



HAL
open science

Sparse Wavelet Representations of Spatially Varying Blurring Operators

Paul Escande, Pierre Weiss

► **To cite this version:**

Paul Escande, Pierre Weiss. Sparse Wavelet Representations of Spatially Varying Blurring Operators. 2014. hal-00971549v6

HAL Id: hal-00971549

<https://hal.science/hal-00971549v6>

Preprint submitted on 9 Oct 2015

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

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

Sparse Wavelet Representations of Spatially Varying Blurring Operators.

Paul Escande [†] Pierre Weiss [‡]

October 11, 2015

Abstract

Restoring images degraded by spatially varying blur is a problem encountered in many disciplines such as astrophysics, computer vision or biomedical imaging. One of the main challenges to perform this task is to design efficient numerical algorithms to approximate integral operators.

We introduce a new method based on a sparse approximation of the blurring operator in the wavelet domain. This method requires $\mathcal{O}(N\epsilon^{-d/M})$ operations to provide ϵ -approximations, where N is the number of pixels of a d -dimensional image and $M \geq 1$ is a scalar describing the regularity of the blur kernel. In addition, we propose original methods to define sparsity patterns when only the operators regularity is known.

Numerical experiments reveal that our algorithm provides a significant improvement compared to standard methods based on windowed convolutions.

Keywords: Image deblurring, spatially varying blur, integral operator approximation, wavelet compression, windowed convolution

1 Introduction

The problem of image restoration in the presence of spatially varying blur appears in many domains. Examples of applications in computer vision, biomedical imaging and astronomy are shown in Figures 1 and 2 respectively. In this paper, we propose new solutions to address one of the main difficulties associated to this problem: the computational evaluation of matrix-vector products.

A spatially variant blurring operator can be modelled as a linear operator and therefore be represented by a matrix \mathbf{H} of size $N \times N$, where N represents the number of pixels of

[†]Département Mathématiques, Informatique, Automatique (DMIA), Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), Toulouse, France, paul.escande@gmail.com

[‡]Institut des Technologies Avancées en Sciences du Vivant, ITAV-USR3505 and Institut de Mathématiques de Toulouse, IMT-UMR5219, CNRS and université de Toulouse, Toulouse, France, pierre.armand.weiss@gmail.com

a d -dimensional image. Sizes of typical images range from $N = 10^6$ for small 2D images, to $N = 10^{10}$ for large 2D or 3D images. Storing matrices and computing matrix-vector products using the standard representation is impossible for such sizes: it amounts to tera or exabytes of data/operations. In cases where the Point Spread Functions (PSF) supports are sufficiently small in average over the image domain, the operator can be coded as a sparse matrix and be applied using traditional approaches. However, in many practical applications this method turns out to be too intensive and cannot be applied with decent computing times. This may be due to i) large PSFs supports or ii) the need for super-resolution applications where the PSFs sizes increase with the resolution. Spatially varying blurring matrices therefore require the development of computational tools to compress them and evaluate them in an efficient way.

Existing approaches

To the best of our knowledge, the first attempts to address this issue appeared at the beginning of the seventies (see e.g. [41]). Since then, many techniques were proposed. We describe them briefly below

Composition of diffeomorphisms and convolutions One of the first method proposed to reduce the computational complexity, is based on first applying a diffeomorphism to the image domain [41, 42, 34, 45, 20] followed by a convolution using FFTs and an inverse diffeomorphism. The diffeomorphism is chosen in order to transform the spatially varying blur into an invariant one. This approach suffers from two important drawbacks:

- first it was shown that not all spatially varying kernel can be approximated by this approach [34],
- second, this method requires good interpolation methods and the use of Euclidean grids with small grid size in order to correctly estimate integrals.

Separable approximations Another common idea is to approximate the kernel of the operator by a separable one that operates in only one dimension. The computational complexity of a product is thus reduced to d applications of one-dimensional operators. It drastically improves the performance of algorithms. For instance, in 3D fluorescence microscopy, the authors of [39, 32, 4, 50] proposed to approximate PSFs by anisotropic Gaussians and assumed that the Gaussian variances only vary along one direction (e.g., the direction of light propagation). The separability assumption implies that both the PSF and its variations are separable. Unfortunately, most physically realistic PSFs are not separable and do not vary in a separable manner (see e.g., Figure 3). This method is therefore usually too crude.

Wavelet or Gabor multipliers Some works [9, 19, 21, 29] proposed to approximate blurring operators \mathbf{H} using operators diagonal in wavelet bases, wavelet packet or Gabor frames. This idea consists of defining an approximation $\tilde{\mathbf{H}}$ of kind $\tilde{\mathbf{H}} = \Psi \Sigma \Psi^*$, where Ψ^* and Ψ are wavelet or Gabor transforms and Σ is a diagonal matrix. These diagonal approximations mimic the fact that shift-invariant operators are diagonal in the Fourier domain. These approaches lead to fast $\mathcal{O}(N)$ or $\mathcal{O}(N \log(N))$ algorithms to compute matrix-vector products. In [19], we proposed to deblur images using diagonal approximations of the blurring operators in redundant wavelet packet bases. This approximation was shown to be fast and efficient in deblurring images when the exact operator was scarcely known or in high noise levels. It is however too coarse for applications with low noise levels. This approach seems however promising. Gabor multipliers are considered the state-of-the-art for 1D signals in OFDM systems for instance (slowly varying smoothing operators).

Weighted convolutions Probably the most commonly used approaches consist of approximating the integral kernel by spatially weighted sum of convolutions. Among these approaches two different ideas have been explored. The first one will be called *windowed convolutions* in this paper and appeared in [36, 37, 25, 28, 16]. The second one was proposed in [22] and consists of expanding the PSFs in a common basis of small dimensionality.

Windowed convolutions consists of locally stationary approximations of the kernel. We advise the reading of [16] for an up-to-date description of this approach and its numerous refinements. The main idea is to decompose the image domain into subregions and perform a convolution on each subregion. The results are then gathered together to obtain the blurred image. In its simplest form, this approach consists in partitioning the domain Ω in squares of equal sizes. More advanced strategies consist in decomposing the domain with overlapping subregions. The blurred image can then be obtained by using windowing functions that interpolate the kernel between subregions (see, e.g., [36, 28, 16]). Various methods have been proposed to interpolate the PSF. In [28], a linear interpolation is performed, and in [16] higher order interpolation of the PSF are handled.

Sparse wavelet approximations The approach studied in this paper was proposed recently and independently in [48, 49, 18]. The main idea is to represent the operator in the wavelet domain by using a change of basis. This change of basis, followed by a thresholding operation allows sparsifying the operator and use sparse matrix-vector products. The main objective of this work is to provide solid theoretical foundations to these approaches.

1.1 Contributions of the paper

Our first contribution is the design of a new approach based on sparse approximation of \mathbf{H} in the wavelet domain. Using techniques initially developed for pseudo-differential operators [5, 35], we show that approximations $\tilde{\mathbf{H}}$ satisfying $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2} \leq \epsilon$, can be obtained with

this new technique, in no more than $\mathcal{O}\left(N\epsilon^{-d/M}\right)$ operations. In this complexity bound, $M \geq 1$ is an integer that describes the smoothness of the blur kernel.

Controlling the spectral norm is usually of little relevance in image processing. Our second contribution is the design of algorithms that iteratively construct sparse matrix patterns adapted to the structure of images. These algorithms rely on the fact that both natural images and operators can be compressed simultaneously in the same wavelet basis.

As a third contribution, we propose an algorithm to design a generic sparsity structure when only the operators regularity is known. This paves the way to the use of wavelet based approaches in blind deblurring problems where operators need to be inferred from the data.

We finish the paper by numerical experiments. We show that the proposed algorithms allow significant speed ups compared to some windowed convolutions based methods.

Let us emphasize that the present paper is a continuation of our recent contribution [18]. The main evolution is that i) we provide all the theoretical foundations of the approach with precise hypotheses, ii) we propose a method to automatically generate adequate sparsity patterns and iii) we conduct a thorough numerical analysis of the method.

1.2 Outline of the paper

The outline of this paper is as follows. We introduce the notation used throughout the paper in Section 2. We propose an original mathematical description of blurring operators appearing in image processing in Section 3. We introduce the proposed method and analyze its theoretical efficiency Section 4. We then propose various algorithms to design good sparsity patterns in Section 5. Finally, we perform numerical tests to analyze the proposed method and compare it to the standard windowed convolutions based methods in Section 6.

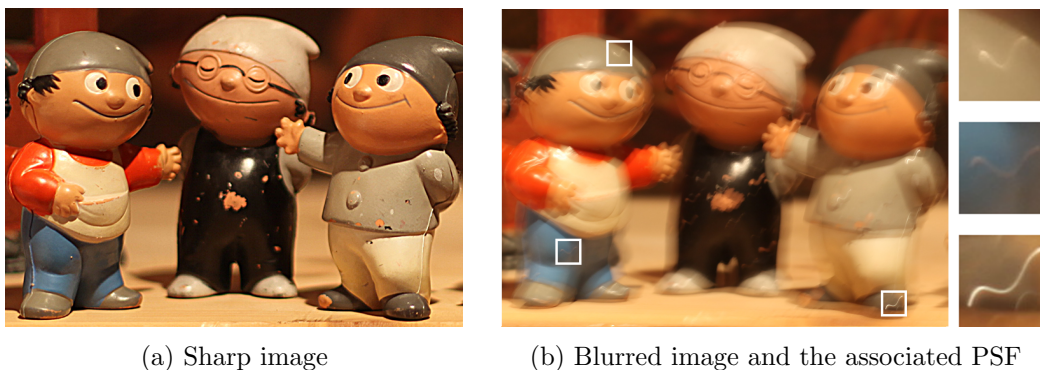


Figure 1: An example in computer vision. Image degraded by spatially varying blur due to a camera shake. Images are from [27] and used here by courtesy of Michael Hirsch.

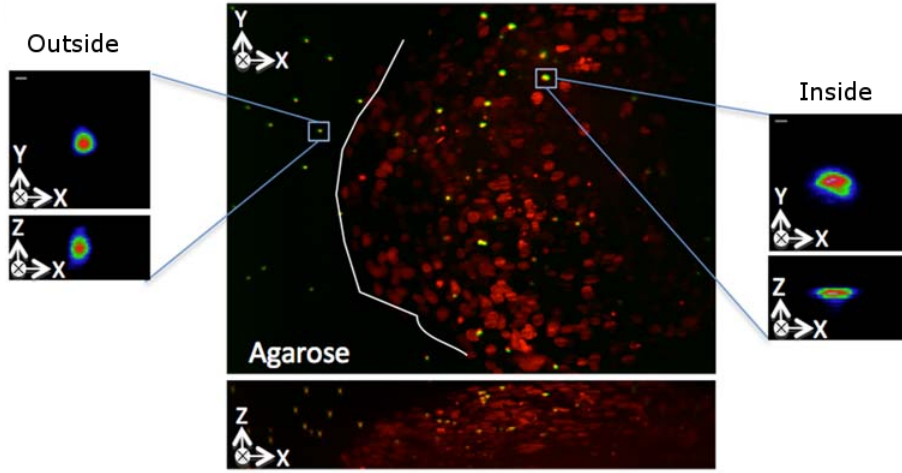


Figure 2: An example in biology. Image of a multicellular tumor spheroid imaged in 3D using Selective Plane Illumination Microscope (SPIM). Fluorescence beads (in green) are inserted in the tumor model and allow the observation of the PSF at different locations. Nuclei are stained in red. On the left-hand-side, 3D PSFs outside the sample are observed. On the right-hand-side, 3D PSFs inside the sample are observed. This image is from [30] and used here by courtesy of Corinne Lorenzo.

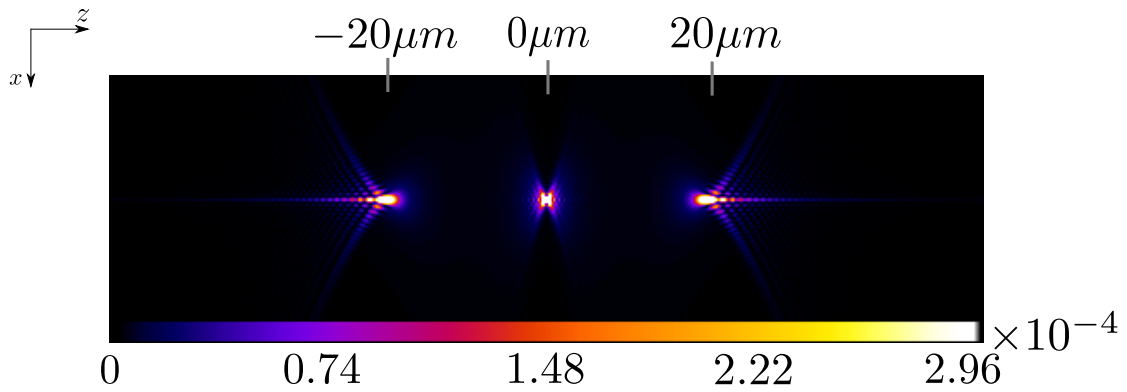


Figure 3: Three PSFs displayed in a XZ plan at different z depths: $-20\mu\text{m}$, $0\mu\text{m}$ and $20\mu\text{m}$. PSFs are generated using Gibson and Lanni 3D optical model from the PSF Generator [31]. The parameters used are $n_i = 1.5$, $n_s = 1.33$, $t_i = 150\mu\text{m}$, $\text{NA} = 1.4$ and a wavelength of 610nm .

2 Notation

In this paper, we consider d dimensional images defined on a domain $\Omega = [0, 1]^d$. The space $\mathbb{L}^2(\Omega)$ will denote the space of squared integrable functions defined on Ω .

Let $\alpha = (\alpha_1, \dots, \alpha_d)$ denote a multi-index. The sum of its components is denoted $|\alpha| = \sum_{i=1}^d \alpha_i$. The Sobolev spaces $W^{M,p}$ are defined as the set of functions $f \in \mathbb{L}^p$ with partial derivatives up to order M in \mathbb{L}^p where $p \in [1, +\infty]$ and $M \in \mathbb{N}$. These spaces, equipped with the following norm are Banach spaces

$$\|f\|_{W^{M,p}} = \|f\|_{\mathbb{L}^p} + |f|_{W^{M,p}}, \quad \text{where,} \quad |f|_{W^{M,p}} = \sum_{|\alpha|=M} \|\partial^\alpha f\|_{\mathbb{L}^p}. \quad (1)$$

In this notation, $\partial^\alpha f = \frac{\partial^{\alpha_1}}{\partial x_1^{\alpha_1}} \dots \frac{\partial^{\alpha_d}}{\partial x_d^{\alpha_d}} f$.

Let X and Y denote two metric spaces endowed with their respective norms $\|\cdot\|_X$ and $\|\cdot\|_Y$. In all the paper $H : X \rightarrow Y$ will denote a linear operator and H^* its adjoint operator. The subordinate operator norm is defined by

$$\|H\|_{X \rightarrow Y} = \sup_{x \in X, \|x\|_X=1} \|Hx\|_Y.$$

The notation $\|H\|_{p \rightarrow q}$ corresponds to the case where X and Y are endowed with the standard \mathbb{L}^p and \mathbb{L}^q norms. In all the paper, operators acting in a continuous domain are written in plain text format H . Finite dimensional matrices are written in bold fonts \mathbf{H} . Approximation operators will be denoted \tilde{H} in the continuous domain or $\tilde{\mathbf{H}}$ in the discrete domain.

In this paper we consider a compactly supported wavelet basis of $\mathbb{L}^2(\Omega)$. We first introduce wavelet basis of $\mathbb{L}^2([0, 1])$. We let ϕ and ψ denote the scaling and mother wavelets. We assume that the mother-wavelet ψ has M vanishing moments, i.e.

$$\text{for all } 0 \leq m < M, \quad \int_{[0,1]} t^m \psi(t) dt = 0.$$

We assume that $\text{supp}(\psi) = [-c(M)/2, c(M)/2]$. Note that $c(M) \geq 2M - 1$, with equality for Daubechies wavelets, see, e.g., [33, Theorem 7.9, p. 294].

We define translated and dilated versions of the wavelets for $j \geq 0$ as follows

$$\begin{aligned} \phi_{j,l} &= 2^{j/2} \phi(2^j \cdot - l), \\ \psi_{j,l} &= 2^{j/2} \psi(2^j \cdot - l), \end{aligned} \quad (2)$$

with $l \in \mathcal{T}_j$ and $\mathcal{T}_j = \{0, \dots, 2^j - 1\}$.

In dimension d , we use separable wavelet bases, see, e.g., [33, Theorem 7.26, p. 348]. Let $m = (m_1, \dots, m_d)$. Define $\rho_{j,l}^0 = \phi_{j,l}$ and $\rho_{j,l}^1 = \psi_{j,l}$. Let $e = (e_1, \dots, e_d) \in \{0, 1\}^d$. For ease of reading, we will use the shorthand notation $\lambda = (j, m, e)$. We also denote

$$\Lambda_0 = \left\{ (j, m, e) \mid j \in \mathbb{Z}, m \in \mathcal{T}_j, e \in \{0, 1\}^d \right\}$$

and

$$\Lambda = \left\{ (j, m, e) \mid j \in \mathbb{Z}, m \in \mathcal{T}_j, e \in \{0, 1\}^d \setminus \{0\} \right\}.$$

Wavelet ψ_λ is defined by $\psi_\lambda(x_1, \dots, x_d) = \psi_{j,m}^e(x_1, \dots, x_d) = \rho_{j,m_1}^{e_1}(x_1) \cdots \rho_{j,m_d}^{e_d}(x_d)$. Elements of the separable wavelet basis consist of tensor products of scaling and mother wavelets at the same scale. Note that if $e \neq 0$ wavelet $\psi_{j,m}^e$ has M vanishing moments in \mathbb{R}^d . We let $I_{j,m} = \cup_e \text{supp } \psi_{j,m}^e$ and $I_\lambda = \text{supp } \psi_\lambda$.

We assume that every function $f \in \mathbb{L}^2(\Omega)$ can be written as

$$\begin{aligned} u &= \langle u, \psi_{0,0}^0 \rangle \psi_{0,0}^0 + \sum_{e \in \{0,1\}^d \setminus \{0\}} \sum_{j=0}^{+\infty} \sum_{m \in \mathcal{T}_j} \langle u, \psi_{j,m}^e \rangle \psi_{j,m}^e \\ &= \langle u, \psi_{0,0}^0 \rangle \psi_{0,0}^0 + \sum_{\lambda \in \Lambda} \langle u, \psi_\lambda \rangle \psi_\lambda \\ &= \sum_{\lambda \in \Lambda_0} \langle u, \psi_\lambda \rangle \psi_\lambda \end{aligned}$$

This is a slight abuse since wavelets defined in (2) do not define a Hilbert basis of $\mathbb{L}^2([0, 1]^d)$. There are various ways to define wavelet bases on the interval [12] and wavelets having a support intersecting the boundary should be given a different definition. We stick to these definitions to keep the proofs simple.

We let $\Psi^* : \mathbb{L}^2(\Omega) \rightarrow l^2(\mathbb{Z})$ denote the wavelet decomposition operator and $\Psi : l^2(\mathbb{Z}) \rightarrow \mathbb{L}^2(\Omega)$ its associated reconstruction operator. The discrete wavelet transform is denoted $\Psi : \mathbb{R}^N \rightarrow \mathbb{R}^N$. We refer to [33, 15, 12] for more details on the construction of wavelet bases.

3 Blurring operators and their mathematical properties

3.1 A mathematical description of blurring operators

In this paper, we consider d -dimensional real-valued images defined on a domain $\Omega = [0, 1]^d$, where d denotes the space dimension. We consider a blurring operator $H : \mathbb{L}^2(\Omega) \rightarrow \mathbb{L}^2(\Omega)$ defined for any $u \in \mathbb{L}^2(\Omega)$ by the following integral operator:

$$\forall x \in \Omega, \quad Hu(x) = \int_{y \in \Omega} K(x, y)u(y)dy. \quad (3)$$

The function $K : \Omega \times \Omega \rightarrow \mathbb{R}$ is a kernel that defines the Point Spread Function (PSF) $K(\cdot, y)$ at each location $y \in \Omega$. The image Hu is the blurred version of u . By the Schwartz kernel theorem, a linear operator of kind (3) can represent any linear operator if K is a generalized function. We thus need to determine properties of K specific to

blurring operators that will allow to design efficient numerical algorithms to approximate the integral (3).

We propose a definition of the class of blurring operators below.

Definition 1 (Blurring operators). Let $M \in \mathbb{N}$ and $f : [0, 1] \rightarrow \mathbb{R}_+$ denote a non-increasing bounded function. An integral operator is called a blurring operator in the class $\mathcal{A}(M, f)$ if it satisfies the following properties:

1. Its kernel $K \in W^{M, \infty}(\Omega \times \Omega)$;
2. The partial derivatives of K satisfy:

$$(a) \quad \forall |\alpha| \leq M, \forall (x, y) \in \Omega \times \Omega, \quad |\partial_x^\alpha K(x, y)| \leq f(\|x - y\|_\infty). \quad (4)$$

$$(b) \quad \forall |\alpha| \leq M, \forall (x, y) \in \Omega \times \Omega, \quad |\partial_y^\alpha K(x, y)| \leq f(\|x - y\|_\infty). \quad (5)$$

Let us justify this model from a physical point of view. Most imaging systems satisfy the following properties:

Spatial decay.

The PSFs usually have a bounded support (e.g. motion blurs, convolution with the CCD sensors support) or at least a fast spatial decay (Airy pattern, Gaussian blurs,...). This property can be modelled as property 2a. For instance, the 2D Airy disk describing the PSF due to diffraction of light in a circular aperture satisfies 2a with $f(r) = \frac{1}{(1+r)^4}$ (see e.g. [6]).

PSF smoothness.

In most imaging applications, the PSF at $y \in \Omega$, $K(\cdot, y)$ is smooth. Indeed it is the result of a convolution with the acquisition device impulse response which is smooth (e.g. Airy disk). This assumption motivates inequality (4).

PSFs variations are smooth

We assume that the PSF does not vary abruptly on the image domain. This property can be modelled by inequality (5). It does not hold true in all applications. For instance, when sharp discontinuities occur in the depth maps, the PSFs can only be considered as piecewise regular. This assumption simplifies the analysis of numerical procedures to approximate H . Moreover, it seems reasonable in many settings. For instance, in fluorescence microscopy, the PSF width (or Strehl ratio) mostly depends on the optical thickness, i.e. the quantity of matter laser light has to go through, and this quantity is intrinsically continuous. Even in cases where the PSFs variations are not smooth, the discontinuities locations are usually known only approximately and it seems important to smooth the transitions in order to avoid reconstruction artifacts [2].

Remark 1. A standard assumption in image processing is that the constant functions are preserved by the operator H . This hypothesis ensures that brightness is preserved on the image domain. In this paper we do not make this assumption and thus encompass image formation models comprising blur and attenuation. Handling attenuation is crucial in domains such as fluorescence microscopy.

Remark 2. The above properties are important to derive mathematical theories, but only represent an approximation of real systems. The methods proposed in this paper may be applied even if the above properties are not satisfied and are likely to perform well. It is notably possible to relax the boundedness assumption.

4 Wavelet representation of the blurring operator

In this section, we show that blurring operators can be well approximated by sparse representations in the wavelet domain. Since H is a linear operator in a Hilbert space, it can be written as $H = \Psi\Theta\Psi^*$, where $\Theta : l^2(\mathbb{Z}) \rightarrow l^2(\mathbb{Z})$ is the (infinite dimensional) matrix representation of the blur operator in the wavelet domain. Matrix Θ is characterized by the coefficients:

$$\theta_{\lambda,\mu} = \langle H\psi_\lambda, \psi_\mu \rangle, \quad \forall \lambda, \mu \in \Lambda. \quad (6)$$

In their seminal papers [35, 13, 5], Y. Meyer, R. Coifman, G. Beylkin and V. Rokhlin prove that the coefficients of Θ decrease fastly away from its diagonal for a large class of pseudo-differential operators. They also show that this property allows to design fast numerical algorithms to approximate H , by thresholding Θ to obtain a sparse matrix. In this section, we detail this approach precisely and adapt it to the class of blurring operators.

This section is organized as follows: first, we discuss the interest of approximating H in a wavelet basis rather than using the standard discretization. Second, we provide various theoretical results concerning the number of coefficients necessary to obtain an ϵ -approximation of H .

4.1 Discretization of the operator by projection

The proposed method relies on a Galerkin discretization of H . The main idea is to use a projection on a finite dimensional linear subspace $V_q = \text{Span}(\varphi_1, \dots, \varphi_q)$ of $\mathbb{L}^2(\Omega)$ where $(\varphi_1, \varphi_2, \dots)$ is an orthonormal basis of $\mathbb{L}^2(\Omega)$. We define a projected operator H_q by $H_q u = P_{V_q} H P_{V_q} u$, where P_{V_q} is the projector on V_q . We can associate a $q \times q$ matrix Θ to this operator defined by $\Theta = (\langle H\varphi_i, \varphi_j \rangle)_{1 \leq i, j \leq q}$.

It is very common in image processing to assume that natural images belong to functional spaces containing functions with some degree of regularity. For instance, images are often assumed to be of bounded total variation [40]. This hypothesis implies that

$$\|u - P_{V_q} u\|_2 = \mathcal{O}(q^{-\alpha}) \quad (7)$$

for a certain $\alpha > 0$. For instance, in 1D, if $(\varphi_1, \varphi_2, \dots)$ is a wavelet or a Fourier basis and $u \in H^1(\Omega)$ then $\alpha = 2$. For $u \in BV(\Omega)$ (the space of bounded variation functions), $\alpha = 1$ in 1D and $\alpha = 1/2$ in 2D [33, 38].

Moreover, if we assume that H is a regularizing operator, meaning that $\|Hu - P_{V_q}Hu\|_2 = \mathcal{O}(q^{-\beta})$ with $\beta \geq \alpha$ for all u satisfying (7), then we have:

$$\begin{aligned} & \|Hu - H_q u\|_2 \\ &= \|Hu - P_{V_q}H(u + P_{V_q}u - u)\|_2 \\ &\leq \|Hu - P_{V_q}Hu\|_2 + \|P_{V_q}H\|_{2 \rightarrow 2} \|P_{V_q}u - u\|_2 \\ &= \mathcal{O}(q^{-\alpha}). \end{aligned}$$

This simple analysis shows that under mild assumptions, the Galerkin approximation of the operator converges and that the convergence rate can be controlled. The situation is not as easy for standard discretization using finite elements for instance (see, e.g., [47, 3] where a value $\alpha = 1/6$ is obtained in 2D for BV functions, while the simple analysis above leads to $\alpha = 1/2$).

4.2 Discretization by projection on a wavelet basis

In order to get a representation of the operator in a finite dimensional setting, we truncate the wavelet representation at scale J . This way, we obtain an operator \tilde{H} acting on a space of dimension N , where $N = 1 + \sum_{j=0}^{J-1} (2^d - 1)2^{dj}$ denotes the numbers of wavelets kept to represent images.

After discretization, it can be written in the following convenient form:

$$\mathbf{H} = \mathbf{\Psi} \mathbf{\Theta} \mathbf{\Psi}^* \tag{8}$$

where $\mathbf{\Psi} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is the discrete separable wavelet transform. Matrix $\mathbf{\Theta}$ is an $N \times N$ matrix which corresponds to a truncated version (also called finite section) of the matrix Θ defined in (6).

4.3 Theoretical guarantees with sparse approximations

Sparse approximations of integral operators have been studied theoretically in [5, 35]. They then have been successfully used in the numerical analysis of PDEs [14, 11, 10]. Surprisingly, they have been scarcely applied to image processing. The two exceptions we are aware of are the paper [9], where the authors show that wavelet multipliers can be useful to approximate foveation operators. More recently, [48] proposed an approach that is very much related to that of our paper.

Let us provide a typical result that motivates the proposed approach.

Lemma 1 (Decay of $\theta_{\lambda,\mu}$). *Assume that H is a blurring operator (see Definition 1) in the class $\mathcal{A}(M, f)$. Assume that the mother wavelet is compactly supported with M vanishing moments.*

Then, the coefficients of Θ satisfy the following inequality for all $\lambda = (j, m, e) \in \Lambda$ and $\mu = (k, n, e') \in \Lambda$:

$$|\theta_{\lambda,\mu}| \leq C_M 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} f_{\lambda,\mu} \quad (9)$$

where $f_{\lambda,\mu} = f(\text{dist}(I_\lambda, I_\mu))$, C_M is a constant that does not depend on λ and μ and

$$\begin{aligned} \text{dist}(I_\lambda, I_\mu) &= \inf_{x \in I_\lambda, y \in I_\mu} \|x - y\|_\infty \\ &= \max \left(0, \left\| 2^{-j}m - 2^{-k}n \right\|_\infty - (2^{-j} + 2^{-k}) \frac{c(M)}{2} \right). \end{aligned} \quad (10)$$

Proof. See Appendix A. □

Lemma 1 is the key to obtain all subsequent complexity estimates.

Theorem 2. *Let Θ_η be the matrix obtained by zeroing all coefficients in Θ such that*

$$2^{-\min(j,k)(M+d)} f_{\lambda,\mu} \leq \eta,$$

with $\lambda = (j, m, e) \in \Lambda$ and $\mu = (k, n, e') \in \Lambda$.

Let $\tilde{\mathbf{H}}_\eta = \Psi \Theta_\eta \Psi^$ denote the resulting operator. Suppose that f is compactly supported in $[0, \kappa]$ and that $\eta \leq \log_2(N)^{-(M+d)/d}$. Then:*

i) *The number of non zero coefficients in Θ_η is bounded above by*

$$C'_M N \kappa^d \eta^{-\frac{d}{M+d}} \quad (11)$$

where $C'_M > 0$ is independent of N .

ii) *The approximation $\tilde{\mathbf{H}}_\eta$ satisfies $\left\| \mathbf{H} - \tilde{\mathbf{H}}_\eta \right\|_{2 \rightarrow 2} \lesssim \eta^{\frac{M}{M+d}}$.*

iii) *The number of coefficients needed to satisfy $\left\| \mathbf{H} - \tilde{\mathbf{H}}_\eta \right\|_{2 \rightarrow 2} \leq \epsilon$ is bounded above by*

$$C''_M N \kappa^d \epsilon^{-\frac{d}{M}} \quad (12)$$

where $C''_M > 0$ is independent of N .

Proof. See Appendix B. □

Let us summarize the main conclusions drawn from this section:

- A discretization in the wavelet domain provides better theoretical guarantees than the standard quadrature rules (see Section 4.1).
- The method is capable of handling *automatically* the degree of smoothness of the integral kernel K since there is a dependency in $\epsilon^{-\frac{d}{M}}$ where M is the smoothness of the integral operator.
- We will see in the next section that the method is quite versatile since different sparsity patterns can be chosen depending on the knowledge of the blur kernel and on the regularity of the signals that are to be processed.
- The method can also handle more general singular operators as was shown in the seminal papers [35, 13, 5].

Remark 3. Similar bounds as (9) can be derived with less stringent assumptions. First, the domain can be unbounded, given that kernels have a sufficiently fast decay at infinity. Second, the kernel can blow up on its diagonal, which is the key to study Calderon-Zygmund operators (see [35, 13, 5] for more details). We stuck to this simpler setting to simplify the proofs.

5 Identification of sparsity patterns

A key step to control the approximation quality is the selection of the coefficients in the matrix Θ that should be kept. For instance, a simple thresholding of Θ leads to sub-optimal and somewhat disappointing results. In this section we propose algorithms to select the most relevant coefficients for images belonging to functional spaces such as that of bounded variation functions. We study the case where Θ is known completely and the case where only an upper-bound such as (9) is available.

5.1 Problem formalization

Let \mathbf{H} be the $N^d \times N^d$ matrix defined in equation (8). We wish to approximate \mathbf{H} by a matrix $\tilde{\mathbf{H}}_K$ of kind $\Psi \mathbf{S}_K \Psi^*$ where \mathbf{S}_K is a matrix with at most K non-zero coefficients. Let \mathbb{S}_K denote the space of $N \times N$ matrices with at most K non-zero coefficients. The problem we address in this paragraph reads

$$\begin{aligned} & \min_{\mathbf{S}_K \in \mathbb{S}_K} \left\| \mathbf{H} - \tilde{\mathbf{H}}_K \right\|_{X \rightarrow 2} \\ & = \min_{\mathbf{S}_K \in \mathbb{S}_K} \max_{\|\mathbf{u}\|_X \leq 1} \left\| \mathbf{H}\mathbf{u} - \Psi \mathbf{S}_K \Psi^* \mathbf{u} \right\|_2. \end{aligned}$$

The solution of this problem provides the best K -sparse matrix \mathbf{S}_K , in the sense that no other choice provides a better SNR uniformly on the unit-ball $\{\mathbf{u} \in \mathbb{R}^N, \|\mathbf{u}\|_X \leq 1\}$.

5.1.1 Theoretical choice of the space X

The norm $\|\cdot\|_X$ should be chosen depending on the type of images that have to be blurred. For instance, it is well-known that natural images are highly compressible in the wavelet domain [33, 43]. This observation is the basis of JPEG2000 compression standard. Therefore, a natural choice could be to set $\|\mathbf{u}\|_X = \|\Psi^*\mathbf{u}\|_1$. This choice will ensure a good reconstruction of images that have a wavelet decomposition with a low ℓ^1 -norm.

Another very common assumption in image processing is that images have a bounded total variation. The space of functions with bounded total variation [1] contains images discontinuous along edges with finite length. It is one of the most successful tools for image processing tasks such as denoising, segmentation, reconstruction, ... Functions in $BV(\Omega)$ can be characterized by their wavelet coefficients [38, 33]. For instance, if $u \in BV(\Omega)$, then

$$\sum_{\lambda \in \Lambda_0} 2^{j(1-\frac{d}{2})} |\langle u, \psi_\lambda \rangle| < +\infty \tag{13}$$

for all wavelet bases. This result is due to embeddings of BV space in Besov spaces which are characterized by their wavelet coefficients (see [10] for more details on Besov spaces). This result motivated us to consider norms defined by

$$\|\mathbf{u}\|_X = \|\Sigma\Psi^*\mathbf{u}\|_1$$

where $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_N)$ is a diagonal matrix. Depending on the regularity level of the images considered, different diagonal coefficients can be used. For instance, for BV signals in 1D, one could set $\sigma_i = 2^{j(i)/2}$ where $j(i)$ is the scale of the i -th wavelet, owing to (13).

5.1.2 Practical choice of the space X

More generally, it is possible to adapt the weights σ_i depending on the images to recover. Most images exhibit a similar decay of wavelet coefficients across subbands. This decay is a characteristic of the functions regularity (see e.g. [26]). To illustrate this fact, we conducted a simple experiment in Figure 4. We evaluate the maximal value of the amplitude of wavelet coefficients of three images with different contents across scales. The wavelet transform is decomposed at level 4 and we normalize the images so that their maximum wavelet coefficient is 1. As can be seen even though the maximal values differ from one image to the next, their overall behavior is the same: amplitudes decay nearly dyadically from one scale to the next. The same phenomenon can be observed with the mean value.

This experiment suggests setting $\sigma_i = 2^{j(i)}$ in order to normalize the wavelet coefficients amplitude in each subband. Once again, the same idea was explored in [48].

5.1.3 An optimization problem

We can now take advantage of the fact that images and operators are sparse in the same wavelet basis. Let $\mathbf{z} = \Psi^*\mathbf{u}$ and $\Delta = \Theta - \mathbf{S}_K$. Since we consider orthogonal wavelet

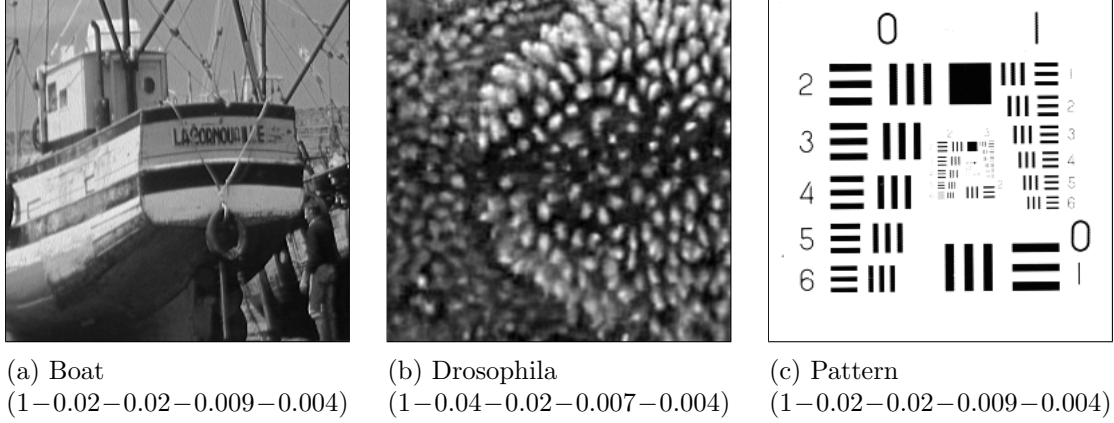


Figure 4: Three pictures and the mean amplitude of their wavelet coefficients at each scale of the wavelet transform.

transforms, we have $\|\Psi\mathbf{u}\|_2 = \|\mathbf{u}\|_2$, for any $\mathbf{u} \in \mathbb{R}^N$ therefore:

$$\begin{aligned} \left\| \mathbf{H} - \tilde{\mathbf{H}}_K \right\|_{X \rightarrow 2} &= \max_{\|\mathbf{u}\|_X \leq 1} \|\Psi(\Theta - \mathbf{S}_K)\Psi^*\mathbf{u}\|_2 \\ &= \max_{\|\Sigma\mathbf{z}\|_1 \leq 1} \|(\Theta - \mathbf{S}_K)\mathbf{z}\|_2 \\ &= \max_{\|\mathbf{z}\|_1 \leq 1} \|\Delta\Sigma^{-1}\mathbf{z}\|_2. \end{aligned}$$

Since the operator norm $\|\mathbf{A}\|_{1 \rightarrow 2} = \max_{1 \leq i \leq N} \|\mathbf{A}^{(i)}\|_2$, where $\mathbf{A}^{(i)}$ denote the i -th column of the $N \times N$ matrix \mathbf{A} and by remarking that $(\Delta\Sigma^{-1})^{(i)} = \Delta^{(i)}\sigma_i^{-1}$, we finally get the following simple expression for the operator norm:

$$\left\| \mathbf{H} - \tilde{\mathbf{H}} \right\|_{X \rightarrow 2} = \max_{1 \leq i \leq N} \frac{1}{\sigma_i} \|\Delta^{(i)}\|_2. \quad (14)$$

Our goal is thus to find the solution of:

$$\min_{\mathbf{S}_K \in \mathcal{S}_K} \max_{1 \leq i \leq N} \frac{1}{\sigma_i} \|\Delta^{(i)}\|_2. \quad (15)$$

5.2 Link with the approach in [48]

In this paragraph, we show that the method proposed in [49, 48], can be interpreted with the formalism given above. In those papers, Θ is approximated by $\tilde{\Theta}$ using the following rule:

$$\tilde{\Theta}_{i,j} = \begin{cases} \Theta_{i,j} & \text{if } \frac{\Theta_{i,j}}{w_j} \text{ is in the } K \text{ largest values of } \Theta\mathbf{W}^{-1} \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

The weights w_i are set as constant by subbands and learned as described in paragraph 5.1.2.

The thresholding rule (16) can be interpreted as the solution of the following problem:

$$\min_{\tilde{\Theta} \in \mathbb{S}_K} \left\| \Theta - \tilde{\Theta} \right\|_{\mathbf{W} \rightarrow \infty},$$

where here $\|x\|_{\mathbf{W}} = \|\mathbf{W}x\|_1$ with $\mathbf{W} = \text{diag}(w_i)$ a diagonal matrix. Indeed, the above problem is equivalent to:

$$\min_{\tilde{\Theta} \in \mathbb{S}_K} \max_{1 \leq i, j \leq N} \left| \frac{1}{w_j} \left(\Theta - \tilde{\Theta} \right)_{i,j} \right|.$$

In other words, the method proposed in [49, 48] constructs a K best-term approximation of Θ in the metric $\|\cdot\|_{\mathbf{W} \rightarrow \infty}$.

Overall, the problem is very similar to (15), except that the image quality is evaluated through an infinite norm in the wavelet domain, while we propose using a Euclidean norm in the spatial domain. We believe that this choice is more relevant for image processing since the SNR is the most common measure of image quality. In practice, we will see in the numerical experiments that both methods lead to very similar practical results.

Finally, let us mention that the authors in [48] have an additional concern of storing the matrix representation with the least memory. They therefore *quantize* the coefficients in Θ . Since the main goal in this paper is the design of fast algorithms for matrix-vector products, we do not consider this extra refinement.

5.3 An algorithm when Θ is known

Finding the minimizer of problem (15) can be achieved using a simple greedy algorithm: the matrix \mathbf{S}_{k+1} is obtained by adding the largest coefficient of the column Δ_i with largest Euclidean norm to \mathbf{S}_k . This procedure can be implemented efficiently by using quick sort algorithms. The complete procedure is described in Algorithm 1. The overall complexity of this algorithm is $\mathcal{O}(N^2 \log(N))$. The most computationally intensive step is the sorting procedure in the initialisation. The loop on k can be accelerated by first sorting the set $(\gamma_j)_{1 \leq j \leq N}$, but the algorithm's complexity remains essentially unchanged.

5.4 An algorithm when Θ is unknown

In the previous paragraph, we assumed that the full matrix Θ was known. There are at least two reasons that make this assumption irrelevant. First, computing Θ is very computationally intensive and it is not even possible to store this matrix in RAM for medium sized images (e.g. 512×512). Second, in blind deblurring problems, the operator \mathbf{H} needs to be inferred from the data and adding priors on the sparsity pattern of \mathbf{S}_K might be an efficient choice to improve the problem identifiability.

Algorithm 1: An algorithm to find the minimizer of (15).

Input:

Θ : $N \times N$ matrix;

Σ : Diagonal matrix;

K : the number of elements in the thresholded matrix;

Output:

\mathbf{S}_K : Matrix minimizing (15)

Initialization:

Set $\mathbf{S}_K = \mathbf{0} \in \mathbb{R}^{N \times N}$;

Sort the coefficients of each column $\Theta^{(j)}$ of Θ in decreasing order;

Obtain $\mathbf{A}^{(j)}$ the sorted columns $\Theta^{(j)}$ and index sets I_j ;

The sorted columns $\mathbf{A}^{(j)}$ and index set I_j satisfy $\mathbf{A}^{(j)}(i) = \Theta^{(j)}(I_j(i))$;

Compute the norms $\gamma_j = \frac{\|\Theta^{(j)}\|_2^2}{\sigma_j^2}$;

Define $\mathbf{O} = (1, \dots, 1) \in \mathbb{R}^N$;

$\mathbf{O}^{(j)}$ is the index of the largest coefficient in $\mathbf{A}^{(j)}$ not yet added to \mathbf{S}_K ;

begin

for $k = 1$ **to** K **do**

 Find $l = \arg \max_{j=1 \dots N} \gamma_j$;

(Find the column l with largest Euclidean norm)

 Set $\mathbf{S}_K(I_l(\mathbf{O}(l)), l) = \Theta(I_l(\mathbf{O}(l)), l)$;

(Add the coefficient in the l -th column at index $I_l(\mathbf{O}(l))$)

 Update $\gamma_l = \gamma_l - \left(\frac{\mathbf{A}^{(l)}(\mathbf{O}(l))}{\sigma_l} \right)^2$;

(Update norms vector)

 Set $\mathbf{O}(l) = \mathbf{O}(l) + 1$;

(The next value to add in l -th column will be at index $\mathbf{O}(l) + 1$)

end

end

When Θ is unknown, we may take advantage of equation (9) to define sparsity patterns. A naive approach would consist in applying Algorithm (1) directly on the upper-bound (9). However, this matrix cannot be stored and this approach is applicable only for small images. In order to reduce the computational burden, one may take advantage of the special structure of the upper-bound: equation (9) indicates that the coefficients $\theta_{\lambda,\mu}$ can be discarded for sufficiently large $|j-k|$ and sufficiently large distance between the wavelet supports. Equation (9) thus means that for a given wavelet ψ_λ , only its spatial neighbours in neighbouring scales have significant correlation coefficients $\langle H\psi_\lambda, \psi_\mu \rangle$. We may thus construct sparsity patterns using the notion of multiscale neighbourhoods defined below.

Definition 2 (Multiscale shift). The multiscale shift $s_{\lambda,\mu} \in \mathbb{Z}^d$ between two wavelets ψ_λ and ψ_μ is defined by

$$s_{\lambda,\mu} = \left\lfloor \frac{n}{2^{\max(k-j,0)}} \right\rfloor - \left\lfloor \frac{m}{2^{\max(j-k,0)}} \right\rfloor. \quad (17)$$

We recall that $\lambda = (j, m, e) \in \Lambda$ and $\mu = (k, n, e') \in \Lambda$. Note that for $k = j$, the multi-scale shift is just $s_{\lambda,\mu} = n - m$ and corresponds to the standard shift between wavelets, measured as a multiple of the characteristic size 2^{-j} . The divisions by $2^{\max(k-j,0)}$ and $2^{\max(j-k,0)}$ allow to rescale the shifts at the coarsest level. This definition is illustrated in Figure 5.

Definition 3 (Multiscale neighborhood). Let

$$\mathcal{N} = \left\{ (j, (k, s)), (j, k) \in \{0, \dots, \log_2(N) - 1\}^2, s \in \{0, \dots, 2^{\min(j,k)} - 1\}^d \right\}$$

denote the set of all neighborhood relationships, i.e. the set of all possible couples of type (scale, (scale, shift)). A multiscale neighborhood \mathcal{N} is an element of the powerset $\mathcal{P}(\mathcal{N})$.

Definition 4 (Multiscale neighbors). Given a multiscale neighborhood \mathcal{N} , two wavelets ψ_λ and ψ_μ will be said to be \mathcal{N} -neighbors if $(j, (k, s_{\lambda,\mu})) \in \mathcal{N}$ where $s_{\lambda,\mu}$ is defined in equation (17).

The problem of finding a sparsity pattern is now reduced to finding a good multiscale neighborhood. In what follows, we let $\mathcal{N}(j) = \{(k, s), (j, (k, s)) \in \mathcal{N}\}$ denote the set of all possible neighborhood relationships at scale j . This is illustrated in Figure 6. Let $\mathcal{N} \in \mathcal{P}(\mathcal{N})$ denote a multiscale neighborhood. We define the matrix $\mathbf{S}_{\mathcal{N}}$ as follows:

$$\mathbf{S}_{\mathcal{N}}(\lambda, \mu) = \begin{cases} \theta_{\lambda,\mu} & \text{if } \psi_\lambda \text{ is an } \mathcal{N}\text{-neighbor of } \psi_\mu \\ 0 & \text{otherwise.} \end{cases}$$

Equation (9) indicates that

$$|\theta_{\lambda,\mu}| \leq u(j, k, s)$$

with

$$u(j, k, s) = C_M 2^{-(M+\frac{d}{2})|j-k|-(M+d)\min(j,k)} f_{j,k,s} \quad (18)$$

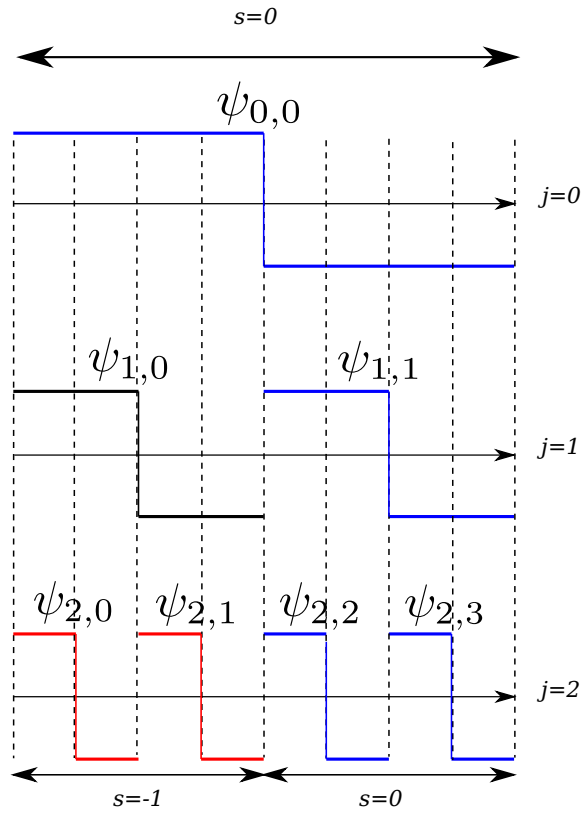


Figure 5: Illustration of a multiscale shift on a 1D signal of size 8 with the Haar basis. The shifts are computed with respect to wavelet $\psi_{1,1}$. Wavelets $\psi_{0,0}$, $\psi_{2,2}$ and $\psi_{2,3}$ have a multiscale shift $s = 0$ with $\psi_{1,1}$ since their support intersects that of $\psi_{1,1}$. Wavelets $\psi_{1,0}$, $\psi_{2,0}$ and $\psi_{2,1}$ have a multiscale shift $s = -1$ with $\psi_{1,1}$ since their support intersects that of $\psi_{1,0}$.

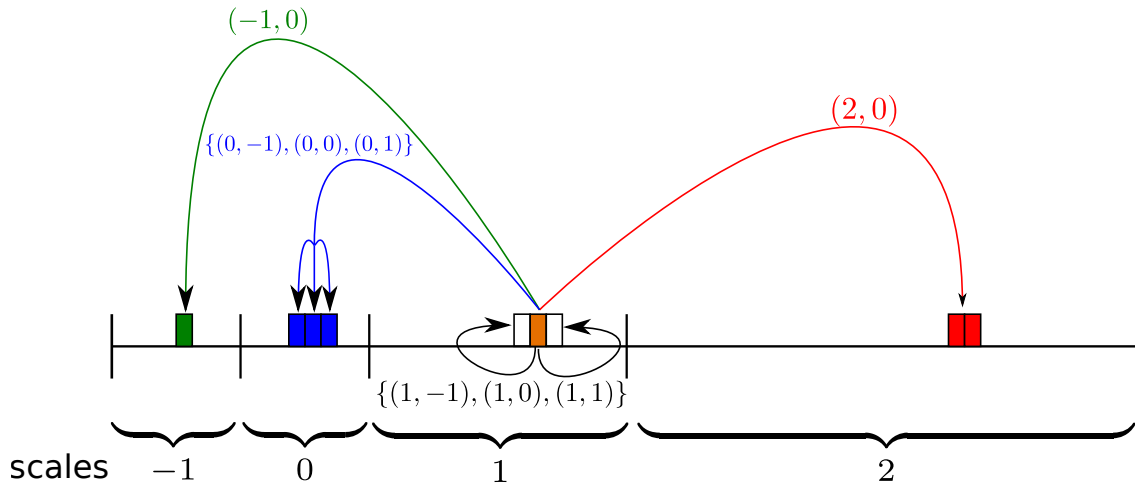


Figure 6: Illustration of a multiscale neighborhood on a 1D signal. In this example, the neighborhood at scale 1 is $\mathcal{N}(1) = \{(-1, 0), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1), (2, 0)\}$. Notice that the two red wavelets at scale 2 are neighbors of the orange wavelet at scale 1 and that this relationship is described through only one shift.

and $f_{j,k,s} = f\left(\max\left(0, 2^{-\min(j,k)} \|s\|_\infty - (2^{-j} + 2^{-k}) \frac{c(M)}{2}\right)\right)$. Let \mathbf{U} be the matrix defined by $\mathbf{U}(\lambda, \mu) = u(j, k, s_{\lambda, \mu})$. Finding a good sparsity pattern can now be achieved by solving the following problem:

$$\min_{\substack{\mathcal{N} \in \mathcal{P}(\mathcal{M}) \\ |\mathcal{N}|=K}} \max_{1 \leq i \leq N} \frac{1}{\sigma_i} \left\| (\mathbf{U} - \mathbf{S}_{\mathcal{N}})^{(i)} \right\|_2 \quad (19)$$

where $(\mathbf{U} - \mathbf{S}_{\mathcal{N}})^{(i)}$ denotes the i -th column of $(\mathbf{U} - \mathbf{S}_{\mathcal{N}})$.

In what follows, we assume that σ_i only depends on the scale $j(i)$ of the i -th wavelet. Similarly to the previous section, finding the optimal sparsity pattern can be performed using a greedy algorithm. A multiscale neighborhood is constructed by iteratively adding the couple (scale, (scale, shift)) that minimizes a residual. This technique is described in Algorithm 2.

Note that the norms γ_k only depend on the scale $j(k)$, so that the initialisation step only requires $\mathcal{O}(N \log_2(N))$ operations. Similarly to Algorithm 1, this algorithm can be accelerated by first sorting the elements of $u(j, k, s)$ in decreasing order. The overall complexity for this algorithm is $\mathcal{O}(N \log(N)^2)$ operations.

6 Numerical experiments

In this section we perform various numerical experiments in order to illustrate the theory proposed in the previous sections and to compare the practical efficiency of wavelet based

Algorithm 2: An algorithm to find the minimizer of (19).

Input:

u : Upper-bound defined in (18);

Σ : Diagonal matrix;

K : the number of elements of the neighborhood;

Output:

\mathcal{N} : multiscale neighborhood minimizing (19)

Initialization:

Set $\mathcal{N} = \emptyset$;

Compute the norms $\gamma_k = \frac{\|\mathbf{U}^{(k)}\|_2^2}{\sigma_k^2}$ using the upper-bound u ;

begin

for $k = 1$ **to** K **do**

 Find $j^* = \arg \max_{j=1 \dots N} \gamma_j$;

(The column with largest norm)

 Find $(k^*, s^*) = \arg \max_{(k,s) \in \mathcal{N}(j^*)} u^2(j^*, k, s) 2^{\max(j^*-k, 0)}$;

(The best scale and shift for this column is (k^, s^*))*

(The number of elements in the neighborhood relationship $(j^, (k, s))$ is $2^{\max(j^*-k, 0)}$)*

 Update $\mathcal{N} = \mathcal{N} \cup \{(j^*, (k^*, s^*))\}$;

 Set $\gamma_k = \gamma_k - u^2(j^*, k^*, s^*) \cdot 2^{\max(j^*-k, 0)}$

end

end

methods against windowed convolutions (WC) based approaches. We first describe the operators and images used in our experiments. Second, we provide numerical experiments for the direct problems. Finally, we provide numerical comparisons for the inverse problem.

6.1 Preliminaries

6.1.1 Test images

We consider a set of 16 images of different natures: standard image processing images (the boat, the house, Lena, Mandrill (see Figure 7a), peppers, cameraman), two satellite images, three medical images, three buildings images, and two test pattern images (see Figure 7b). Due to memory limitations, we only consider images of size $N = 256 \times 256$. Note that a full matrix of size $N \times N$ stored in double precision weighs around 32 gigabytes.

6.1.2 Test operators

Three different blur kernels of different complexities are considered, see Figure 8. The PSFs in Figure 8a and 8b modeled for all $x \in [0, 1]^2$ by 2D Gaussians. Therefore the associated kernel is defined for all $(x, y) \in [0, 1]^2 \times [0, 1]^2$ by

$$K(x, y) = \frac{1}{2\pi |C(y)|} \exp \left[\frac{1}{2} (y - x)^T C^{-1}(y) (y - x) \right].$$

The covariance matrices C are defined as:

- In Figure 8a: $C(y) = \begin{pmatrix} f(y_1) & 0 \\ 0 & f(y_1) \end{pmatrix}$ with $f(t) = 2t$, for $t \in [0, 1]$. The PSFs are truncated out of a 11×11 support.
- In Figure 8b: $C(y) = R(y)^T D(y) R(y)$ where $R(y)$ is a rotation matrix of angle $\theta = \arctan \left(\frac{y_1 - 0.5}{y_2 - 0.5} \right)$ and $D(y) = \begin{pmatrix} g(y) & 0 \\ 0 & h(y) \end{pmatrix}$ with $g(y) = 10 \|y - (0.5, 0.5)^T\|_2$ and $h(y) = 2 \|y - (0.5, 0.5)^T\|_2$. The PSFs are truncated out of a 21×21 support.

The PSFs in Figure 8c were proposed in [44] as an approximation of real spatially optical blurs.

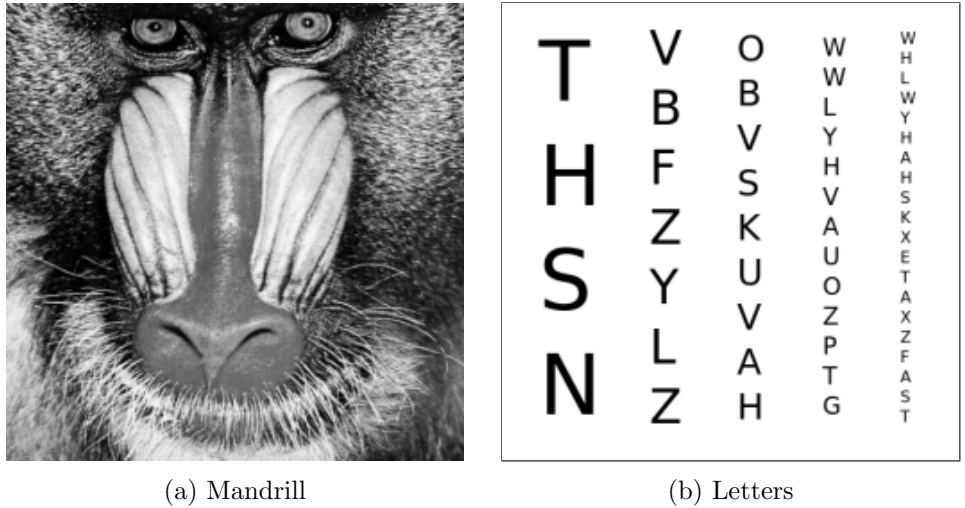


Figure 7: The two images of size 256×256 used in these numerical experiments

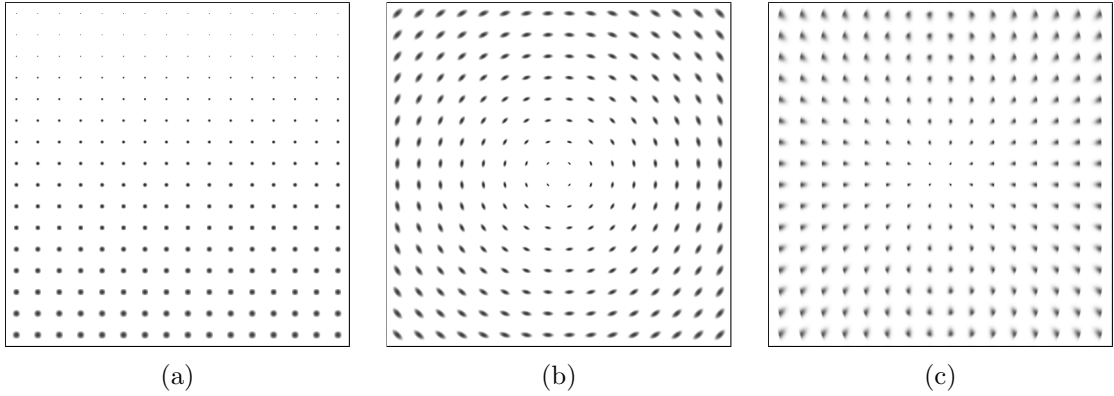


Figure 8: PSFs maps used in the paper. The PSFs in Figure 8a are Gaussians with equal variances increasing in the vertical direction. The PSFs in Figure 8b are anisotropic Gaussians with covariance matrices that depend on the polar coordinates. The PSFs in Figure 8c are based on paper [44].

6.1.3 Computation of the full Θ matrix

Before applying our approximation methods, matrix Θ needs to be computed explicitly. The coefficients $\langle H\psi_\lambda, \psi_\mu \rangle$ are approximated by their discrete counterparts. If ψ_λ and ψ_μ denote discrete wavelets, we simply compute the wavelet transform of $\mathbf{H}\psi_\lambda$ and store it into the λ -th column of Θ . This computation scheme is summarized in Algorithm 3. This algorithm corresponds to the use of rectangle methods to evaluate the dot-products:

$$\int_{\Omega} \int_{\Omega} K(x, y) \psi_\lambda(y) \psi_\mu(x) dy dx \simeq \frac{1}{N^{2d}} \sum_{x \in X} \sum_{y \in X} K(x, y) \psi_\lambda(y) \psi_\mu(x). \quad (20)$$

Algorithm 3: An algorithm to compute Θ

Output:

Θ : the full matrix of \mathbf{H}

begin

forall the λ do

 Compute the wavelet ψ_λ using an inverse wavelet transform

 Compute the blurred wavelet $\mathbf{H}\psi_\lambda$

 Compute $(\langle \mathbf{H}\psi_\lambda, \psi_\mu \rangle)_\mu$ using one forward wavelet transform

 Set $(\langle \mathbf{H}\psi_\lambda, \psi_\mu \rangle)_\mu$ in the λ -th column of Θ .

end

end

6.2 Application to direct problems

In this section, we investigate the approximation properties of the proposed approaches in the aim of computing matrix-vector products. In all numerical experiments, we use an orthogonal wavelet transform with 4 decomposition levels. We always use Daubechies wavelets.

6.2.1 Influence of vanishing moments

First of all we demonstrate the influence of vanishing moments on the quality of approximations. For each number of vanishing moments $M \in \{1, 2, 4, 6, 10\}$, a sparse approximation $\tilde{\mathbf{H}}$ is constructed by thresholding Θ , keeping the $K = l \times N$ largest coefficients with $l \in \{0 \dots 40\}$. Then for each \mathbf{u} in the set of 16 images, we compare $\tilde{\mathbf{H}}\mathbf{u}$ to $\mathbf{H}\mathbf{u}$ computing the pSNR. We then plot the average of pSNRs over the set of images with respect to the number of operations needed for a matrix-vector product. The results of this experiment are displayed in Figure 9. It appears that for the considered operators, using as many vanishing moments as possible was preferable. Using more than 10 vanishing moments

however led to insignificant performance increase while making the numerical complexity higher. Therefore, in all the following numerical experiments we will use Daubechies wavelets with 10 vanishing moments. Note that paper [48] only explored the use of Haar wavelets. This experiment shows that very significant improvements can be obtained by leveraging regularity of the integral kernel using vanishing moments. The behavior was predicted by Theorem 2.

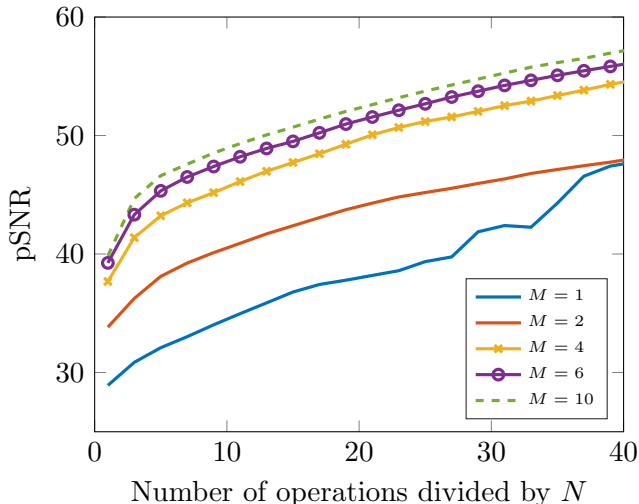


Figure 9: pSNR of the blurred image using the approximated operator $\tilde{\mathbf{H}}\mathbf{u}$ with respect to the blurred image using the exact operator $\mathbf{H}\mathbf{u}$. pSNRs have been averaged over the set of test images. Daubechies wavelets have been used with different number vanishing moments $M \in \{1, 2, 4, 6, 10\}$. The case $M = 1$ corresponds to Haar wavelets.

6.2.2 Comparison of different methods

Wavelets VS windowed convolutions. In this first numerical experiment, we evaluate $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2}$ where $\tilde{\mathbf{H}}$ is obtained by windowed convolutions method or sparse approximations in the wavelet domain.

The sparse approximation of the operator is constructed by thresholding the matrix Θ in order to keep the K largest coefficients. We have set $K = 2^l \times N$ with $l \in \{0 \dots 2 \log_2 N\}$. This way K is a multiple of the number of pixels in the image. The windowed convolutions method is constructed by partitioning the image into $2^l \times 2^l$ sub-images where $l \in \{1 \dots \log_2 N\}$. We also studied the case where sub-images overlap and linearly interpolated the blur between sub-images as proposed in [36, 28]. The overlap has been fixed to 50% of the sub-images sizes.

For each sub-image size, and each overlap, the norm $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2}$ is approximated

using a power method [23]. We stop the iterative process when the difference between the eigenvalues of two successive iterations is smaller than $10^{-8}\|\mathbf{H}\|_{2\rightarrow 2}$. The number of operations associated to each type of approximation is computed using theoretical complexities. For sparse matrix-vector product the number of operations is proportional to the number of non-zero coefficients in the matrix. For windowed convolutions methods, the number of operations is proportional to the number of windows ($2^l \times 2^l$) multiplied by the cost of a discrete convolution over a window $(\frac{N}{2^l} + N\kappa)^2 \log_2 (\frac{N}{2^l} + N\kappa)$.

Figure 10 shows the results of this experiment. The wavelet based method seems to perform much better than windowed convolutions methods for both operators. The gap is however significantly larger for the rotation blur in Figure 8b. This experiment therefore suggests that the advantage of wavelet based approaches will depend on the type of blur considered.

The influence of sparsity patterns. In this numerical experiment, we obtain a K -sparse matrix Θ_K using either a simple thresholding strategy, Algorithm 1 or Algorithm 2. We evaluate the error $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{X\rightarrow 2}$ defined in (14) for each methods. We set $\sigma_i = 2^{j(i)}$, where $j(i)$ corresponds to the scale of the i -th wavelet. As can be seen from Figure 11, Algorithm 1 provides a much better error decay for each operator than the simple thresholding strategy. This fact will be verified for real images in next paragraph. Algorithm 2 has a much slower decay than both thresholding algorithm. Notice that this algorithm is essentially blind, in the sense that it does not require knowing the exact matrix Θ to select the pattern. It would therefore work for a whole class of blur kernels, whereas the simple thresholding and Algorithm 1 work only for a specific matrix.

Figure 12 shows the sparsity patterns of matrices obtained with Algorithms 1 and 2 for $K = 30N$ and $K = 128N$ coefficients. The sparsity patterns look quite similar. However, Algorithm 1 selects subbands that are not selected by Algorithm 2, which might explain the significant performance differences. Similarly, Algorithm 2 select subbands that would probably be crucial for some blur kernels, but which are not significant for this particular blur kernel.

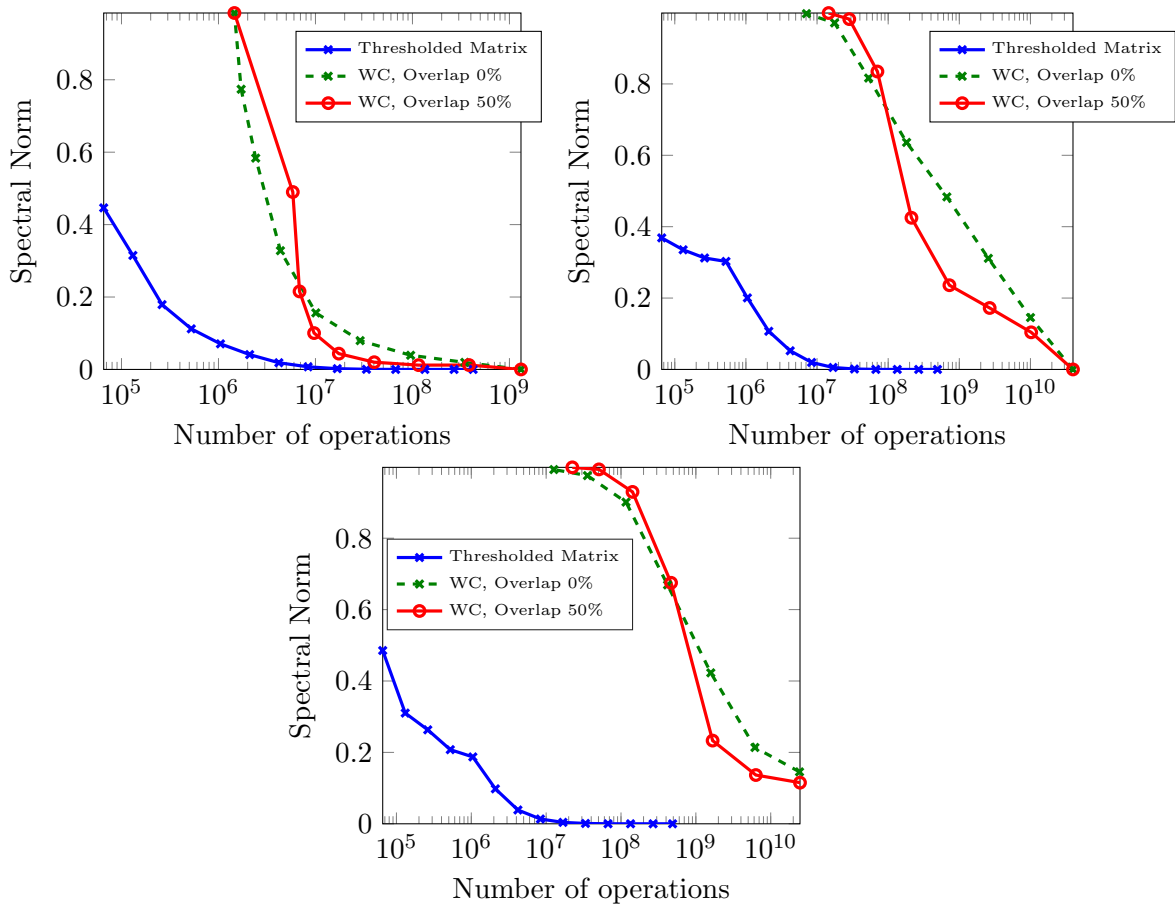


Figure 10: The operator norms $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2}$ are displayed for the three proposed kernels. (Left: kernel Figure 8a, middle: kernel in Figure 8b, right: kernel in Figure 8c). Norms are plotted with respect to the number of operations needed to compute $\tilde{\mathbf{H}}\mathbf{u}$. The abscissas are in log scale.

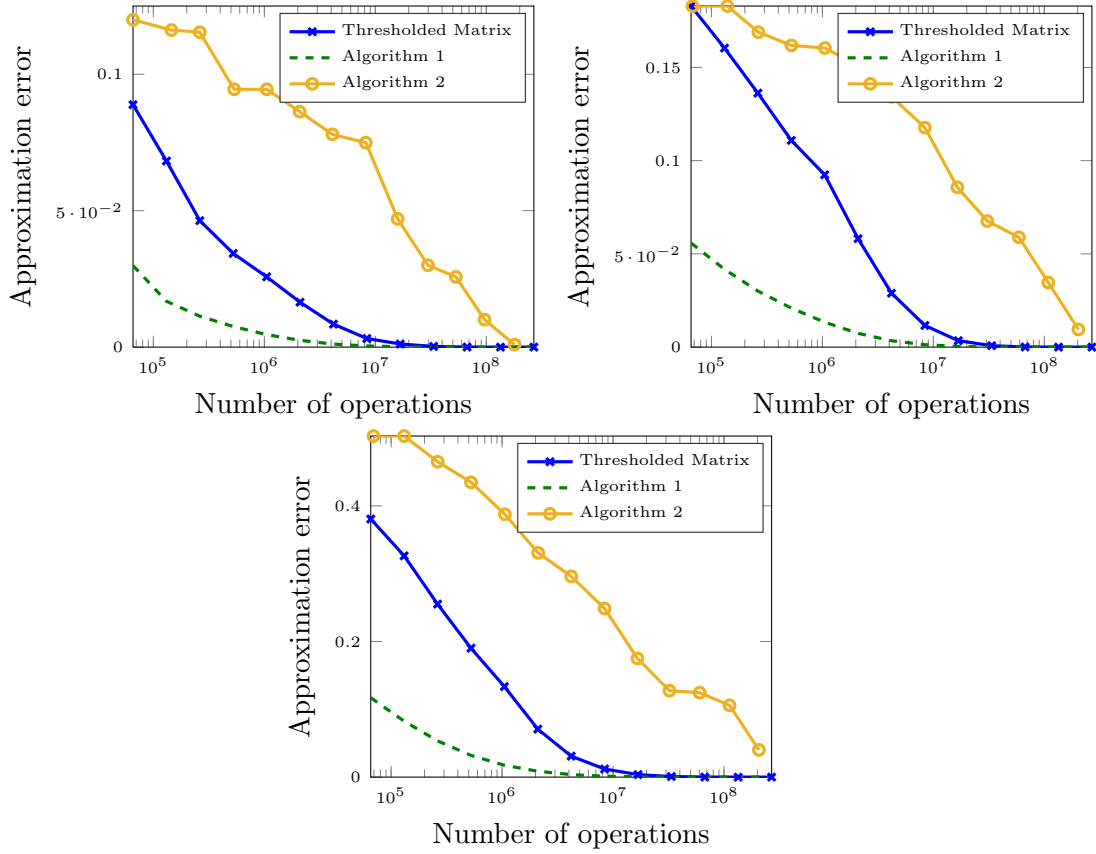


Figure 11: The operator norms $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{X \rightarrow 2}$ are displayed for kernels Figure 8a (left) and Figure 8b (right); and with respect to the number of operations needed to compute $\tilde{\mathbf{H}}u$. The abscissas are in log scale. Daubechies wavelets with 10 vanishing moments have been used.

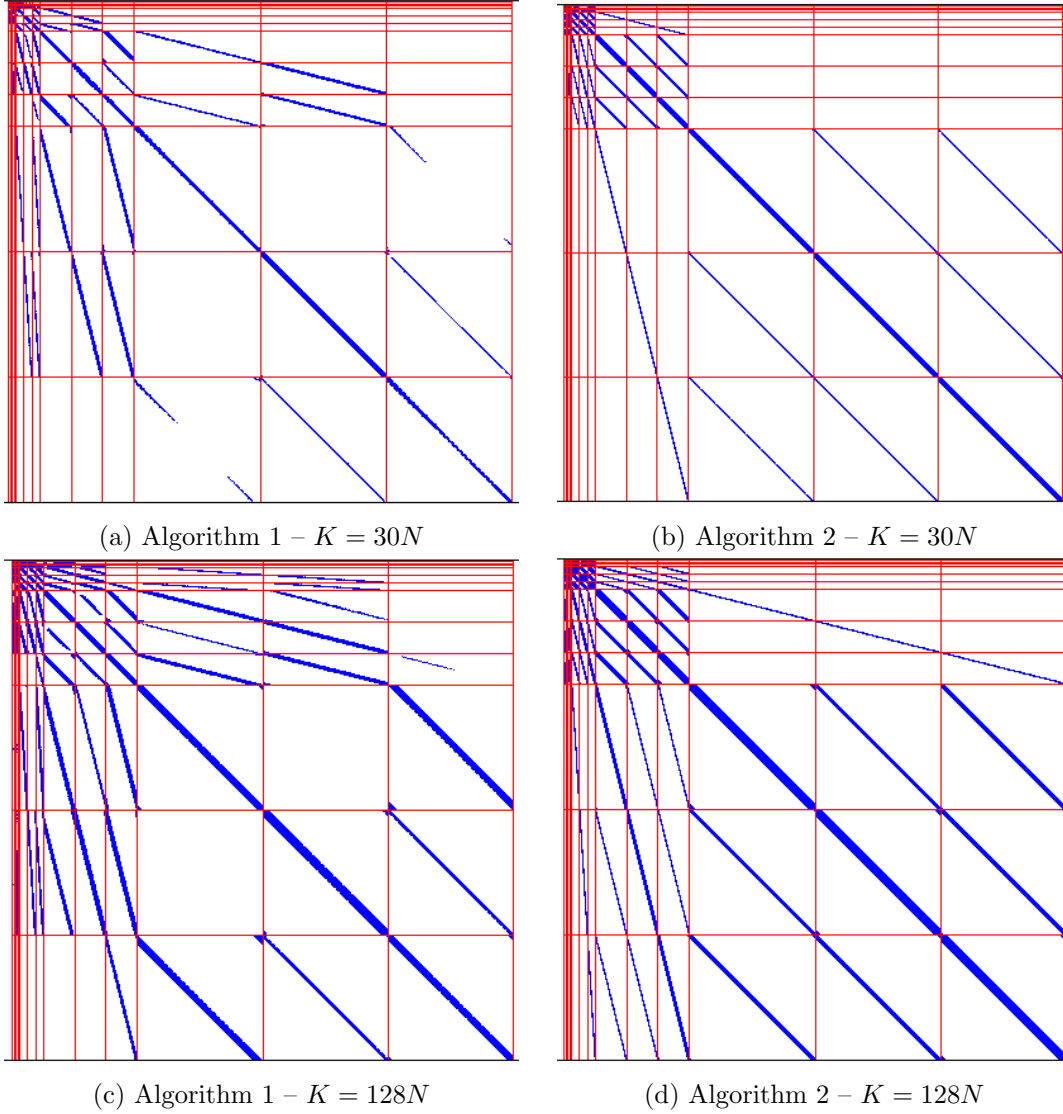


Figure 12: The structure of the wavelet matrices of Θ_K are displayed for Algorithms 1 and 2 and for $K = 30N$ and $K = 128N$ coefficients. Algorithm 1 has been applied using the second $\Sigma = \text{diag}(2^{j(i)})_i$ matrix. This experiment corresponds to the blur in Figure 8b

6.2.3 Quality of matrix-vector products for real images

In this section, we evaluate the performance of wavelet based methods for matrix-vector products with real images.

Quality VS complexity. We compare $\tilde{\mathbf{H}}\mathbf{u}$ to $\mathbf{H}\mathbf{u}$, where \mathbf{u} is the image in Figure 7b and where $\tilde{\mathbf{H}}$ is obtained either by windowed convolutions methods or by sparse wavelet approximations. We plot the pSNR between the exact blurred image $\mathbf{H}\mathbf{u}$ and the blurred image using the approximated operator $\tilde{\mathbf{H}}\mathbf{u}$ in Figure 13. Different approximation methods are tested:

Thresholded matrix: This corresponds to a simple thresholding of the wavelet matrix Θ .

Σ $n^\circ 1$: This corresponds to applying Algorithm 1 with $\sigma_i = 1, \forall i$ where $j(i)$ corresponds to the scale of the i -th wavelet.

Σ $n^\circ 2$: This corresponds to applying Algorithm 1 with $\sigma_i = 2^{j(i)} \forall i$.

[48]: The method presented in [48] with $K = l \times N$ coefficients in the matrix, with $l \in \{1, \dots, 100\}$.

WC, Overlap 50%: This corresponds to the windowed convolution with 50% overlap. We use this overlap since it produces better pSNRs.

Algo 2: The algorithm finds multi-scale neighbourhoods until $K = l \times N$ coefficients populate the matrix, with $l \in \{1, \dots, 100\}$. In this experiment, we set $M = 1, f(t) = \frac{1}{1+t}$ and $\sigma_i = 2^{j(i)}, \forall i$.

The pSNRs are averaged over the set of 16 images. The results of this experiment are displayed in Figure 13 for the two kernels from Figures 8b and 8a. Let us summarize the conclusions from this experiment:

- A clear fact is that windowed convolution methods are significantly outperformed by wavelet based methods for all blur kernels. Moreover, the differences between wavelet and windowed convolution based methods get larger as the blurs regularity decreases.
- A second result is that wavelet based methods with fixed sparsity patterns (Algo 2) are quite satisfactory for very sparse patterns (i.e. less than $20N$ operations) and kernels 8a and 8b. We believe that the most important regime for applications is in the range $[N, 20N]$, so that this result is rather positive. However, Algo 2 suffers from two important drawbacks: first, the increase in SNR after a certain value becomes very slow. Second, this algorithm provides very disappointing results for the last blur map 8c. These results suggest that this method should be used with caution if one aims at obtaining very good approximations. In particular, the algorithm is

dependent on the bound (9) which itself depends on user given parameters such as function f in (2a). Modifying those parameters might result in better results, but is usually hard to tweak manually.

- The methods Σ n° 1, Σ n° 2, *Thresholded matrix* all behave similarly. Method Σ n° 1 is however significantly better, showing the importance of choosing the weights σ_i in equation (15) carefully.
- The methods Σ n° 1, Σ n° 2, *Thresholded matrix* outperform the method proposed in [48] for very sparse patterns ($< 20N$) and get outperformed for mid-range sparsification $> 40N$. The main difference between algorithm [48] and the methods proposed in this paper is the number of vanishing moments. In [48], the authors propose using the Haar wavelet (i.e. 1 vanishing moment), while we use Daubechies wavelets with 10 vanishing moments. In practice, this results in better approximation properties in the very sparse regime, which might be the most important in applications. For mid-range sparsification, the Haar wavelet provides better results. Two reasons might explain this phenomenon: first, Haar wavelets have a small spatial support, therefore matrix Θ contains less non-zero coefficients when expressed with Haar wavelets than Daubechies wavelets. Second, the constants C'_M and C''_M in Theorem (2) are increasing functions of the number of vanishing moments.

Illustration of artefacts. Figure 14 provides a comparison of the windowed convolutions methods and the wavelet based approach in terms of approximation quality and computing times. The following conclusions can be drawn from this experiment:

- The residual artefacts appearing in the windowed convolutions approach and wavelet based approach are different. They are localized at the interfaces between sub-images for the windowed convolutions approach while they span the whole image domain for the wavelet based approach. It is likely that using translation and/or rotation invariant wavelet would improve the result substantially.
- The approximation using the second Σ matrix produces the best results and should be preferred over more simple approaches.
- In our implementation, the windowed convolutions approach (implemented in C) is outperformed by the wavelet based method (implemented in Matlab with C-mex files). For instance, for a precision of 45dBs, the wavelet based approach is about 10 times faster.
- The computing time of 1.21 seconds for the windowed convolutions approach with a 2×2 partition might look awkward since the computing times are significantly lower for finer partitions. This is because the efficiency of FFT methods depend greatly on the image size. The time needed to compute an FFT is usually lower for sizes

that have a prime factorization comprising only small primes (e.g. less than 7). This phenomenon explains the fact that the practical complexity of windowed convolutions algorithms may increase in a chaotic manner with respect to m .

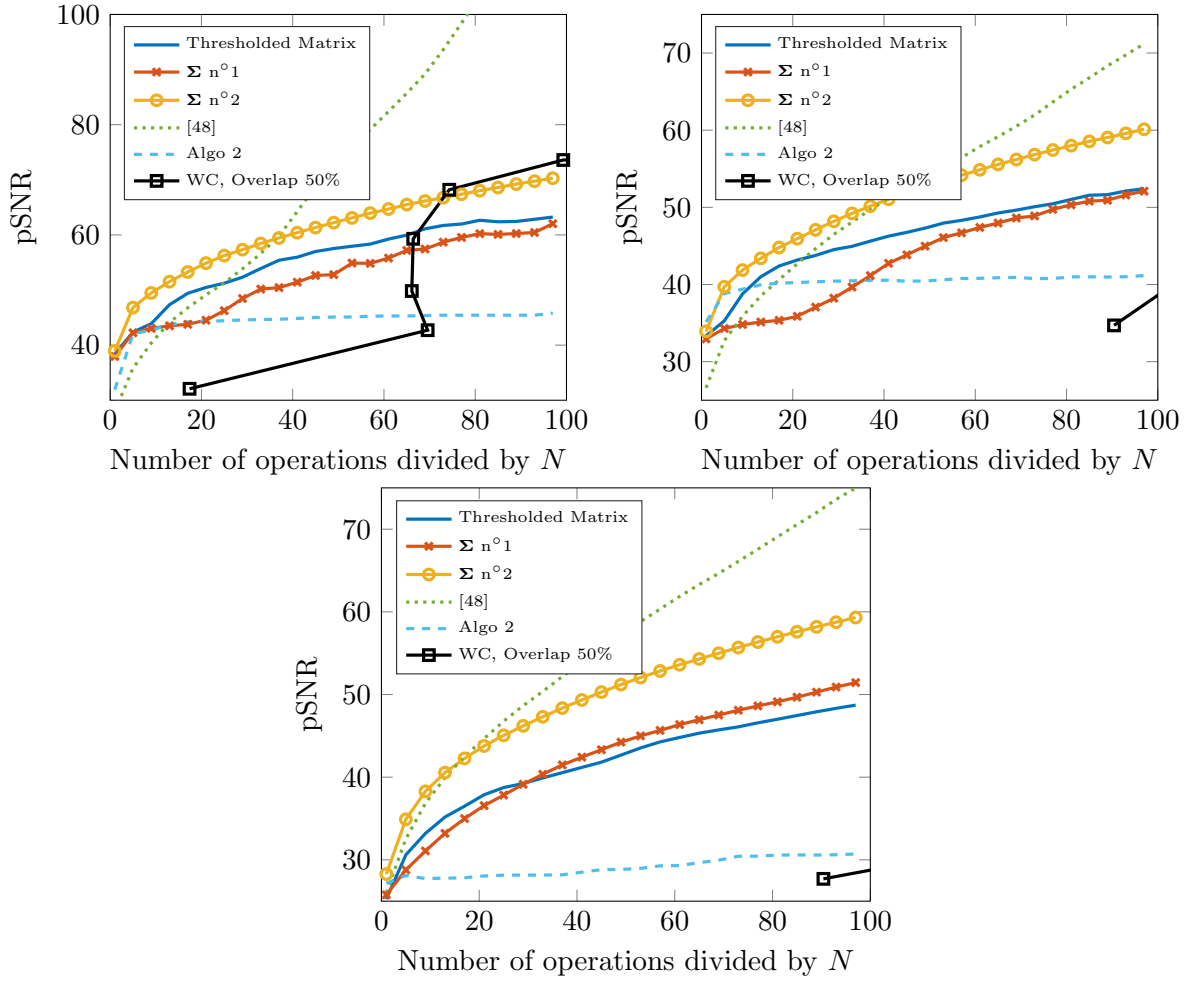


Figure 13: pSNR of the blurred image using the approximated operators $\tilde{\mathbf{H}}\mathbf{u}$ with respect to the blurred image using the exact operator $\mathbf{H}\mathbf{u}$. The results have been obtained with blur Figure 8a for top-left graph, blur Figure 8b for top-right graph and blur Figure 8c for the bottom. pSNR are averaged over the set of 16 images.









	Piece. Conv.	Difference	Algorithm 1	Difference	$l =$
2×2	31.90 dB		36.66 dB		5
1.21 sec					0.039 sec
4×4	38.49 dB		45.87 dB		30
0.17 sec					0.040 sec
8×8	44.51 dB		50.26 dB		50
0.36 sec					0.048 sec
16×16	53.75 dB		57.79 dB		100
0.39 sec					0.058 sec

Figure 14: Blurred images and the differences $\mathbf{Hu} - \tilde{\mathbf{H}}\mathbf{u}$ for the kernel Figure 8b. Results on the left are obtained using windowed convolutions approximations with 2×2 , 4×4 , 8×8 and 16×16 partitionings all with 50% overlap. Results on the right are obtained using Algorithm 1 with the second $\Sigma = \text{diag}(2^{j(i)})_i$ matrix keeping $K = lN$ coefficients. The pSNR and the time needed for the computation for the matrix-vector product are shown.

6.3 Application to inverse problems

In this experiment we compare the methods efficiency in deblurring problems. We assume the following classical image degradation model

$$\mathbf{v} = \mathbf{H}\mathbf{u} + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2 \text{Id}), \quad (21)$$

where \mathbf{v} is the degraded image observed, \mathbf{u} is the image to restore, \mathbf{H} in the blurring operator and σ^2 is the noise variance. A standard TV-L2 optimization problem is solved to restore the image \mathbf{u} :

$$\text{Find } \mathbf{u}^* \in \arg \min_{\mathbf{u} \in \mathbb{R}^N, \|\tilde{\mathbf{H}}\mathbf{u} - \mathbf{v}\|_2 \leq \alpha} TV(\mathbf{u}), \quad (22)$$

where $\tilde{\mathbf{H}}$ is an approximating operator and TV is the isotropic total variation of \mathbf{u} . The optimization problem is solved using the primal-dual algorithm proposed in [8]. We do not detail the resolution method since it is now well documented in the literature.

An important remark is that the interest of the total variation term is not only to regularize the ill-posed inverse problem, but also to handle the errors in the operator approximation. In practice we found that setting $\alpha = (1 + \epsilon)\sigma^2 N$ where $\epsilon > 0$ is a small parameter provides good experimental results.

In Figures 15 and 16, we present deblurring results using Figure 7b with kernel 8b.

In both the noisy and noiseless cases, the 4×4 windowed convolutions method performs worst reconstructions than wavelet approaches with $30N$. Moreover, they are between 4 and 6 times significantly slower. Surprisingly even the implementation in the space domain is faster. The reason for that is probably a difference in the quality of implementation: we use Matlab sparse matrix-vector products for space and wavelet methods. This routine is cautiously optimized while our c implementation of windowed convolutions can probably be improved. In addition, let us mention that two wavelet transforms need to be computed at each iteration with the wavelet based methods, while this is not necessary with the space implementation. It is likely that the acceleration factor would have been significantly higher if wavelet based regularizations had been used.

In the noiseless case, the simple thresholding approach provides significantly better SNRs than the more advanced proposed in this paper and in [48]. Note however that it produces more significant visual artefacts. This result might come as a surprise at first sight. However, as was explained in section 5, our aim to design sparsity patterns was to minimize an operator norm $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{X \rightarrow 2}$. When dealing with an inverse problem, approximating the direct operator is not as relevant as approximating its inverse. This calls for new methods specific to inverse problems.

In the noisy case, all three thresholding strategies produce results of a similar quality. The Haar wavelet transform is however about twice faster since the Haar wavelet support is smaller. Moreover, the results obtained with the approximated matrices are nearly as good

as the ones with the true operator. It suggests that it is not necessary to construct accurate approximations of the operators in practical problems. This observation is also supported by the experiment in Figure 17. In this experiment, we plot the pSNR of the deblurred image in presence of noise with respect to the number of elements in Θ_K . Interestingly, a matrix containing only $20N$ coefficients leads to deblurred images close to the results obtained with the exact operator. In this experiment, a total of $K = 5N$ coefficients in Θ_K is enough to retrieve satisfactory results. This is a very encouraging result for blind deblurring problems.

7 Conclusion

7.1 Brief summary

In this paper, we introduced an original method to represent spatially varying blur operators in the wavelet domain. We showed that this new technique has a great adaptivity to the smoothness of the operator and exhibit an $\mathcal{O}(N\epsilon^{-d/M})$ complexity, where M denotes the kernel regularity. This method is versatile since it is possible to adapt it to the kind of images that have to be treated. We showed that much better performance to approximate the direct operator can be obtained by leveraging the fact that natural signals exhibit some structure in the wavelet domain. Moreover, we proposed a original method to design sparsity patterns for class of blurring operators when only the operator regularity is known. These theoretical results were confirmed by practical experiments on real images. Even though our conclusions are still preliminary since we tested only small 256×256 images, the wavelet based methods seem to significantly outperform standard windowed convolutions based approaches. Moreover, they seem to provide satisfactory deblurring results on practical problems with a complexity no greater than $5N$ operations, where N denotes the pixels number.

7.2 Outlook

We provided a simple complexity analysis based solely on the *global* regularity of the kernel function. It is well known that wavelets are able to adapt locally to the structures of images or operators [11]. The method should thus provide an efficient tool for piecewise regular blurs appearing in computer vision for instance. It could be interesting to evaluate precisely the complexity of wavelet based approximations for piecewise regular blurs.

A key problem of the wavelet based approach is the need to project the operator on a wavelet basis. In this paper we performed this operation using the computationally intensive Algorithm 3. It could be interesting to derive fast projection methods. Let us note that such methods already exist in the literature [5]. A similar procedure was used in the specific context of spatially varying blur in [48].



(a) Degrated image
21.85dB



(b) Exact operator
34.53dB – 64.87 sec



(c) Simple thresh
31.68dB – 21.68 sec



(d) Algorithm 1
30.57dB – 21.16 sec

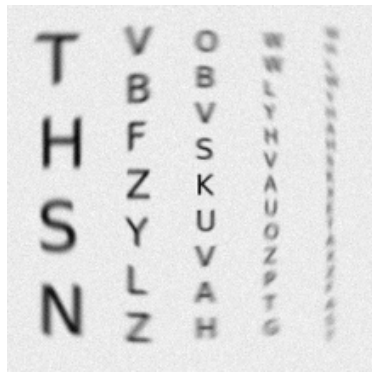


(e) WC 4×4
28.37dB – 85.60 sec



(f) [48]
30.53dB – 14.12 sec

Figure 15: Deblurring results for kernel Figure 8b and without noise. Top-left: degraded image. Top-right: deblurred using the exact operator. Middle-left: deblurred by the wavelet based method and a simple thresholding. Middle-right: deblurred by the wavelet based method and Algorithm 2 with the second $\Sigma = \text{diag}(2^{j(i)});$ matrix. Bottom: deblurred using a 4×4 windowed convolutions algorithm with 50% overlap. For wavelet methods $K = 30N$ coefficients are kept in matrices. pSNR are displayed for each restoration.



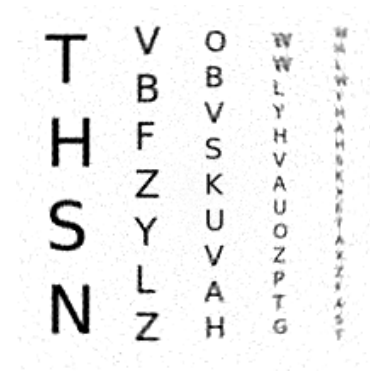
(a) Degraded image
21.62dB



(b) Exact operator
29.09dB – 64.87 sec



(c) Simple thresh
28.64dB – 21.68 sec



(d) Algorithm 1
28.24dB – 21.16 sec



(e) WC 4×4
27.62dB – 85.60 sec



(f) [48]
28.37dB – 14.12 sec

Figure 16: Deblurring results for kernel Figure 8b and with $\sigma = 0.02$ noise. Top-left: degraded image. Top-right: deblurred using the exact operator. Middle-left: deblurred by the wavelet based method and a simple thresholding. Middle-right: deblurred by the wavelet based method and Algorithm 2 with the second $\Sigma = \text{diag}(2^{j(i)})_i$ matrix. Bottom: deblurred using a 4×4 windowed convolutions algorithm with 50% overlap. For wavelet methods $K = 30N$ coefficients are kept in matrices. pSNR are displayed for each restoration.

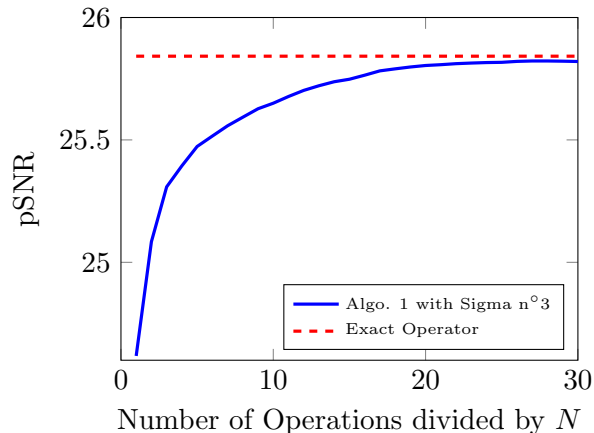


Figure 17: pSNR of the deblurred image with respect to the number of coefficients in the matrix divided by N for the image Figure 7a and the kernel Figure 8a. The matrix is constructed using Algorithm 1 with the second $\Sigma = \text{diag}(2^{j(i)})_i$ matrix with $K = lN$ coefficients for l from 1 to 30. Deblurred images using these matrices are compared with the one obtained with the exact operator.

Moreover, the proposed method can already be applied to situations where the blur mostly depends on the instrument: the wavelet representation has to be computed once for all off-line, and then all deblurring operations can be handled much faster. This situation occurs in satellite imaging or for some fluorescence microscopes (see e.g. [24, 46, 32]).

The design of good sparsity patterns is an open and promising research avenue. In particular, designing patterns adapted to specific inverse problems could have some impact as was illustrated in section 6.3.

Another exciting research perspective is the problem of blind deconvolution. Expressing the unknown operator as a sparse matrix in the wavelet domain is a good way to improve the problem identifiability. This is however far from being sufficient since the blind deconvolution problem has far more unknowns (a full operator and an image) than data (a single image). Further assumptions should thus be made on the wavelet coefficients regularity, and we plan to study this problem in a forthcoming work.

Finally let us mention that we observed some artefacts when using the wavelet based methods with high sparsity levels. This is probably due to their non translation and rotation invariance. It could be interesting to study sparse approximations in redundant wavelet bases or other time-frequency bases. It was shown for instance in [7] that curvelets are near optimal to represent Fourier integral operators. Similarly, Gabor frames are known to be very efficient to describe smoothly varying integral operators in the 1D setting [29].

Acknowledgements

The authors thank François Malgouyres and Mathieu Bouyrie for stimulating discussions in the early preparation of this work. They thank Jérémie Bigot and Guillermo Cabrera for providing the SDSS database address as an illustration of a deblurring problem with a spatially variant PSF. They thank Jérôme Fehrenbach and the IP3D team of ITAV for their comments and support. They thank Sandrine Anthoine, Caroline Chaux, Hans Feichtinger and Clothilde Mélot for their comments on the manuscript. They thank Michael Hirsch for providing the pictures in Figure 1 and indicating reference [28]. They thank Loïc Denis and Ferréol Soulez for indicating references [16, 48]. This work was supported by ANR SPH-IM-3D (ANR-12-BSV5-0008). Paul Escande is pursuing a PhD degree supported by the MODIM project funded by the PRES of Toulouse University and Midi-Pyrénées region.

References

- [1] G. Aubert and P. Kornprobst. *Mathematical problems in image processing: partial differential equations and the calculus of variations*, volume 147. Springer Science & Business Media, 2006.
- [2] L. Bar, N. Sochen, and N. Kiryati. Restoration of images with piecewise space-variant blur. In *Scale Space and Variational Methods in Computer Vision*, pages 533–544. Springer, 2007.
- [3] S. Bartels. Total variation minimization with finite elements: Convergence and iterative solution. *SIAM Journal on Numerical Analysis*, 50:1162–1180, 2012.
- [4] S. Ben Hadj and L. Blanc-Féraud. Restoration method for spatially variant blurred images. Research report, INRIA, June 2011.
- [5] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transform and numerical algorithm. *Commun. Pure and Applied Math.*, 44:141–183, 1991.
- [6] M. Born and E. Wolf. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. CUP Archive, 1999.
- [7] E. Candes and L. Demanet. Curvelets and fourier integral operators. *Comptes Rendus Mathématique*, 336:395–398, 2003.
- [8] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011.
- [9] E.-C. Chang, S. Mallat, and C. Yap. Wavelet foveation. *Applied and Computational Harmonic Analysis*, 9:312–335, 1999.

- [10] A. Cohen. *Numerical analysis of wavelet methods*, volume 32. Elsevier, 2003.
- [11] A. Cohen, W. Dahmen, and R. A. DeVore. Adaptive wavelet methods ii-beyond the elliptic case. *Foundations of Computational Mathematics*, 2:203–245, 2002.
- [12] A. Cohen, I. Daubechies, and P. Vial. Wavelets on the interval and fast wavelet transforms. *Applied and Computational Harmonic Analysis*, 1(1):54–81, 1993.
- [13] R. Coifman and Y. Meyer. Wavelets, Calderón-Zygmund and multilinear operators. *Cambridge Studies in Advanced Math*, 48, 1997.
- [14] W. Dahmen, S. Prössdorf, and R. Schneider. Wavelet approximation methods for pseudodifferential equations II: matrix compression and fast solution. *Advances in Computational Mathematics*, 1:259–335, Oct. 1993.
- [15] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, June 1992.
- [16] L. Denis, E. Thiébaud, F. Soulez, J.-M. Becker, and R. Mourya. Fast approximations of shift-variant blur. 2014.
- [17] J. Deny and J.-L. Lions. Les espaces du type de beppo levi. In *Annales de l’institut Fourier*, volume 5, pages 305–370. Institut Fourier, 1954.
- [18] P. Escande, P. Weiss, and F. Malgouyres. Image restoration using sparse approximations of spatially varying blur operators in the wavelet domain. *Proc. NCMIP*, 2013.
- [19] P. Escande, P. Weiss, and F. Malgouyres. Spatially varying blur recovery. diagonal approximations in the wavelet domain. *Proceedings of ICPRAM*, 2013.
- [20] C. Estatico and F. Di Benedetto. Shift-invariant approximations of structured shift-variant blurring matrices. *Numerical Algorithms*, 62(4):615–635, 2013.
- [21] H. G. Feichtinger and K. Nowak. A first survey of gabor multipliers. In *Advances in Gabor analysis*, pages 99–128. Springer, 2003.
- [22] R. C. Flicker and R. F. J. Anisoplanatic deconvolution of adaptive optics images. *Journal of the Optical Society of America*, 22:504–513, 2005.
- [23] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.
- [24] N. Hajlaoui, C. Chaux, G. Perrin, F. Falzon, and A. Benazza-Benyahia. Satellite image restoration in the context of a spatially varying point spread function. *JOSA A*, 27(6):1473–1481, 2010.
- [25] P. C. Hansen, J. G. Nagy, and D. P. O’leary. *Deblurring images: matrices, spectra, and filtering*. Siam, 2006.

- [26] E. Hernández and G. Weiss. *A first course on wavelets*. CRC press, 1996.
- [27] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Scholkopf. Fast removal of non-uniform camera shake. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 463–470. IEEE, 2011.
- [28] M. Hirsch, S. Sra, B. Scholkopf, and S. Harmeling. Efficient filter flow for space-variant multiframe blind deconvolution. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 607–614. IEEE, 2010.
- [29] T. Hrycak, S. Das, G. Matz, and H. G. Feichtinger. Practical estimation of rapidly varying channels for ofdm systems. *Communications, IEEE Transactions on*, 59(11):3040–3048, 2011.
- [30] R. Jorand, G. Le Corre, J. Andilla, A. Maandhui, C. Frongia, V. Lobjois, B. Ducommun, and C. Lorenzo. Deep and clear optical imaging of thick inhomogeneous samples. *PLoS ONE*, 7, 04 2012.
- [31] H. Kirshner, D. Sage, and M. Unser. 3D PSF models for fluorescence microscopy in ImageJ. In *Proceedings of the Twelfth International Conference on Methods and Applications of Fluorescence Spectroscopy, Imaging and Probes (MAF’11)*, page 154, 2011.
- [32] E. Maalouf, B. Colicchio, and A. Dieterlen. Fluorescence microscopy three-dimensional depth variant point spread function interpolation using zernike moments. *JOSA A*, 28:1864–1870, 2011.
- [33] S. Mallat. *A Wavelet Tour of Signal Processing – The Sparse Way*. Third Edition. Academic Press, 2008.
- [34] S. R. McNown and B. R. Hunt. Approximate shift-invariance by warping shift-variant systems. In *SPIE’s 1994 International Symposium on Optics, Imaging, and Instrumentation*, pages 156–167. International Society for Optics and Photonics, 1994.
- [35] Y. Meyer. *Wavelets and operators*, volume 2. Cambridge Univ Press, 1992.
- [36] J. G. Nagy and D. P. O’leary. Fast iterative image restoration with a spatially varying psf. In *Optical Science, Engineering and Instrumentation’97*, pages 388–399. International Society for Optics and Photonics, 1997.
- [37] J. G. Nagy and D. P. O’Leary. Restoring images degraded by spatially variant blur. *SIAM Journal on Scientific Computing*, 19:1063, 1998.
- [38] P. Petrushev, A. Cohen, H. Xu, and R. A. DeVore. Nonlinear approximation and the space $bv(r, 2)$. *American Journal of Mathematics*, 121:587–628, 1999.

- [39] C. Preza and J.-A. Conchello. Depth-variant maximum-likelihood restoration for three-dimensional fluorescence microscopy. *JOSA A*, 21:1593–1601, 2004.
- [40] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992.
- [41] A. Sawchuk. Space-variant image motion degradation and restoration. *Proceedings of the IEEE*, 60:854–861, 1972.
- [42] A. A. Sawchuk. Space-variant image restoration by coordinate transformations. *Journal of the Optical Society of America*, 64:138–144, Feb. 1974.
- [43] E. P. Simoncelli. Modeling the joint statistics of images in the wavelet domain. In *SPIE’s International Symposium on Optical Science, Engineering, and Instrumentation*, pages 188–195. International Society for Optics and Photonics, 1999.
- [44] J. Simpkins and R. L. Stevenson. Parameterized modeling of spatially varying optical blur. *Journal of Electronic Imaging*, 23(1):013005–013005, 2014.
- [45] A. Taberero, J. Portilla, and R. Navarro. Duality of log-polar image representations in the space and spatial-frequency domains. *Signal Processing, IEEE Transactions on*, 47:2469–2479, 1999.
- [46] M. Temerinac-Ott, O. Ronneberger, P. Ochs, W. Driever, T. Brox, and H. Burkhardt. Multiview deblurring for 3-d images from light-sheet-based fluorescence microscopy. *Image Processing, IEEE Transactions on*, 21(4):1863–1873, 2012.
- [47] J. Wang and B. J. Lucier. Error bounds for finite-difference methods for rudin-osher-fatemi image smoothing. *SIAM Journal on Numerical Analysis*, 49:845–868, 2011.
- [48] J. Wei, C. A. Bouman, and J. P. Allebach. Fast space-varying convolution using matrix source coding with applications to camera stray light reduction. *IEEE Transactions on Image Processing*, 23:1965–1979, 2014.
- [49] J. Wei, G. Cao, C. A. Bouman, and J. P. Allebach. Fast space-varying convolution and its application in stray light reduction. In *IS&T/SPIE Electronic Imaging*, pages 72460B–72460B. International Society for Optics and Photonics, 2009.
- [50] B. Zhang, J. Zerubia, and J.-C. Olivo-Marin. Gaussian approximations of fluorescence microscope point-spread function models. *Applied Optics*, 46:1819–1829, 2007.

A Proof of Lemma 1

We let Π_M denote the set of polynomials of degree less or equal to M .

Lemma 3 below is a common result in numerical analysis [17] (see also Theorem 3.2.1 in [10]). It ensures that the approximation error of a function by a polynomial of degree M is bounded by the Sobolev semi-norm $W^{M,p}$.

Lemma 3 (Polynomial approximation). *For $1 \leq p \leq +\infty$, $M \in \mathbb{N}^*$ and $\Omega \subset \mathbb{R}^d$ a bounded domain, the following bound holds*

$$\inf_{g \in \Pi_M} \|f - g\|_{\mathbb{L}^p(\Omega)} \leq C |f|_{W^{M+1,p}(\Omega)}, \quad (23)$$

where C is a constant that depends on d, M, p and Ω only.

Moreover, if $I_h \subset \Omega \subset \mathbb{R}^d$ is a cube of sidelength h , the following estimate holds

$$\inf_{g \in \Pi_M} \|f - g\|_{\mathbb{L}^p(I_h)} \leq Ch^{M+1} |f|_{W^{M+1,p}(I_h)}, \quad (24)$$

where C is a constant only depending on d, M, p and Ω .

Let $I_\lambda = \text{supp}(\psi_\lambda)$. From the wavelets definition, we get

$$I_\lambda = 2^{-j}(m + [-c(M)/2, c(M)/2]^d)$$

therefore $|I_\lambda| = c(M)^d \cdot 2^{-jd}$. We will now prove Lemma 1.

Proof of Lemma 1. Since the mapping $(x, y) \mapsto K(x, y)\psi_\lambda(y)\psi_\mu(x)$ is bounded, it is also absolutely integrable on compact domains. Therefore $\langle H\psi_\lambda, \psi_\mu \rangle$ is well-defined for all (λ, μ) . Recall that $\lambda = (j, m, e) \in \Lambda$ and $\mu = (k, n, e') \in \Lambda$. Moreover Fubini's theorem can be applied and we get

$$\begin{aligned} \langle H\psi_\lambda, \psi_\mu \rangle &= \int_{I_\mu} \int_{I_\lambda} K(x, y)\psi_\lambda(y)\psi_\mu(x)dydx \\ &= \int_{I_\lambda} \int_{I_\mu} K(x, y)\psi_\lambda(y)\psi_\mu(x)dx dy. \end{aligned}$$

To prove the result, we distinguish the cases $j \leq k$ and $j > k$. In this proof, we focus on the case $j \leq k$. The other one can be obtained by symmetry, using the facts that $\langle H\psi_\lambda, \psi_\mu \rangle = \langle \psi_\lambda, H^*\psi_\mu \rangle$ and that H and H^* are both blurring operators in the same class.

To exploit the regularity of K and ψ , note that for all $g \in \Pi_{M-1}$, $\int_{I_\mu} g(x)\psi_\mu(x)dx = 0$ since ψ has M vanishing moments. Therefore,

$$\langle H\psi_\lambda, \psi_\mu \rangle = \int_{I_\lambda} \inf_{g \in \Pi_{M-1}} \int_{I_\mu} (K(x, y) - g(x))\psi_\lambda(y)\psi_\mu(x)dx dy,$$

and

$$|\langle H\psi_\lambda, \psi_\mu \rangle| \leq \int_{I_\lambda} \inf_{g \in \Pi_{M-1}} \int_{I_\mu} |K(x, y) - g(x)| |\psi_\lambda(y)| |\psi_\mu(x)| dx dy$$

$$\leq \int_{I_\lambda} \inf_{g \in \Pi_{M-1}} \|K(\cdot, y) - g\|_{\mathbb{L}^\infty(I_\mu)} \|\psi_\mu\|_{\mathbb{L}^1(I_\mu)} |\psi_\lambda(y)| dy.$$

By Lemma 3, $\inf_{g \in \Pi_{M-1}} \|K(\cdot, y) - g\|_{\mathbb{L}^\infty(I_\mu)} \lesssim 2^{-kM} |K(\cdot, y)|_{W^{M,\infty}(I_\mu)}$ since I_μ is a cube of sidelength $c(M) \cdot 2^{-k}$. We thus obtain

$$\begin{aligned} |\langle H\psi_\lambda, \psi_\mu \rangle| &\lesssim 2^{-kM} \|\psi_\mu\|_{\mathbb{L}^1(I_\mu)} \|\psi_\lambda\|_{\mathbb{L}^1(I_\lambda)} \operatorname{ess\,sup}_{y \in I_{j,m}} |K(\cdot, y)|_{W^{M,\infty}(I_\mu)} \\ &\lesssim 2^{-kM} 2^{-\frac{dj}{2}} 2^{-\frac{dk}{2}} \operatorname{ess\,sup}_{y \in I_\lambda} |K(\cdot, y)|_{W^{M,\infty}(I_\mu)} \end{aligned}$$

since $\|\psi_\lambda\|_{\mathbb{L}^1} = 2^{-\frac{dj}{2}} \|\psi\|_{\mathbb{L}^1}$.

Since $H \in \mathcal{A}(M, f)$

$$\begin{aligned} \operatorname{ess\,sup}_{y \in I_\lambda} |K(\cdot, y)|_{W^{M,\infty}(I_\mu)} &= \operatorname{ess\,sup}_{y \in I_\lambda} \sum_{|\alpha|=M} \operatorname{ess\,sup}_{x \in I_\mu} |\partial_x^\alpha K(x, y)| \\ &\leq \sum_{|\alpha|=M} \operatorname{ess\,sup}_{(x,y) \in I_\lambda \times I_\mu} f(\|x - y\|_\infty) \\ &\lesssim \operatorname{ess\,sup}_{(x,y) \in I_\lambda \times I_\mu} f(\|x - y\|_\infty). \end{aligned}$$

Because f is a non-increasing function, $f(\|x - y\|_\infty) \leq f(\operatorname{dist}(I_\lambda, I_\mu))$ since $\operatorname{dist}(I_\lambda, I_\mu) = \inf_{(x,y) \in I_\lambda \times I_\mu} \|x - y\|_\infty$. Therefore

$$\begin{aligned} |\langle H\psi_\lambda, \psi_\mu \rangle| &\lesssim 2^{-kM} 2^{-\frac{dj}{2}} 2^{-\frac{dk}{2}} f(\operatorname{dist}(I_\lambda, I_\mu)) \\ &= 2^{-(M+\frac{d}{2})|j-k|} 2^{-j(M+d)} f(\operatorname{dist}(I_\lambda, I_\mu)). \end{aligned}$$

The case $k < j$ gives

$$|\langle H\psi_\lambda, \psi_\mu \rangle| \lesssim 2^{-(M+\frac{d}{2})|j-k|} 2^{-k(M+d)} f(\operatorname{dist}(I_\lambda, I_\mu)),$$

which allows to conclude that

$$|\langle H\psi_\lambda, \psi_\mu \rangle| \lesssim 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} f(\operatorname{dist}(I_\lambda, I_\mu)),$$

□

B Proof of Theorem 2

Let us begin with some preliminary results. Recall that $\lambda = (j, m, e) \in \Lambda$ and $\mu = (k, n, e') \in \Lambda$. Since f is compactly supported on $[0, \kappa]$ and bounded by c_f , we have $f_{\lambda,\mu} = f(\operatorname{dist}(I_\lambda, I_\mu)) \leq c_f \mathbb{1}_{\operatorname{dist}(I_\lambda, I_\mu) \leq \kappa}$. By equation (10), $\operatorname{dist}(I_\mu, I_\lambda) \leq \kappa$ if $\|2^{-j}m - 2^{-k}n\|_\infty \leq R_{j,k}^\kappa$, where $R_{j,k}^\kappa = (2^{-j} + 2^{-k})c(M)/2 + \kappa$.

Lemma 4. *Define*

$$\mathcal{G}_{j,k}^{e,e'} = \{(m, n) \in \mathcal{T}_j \times \mathcal{T}_k \mid \mathbf{1}_{\text{dist}(I_\lambda, I_\mu) \leq \kappa} = 1\}.$$

Then $|\mathcal{G}_{j,k}^{e,e'}| \leq (2^j 2^{k+1} R_{j,k}^\kappa)^d$.

Proof. First note that

$$\mathcal{G}_{j,k}^{e,e'} = \{(m, n) \in \mathcal{T}_j \times \mathcal{T}_k \mid |2^{-j} m_i - 2^{-k} n_i| \leq R_{j,k}^\kappa, \quad \forall i \in \{1, \dots, d\}\}.$$

Now, define $\mathcal{G}_{j,k,m}^{e,e'} = \{n \in \mathcal{T}_k \mid (m, n) \in \mathcal{G}_{j,k}^{e,e'}\}$. For a fixed (j, k, m, e, e') the set $\mathcal{G}_{j,k,m}^{e,e'}$ is a discrete hyper-cube of sidelength bounded above by $2^{k+1} R_{j,k}^\kappa$. Therefore $|\mathcal{G}_{j,k,m}^{e,e'}| \leq (2^{k+1} R_{j,k}^\kappa)^d$ coefficients. Moreover, $|\mathcal{T}_j| = 2^{jd}$, hence the number of coefficients in $\mathcal{G}_{j,k}^{e,e'}$ is bounded above by $(2^j 2^k R_{j,k}^\kappa)^d$. \square

Proof of i). We denote $J_{\max} = \log_2(N)/d$ the highest scale of decomposition. First note that a sufficient condition for $2^{-\min(j,k)(M+d)} f_{\lambda,\mu} \leq \eta$ is that $\min(j, k) \geq J(\eta)$ with $J(\eta) = \frac{-\log_2(\eta/c_f)}{M+d}$. In the following, we let $\tilde{J}(\eta) = \min(J(\eta), J_{\max})$ and define

$$\mathcal{G} = \bigcup_{\min(j,k) < J(\eta)} \bigcup_{e, e' \in \{0,1\}^d \setminus \{0\}} \mathcal{G}_{j,k}^{e,e'}.$$

The overall number of non zero coefficients $|\mathcal{G}|$ in Θ_η satisfies

$$\begin{aligned} \#\mathcal{G} &= \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \sum_{e, e' \in \{0,1\}^d} \#\mathcal{G}_{j,k}^{e,e'} \mathbf{1}_{\min(j,k) < J(\eta)} \\ &\lesssim (2^d - 1)^2 \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} 2^{kd} \left(\frac{c(M)}{2} (2^{-j} + 2^{-k}) + \kappa \right)^d \\ &\lesssim \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} 2^{kd} \left(\frac{c(M)^d}{2^d} 2^{-dj} + \frac{c(M)^d}{2^d} 2^{-dk} + \kappa^d \right) \\ &\lesssim \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} + \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} \\ &\quad + \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} 2^{jd} \kappa^d. \end{aligned}$$

The first sum yields

$$\begin{aligned}
& \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} \\
&= \left(\sum_{j=0}^{\tilde{J}(\eta)-1} \sum_{k=j}^{J_{\max}-1} 2^{kd} + \sum_{k=0}^{\tilde{J}(\eta)-1} 2^{kd} \sum_{j=k}^{J_{\max}-1} 1 \right) \\
&\lesssim \tilde{J}(\eta)N + 2^{d\tilde{J}(\eta)} \log_2(N) \lesssim \log_2(N)N.
\end{aligned}$$

The second sum is handled similarly and the third sum gives

$$\begin{aligned}
& \sum_{j=0}^{J_{\max}-1} \sum_{k=0}^{J_{\max}-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} 2^{kd} \kappa^d \\
&= \kappa^d \sum_{j=0}^{\tilde{J}(\eta)-1} 2^{jd} \sum_{k=j}^{J_{\max}-1} 2^{kd} + \sum_{k=0}^{\tilde{J}(\eta)-1} 2^{kd} \sum_{j=k}^{J_{\max}-1} 2^{jd} \\
&\lesssim \kappa^d N 2^{d\tilde{J}(\eta)}.
\end{aligned}$$

Overall $|\mathcal{G}| \lesssim \log_2(N)N + \eta^{-\frac{d}{M+d}}N$. For $\eta \leq \log_2(N)^{-(M+d)/d}$, the dominating terms are of kind $\eta^{-\frac{d}{M+d}}$, hence $|\mathcal{G}| \lesssim \eta^{-\frac{d}{M+d}}N\kappa^d$. \square

Proof of ii). Since Ψ is an orthogonal wavelet transform

$$\left\| \mathbf{H} - \tilde{\mathbf{H}}_\eta \right\|_{2 \rightarrow 2} = \left\| \Theta - \Theta_\eta \right\|_{2 \rightarrow 2}.$$

Let $\Delta_\eta = \Theta - \Theta_\eta$. We will make use of the following version of Shur inequality

$$\left\| \Delta_\eta \right\|_{2 \rightarrow 2}^2 \leq \left\| \Delta_\eta \right\|_{1 \rightarrow 1} \left\| \Delta_\eta \right\|_{\infty \rightarrow \infty}. \quad (25)$$

Since the upper-bound (9) is symmetric,

$$\left\| \Delta_\eta \right\|_{\infty \rightarrow \infty} = \left\| \Delta_\eta \right\|_{1 \rightarrow 1} = \max_{\lambda \in \Lambda} \sum_{\mu \in \Lambda} |\Delta_{\lambda, \mu}|$$

By definition of Θ_η we get that

$$\sum_{\mu \in \Lambda} |\Delta_{\lambda, \mu}| = \sum_{k=0}^{J_{\max}-1} \sum_{e' \in \{0,1\}^d \setminus \{0\}} \sum_{n \in \mathcal{G}_{j,k,m}^{e,e'}} |\theta_{\lambda, \mu}| \mathbf{1}_{\min(j,k) > J(\eta)}$$

$$\lesssim \sum_{k=0}^{J_{\max}-1} \sum_{e' \in \{0,1\}^d \setminus \{0\}} \sum_{n \in \mathcal{G}_{j,k,m}^{e,e'}} 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} \mathbf{1}_{\min(j,k) > J(\eta)}.$$

Then

$$\begin{aligned} \sum_{\mu \in \Lambda} |\Delta_{\lambda, \mu}| &\lesssim \sum_{k=0}^{J_{\max}-1} 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} \mathbf{1}_{\min(j,k) > J(\eta)} \left| \mathcal{G}_{j,k}^{e,e'} \right| \\ &\lesssim \sum_{k=0}^{j-1} (2^k R_{j,k}^\kappa)^d 2^{(k-j)(M+d/2)} 2^{-k(M+d)} \mathbf{1}_{k > J(\eta)} \\ &\quad + \sum_{k=j}^{J_{\max}-1} (2^k R_{j,k}^\kappa)^d 2^{(j-k)(M+d/2)} 2^{-j(M+d)} \mathbf{1}_{j > J(\eta)}. \end{aligned}$$

The first sum on $k < j$ is equal to

$$\begin{aligned} A_1 &= 2^{-jM} 2^{-jd/2} \sum_{k=0}^{j-1} (2^{k/2} R_{j,k}^\kappa)^d \mathbf{1}_{k > J(\eta)} \\ &= 2^{-jM} 2^{-jd/2} \mathbf{1}_{j > J(\eta)} \sum_{k=J(\eta)}^{j-1} (2^{k/2} R_{j,k}^\kappa)^d. \end{aligned}$$

The second sum on $k \geq j$ is:

$$A_2 = \mathbf{1}_{j > J(\eta)} 2^{-jd/2} \sum_{k=j}^{J_{\max}-1} (R_{j,k}^\kappa)^d 2^{-k(M-d/2)}.$$

Now, notice that $(R_{j,k}^\kappa)^d \lesssim 2^{-jd} + 2^{-kd} + \kappa^d$. Thus

$$\begin{aligned} A_1 &\lesssim 2^{-jM} 2^{-jd/2} \mathbf{1}_{j > J(\eta)} \sum_{k=J(\eta)}^{j-1} \left(2^{dk/2} 2^{-jd} + 2^{-dk/2} + 2^{kd/2} \kappa^d \right) \\ &\lesssim 2^{-jM} 2^{-jd/2} \mathbf{1}_{j > J(\eta)} \left(2^{-jd} 2^{jd/2} + 2^{-\frac{d}{2}J(\eta)} + \kappa^d 2^{jd/2} \right) \\ &= 2^{-jM} \mathbf{1}_{j > J(\eta)} \left(2^{-jd} + 2^{-\frac{d}{2}(J(\eta)+j)} + \kappa^d \right). \end{aligned}$$

And

$$A_2 \lesssim \mathbf{1}_{j > J(\eta)} 2^{-jd/2} \sum_{k=j}^{J_{\max}-1} \left(2^{-jd} + 2^{-kd} + \kappa^d \right) 2^{-k(M-d/2)}$$

$$\begin{aligned}
&\lesssim \mathbf{1}_{j>J(\eta)} 2^{-jd/2} \left(2^{-jd} 2^{-j(M-d/2)} + 2^{-j(M+d/2)} + \kappa^d 2^{-j(M-d/2)} \right) \\
&\lesssim \mathbf{1}_{j>J(\eta)} 2^{-jM} \left(2^{-jd} + \kappa^d \right).
\end{aligned}$$

Hence

$$\sum_{\mu \in \Lambda} |\Delta_{\lambda, \mu}| \lesssim \mathbf{1}_{j>J(\eta)} 2^{-jM} \left(2^{-jd} + \kappa^d + 2^{-\frac{d}{2}(J(\eta)+j)} \right).$$

Therefore

$$\begin{aligned}
\|\Delta_\eta\|_{1 \rightarrow 1} &\lesssim 2^{-J(\eta)M} \left(2^{-J(\eta)d} + \kappa^d + 2^{-dJ(\eta)} \right) \\
&\lesssim 2^{-J(\eta)M} \left(2^{-J(\eta)d} + \kappa^d \right) \\
&\lesssim \eta + \kappa^d \eta^{\frac{M}{M+d}} \\
&\lesssim \kappa^d \eta^{\frac{M}{M+d}} \quad \text{for small } \eta.
\end{aligned}$$

Finally, we can see that there exists a constant C_M independent of N such that

$$\|\Delta_\eta\|_{1 \rightarrow 1} \leq C_M \kappa^d \eta^{\frac{M}{M+d}} \quad \text{and} \quad \|\Delta_\eta\|_{\infty \rightarrow \infty} \leq C_M \kappa^d \eta^{\frac{M}{M+d}}.$$

It suffices to use inequality (25) to conclude. □

Proof of iii). This is a direct consequence of point i) and ii). □