forward-backward. Proposed in [19], it leads to Algorithm 1 when customized to Problem (2). Convergence of the sequence $(h^{[k]})_{k \in \mathbb{N}}$ towards the unique solution to (2) is secured if

$$(\forall \ell \in \{1, \ldots, L\}) \quad 0 < \gamma_\ell < 2/\|\mathbf{D}_\ell\|^2, \tag{6}$$

where $\|\mathbf{D}_\ell\|$ designates the spectral norm of $\mathbf{D}_\ell$.

Algorithm 1: Block dual forward-backward

Data: $\forall \ell \in \{1, \ldots, L\}, y_\ell^{[0]} \in \mathbb{R}^{N^2/2}$,
  $h^{[0]} = \widehat{h} - \sum_{\ell=1}^{L} \mathbf{D}_\ell^* y_\ell^{[0]}$
for $k \in \mathbb{N}$ **do**
  for $\ell \in \{1, \ldots, L\}$ **do**
  $\quad \tilde{y}_\ell^{[k]} = y_\ell^{[k]} + \gamma_\ell \mathbf{D}_\ell h^{[k]}$
  $\quad y_\ell^{[k+1]} = \tilde{y}_\ell^{[k]} - \gamma_\ell \mathrm{prox}_{\gamma_\ell^{-1}\lambda\|\cdot\|_{2,1}}\left(\gamma_\ell^{-1}\tilde{y}_\ell^{[k]}\right)$
  $\quad h^{[k+1]} = h^{[k]} - \mathbf{D}_\ell^*\left(y_\ell^{[k+1]} - y_\ell^{[k]}\right)$

Block primal-dual algorithm. Proposed in [23], it leads to Algorithm 2 when customized to Problem (2), with p_ℓ the probability (unchanged from one iteration to another) that block $y_\ell^{[k]}$ of the dual variable is updated at iteration k. The sequence $(h^{[k]})_{k \in \mathbb{N}}$ defined in Algorithm 2 converges in an ergodic expected sense towards the solution to Problem (2) if

$$(\forall \ell \in \{1, \ldots, L\}) \qquad \sigma_\ell \tau < p_\ell/\|\mathbf{D}_\ell\|^2. \tag{7}$$

3. CHOICE OF THE BLOCKS

Computational cost decrease is expected from splitting the dual variables $y^{[k]}$ into blocks. Moreover, the speed of convergence of the iterates $(h^{[k]})_{k \in \mathbb{N}}$ toward the solution $\widehat{\widehat{h}}$ is controlled by the descent steps γ_ℓ for the block DFB in Alg. 1, and by σ_ℓ and τ for the block PD in Alg. 2. The larger the descent step, the faster $(h^{[k]})_{k \in \mathbb{N}}$ is expected to converge to $\widehat{\widehat{h}}$. The choice of these parameters fulfills (6) for Alg. 1 and (7) for Alg. 2, hence the smaller $\|\mathbf{D}_\ell\|$, the larger γ_ℓ and σ_ℓ. The goal is thus to build sub-operators $\mathbf{D}_\ell$ having spectral norms as small as possible.

Algorithm 2: Block primal-dual algorithm.

Data: $\forall \ell \in \{1, \ldots, L\}$, $y_\ell^{[0]} \in \mathbb{R}^{N^2/2}$, $\overline{y}_\ell^{[0]} \in \mathbb{R}^{N^2/2}$,
$\quad h^{[0]} \in \mathbb{R}^{N^2}$

for $k \in \mathbb{N}^*$ **do**

$\quad$ **Choose** $\ell \in \{1, \ldots, L\}$ with probability p_ℓ,

$\quad h^{[k+1]} = \left(h^{[k]} - \tau \sum_{r=1}^{L} \mathbf{D}_r^* \overline{y}_r^{[k]} + \tau \widehat{h} \right) / (1 + \tau)$

$\quad \tilde{y}_\ell^{[k]} = y_\ell^{[k]} + \sigma_\ell \mathbf{D}_\ell h^{[k+1]}$

$\quad y_\ell^{[k+1]} = \tilde{y}_\ell^{[k]} - \sigma_\ell \text{prox}_{\sigma_\ell^{-1} \lambda \|.\|_{2,1}} \left(\sigma_\ell^{-1} \tilde{y}_\ell^{[k]} \right)$

$\quad \overline{y}_\ell^{[k+1]} = y_\ell^{[k+1]} + \frac{1}{p_\ell} \left(y_\ell^{[k+1]} - y_\ell^{[k]} \right)$

$\quad \overline{y}_r^{[k+1]} = \overline{y}_r^{[k]}$ if $r \neq \ell$

3.1. General principle

Since 2D gradient operators **H** and **V** have more intricate structures, we propose first to illustrate the splitting strategies on the toy example of the 1D difference operator. Let $\mathbf{A} \in \mathbb{R}^{K \times I}$ denote a matrix corresponding to this operator and let $z = \mathbf{A}x$ with $x \in \mathbb{R}^I$. Sub-operator $\mathbf{A}_\ell$ computes selected variables $\{z_{k_\ell}, k_\ell \in \mathbb{I}_\ell\}$, where $(\mathbb{I}_\ell)_{1 \le \ell \le L}$ forms a partition of $\{1, \ldots, K\}$. This is equivalent to selecting K_ℓ rows of the matrix **A** so as to built sub-matrix $\mathbf{A}_\ell \in \mathbb{R}^{K_\ell \times I}$, K_ℓ being the cardinality of $\mathbb{I}_\ell$.

Example 1. Let $\mathbf{A} = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & -1 \end{pmatrix} \in \mathbb{R}^{4 \times 4}$. One can easily check that $\|\mathbf{A}\| = 2$ and we aim to split **A** into submatrices $\mathbf{A}_\ell$ with lower norms. Table 1 compares two types of splitting and the resulting norms of sub-operators. The second splitting obviously yields a lower operator norm for each sub-operator and is a good candidate for lattice splitting as explained in the next section.

Split 1. (Successive lines)	Split 2. (Alternating lines)
$\mathbf{A}_1 = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \end{pmatrix}$	$\mathbf{A}_1 = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}$
$\mathbf{A}_2 = \begin{pmatrix} 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & -1 \end{pmatrix}$	$\mathbf{A}_2 = \begin{pmatrix} 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & -1 \end{pmatrix}$
$\|\mathbf{A}_\ell\| = \sqrt{3}, \quad \ell \in \{1, 2\}$	$\|\mathbf{A}_\ell\| = \sqrt{2}, \quad \ell \in \{1, 2\}$

Table 1: Comparison of two basic splitting strategies.

3.2. Specificity of Problem (2)

To built the sub-operators $\mathbf{D}_\ell$ in 2D we specify pixels at which the horizontal and vertical gradients are computed, in other words $\mathbf{D}_\ell$ evaluates the differences only at pixels in $\mathbb{I}_\ell$. Duality implies that this is equivalent to split the dual variables into blocks (cf. Fig. 1).

Regions. A first possible choice is to split the analyzed image into *regions*, following [21]. This is illustrated for $L = 4$ blocks in Fig. 1 (left) and the dual variables partition for region splitting is:

$$\begin{cases} \mathbb{I}_1 = \{1, \ldots, N/2\} \times \{1, \ldots, N/2\} \\ \mathbb{I}_2 = \{1, \ldots, N/2\} \times \{N/2 + 1, \ldots, N\} \\ \mathbb{I}_3 = \{N/2 + 1, \ldots, N\} \times \{1, \ldots, N/2\} \\ \mathbb{I}_4 = \{N/2 + 1, \ldots, N\} \times \{N/2 + 1, \ldots, N\}. \end{cases} \quad (8)$$

Lattices. We propose here an alternative splitting into sub-lattices, illustrated in Fig. 1 (right). This splitting drastically changes the
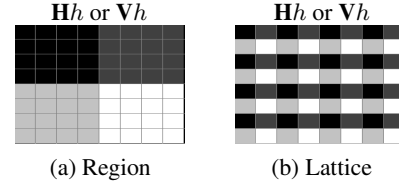

(a) Region $\qquad$ (b) Lattice

Fig. 1: Two different splittings for block coordinate strategy for $N = 8$ and thus a dual variable of size $2 \times 8 \times 8$, corresponding to **H**h and **V**h. $\mathbb{I}_1$ is in black, $\mathbb{I}_2$ is in dark grey, $\mathbb{I}_3$ is in light grey, and $\mathbb{I}_4$ is in white.

structure of the difference sub-operators $\mathbf{D}_\ell$, which results in a smaller sub-operator norm, hence allowing larger descent steps and thus leading to faster convergence. The dual variables partitioning into $L = 4$ blocks is given by

$$\begin{cases} \mathbb{I}_1 = \left\{ (2i_1 + 1, 2i_2 + 1) \mid (i_1, i_2) \in \{0, \ldots, N/2 - 1\}^2 \right\} \\ \mathbb{I}_2 = \left\{ (2i_1 + 1, 2i_2 + 2) \mid (i_1, i_2) \in \{0, \ldots, N/2 - 1\}^2 \right\} \\ \mathbb{I}_3 = \left\{ (2i_1 + 2, 2i_2 + 1) \mid (i_1, i_2) \in \{0, \ldots, N/2 - 1\}^2 \right\} \\ \mathbb{I}_4 = \left\{ (2i_1 + 2, 2i_2 + 2) \mid (i_1, i_2) \in \{0, \ldots, N/2 - 1\}^2 \right\}. \end{cases} \quad (9)$$

3.3. Region versus lattice

Operator norm. Table 2 compares both the computational complexity and the norms of the difference operators for the global, region and lattice splitting cases. It shows that region splitting does not yield sub-operators with lower norms. In contrast, the proposed lattice splitting yields sub-operators with a significantly decreased norm, while having a small complexity.

Operator	Global $\mathbf{D}$	Region $\mathbf{D}_\ell$	Lattice $\mathbf{D}_\ell$
Complex.	$2N^2 - 2N$	$N^2/2$	$N^2/2$
Adj. Compl.	$3N^2 - 2N$	$3N^2/4$	$N^2/4$
Norm $\| \cdot \|$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{3}/2$

Table 2: Norms and complexity (for an $N \times N$ image) when applying (sub-)operators and their adjoints.

Complexity and memory. For images of size $N \times N$ pixels, Table 3 compares: (i) the number of elementary operations (addition, subtraction, multiplication or division of *non-zero coefficients*, allocations are not taken into account) for one iteration as well as (ii) the working memory (defined as the total number of stored coefficients). Various algorithms are benchmarked. First the standard dual forward-backward algorithm (DFB), the region-block DFB algorithm (R-DFB) and the lattice-block DFB algorithm (L-DFB) (cf. Alg. 1 for block version). Second, the standard primal-dual algorithm (PD), the lattice-block PD algorithm (L-PD, cf. Alg. 2 where $(\forall \ell) \; p_\ell = 1/4$) and what we call the double lattice-block PD algorithm (DL-PD), proposed in [23]. The latter algorithm constitutes an alternative to Alg. 2 in which two blocks are updated at each iteration with $p_1 = p_3 = 3/5$ and $p_2 = p_4 = 2/5$.

Table 3 shows that, among the class of DFB algorithms, L-DFB benefits both from the lowest complexity per iteration and the largest descent step γ_ℓ. We thus expect that L-DFB will converge faster than R-DFB and DFB. Concerning the class of PD algorithms the situation is more complex since the lowest complexity is achieved in L-PD but DL-PD permits larger descent steps. Moreover, the resulting gains in terms of memory and step values are quite limited. Thus, PD algorithms should be more complicated to accelerate.

	Descent step(s)	Complexity	Memory
DFB	$\gamma < 1$	$24N^2 - 4N$	$5N^2$
R-DFB	$\gamma_\ell < 1$	$13N^2/2$	$\mathbf{7N^2/2}$
L-DFB	$\gamma_\ell < \mathbf{8/3}$	$\mathbf{6N^2}$	$\mathbf{7N^2/2}$
PD	$\sigma\tau < 1/2$	$24N^2 - 4N$	$6N^2$
L-PD	$\sigma_\ell\tau < 1/3$	$\mathbf{63N^2/4 - 2N}$	$\mathbf{9N^2/2}$
DL-PD	$\sigma_\ell\tau < \mathbf{8/15}, \ell \in \{2,4\}$	$39N^2/2 - 2N$	$5N^2$
	$\sigma_\ell\tau < \mathbf{4/5}, \ell \in \{1,3\}$		

Table 3: Conditions on descent step(s), complexity per iteration, and memory used for the studied algorithms (images of size $N \times N$). Best results (largest descent step, and lowest complexity and memory) for each type of algorithms are highlighted.

4. APPLICATION TO SYNTHETIC TEXTURES

Synthetic textures. To mimic real textured images, such as e.g. mammography, composed of two regions (e.g. one corresponding to wealthy tissues and the other to cancerous tissues, cf. [1]), while having ground truth, synthetic texture images of size 256×256 are generated, using realistic random field models. They consist of $Q = 2$ piecewise constant local regularity areas, with $h = 0.7$ for the central area and $h = 0.5$ for the background (cf. Fig. 2). A local estimation $\widehat{h}_n$ is conducted as in Section 2.1, using scales $2^2 \leq 2^j \leq 2^4$, followed by a TV-denoising estimation $\widehat{\widehat{h}}$, as proposed in (2). Performances reported here are obtained as averages across 10 independent realizations.

Performance comparisons. Performance of the different algorithms compared here are quantified in terms of *duality gap* δ:

$$\delta(h,y) = \frac{1}{2}\|h - \widehat{h}\|_2^2 + \|\mathbf{D}h\|_{2,1} + \frac{1}{2}\|-\mathbf{D}^*y + \widehat{h}\|_2^2 + \iota_{2,\infty(\lambda)}(y)$$

where $\iota_{2,\infty(\lambda)}$ is the indicator function of the ball of radius λ for the $\|.\|_{2,\infty}$ norm. In Problem (2), the duality gap $\delta(h^{[k]}, y^{[k]})$ tends to zero as $h^{[k]} \to \widehat{\widehat{h}}$ and $y^{[k]}$ tends to the solution to the dual problem of (2) (cf. [9] for further details). Convergence speeds for the duality gap are plotted in Fig. 3 as functions of the computational time (left) and the number of operations actually executed (right). The threshold $\delta = 1$ (Fig. 3, black solid line) indicates the value below which convergence can be considered as satisfactory, the output image displaying sharp edges which are consistent with our segmentation goal.

Dual forward-backward. L-DFB algorithm (Fig. 3, top row, red-mixed lines) yields a significant acceleration in time (top-left) and number of elementary operations (top-right) compared to DFB algorithms (Fig.3, top row, blue-dashed lines), while no acceleration is obtained for R-DFB algorithm (Fig. 3, top row, solid-green lines).

Primal-dual. It turns out to be far more difficult to accelerate primal-dual algorithms. The L-PD algorithm (Fig. 3, bottom row, red-mixed lines) yields no acceleration with respect to the standard PD algorithm (Fig.3, bottom row, blue-dashed lines). Yet, the DL-PD (Fig.3, bottom row, yellow-solid lines) algorithm appears to be a bit faster than the L-PD one.

DFB Vs PD. Among all DFB and PD benchmarked algorithms, the fastest turns out to be the L-DFB algorithm (Fig. 3, top row, red-mixed lines), that benefits simultaneously from a gain in complexity per iteration and larger descent step.
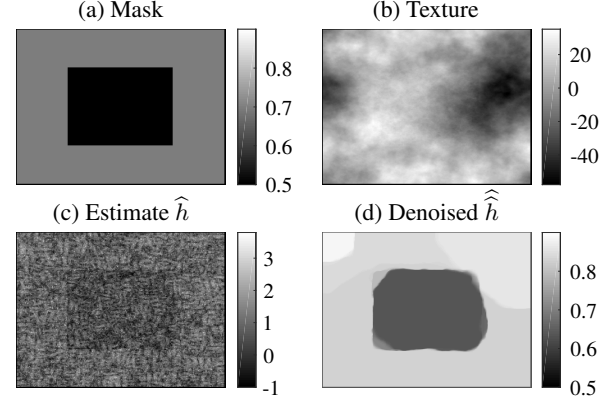

Fig. 2: (a) Prior texture areas: in the black area Ω_1, $h = 0.5$, in the gray area Ω_2, $h = 0.7$. (b) Synthetic texture generated by fBm with two textures corresponding to the prior mask. (c) Local $\widehat{h}$ estimated by linear regression (cf. Sec. 2.1). (d) TV-denoised estimate $\widehat{\widehat{h}}$ obtained with a duality gap $\delta = 1$.
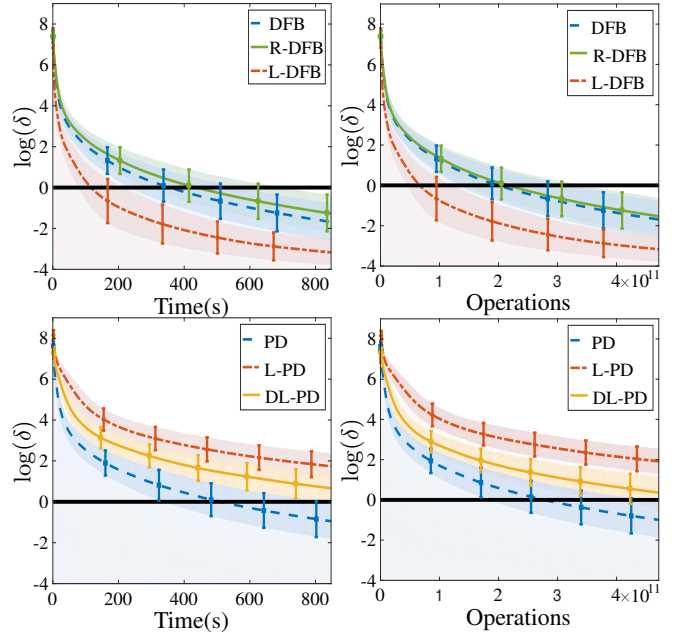

Fig. 3: Convergence to zero of the duality gap (for 10 realizations) as a function of computational cost (left) and of the number of elementary operations (right). Top line: For global (DFB), region-block (R-DFB) and lattice-block (L-DFB) DFB algorithms. Bottom line: For global (PD), lattice-block (L-PD) and double lattice-block (DL-PD) PD algorithms. Duality gap $\delta = 1$ is marked by the solid black line.

5. CONCLUSION

All block-coordinate algorithms proposed here yield significant gains in memory cost, which makes possible to process large size images. In addition, the lattice splitting developed here leads to a significant acceleration of the dual forward-backward algorithm for TV denoising. Applying the lattice splitting strategy to primal-dual algorithms is less effective. This suggests that, in order to accelerate TV-denoising algorithms, larger descent steps and lower complexity per iteration are simultaneously needed. Moreover, further acceleration can be expected from a parallel implementation of the L-DFB algorithm.

6. REFERENCES

[1] Z. Marin, K. A. Batchelder, B. C. Toner, L. Guimond, E. Gerasimova-Chechkina, A. R. Harrow, A. Arneodo, and A. Khalil, "Mammographic evidence of microenvironment changes in tumorous breasts," *Medical physics*, vol. 44, no. 4, pp. 1324–1336, 2017.

[2] M. J. Ehrhardt, P. Markiewicz, A. Chambolle, P. Richtárik, J. Schott, and C.-B. Schönlieb, "Faster PET reconstruction with a stochastic primal-dual hybrid gradient method," in *Wavelets and Sparsity XVII*, San Diego, California, United States, 2017, vol. 10394, p. 103941O, International Society for Optics and Photonics.

[3] J. Yuan, D. Wang, and A. M. Cheriyadat, "Factorization-based texture segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3488–3497, Nov. 2015.

[4] E. Gerasimova, B. Audit, S. G. Roux, A. Khalil, O. Gileva, F. Argoul, O. Naimark, and A. Arneodo, "Wavelet-based multifractal analysis of dynamic infrared thermograms to assist in early breast cancer diagnosis," *Frontiers in Physiology*, vol. 5, pp. 176, May 2014.

[5] N. Pustelnik, H. Wendt, P. Abry, and N. Dobigeon, "Combining local regularity estimation and total variation optimization for scale-free texture segmentation," *IEEE Trans. Computational Imaging*, vol. 2, no. 4, pp. 468–479, 2016.

[6] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-point algorithms for inverse problems in science and engineering*, pp. 185–212. Springer, 2011.

[7] P. L. Combettes, L. Condat, J.-C. Pesquet, and B.C. Vũ, "A forward-backward view of some primal-dual optimization methods in image recovery," in *Proc. Int. Conf. Image Process.*, Paris, France, Oct. 27-30, 2014, IEEE, pp. 4141–4145.

[8] N. Komodakis and J.-C. Pesquet, "Playing with duality: An overview of recent primal dual approaches for solving large-scale optimization problems," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 31–54, Nov. 2015.

[9] H. H. Bauschke and P. L. Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*, Springer, New York, second edition, 2017.

[10] H. Wendt, S. G. Roux, P. Abry, and S. Jaffard, "Wavelet leaders and bootstrap for multifractal analysis of images," *Signal Process.*, vol. 89, no. 6, pp. 1100–1114, 2009.

[11] A. Chambolle, "An algorithm for total variation minimization and applications," *J. Math. Imag. Vis.*, vol. 20, no. 1-2, pp. 89–97, Jan. 2004.

[12] Z Harchaoui, A. Juditsky, and A. Nemirovski, "Conditional gradient algorithms for machine learning," in *Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, Lake Tahoe, Nevada, United States, Dec. 8, 2012, vol. 3, pp. 3–2.

[13] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Model. and Simul.*, vol. 4, no. 4, pp. 1168–1200, 2005.

[14] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problem with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, May 2011.

[15] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optim. Theory Appl.*, vol. 158, no. 2, pp. 460–479, 2013.

[16] F. Iutzeler, P. Bianchi, P. Ciblat, and W. Hachem, "Asynchronous distributed optimization using a randomized alternating direction method of multipliers," in *52nd IEEE Conference on Decision and Control*, Firenze, Italy, Dec. 10-13, 2013, pp. 3671–3676.

[17] A. Chambolle and C. Dossal, "On the convergence of the iterates of "FISTA"," *J. Optim. Theory Appl.*, vol. 166, no. 3, pp. 25, Aug. 2015.

[18] P. L. Combettes and J.-C. Pesquet, "Stochastic quasi-Fejér block-coordinate fixed point iterations with random sweeping," *SIAM J. Opt.*, vol. 25, no. 2, pp. 1221–1248, 2015.

[19] F. Abboud, E. Chouzenoux, J.-C. Pesquet, J.-H. Chenot, and L. Laborelli, "Dual block-coordinate forward-backward algorithm with application to deconvolution and deinterlacing of video sequences," *J. Math. Imag. Vis.*, vol. 59, no. 3, pp. 415–431, Nov. 2017.

[20] O. Fercoq and P. Bianchi, "A coordinate descent primal-dual algorithm with large step size and possibly non separable functions," *preprint arXiv:1508.04625*, 2015.

[21] A. Repetti, E. Chouzenoux, and J.-C. Pesquet, "A parallel block-coordinate approach for primal-dual splitting with arbitrary random block selection," in *European Signal Processing Conference*, Nice, France, Aug. 31-Sept. 4, 2015, pp. 235–239.

[22] G. Chierchia, N. Pustelnik, and J.-C. Pesquet, "Random primal-dual proximal iterations for sparse multiclass SVM," in *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*, Vietri sul Mare, Italy, Sept. 13-16, 2016, pp. 1–6, IEEE.

[23] A. Chambolle, M. J. Ehrhardt, P Richtárik, and C.-B. Schönlieb, "Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging application," *preprint arXiv:1706.04957*, 2017.

[24] A. Chambolle and T. Pock, "A remark on accelerated block coordinate descent for computing the proximity operators of a sum of convex functions," *SMAI Journal of Computational Mathematics*, vol. 1, pp. 29–54, Jan. 2015.