



Complexity of Spatially Varying Blur Approximations. Piecewise Convolutions VS Sparse Wavelet Representations.

Paul Escande, Pierre Weiss

► To cite this version:

Paul Escande, Pierre Weiss. Complexity of Spatially Varying Blur Approximations. Piecewise Convolutions VS Sparse Wavelet Representations.. 2014. hal-00971549v5

HAL Id: hal-00971549

<https://hal.science/hal-00971549v5>

Preprint submitted on 23 Jan 2015 (v5), last revised 9 Oct 2015 (v6)

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.

Complexity of Spatially Varying Blur Approximations. Piecewise Convolutions VS Sparse Wavelet Representations.

Paul Escande [†]

Pierre Weiss [‡]

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 review the main approaches developed so far and detail their pros and cons. We then analyze the numerical complexity of the mainstream approach based on piecewise convolutions. We show that this method provides an ϵ -approximation of the matrix-vector product in $\mathcal{O}(N^d \log(N) \epsilon^{-d})$ operations where N^d is the number of pixels of a d -dimensional image. Moreover, we show that this bound cannot be improved even if further assumptions on the kernel regularity are made.

We then introduce a new method based on a sparse approximation of the blurring operator in the wavelet domain. This method requires $\mathcal{O}(N^d \epsilon^{-d/M})$ operations to provide ϵ -approximations, where $M \geq 1$ is a scalar describing the regularity of the blur kernel. We then propose variants to further improve the method by exploiting the fact that both images and operators are sparse in the same wavelet basis.

We finish by numerical experiments to illustrate the practical efficiency of the proposed algorithms.

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

[†]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

are shown in Figures 1, 2 and 3 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 $N^d \times N^d$, where N^d represents the number of pixels of a square d -dimensional image. Sizes of typical images range from $N^d = 10^6$ for small 2D images, to $N^d = 10^{10}$ 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, this assumption is not satisfied in many practical applications and it cannot be applied whenever super-resolution is targeted, since the PSFs supports depends on the image resolution. Spatially varying blurring matrices therefore require the development of computational tools to compress and evaluate them in an efficient way.

To the best of our knowledge, the first attempts to address this issue appeared at the beginning of the seventies (see e.g. [37]). Since then, many techniques were proposed and we will describe them in a short account in Section 4. The most broadly applied approach nowadays is based on what will be coined “piecewise convolutions” in this paper. In a nutshell, this approach consists in decomposing images into squares and to apply a space invariant blur on each element of the decomposition separately. This approach was described precisely in [33] and is still an active topic of research (see e.g. [23, 15]).

Contributions of the paper Our first contribution is a theoretical analysis of a simple version of this technique. We address the following question: how many floating point operations are necessary to construct a piecewise convolution approximation $\tilde{\mathbf{H}}$ of \mathbf{H} such that the spectral norm $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2} \leq \epsilon$? Under mild regularity assumptions on the PSF, we show that the answer is $\mathcal{O}(N^d \log(N) \epsilon^{-d})$ operations. Moreover, we show that this bound is tight: it holds whatever the regularity of the PSFs variations.

Our second 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 [4, 31], 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 $\mathcal{O}(N^d \epsilon^{-d/M})$ operations. In this complexity bound, $M \geq 1$ is an integer that describes the smoothness of the blur kernel. For $M \geq 2$, this new bound is significantly better than the one for piecewise convolutions.

Controlling the spectral norm is usually of little relevance in image processing. Our third 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.

We finish the paper by numerical experiments. We show that the proposed algorithms

allow significant speed ups compared to some versions of piecewise convolutions.

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 review the main existing computational strategies and analyse their pros and cons in Section 4. By far, in the literature, the most commonly approach used to blur images consists in using piecewise convolutions [32, 33, 23, 15]. Since this technique is so widespread, we propose an original analysis of its theoretical efficiency in Section 5. We introduce the proposed method and analyze its theoretical efficiency Section 6. We then propose various algorithms to design good sparsity patterns in Section 7. Finally, we perform numerical tests to analyze the proposed method and compare it to the standard methods based on piecewise convolutions in Section 8.

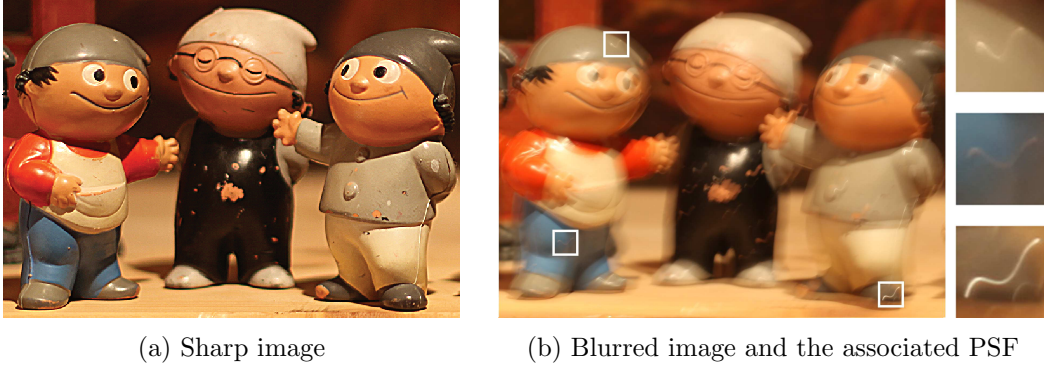


Figure 1: An example in computer vision. Image degraded by spatially varying blur due to a camera shake. Images are from [22] and used here by courtesy of Michael Hirsch.

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

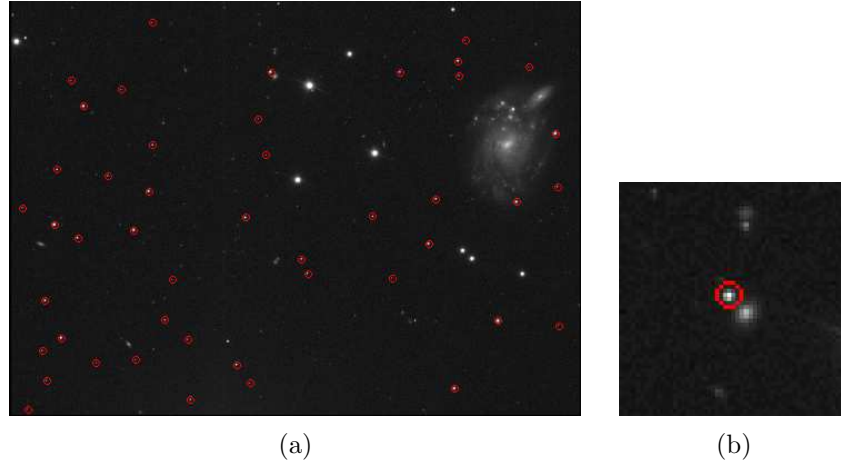


Figure 2: An example in astronomy. Image degraded by spatially varying blur due to atmosphere turbulence <http://www.sdss.org/>.

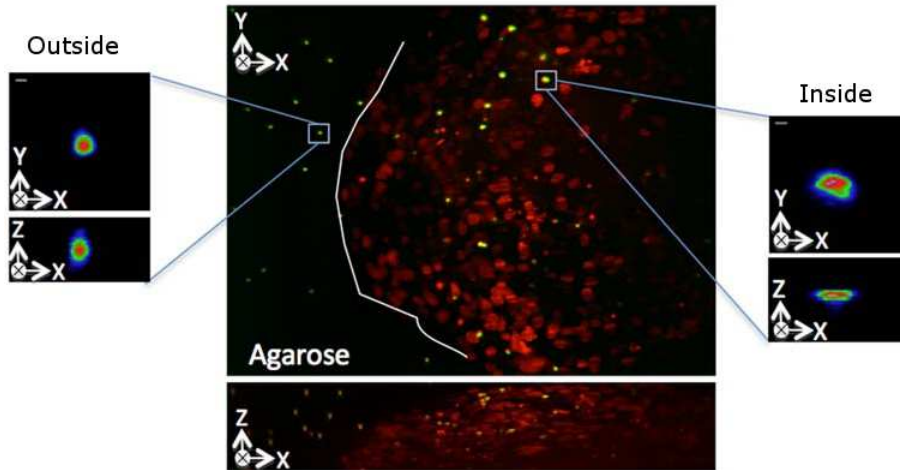


Figure 3: An example in biology. Image of a multicellular tumor spheroid imaged in 3D using Selective Plane Illumination Microscope (SPIM). Fluorescence beads (in green) are inserted in the tumor model and allow the observation of the PSF at different locations. Nuclei are stained in red. On the left-hand-side the 3D PSF outside of the sample is observed. On the right-hand-side the 3D PSF inside the sample is observed. This image is from [25] and used here by courtesy of Corinne Lorenzo.

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., [29, Theorem 7.9, p. 294].

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

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

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., [29, 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 $e = (e_1, \dots, e_d) \in \{0, 1\}^d$. Wavelet $\psi_{j,m}^e$ is defined by $\psi_{j,m}^e(x_1, \dots, x_d) = \rho_{j,m_1}^{e_1}(x_1) \dots \rho_{j,m_d}^{e_d}(x_d)$. 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 $I_{j,m} = \cup_e \text{supp } \psi_{j,m}^e$.

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

$$u = \langle u, \psi_{0,0}^0 \rangle \psi_{0,0}^0 + \sum_{e \in \{0,1\}^d \setminus \{0\}} \sum_{j=0}^{+\infty} \sum_{m \in \mathcal{T}_j} \langle u, \psi_{j,m}^e \rangle \psi_{j,m}^e.$$

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 [11] 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 [29, 14, 11] 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 $K \in \mathcal{C}^M(\Omega \times \Omega)$;
2. The partial derivatives of K satisfy:

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

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

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. [5]).

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 objects move in front of a still background, the PSF 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 (e.g. spatially varying motion blur), the discontinuities locations are usually known only approximately and it seems important to smooth the transitions in order to avoid reconstruction artifacts.

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 A brief review of existing approximation methods

Various approaches have been proposed in the literature to approximate the integral operator (3). In this section, we review some of these methods. We briefly discuss their pros and cons.

In many situations, $K(x, \cdot)$ and $K(\cdot, y)$ have a bounded support and this property can be exploited to accelerate computations. In the following we therefore assume that the integral operator $H \in \mathcal{A}(0, f_\kappa)$ where f_κ is defined by:

$$f_\kappa(r) = \begin{cases} 1 & \text{if } r \leq \kappa \\ 0 & \text{otherwise,} \end{cases}$$

and $\kappa \in]0, 1]$ is the maximal radius of the PSFs of the operator and its adjoint.

4.1 Direct discretization

The most straightforward approach to approximate the product Hu numerically consists of discretizing (3) using the rectangle rule. Let $\mathbf{\Omega} = \{k/N, k \in \{0, \dots, N-1\}^d\}$ denote a Euclidean discretization of Ω and $\mathbf{u} \in \mathbb{R}^{N^d}$ denote a discretization of u on $\mathbf{\Omega}$. The product Hu can be approximated by $\mathbf{H}\mathbf{u}$ where \mathbf{H} is an $N^d \times N^d$ matrix defined by

$$\mathbf{H} = \frac{1}{N^d} (K(x, y))_{x \in \mathbf{\Omega}, y \in \mathbf{\Omega}}. \quad (6)$$

The rationale behind this discretization is that if u and H are sufficiently smooth

$$\forall x \in \mathbf{\Omega}, \quad Hu(x) \simeq \frac{1}{N^d} \sum_{y \in \mathbf{\Omega}} K(x, y) u(y). \quad (7)$$

This approach is simple to implement but a matrix vector product costs $\mathcal{O}(N^{2d})$ arithmetic operations. By taking into account the boundedness of the kernel support, one can easily reduce the complexity to $\mathcal{O}(\kappa^d N^{2d})$. Indeed, for each of the N^d pixels, the method performs $(\kappa N)^d$ operations (additions and multiplications).

This method has the advantages of being straightforward to understand and implement. It is also easily parallelizable. It is thus a suitable method when the PSF width remains small over the image domain. However, it becomes unusable whenever the image size and the PSF sizes become large, i.e., in most practical settings.

Note that other quadrature formulas can be used and improve the approximation quality.

4.2 Composition of a diffeomorphism and a convolution

One of the first alternative method proposed to reduce the computational complexity, is based on first applying a diffeomorphism to the image domain [37, 38, 30, 39] 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 [30],
- second, this method requires good interpolation methods and the use of Euclidean grids with small grid size in order to correctly estimate integrals.

4.3 Approximation by separable kernels

This approach was described in [1, 26]. The main idea is to approximate the kernel K of H by a separable kernel \tilde{K} that reads:

$$\tilde{K}(x, y) = \prod_{k=1}^d \tilde{K}_k(x_k, y_k)$$

where each $\tilde{K}_k : [0, 1] \times [0, 1] \rightarrow \mathbb{R}$ operates in only one direction. With this assumption, the approximation operator \tilde{H} can be decomposed as the product of d one-dimensional operators $\tilde{H}^{(k)}$

$$\tilde{H} = \tilde{H}^{(1)} \circ \dots \circ \tilde{H}^{(d)}. \quad (8)$$

with

$$\forall x \in \Omega, \quad \tilde{H}^{(k)}u(x) = \int_{y_k \in [0,1]} \tilde{K}_k(x_k, y_k) u(x_1, \dots, y_k, \dots, x_d) dy_k.$$

In the discrete setting, the computational complexity of a product $\tilde{\mathbf{H}}\mathbf{u}$ is $\mathcal{O}(d\kappa N^{d+1})$ operations where κ denotes the PSF extent in pixels. The complexity of a product is thus reduced by a factor $\kappa^{d-1}N^{d-1}$ compared to the standard discretization described in paragraph 4.1.

The separability assumption (8) implies that $y \mapsto K(\cdot, y)$ is a separable function, meaning that the PSFs are separable. Moreover, it implies that $x \mapsto K(x, \cdot)$ is a separable function, meaning that the PSFs variations are also separable. Unfortunately, most physically realistic PSFs are not separable (see, e.g., Figure 4). Furthermore, the separability of the PSFs variations is an extremely restrictive assumption. There are however a few cases where this approximation might be sound. For instance, in 3D fluorescence microscopy, it is common to approximate the PSFs by anisotropic Gaussians [43], and to assume that the Gaussian variances only vary along one direction (e.g., the propagation of light direction) [35, 28, 3].

4.4 Diagonal approximation of blurring operators in wavelet or wavelet packet bases

Some works [8, 17] proposed to approximate blurring operators \mathbf{H} using operators diagonal in wavelet or wavelet packet bases. This idea consists in defining an approximation $\tilde{\mathbf{H}}$ of kind $\tilde{\mathbf{H}} = \mathbf{\Psi}\mathbf{\Sigma}\mathbf{\Psi}^*$, where $\mathbf{\Psi}^*$ and $\mathbf{\Psi}$ are wavelet (packet) transforms and $\mathbf{\Sigma}$ is a diagonal matrix. The wavelet transforms might be redundant in order to ensure translation invariance. This diagonal approximation mimics the fact that shift-invariant operators are diagonal in the Fourier domain. These approaches lead to fast $\mathcal{O}(N^d)$ algorithms to compute matrix vector products. In [17], 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

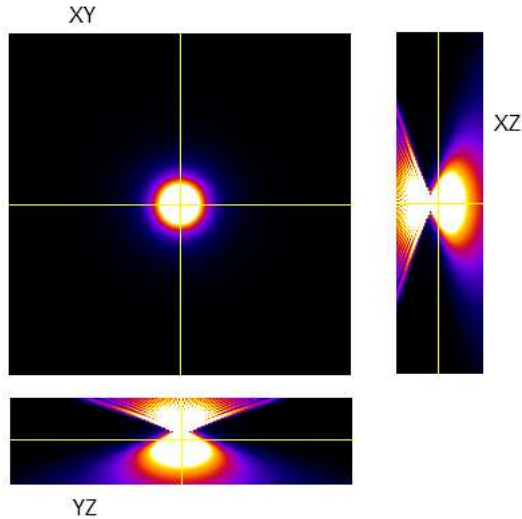


Figure 4: An orthogonal view of a Variable Refractive Index Gibson and Lanni PSF obtained with the PSF Generator [27]

or in high noise levels. It is however too coarse for applications with low noise levels. Let us note that similar approaches based on Gabor multipliers were shown to provide very good results in ODFM systems (slowly varying smoothing operators) [18, 24].

4.5 Piecewise convolutions

Probably the most commonly used approach is the piecewise convolution approximation of the kernel [32, 33, 21, 23, 15]. We advise the reading of [15] for an up-to-date description of this approach and its numerous refinements. Due to its wide use, we propose a detailed analysis of the simplest version of the method in the next section.

5 An analysis of piecewise convolution algorithms

The main idea of this approach 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., [32, 23, 15]).

In this section, we analyze this approach from a theoretical point of view: we derive upper and lower complexity bounds on the approximation error with respect to the subregions

sizes.

5.1 Description of the method

We consider the simple case where the region Ω is partitioned into non-overlapping subregions. We decompose the domain $\Omega = [0, 1]^d$ in m^d subregions of edge length $1/m$. We denote the subregions $\omega_k = \left[\frac{k_1-1}{m}, \frac{k_1}{m}\right] \times \dots \times \left[\frac{k_d-1}{m}, \frac{k_d}{m}\right]$ for $k \in \{1, \dots, m\}^d$ (see Figure 5 for an illustration in 1D). The operator H is approximated by \tilde{H}_m defined by

$$\tilde{H}_m = \sum_{k_1, \dots, k_d=1}^m \mathbb{1}_{\omega_k} \tilde{H}^{(k)} \quad (9)$$

where $\tilde{H}^{(k)}$ is a convolution operator representing the spatially invariant blur in the subregion indexed by k and

$$\mathbb{1}_{\omega_k}(x) = \begin{cases} 1 & \text{if } x \in \omega_k \\ 0 & \text{otherwise.} \end{cases}$$

The convolution kernel associated to $\tilde{H}^{(k)}$ is set to $\tilde{h}_k(y) = K(c_k, c_k - y)$ where c_k denotes the center of the set ω_k . This choice ensures that $\tilde{H}_m u(c_k) = H u(c_k)$.

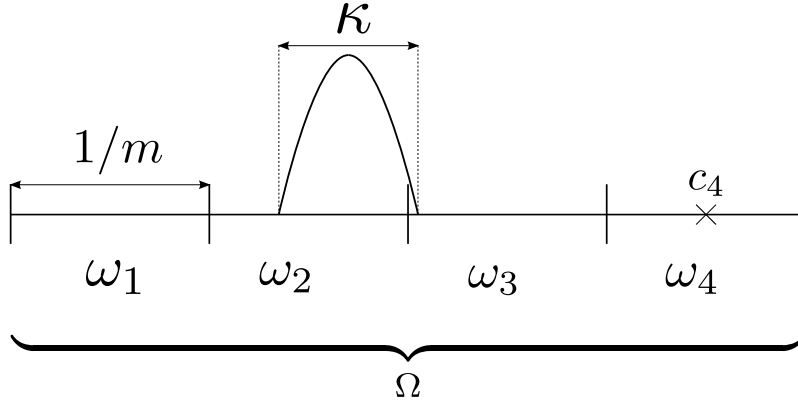


Figure 5: Illustration of the notations in 1D. The domain Ω is decomposed in $m^d = 4$ subregions of length $1/m$. The PSF support has a maximal width κ . The center of each subregion ω_k is denoted c_k .

5.2 Theoretical analysis

We first study this approach in the continuous setting. We assume that the kernel K is L -Lipschitz:

$$\forall (x_1, y_1), (x_2, y_2) \in \Omega, \quad |K(x_1, y_1) - K(x_2, y_2)| \leq L \|(x_1, y_1) - (x_2, y_2)\|_\infty. \quad (10)$$

Under this assumption, we obtain the following result.

Lemma 1 (Upper-Bounds). *For all kernels K satisfying (10), the approximation error satisfies:*

$$\left\| H - \tilde{H}_m \right\|_{2 \rightarrow 2} \leq \frac{L}{m}. \quad (11)$$

Proof. Proof in Appendix A. \square

Remark 3. The same result holds if instead of the global Lipschitz assumption (10), we assume the less stringent

$$\forall (x, y) \in \Omega \times \Omega \text{ and for all } h, \quad |K(x + h, y + h) - K(x, y)| \leq L \|h\|_\infty.$$

This assumption basically indicates that the kernel K can be considered as a convolution locally, since convolution kernels satisfy, $K(x + h, y + h) = K(x, y)$ for all x, y, h .

Lemma 2 indicates that the bound in Lemma 1 is tight for all \mathcal{C}^1 kernels that are not convolutions.

Lemma 2 (Lower-Bounds). *Let K be a $\mathcal{C}^1(\Omega \times \Omega)$ kernel of a space varying integral operator. Then there exists a constant $c > 0$ such that,*

$$\left\| H - \tilde{H}_m \right\|_{2 \rightarrow 2} \geq \frac{c}{m}. \quad (12)$$

Proof. Proof in Appendix A. \square

5.3 Numerical complexity

In this paragraph, we assume that the image consists of $N^d \in \mathbb{N}$ pixels. The PSF volume in the discrete setting is thus approximately equal to $(\kappa N)^d$. We let $\tilde{\mathbf{H}}_m$ denote the discrete piecewise convolution approximation of \mathbf{H} . It is defined similarly to (9). The complexity of the piecewise convolution approach in the discrete setting is summarized in the following theorem.

Theorem 3. *Let K denote a Lipschitz kernel that is not a convolution. Let \mathbf{H} denote the discretized operator defined in equation (6). Let $\tilde{\mathbf{H}}_m$ denote the discrete piecewise convolution approximation of \mathbf{H} . The following results hold:*

- i) A product $\tilde{\mathbf{H}}_m \mathbf{u}$ implemented with FFTs with $\mathbf{u} \in \mathbb{R}^{N^d}$ is performed in a number of operations proportional to*

$$d(N + \kappa N m)^d \log \left(\frac{N}{m} + \kappa N \right). \quad (13)$$

ii) For sufficiently large m and N with $m < N$, there exists constants $0 < c_1 \leq c_2$ such that

$$\frac{c_1}{m} \leq \left\| \mathbf{H} - \tilde{\mathbf{H}}_m \right\|_{2 \rightarrow 2} \leq \frac{c_2}{m}. \quad (14)$$

iii) For sufficiently large $N \in \mathbb{N}$ and sufficiently small $\epsilon > 0$, the number of operations required to obtain $\left\| \mathbf{H} - \tilde{\mathbf{H}}_m \right\|_{2 \rightarrow 2} \leq \epsilon$ is proportional to

$$\frac{d(L\kappa N)^d \log(\kappa N)}{\epsilon^d}. \quad (15)$$

Proof. Proof in Appendix A. □

One cannot hope to improve this complexity result due to Lemma 2. In particular, the method efficiency is insensitive to higher degrees of regularity of the kernel.

Remark 4. Many authors proposed to decompose the domain into overlapping sub-regions and use linear interpolations of PSF on the overlapping domains. We do not analyze this technique in this paper. It is not clear yet whether it would improve the complexity result (15).

6 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:

$$\theta_{j,m,k,n}^{e,e'} = \left\langle H \psi_{j,m}^e, \psi_{k,n}^{e'} \right\rangle, \quad \forall j, k \in \mathbb{N}, \forall (m, n) \in \mathcal{T}_j \times \mathcal{T}_k, \forall e, e' \in \{0, 1\}^d. \quad (16)$$

In their seminal papers [31, 12, 4], 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. Let us acknowledge that a similar algorithm has been developed independently in [42]. This paper contains however no theoretical analysis of the method's complexity.

This section is organized as follows: first, we discuss the interest of approximating H in a wavelet basis rather than using the standard discretization described in paragraph 4.1. Second, we provide various theoretical results concerning the number of coefficients necessary to obtain an ϵ -approximation of H .

6.1 Discretization of the operator by projection

The proposed method relies on a discretization of H different from that of paragraph 4.1. The main idea is to use a projection on a finite dimensional linear subspace $V_N = \text{Span}(\varphi_1, \dots, \varphi_N)$ 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_N by $H_N u = P_{V_N} H P_{V_N} u$, where P_{V_N} is the projector on V_N . We can associate an $N \times N$ matrix Θ to this operator defined by $\Theta = (\langle H \varphi_i, \varphi_j \rangle)_{1 \leq i, j \leq N}$.

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 [36]. This hypothesis implies that

$$\|u - P_{V_N} u\|_2 = \mathcal{O}(N^{-\alpha}) \quad (17)$$

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 [29, 34].

Moreover, if we assume that H is a regularizing operator, meaning that $\|Hu - P_{V_N} Hu\|_2 = \mathcal{O}(N^{-\beta})$ with $\beta \geq \alpha$ for all u satisfying (17), then we have:

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

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., [41, 2] where a value $\alpha = 1/6$ is obtained in 2D for BV functions, while the simple analysis above leads to $\alpha = 1/2$).

6.2 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^d , where $N^d = 1 + \sum_{j=0}^{J-1} (2^d - 1)2^{dj}$ denotes the numbers of wavelets kept to represent images.

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

$$\mathbf{H} = \Psi \Theta \Psi^* \quad (18)$$

where $\Psi : \mathbb{R}^{N^d} \rightarrow \mathbb{R}^{N^d}$ is the discrete separable wavelet transform. Matrix Θ is an $N^d \times N^d$ matrix which corresponds to a truncated version (also called finite section) of the matrix Θ defined in (16).

6.3 Theoretical guarantees with sparse approximations

Sparse approximations of integral operators have been studied theoretically in [4, 31]. They then have been successfully used in the numerical analysis of PDEs [13, 10, 9]. Surprisingly, they have been scarcely applied to image processing. The two exceptions we are aware of are the paper [8], where the authors show that wavelet multipliers can be useful to approximate foveation operators. More recently, [42] proposed an approach that is very much related to that of our paper. It contains however no analysis of complexity with respect to the precision ϵ .

Let us recall a typical result that motivates the proposed approach. We stick to the one-dimensional case for the ease of exposition.

Lemma 4 (Decay of $\theta_{j,m,k,n}^{e,e'}$). *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:

$$\left| \theta_{j,m,k,n}^{e,e'} \right| \leq C_M 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} f_{j,m,k,n} \quad (19)$$

where $f_{j,m,k,n} = f(\text{dist}(I_{j,m}, I_{k,n}))$, C_M is a constant that does not depend on j, k, m, n, e and e' and

$$\begin{aligned} \text{dist}(I_{j,m}, I_{k,n}) &= \inf_{x \in I_{j,m}, y \in I_{k,n}} \|x - y\|_\infty \\ &= \min \left(0, \left\| 2^{-j}m - 2^{-k}n \right\|_\infty - (2^{-j} + 2^{-k}) \frac{c(M)}{2} \right). \end{aligned} \quad (20)$$

Proof. See Appendix C. □

Lemma 4 is the key to obtain all subsequent complexity estimates.

Theorem 5. *Let Θ_η be the matrix obtained by zeroing all coefficients in Θ such that*

$$2^{-\min(j,k)(M+d)} f_{j,m,k,n} \leq \eta.$$

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*

$$C'_M (N\kappa)^d \eta^{-\frac{d}{M+d}} \quad (21)$$

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 $\|\mathbf{H} - \tilde{\mathbf{H}}_\eta\|_{2 \rightarrow 2} \leq \epsilon$ is bounded above by

$$C_M'' (N\kappa)^d \epsilon^{-\frac{d}{M}} \quad (22)$$

where $C_M'' > 0$ is independent of N .

Proof. See Appendix D. □

Remark 5. There are a few differences making the wavelet approach more attractive than piecewise convolutions from a theoretical point of view:

- A discretization in the wavelet domain provides better theoretical guarantees than the standard quadrature rules (see Section 6.1).
- A comparison between the upper-bound (22) and the bound (15) is instructive. In the piecewise convolution approach, there is no hope to obtain a better approximation rate than $\mathcal{O}(\frac{1}{m})$ (see Proposition 12). For the wavelet approach, the situation is different: 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 [31, 12, 4].

7 How to define 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 (19) is available.

7.1 Problem formalization

Let \mathbf{H} be the $N^d \times N^d$ matrix defined in equation (18). 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 $N^d \times N^d$ matrices with at most K non zeros coefficients. The problem we address in this paragraph reads

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

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 $\{\mathbf{u} \in \mathbb{R}^N, \|\mathbf{u}\|_X \leq 1\}$.

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. 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. Functions in $BV(\Omega)$ can be characterized by their wavelet coefficients [34]. For instance, if $u \in BV([0, 1])$, then

$$|\langle u, \phi_0 \rangle| + \sum_{j=0}^{+\infty} \sum_{m=0}^{2^j-1} 2^j |\langle u, \psi_{j,m} \rangle| < +\infty \quad (23)$$

for all wavelet bases. 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 images in 1D, one could set $\sigma_i = 2^{j(i)}$ where $j(i)$ is the scale of the i -th wavelet, owing to (23).

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 transforms:

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

By letting $\Delta^{(i)}$ denote the i -th column of Δ , we finally get the following simple expression for the operator norm:

$$\left\| \mathbf{H} - \tilde{\mathbf{H}} \right\|_{X \rightarrow 2} = \max_{1 \leq i \leq N^d} \frac{1}{\sigma_i} \left\| \Delta^{(i)} \right\|_2. \quad (24)$$

Our goal is thus to find the solution of:

$$\min_{\mathbf{S}_K \in \mathbb{S}_K} \max_{1 \leq i \leq N^d} \frac{1}{\sigma_i} \left\| \mathbf{\Delta}^{(i)} \right\|_2. \quad (25)$$

7.2 An algorithm when $\mathbf{\Theta}$ is known

Finding the minimizer of problem (25) can be achieved using a simple greedy algorithm: the matrix \mathbf{S}_{k+1} is obtained by adding the largest coefficient of the column $\mathbf{\Delta}_i$ 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 $\mathcal{O}(N^{2d} \log(N))$. 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.

Algorithm 1: An algorithm to find the minimizer of (25).

Input:

$\mathbf{\Theta}$: $N^d \times N^d$ matrix;

$\mathbf{\Sigma}$: Diagonal matrix;

K : the number of elements in the thresholded matrix;

Output:

\mathbf{S}_K : Matrix minimizing (25)

Initialization:

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

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

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

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

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

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

begin

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

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

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

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

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

end

end

7.3 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.

When Θ is unknown, we may take advantage of equation (19) to define sparsity patterns. A naive approach would consist in applying Algorithm (1) directly on the upper-bound (19). 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 (19) indicates that the coefficients $\theta_{j,m,k,n}$ can be discarded for sufficiently large $|j - k|$ and sufficiently large distance between the wavelet supports. Equation (19) thus means that for a given wavelet $\psi_{j,m}$, only its spatial neighbours in neighbouring scales have significant correlation coefficients $\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle$. We may thus construct sparsity patterns using the notion of multiscale neighbourhoods defined below.

Definition 2 (Multiscale shift). The multiscale shift $s \in \mathbb{Z}^d$ between two wavelets $\psi_{j,m}^e$ and $\psi_{k,n}^{e'}$ is defined by

$$s = \left\lfloor \frac{n}{2^{\max(k-j,0)}} \right\rfloor - \left\lfloor \frac{m}{2^{\max(j-k,0)}} \right\rfloor. \quad (26)$$

Note that for $k = j$, the multi-scale shift is just $s = n - m$ 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 6.

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 $\psi_{j,m}^e$ and $\psi_{k,n}^{e'}$ will be said to be \mathcal{N} -neighbors if $(j, (k, s)) \in \mathcal{N}$ where s is defined in equation (26).

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

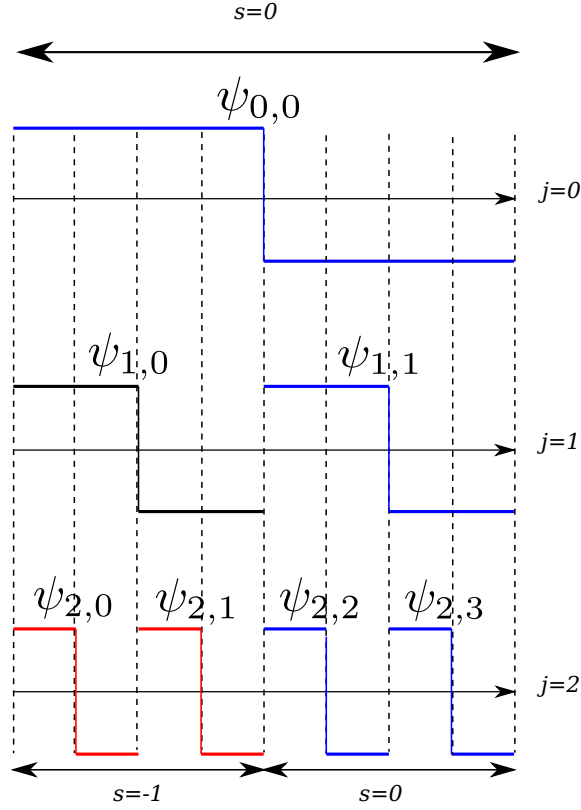


Figure 6: Illustration of a multiscale shift on a 1D signal of size 8 with the Haar basis. The shifts are computed with respect to wavelet $\psi_{1,1}$. Wavelets $\psi_{0,0}$, $\psi_{2,2}$ and $\psi_{2,3}$ have a multiscale shift $s = 0$ with $\psi_{1,1}$ since their support intersects that of $\psi_{1,1}$. 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 $\psi_{1,0}$.

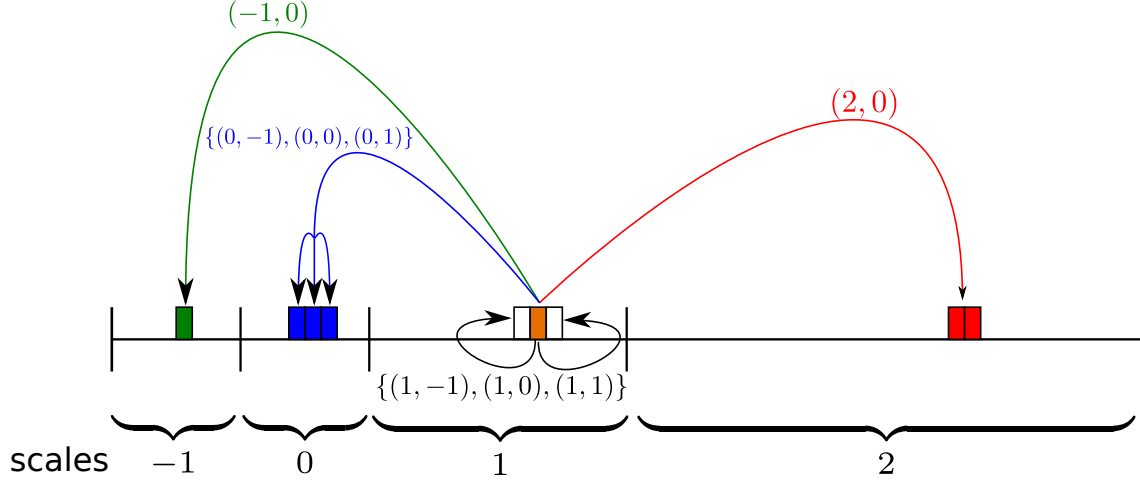


Figure 7: Illustration of a multiscale neighborhood on a 1D signal. In this example, the neighborhood at scale 1 is $\mathcal{N}(1) = \{(-1, 0), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1), (2, 0)\}$. Notice that the two red wavelets at scale 2 are neighbors of the orange wavelet at scale 1 and that this relationship is described through only one shift.

all possible neighborhood relationships at scale j . This is illustrated in Figure 7. Let $\mathcal{N} \in \mathcal{P}(\mathcal{N})$ denote a multiscale neighborhood. We define the matrix $\mathbf{S}_{\mathcal{N}}$ as follows:

$$\mathbf{S}_{\mathcal{N}}((j, m, e), (k, n, e')) = \begin{cases} \theta_{j,m,k,n}^{e,e'} & \text{if } \psi_{j,m}^e \text{ is an } \mathcal{N}\text{-neighbor of } \psi_{k,n}^{e'} \\ 0 & \text{otherwise.} \end{cases}$$

Equation (19) indicates that

$$|\theta_{j,m,k,n}^{e,e'}| \leq u(j, k, s)$$

with

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

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

$$\min_{\substack{\mathcal{N} \in \mathcal{P}(\mathcal{N}) \\ |\mathcal{N}|=K}} \max_{1 \leq i \leq N} \frac{1}{\sigma_i} \left\| (\mathbf{U} - \mathbf{S}_{\mathcal{N}})^{(i)} \right\|_2 \quad (27)$$

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.

Algorithm 2: An algorithm to find the minimizer of (27).

Input:

u : Upper-bound ;

Σ : Diagonal matrix;

K : the number of elements of the neighborhood;

Output:

\mathcal{N} : multiscale neighborhood minimizing (27)

Initialization:

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

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

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

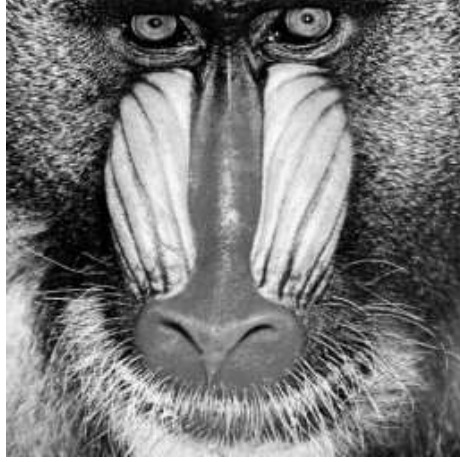
Note that the norms γ_k only depend on the scale $j(k)$, so that the initialisation step only requires $\mathcal{O}(N^d \log_2(N))$ operations. Similarly to Algorithm 1, this algorithm can be accelerated by first sorting the elements of $u(j, k, s)$ in decreasing order. The overall complexity for this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

8 Numerical experiments

In this section we perform various numerical experiments in order to evaluate the practical efficiency of wavelet based methods. We also perform comparisons between the piecewise convolution approach and the wavelet based approach. We first evaluate the method's efficiency on the direct problem:

- we analyze $\|\mathbf{H} - \tilde{\mathbf{H}}\|$ for various operator norms.
- we compare $\mathbf{H}\mathbf{u}$ and $\tilde{\mathbf{H}}\mathbf{u}$ for real images \mathbf{u} .

We then study the performance of the different methods for deblurring problems. Two different blur kernels and two images will be considered, see Figures 8 and 9. The images are rescaled in $[0, 1]$ to ease the visualization of residuals. Due to memory limitations, we only consider images of size $N = 256 \times 256$. Note that a full matrix of size N^2 stored in double precision weighs around 32 gigabytes.



(a) Mandrill



(b) Letters

Figure 8: The two images of size 256×256 used in these numerical experiments

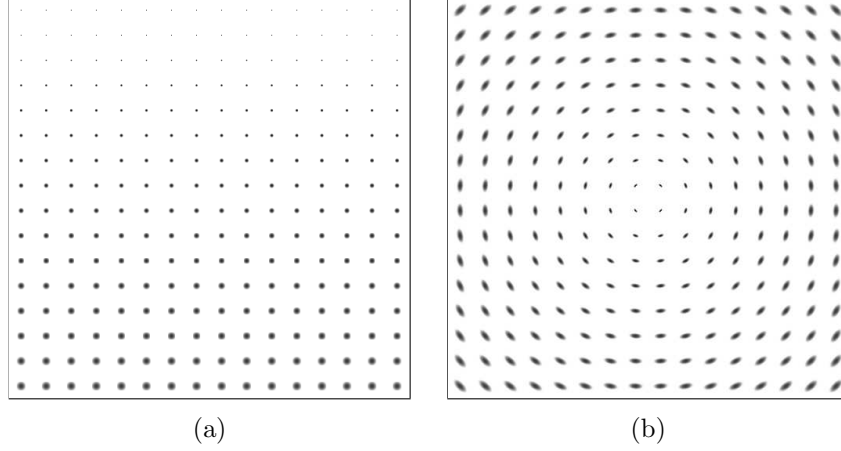


Figure 9: The PSFs associated to the blur kernels. The PSFs in Figure (9a) are Gaussians with equal variances increasing in the vertical direction. The PSFs in Figure (9b) are anisotropic Gaussians with covariance matrices that depend on the polar coordinates.

8.1 Computation of the full Θ matrix

Before applying our approximation methods, matrix Θ needs to be computed explicitly. The coefficients $\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle$ are approximated by their discrete counterparts. If $\psi_{j,m}^e$ and $\psi_{k,n}^{e'}$ denote discrete wavelets, we simply compute the wavelet transform of $\mathbf{H}\psi_{j,m}^e$ and store it into the (j, e, m) -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:

$$\int_{\Omega} \int_{\Omega} K(x, y) \psi_{j,m}^e(y) \psi_{k,n}^{e'}(x) dy dx \simeq \frac{1}{N^{2d}} \sum_{x \in X} \sum_{y \in X} K(x, y) \psi_{j,m}^e(y) \psi_{k,n}^{e'}(x). \quad (28)$$

In all our numerical experiments, we used Daubechies wavelets with 10 vanishing moments decomposed at the fourth level. We made several tests not reported here to find the best combination. 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.

8.2 Evaluation of $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2}$ with a simple thresholding strategy

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

Algorithm 3: An algorithm to compute Θ

Output:

Θ : the full matrix of \mathbf{H}

begin

forall the (j, e, m) **do**

 Compute the wavelet $\psi_{j,m}^e$ using an inverse wavelet transform

 Compute the blurred wavelet $\mathbf{H}\psi_{j,m}^e$

 Compute $\left(\left\langle \mathbf{H}\psi_{j,m}^e, \psi_{k,n}^{e'} \right\rangle\right)_{k,e',n}$ using one forward wavelet transform

 Set $\left(\left\langle \mathbf{H}\psi_{j,m}^e, \psi_{k,n}^{e'} \right\rangle\right)_{k,e',n}$ in the (j, e, m) -th column of Θ .

end

end

The sparse approximation of the operator is constructed by thresholding the matrix Θ in order to keep the K largest coefficients. The values $K = 2^l \times N^2$ with $l \in \{0 \dots 2 \log_2 N\}$ have been chosen. This way K is a multiple of the number of pixels in the image. The piecewise convolution approximation is constructed by partitioning the image into $2^l \times 2^l$ sub-images where $l \in \{0 \dots \log_2 N\}$. We also studied the case where sub-images overlap and linearly interpolated the blur between sub-images as proposed in [32, 23]. 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 [19]. 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 piecewise convolutions, 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 piecewise convolutions. The total number of operations is reduced by a factor roughly equal to 100 for a precision of 0.1.

8.3 Numerical comparison of different sparsity patterns.

In this numerical experiment, we obtain a K -sparse matrix Θ_K using either a simple thresholding strategy or Algorithm 1. We evaluate the error $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{X \rightarrow 2}$ defined in (24)

*We also carried out numerical computations of the operator norm error when images are supposed to be in Sobolev spaces $H^s(\Omega)$ for different s . However, since the results do not differ too much, we do not present them.

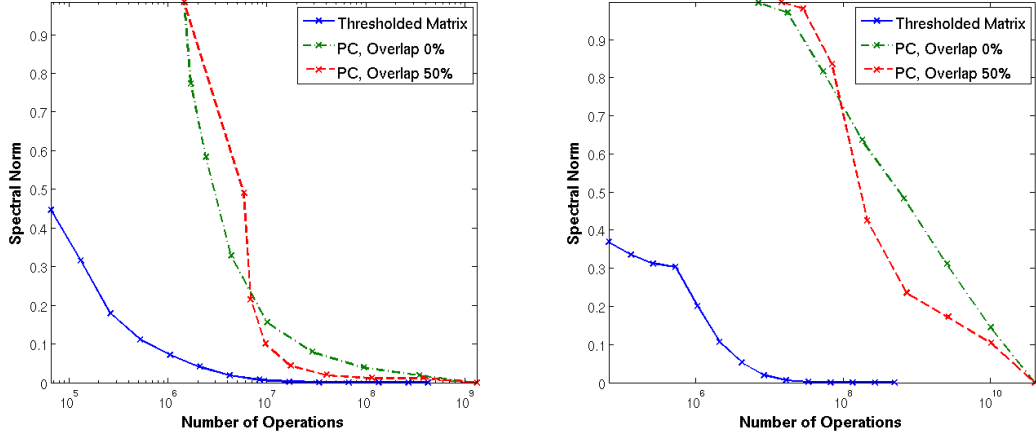


Figure 10: The operator norms $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{2 \rightarrow 2}$ are displayed for the three proposed kernels. (From left to right, kernels corresponding to Figures 9a and 9b) and with respect to the number of operations needed to compute $\tilde{\mathbf{H}}\mathbf{u}$. Plots are displayed in a semilogx scale.

for both methods. In this experiment, we set $\sigma_i = 2^{j(i)}$. It is readily seen from Figure 11 that Algorithm provides a much better error decay for both operators.

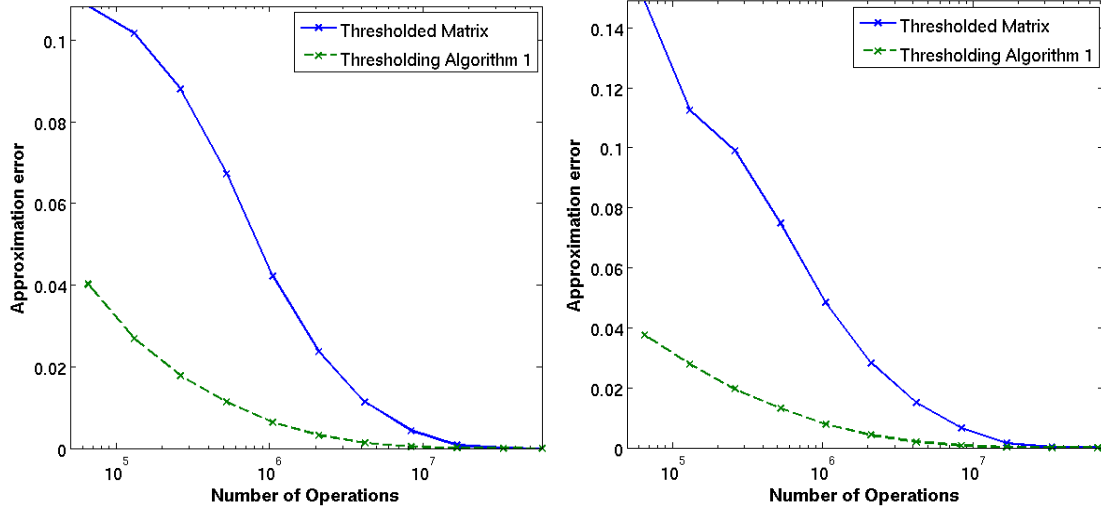


Figure 11: The operator norms $\|\mathbf{H} - \tilde{\mathbf{H}}\|_{X \rightarrow 2}$ are displayed for kernels Figure 9a (left) and Figure 9b (right); and with respect to the number of operations needed to compute $\tilde{\mathbf{H}}\mathbf{u}$. Plots are displayed in a semilogx scale. Daubechies wavelets with 10 vanishing moments have been used.

8.4 Quality of matrix vector products for real images

In this experiment, we compare $\tilde{\mathbf{H}}\mathbf{u}$ to $\mathbf{H}\mathbf{u}$, where \mathbf{u} is the image in Figure 8b and where $\tilde{\mathbf{H}}$ is obtained either by piecewise convolutions 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 $\tilde{\mathbf{H}}\mathbf{u}$. The two methods are tested with the following choices:

- We only test the piecewise-convolution with overlap since it produces better pSNR.
- Different sparsity patterns are tested. The first one is obtained by thresholding of Θ . The second one is obtained using Algorithm 1. We test three different diagonal matrices $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_{N^2})$:
 1. Σ contains only ones on its diagonal: $\sigma_i = 1, \forall i$.
 2. σ is piecewise constant with a value increasing with respect to the scale of the wavelet coefficients: $\sigma_i = 2^{j(i)}$.
 3. $\sigma_i = 2^{j(i)}$, and $\sigma_i = \frac{1}{2}$ for indices i corresponding to the low frequency sub-band.

The third one is obtained using Algorithm 2. The algorithm finds multi-scale neighbourhoods until $K = l \times N^2$ coefficients populate the matrix, with $l \in \{1, \dots, 200\}$.

We also performed experiments with $\sigma_i = 4^{j(i)}$, which corresponds to the assumption that images belong to $BV(\Omega)$. We do not report the results since they were not good.

The results of this experiment are displayed in Figure 12 for the two kernels from Figures 9b and 9a. Approximations built with Algorithm 1 and with $\sigma_i = 2^{j(i)}$ increase the pSNR of the blurred image $\tilde{\mathbf{H}}\mathbf{u}$ by almost 10dBs compared to the naive thresholding approaches. This experiment highlights the relevance of Algorithm 1, used with the second and third Σ matrices. It allows to construct accurate sparse approximations of operators when applied to images in $BV(\Omega)$.

The piecewise convolution approach performs slightly better than wavelet based methods for the simple kernel 9a on a small interval. It corresponds to the partitioning of Ω in 16×16 and 32×32 sub-windows (i.e. sub-windows of size 16×16 or 8×8 pixels). However, for more complex kernels, wavelet methods perform better.

This experiment also shows the qualities and limits of the “blind” Algorithm 2. In this algorithm, the structure of the approximating matrix is deduced from the upper-bound (19). Matrices constructed using Algorithm 2 perform similarly to Algorithm 1 (that has a full knowledge of Θ) up to approximately $K = 30 \times N^2$ coefficients. Above this number, the approximation quality increases very slowly. This is probably due to the fact that the upper-bound (19) is too rough: the operator might be much sparser than what is predicted by the theory. We will see that in deblurring applications, an approximation made of $K = 30 \times N^2$ non zero coefficients is more than enough.

Figure 16 shows the sparsity patterns of matrices obtained with Algorithms 1 and 2 for $K = 30N^2$ and $K = 128N^2$ coefficients. It is readily seen that the sparsity patterns look very similar and tend to confirm the soundness of Algorithm 2.

Finally, for kernel Figure 9b, we show blurred images $\tilde{\mathbf{H}}\mathbf{u}$ in Figures 13 and 14 for the different sparsity patterns. Figure 13 shows the blurred images $\tilde{\mathbf{H}}\mathbf{u}$ obtained with Algorithm 1 and with the three different Σ matrices. Figure 14 displays the blurred images $\mathbf{H}\mathbf{u}$ obtained with the simple thresholding scheme and Algorithm 2. Finally, Figure 15 provides a comparison of the piecewise convolution approach 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 piecewise convolution and wavelet based approach are different. They are localized at the interfaces between sub-images for the piecewise convolution 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 substantially the reconstruction.
- The approximation using the third Σ matrix produces the best results and should be preferred over more simple approaches.
- The sparsity pattern obtained using Algorithm 2 suffers from more artifacts than the other approaches. The quality is however acceptable from a visual point of view.

- In our implementation, the piecewise convolution 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. Note that no method is multi-threaded, so that the comparison seems fair.
- The computing time of 1.21 seconds for the piecewise convolution 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 piecewise convolution algorithms may increase in a chaotic manner with respect to m . In our numerical experiments we implemented two versions of the piecewise convolution approach: one based on the FFTW <http://www.fftw.org/> and the other based on Kiss FFT <http://sourceforge.net/projects/kissfft/>. The latter revealed to provide lower computing times (probably to the fact that no plan is pre-computed) and we thus reported computing times using this method.

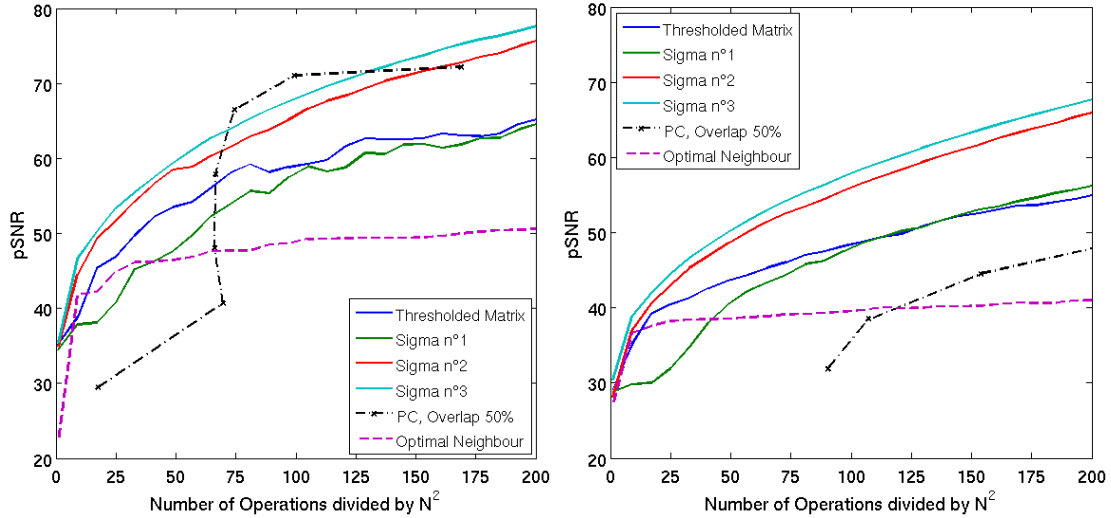


Figure 12: pSNR of the blurred image using the approximated operators $\tilde{\mathbf{H}}\mathbf{u}$ with respect to the blurred image using the exact operator $\mathbf{H}\mathbf{u}$. The results have been obtained using the letters image Figure 8b.

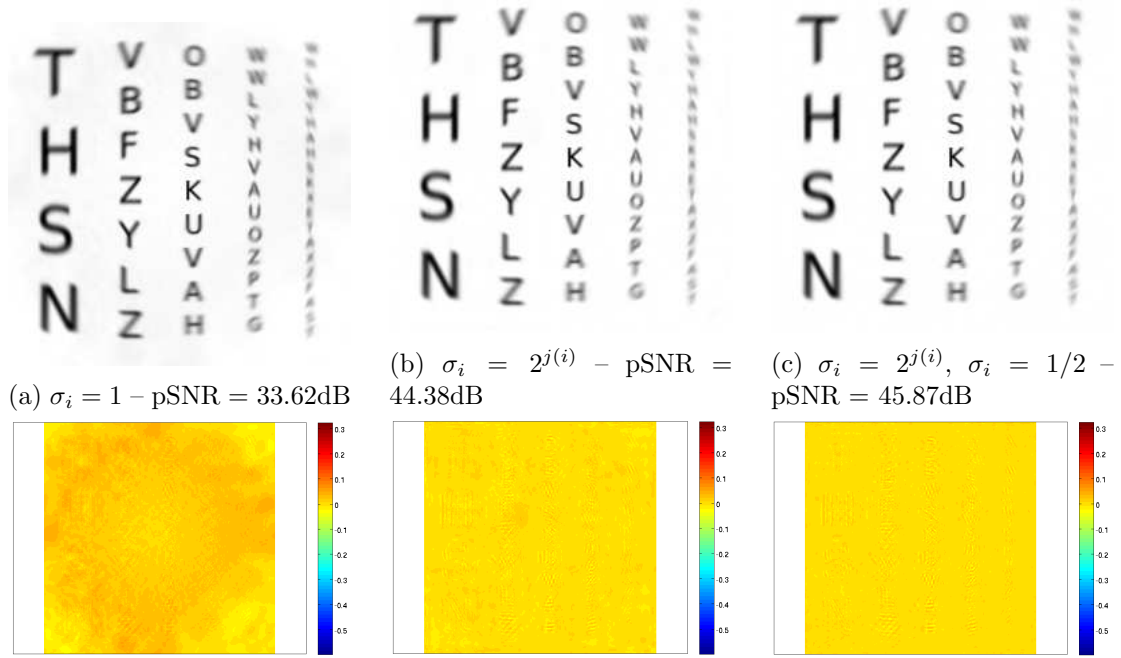


Figure 13: Blurred images using matrices formed with Algorithm 1 and for the kernel Figure 9b. $K = 30N^2$ coefficients are kept in the matrices. Algorithm 1 has been applied with the three different Σ matrices. The blurred images $\tilde{\mathbf{H}}\mathbf{u}$ are shown on top. The differences $\mathbf{H}\mathbf{u} - \tilde{\mathbf{H}}\mathbf{u}$ are also displayed on the bottom. They all have the same color range.

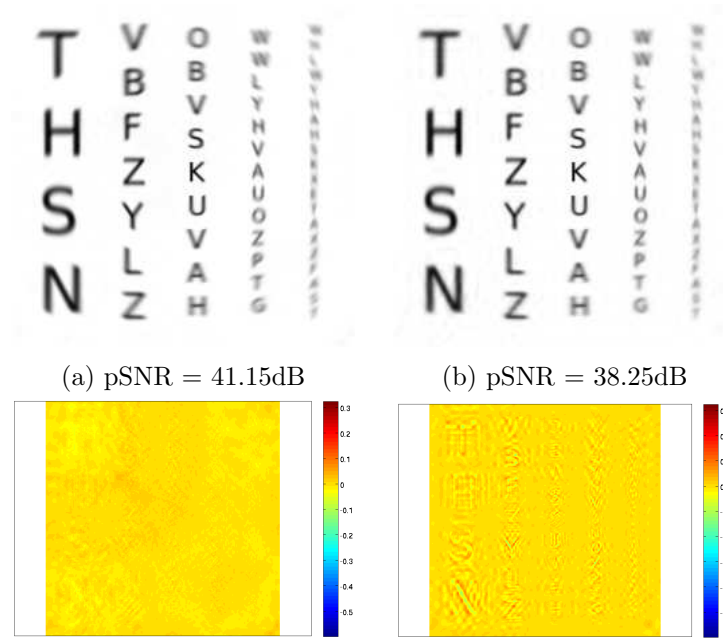


Figure 14: Blurred images using the wavelet based method with two different sparsity patterns and the kernel in Figure 9b. $K = 30N^2$ coefficients are kept in the matrices. Matrices are obtained with the simple thresholding (left) and Algorithm 2 (right). The differences $\mathbf{Hu} - \tilde{\mathbf{H}}\mathbf{u}$ are displayed within the same color range.









	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 15: Blurred images and the differences $\mathbf{Hu} - \tilde{\mathbf{H}}\mathbf{u}$ for the kernel Figure 9b. Results on the left are obtained using piecewise convolution 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 third Σ matrix keeping $K = lN^2$ coefficients. The pSNR and the time needed for the computation for the matrix-vector product are shown. The differences are displayed within the same color range as Figures 13 and 14

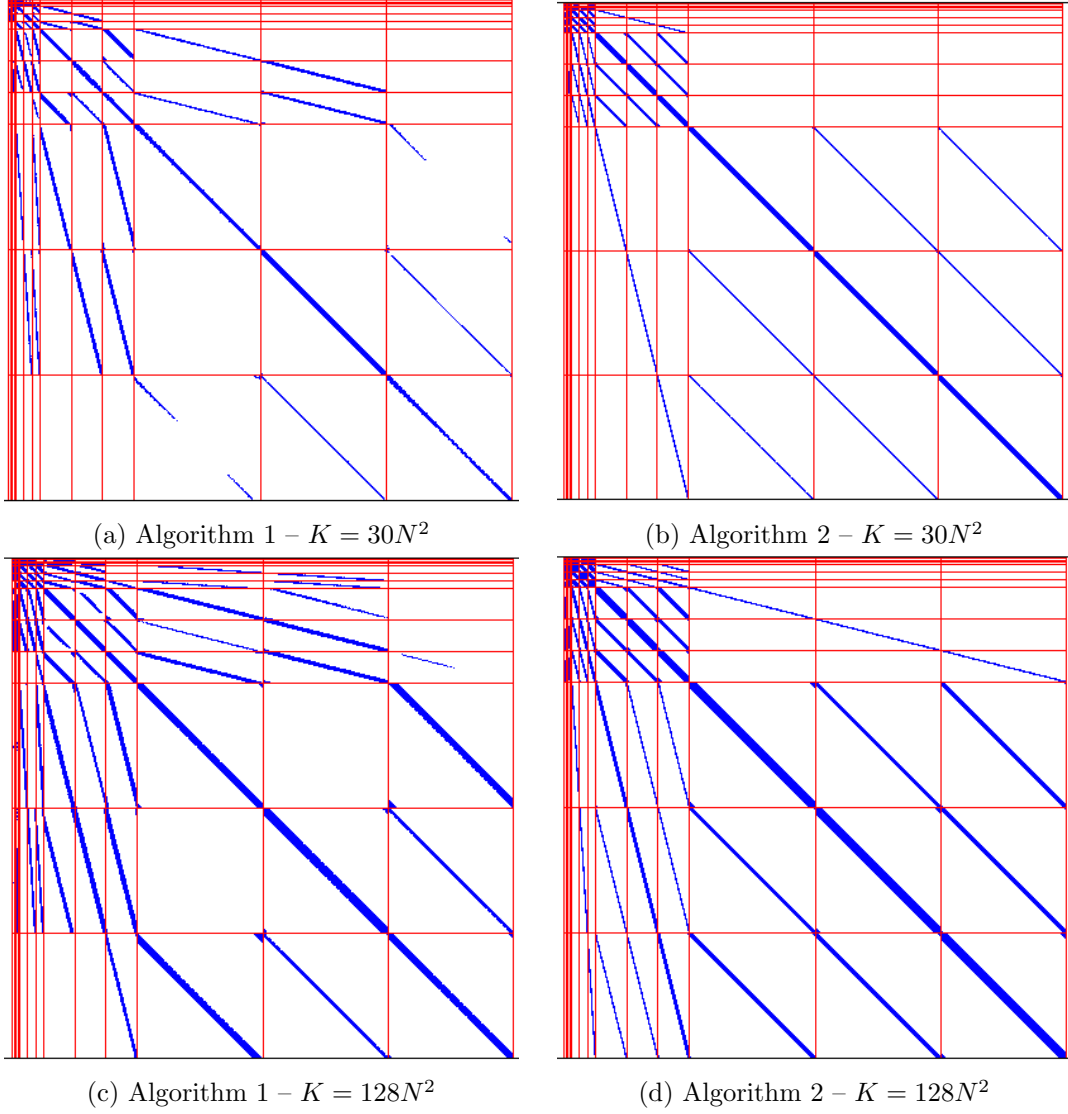


Figure 16: The structure of the wavelet matrices of Θ_K are displayed for Algorithms 1 and 2 and for $K = 30N^2$ and $K = 128N^2$ coefficients. Algorithm 1 has been applied using the third Σ matrix.

8.5 Deblurring problem

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

$$\mathbf{v} = \mathbf{H}\mathbf{u} + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2 \text{Id}), \quad (29)$$

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} :

$$\text{Find } \mathbf{u}^* \in \arg \min_{\mathbf{u} \in \mathbb{R}^{N^d}, \|\tilde{\mathbf{H}}\mathbf{u} - \mathbf{v}\|_2^2 \leq \alpha} TV(\mathbf{u}), \quad (30)$$

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 [7]. 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 used 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 17 to 20, we present some deblurring results with or without noise for image Figure 8a with kernel Figure 9a and for image Figure 8b with kernel 9b. Figure 17 shows that without noise and the simple kernel 9a, 4×4 piecewise convolutions perform better than wavelet approaches with $30N^2$. Piecewise convolution achieve better pSNR but are 4 times slower. For equivalent computation times, wavelet approaches should be constructed with $100N^2$ coefficients.

In Figures 19 and 20 we can notice that wavelet methods perform better than piecewise convolution. Also notice that Algorithm 1 is strongly preferable to a simple thresholding since it reduces deblurring artefacts.

With noise, all methods perform nearly the same as the exact 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 21. 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 $20N^2$ coefficients leads to deblurred images close to the results obtained with the exact operator. In this experiment, a total of $K = 5N^2$ coefficients in Θ_K is enough to retrieve satisfactory results. This is a very encouraging result for blind deblurring problems.



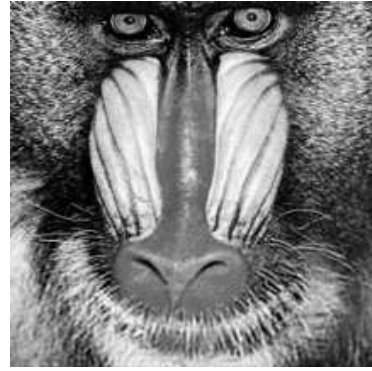
(a) Degraded image
25.04dB



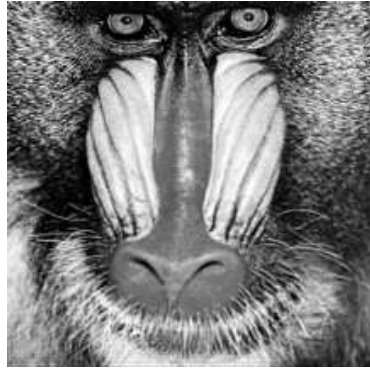
(b) Exact operator
35.38dB – 2 hours



(c) Simple thresholding
31.54dB – 8 sec



(d) Algorithm 1
31.03dB – 8 sec

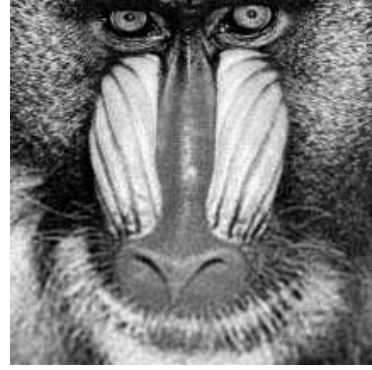


(e) Piecewise convolutions
33.81dB – 35 sec

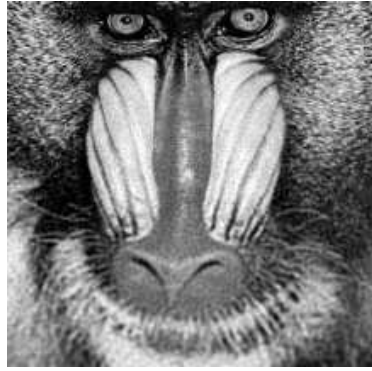
Figure 17: Deblurring results for kernel Figure 9a and without noise. Top-left: degraded image. Top-right: deblurred using the exact operator. Middle-left: deblurred by the wavelet based method and a simple thresholding. Middle-right: deblurred by the wavelet based method and Algorithm 2 with the third Σ matrix. Bottom: deblurred using a 4×4 piecewise convolution algorithm with 50% overlap. For wavelet methods $K = 30N^2$ coefficients are kept in matrices. pSNR are displayed for each restoration.



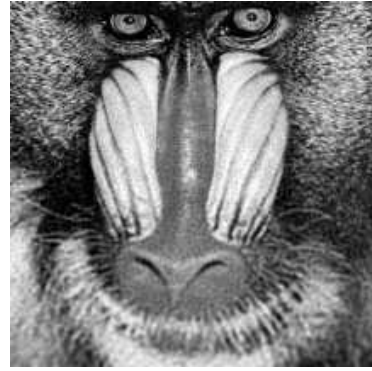
(a) Degraded image
24.52dB



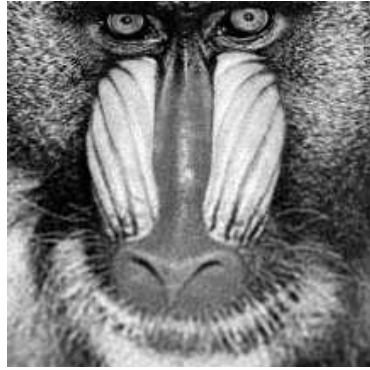
(b) Exact operator
25.85dB – 2 hours



(c) Simple thresholding
25.83dB – 8sec



(d) Algorithm 1
25.83dB – 8 sec



(e) Piecewise convolutions
25.81dB – 35 sec

Figure 18: Deblurring results for kernel Figure 9a and with $\sigma = 0.02$. 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 third Σ matrix. Bottom: deblurred using a 4×4 piecewise convolution algorithm with 50% overlap. For wavelet methods $K = 30N^2$ coefficients are kept in matrices. pSNR are displayed for each restoration.



(a) Degraded image
21.85dB



(b) Exact operator
34.53dB – 2 hours



(c) Simple thresholding
30.96dB – 8 sec

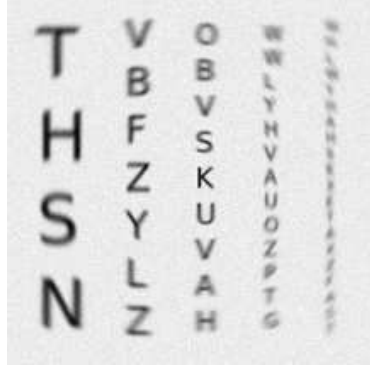


(d) Algorithm 1
30.56dB – 8 sec



(e) Piecewise convolutions
28.37dB – 35 sec

Figure 19: Deblurring results for kernel Figure 9b 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 third Σ matrix. Bottom: deblurred using a 4×4 piecewise convolution algorithm with 50% overlap. For wavelet methods $K = 30N^2$ coefficients are kept in matrices. pSNR are displayed for each restoration.



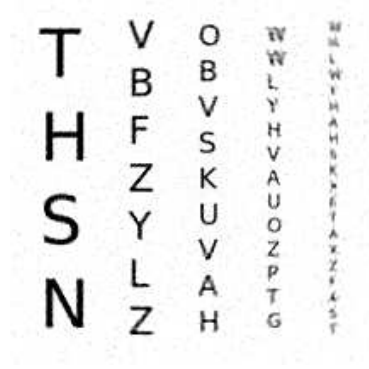
(a) Degraded image
21.62dB



(b) Exact operator
28.97dB – 2 hours



(c) Simple thresholding
27.89dB – 8 sec



(d) Algorithm 1
28.02dB – 8 sec



(e) Piecewise convolutions
27.12dB – 35 sec

Figure 20: Deblurring results for kernel Figure 9b 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 third Σ matrix. Bottom: deblurred using a 4×4 piecewise convolution algorithm with 50% overlap. For wavelet methods $K = 30N^2$ coefficients are kept in matrices. pSNR are displayed for each restoration.

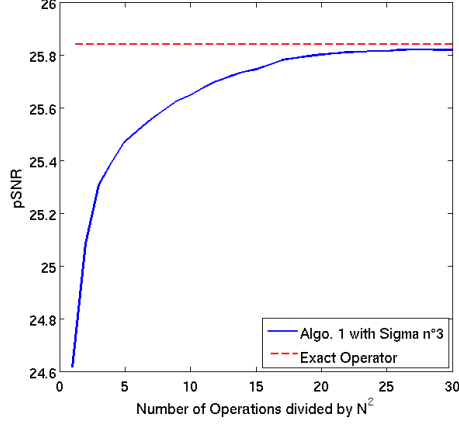


Figure 21: pSNR of the deblurred image with respect to the number of coefficients in the matrix divided by N^2 for the image Figure 8a and the kernel Figure 9a. The matrix is constructed using Algorithm 1 with the third Σ matrix with $K = lN^2$ coefficients for l from 1 to 30. Deblurred images using these matrices are compared with the one obtained with the exact operator.

9 Conclusion

9.1 Brief summary

In this paper, we analyzed the piecewise convolution approach to approximate spatially varying blur. We showed that it has an $\mathcal{O}(N^d \log(N) \epsilon^{-d})$ complexity where ϵ denotes the desired accuracy in l^2 . We then introduced a new method based on sparse wavelet representation of the operator. We showed that this new technique has a greater adaptivity to the smoothness of the operator and exhibit an $\mathcal{O}(N^d \epsilon^{-d/M})$ 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 can be obtained by leveraging the fact that natural signals exhibit some structure in the wavelet domain. 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 the piecewise convolutions approaches with or without overlap. Moreover, they seem to provide satisfactory deblurring results on practical problems with a complexity no greater than $5N^2$ operations, where N^2 denotes the pixels number.

9.2 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 [10]. 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.

An 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 for certain blur families. A top-down approach has already been proposed in [42]. 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. [20, 40, 28]).

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 artifacts 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 [6] that curvelets are near optimal to represent Fourier integral operators.

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 [23]. They thank Loïc Denis and Ferréol Soulez for indicating references [15, 42]. 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] E. Angel and A. K. Jain. Restoration of images degraded by spatially varying pointspread functions by a conjugate gradient method. *Applied Optics*, 17:2186–2190,

1978.

- [2] S. Bartels. Total variation minimization with finite elements: Convergence and iterative solution. *SIAM Journal on Numerical Analysis*, 50:1162–1180, 2012.
- [3] S. Ben Hadj and L. Blanc-Féraud. Restoration method for spatially variant blurred images. Research report, INRIA, June 2011.
- [4] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transform and numerical algorithm. *Commun. Pure and Applied Math.*, 44:141–183, 1991.
- [5] M. Born and E. Wolf. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. CUP Archive, 1999.
- [6] E. Candes and L. Demanet. Curvelets and fourier integral operators. *Comptes Rendus Mathématique*, 336:395–398, 2003.
- [7] 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.
- [8] E.-C. Chang, S. Mallat, and C. Yap. Wavelet foveation. *Applied and Computational Harmonic Analysis*, 9:312–335, 1999.
- [9] A. Cohen. *Numerical analysis of wavelet methods*, volume 32. Elsevier, 2003.
- [10] A. Cohen, W. Dahmen, and R. A. DeVore. Adaptive wavelet methods ii-beyond the elliptic case. *Foundations of Computational Mathematics*, 2:203–245, 2002.
- [11] 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.
- [12] R. Coifman and Y. Meyer. Wavelets, Calderón-Zygmund and multilinear operators. *Cambridge Studies in Advanced Math*, 48, 1997.
- [13] 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.
- [14] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, June 1992.
- [15] L. Denis, E. Thiébaud, F. Soulez, J.-M. Becker, and R. Mourya. Fast approximations of shift-variant blur. 2014.
- [16] 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.

- [17] P. Escande, P. Weiss, and F. Malgouyres. Spatially varying blur recovery. diagonal approximations in the wavelet domain. *Proceedings of ICPRAM*, 2013.
- [18] H. G. Feichtinger and K. Nowak. A first survey of gabor multipliers. In *Advances in Gabor analysis*, pages 99–128. Springer, 2003.
- [19] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.
- [20] 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.
- [21] P. C. Hansen, J. G. Nagy, and D. P. O’leary. *Deblurring images: matrices, spectra, and filtering*. Siam, 2006.
- [22] 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.
- [23] 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.
- [24] 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.
- [25] 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.
- [26] J. Kamm and J. G. Nagy. Kronecker product and svd approximations for separable spatially variant blurs. In *SPIE’s International Symposium on Optical Science, Engineering, and Instrumentation*, pages 358–369. International Society for Optics and Photonics, 1998.
- [27] 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.
- [28] 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.

- [29] S. Mallat. *A Wavelet Tour of Signal Processing – The Sparse Way*. Third Edition. Academic Press, 2008.
- [30] 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.
- [31] Y. Meyer. *Wavelets and operators*, volume 2. Cambridge Univ Press, 1992.
- [32] 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.
- [33] J. G. Nagy and D. P. O’Leary. Restoring images degraded by spatially variant blur. *SIAM Journal on Scientific Computing*, 19:1063, 1998.
- [34] 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.
- [35] C. Preza and J.-A. Conchello. Depth-variant maximum-likelihood restoration for three-dimensional fluorescence microscopy. *JOSA A*, 21:1593–1601, 2004.
- [36] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60:259–268, 1992.
- [37] A. Sawchuk. Space-variant image motion degradation and restoration. *Proceedings of the IEEE*, 60:854–861, 1972.
- [38] A. A. Sawchuk. Space-variant image restoration by coordinate transformations. *Journal of the Optical Society of America*, 64:138–144, Feb. 1974.
- [39] 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.
- [40] 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.
- [41] 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.
- [42] 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.

- [43] 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 and 2

Before proving lemmas 1 and 2, we need additional results.

Lemma 6. *The integral operator \tilde{H}_m can be written as follows*

$$\tilde{H}_m u = \int_{\Omega} \tilde{K}(x, y) u(y) dy$$

where

$$\tilde{K}(x, y) = K(c(x), c(x) - x + y) \quad (31)$$

and where $c(x)$ denotes the center of the subregion containing x .

$$c(x) = \frac{\lfloor mx \rfloor}{m} + \frac{1}{2m} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

Proof. By construction, \tilde{K} should lead to piecewise convolutions. Therefore, we can associate a convolution kernel \tilde{h}_k for each ω_k defined for all $x \in \omega_k$ by $\tilde{K}(x, y) = \tilde{h}_k(x - y)$. Assuming that \tilde{h} corresponds to the kernel K at the center of ω_k , we get in each ω_k ,

$$\tilde{K}(c(x), \cdot) = K(c(x), \cdot) \Leftrightarrow K(c(x), \cdot) = \tilde{h}_k(c(x) - \cdot).$$

Hence, $\tilde{h}_k(y) = K(c(x), c(x) - y)$. □

Lemma 7 (Convolution). *An integral operator H with a $C^1(\Omega \times \Omega)$ kernel K is a convolution if and only if*

$$\forall (x, y) \in \Omega \times \Omega, \quad \nabla_x K(x, y) + \nabla_y K(x, y) = 0.$$

Proof. First suppose that H is a convolution. It means that $K(x, y) = k(x - y)$. A simple differentiation leads to

$$\nabla_x K(x, y) + \nabla_y K(x, y) = \nabla k(x - y) - \nabla k(x - y) = 0.$$

Second, suppose that

$$\forall (x, y) \in \Omega \times \Omega, \quad \nabla_x K(x, y) + \nabla_y K(x, y) = 0.$$

It implies that $K(x + h, y + h) = K(x, y)$ for all x, y, h . Therefore, choosing $h = -y$ leads to

$$K(x, y) = K(x - y, 0) = k(x - y), \quad \forall x, y.$$

□

Proof of Lemma 1. Since K is L -Lipschitz we get

$$\begin{aligned}
|K(x, y) - \tilde{K}(x, y)| &= |K(x, y) - K(c(x), c(x) - x + y)| \\
&= |K(x, y) - K(c(x), y) + K(c(x), y) - K(c(x), c(x) - x + y)| \\
&\leq |K(x, y) - K(c(x), y)| + |K(c(x), y) - K(c(x), c(x) - x + y)| \\
&\leq 2L \|x - c(x)\|_\infty.
\end{aligned}$$

To study the spectral norm, we consider a function u with $\|u\|_2 = 1$.

$$\begin{aligned}
\|Hu - \tilde{H}_m u\|_2^2 &= \int_{x \in \Omega} \left(\int_{y \in \Omega} (K(x, y) - \tilde{K}(x, y)) u(y) dy \right)^2 dx \\
&\leq \int_{x \in \Omega} \left(\int_{y \in \Omega} (K(x, y) - \tilde{K}(x, y))^2 dy \right) \left(\int_{y \in \Omega} u^2(y) dy \right) dx \\
&= \int_{x \in \Omega} \int_{y \in \Omega} (K(x, y) - \tilde{K}(x, y))^2 dy dx \\
&\leq \int_{x \in \Omega} \int_{y \in \Omega} 4L^2 \|x - c(x)\|_\infty^2 dy dx \\
&= 4L^2 \int_{x \in \Omega} \|x - c(x)\|_\infty^2 dx, \quad \text{since } |\Omega| = 1 \\
&\leq 4L^2 \frac{1}{4m^2} \\
&= \frac{L^2}{m^2}.
\end{aligned}$$

□

Proof of Lemma 2. The idea of the proof is to exhibit a pathological unit vector $u \in L^2(\Omega)$ such that $\|Hu - \tilde{H}_m u\|_2$ is of order $\frac{1}{m}$.

If H is a \mathcal{C}^1 kernel that is not a convolution, Lemma 7 ensures that there exists $(x^0, y^0) \in \Omega \times \Omega$, $v \in \mathbb{R}^d$ with $\|v\|_\infty = 1$ and $\epsilon > 0$ such that $|\langle \nabla_x K(x^0, y^0) + \nabla_y K(x^0, y^0), v \rangle| \geq 2\epsilon$. We assume, without loss of generality, that the sign of the difference is positive, i.e. $\langle \nabla_x K(x^0, y^0) + \nabla_y K(x^0, y^0), v \rangle \geq 2\epsilon$. Since $K \in \mathcal{C}^1(\Omega \times \Omega)$, ∇K is continuous and therefore, there exists $h > 0$ such that

$$\langle \nabla_x K(x, y) + \nabla_y K(x, y), v \rangle \geq \epsilon, \quad \forall (x, y) \in I \times J, \quad (32)$$

with $I = x^0 + [-h, h]^d$ and $J = y^0 + [-\frac{3}{2}h, \frac{3}{2}h]^d$.

Now, set $J_- = y^0 + [-h, h]^d$ and $u = \frac{\mathbf{1}_{J_-}}{\sqrt{|J_-|}}$. We have $\|u\|_2 = 1$ and therefore:

$$\|\tilde{H}_m - H\|_{2 \rightarrow 2}^2 \geq \|\tilde{H}_m u - Hu\|_2^2$$

$$\begin{aligned}
&= \int_{x \in \Omega} \left(\int_{y \in \Omega} (\tilde{K}(x, y) - K(x, y)) u(y) dy \right)^2 dx \\
&\geq \frac{1}{|J_-|} \int_{x \in I} \left(\int_{y \in J_-} \tilde{K}(x, y) - K(x, y) dy \right)^2 dx. \tag{33}
\end{aligned}$$

We now need to bound the difference $\tilde{K}(x, y) - K(x, y)$ from below on $I \times J_-$.

Let $E = \{k \in \mathbb{N}^d, \omega_k \subset I\}$. If $mh \geq 1$ (i.e. m is large enough) then E is non empty, and $|E| \gtrsim m^d$. Now, let $k \in E$, $x \in \omega_k$ and $y \in \Omega$. By Lemma 6:

$$\begin{aligned}
K(x, y) - \tilde{K}(x, y) &= K(x, y) - K(c_k, c_k - x + y) \\
&= \int_{t=0}^1 \left\langle \nabla K(c_k + t(x - c_k), c_k - x + y + t(x - c_k)), \begin{pmatrix} x - c_k \\ x - c_k \end{pmatrix} \right\rangle dt.
\end{aligned}$$

Any $x \in \omega_k$ can be written as $x = c_k + \alpha v + w$, where $w \in \text{span}(v)^\perp$ and $\|\alpha v + w\|_\infty \leq \frac{1}{2m}$. Thus, for $y \in J_-$:

$$K(x, y) - \tilde{K}(x, y) = \int_{t=0}^1 \left\langle \nabla K(c_k + t(\alpha v + w), y + (t-1)(\alpha v + w)), \begin{pmatrix} \alpha v + w \\ \alpha v + w \end{pmatrix} \right\rangle dt.$$

Since $\omega_k \subset I$, $c_k + t(\alpha v + w) \in I, \forall t \in [0, 1]$. Similarly, since $y \in J_-$, $\|\alpha v + w\|_\infty \leq \frac{1}{2m}$ and $h \geq 1/m$, point $y + (t-1)(\alpha v + w) \in J, \forall t \in [0, 1]$. Therefore, the whole segment $\{(c_k + t(\alpha v + w), y + (t-1)(\alpha v + w)), t \in [0, 1]\} \subset I \times J$. As a consequence, inequality (32) holds on all the integration domain and

$$K(x, y) - \tilde{K}(x, y) \geq \alpha \epsilon - M \|w\|_\infty,$$

where $M = d \sup_{(x, y) \in \Omega \times \Omega} \|\nabla K(x, y)\|_\infty$. By setting $M' = \max(M, \epsilon)$ we still have:

$$\tilde{K}(x, y) - K(x, y) \geq \alpha \epsilon - M' \|w\|_\infty. \tag{34}$$

Now, define $S_k = \{c_k + \alpha v + w, \alpha \in [\frac{1}{4m}, \frac{1}{2m} - \frac{\epsilon}{8M'm}], \|w\|_\infty \leq \frac{\epsilon}{8M'm}\}$. By construction, $\|\alpha v + w\|_\infty \leq \alpha + \|w\|_\infty \leq \frac{1}{2m}$, therefore $S_k \subset \omega_k$. Moreover, $|S_k| \geq \frac{1}{8m} (\frac{\epsilon}{8M'm})^{d-1} \gtrsim \frac{1}{m^d}$. Finally, from bound (34), $K(x, y) - \tilde{K}(x, y) \geq \frac{\epsilon}{8m}$.

To finish the proof, we go back to inequality (33) and obtain:

$$\begin{aligned}
\|\tilde{H}_m - H\|_{2 \rightarrow 2}^2 &\geq \frac{1}{|J_-|} \int_{x \in \cup_{k \in E} S_k} \left(\int_{y \in J_-} \tilde{K}(x, y) - K(x, y) dy \right)^2 dx \\
&= \frac{1}{|J_-|} \sum_{k \in E} \int_{x \in S_k} \left(\int_{y \in J_-} \tilde{K}(x, y) - K(x, y) dy \right)^2 dx \\
&\gtrsim \frac{1}{|J_-|} \sum_{k \in E} \int_{x \in S_k} \left(\int_{y \in J_-} \frac{1}{m} dy \right)^2 dx
\end{aligned}$$

$$\gtrsim \frac{1}{|J_-|} m^d \frac{1}{m^d} \left(\frac{|J_-|}{m} \right)^2 \gtrsim \frac{1}{m^2}.$$

□

B Proof of Theorem 3

Proof of i). The discrete convolutions have to be performed on extended regions of size $(N/m + \kappa N)^d$ in order to correctly handle the boundary conditions. Each convolution is thus performed on a sub-region with $(N/m + \kappa N)^d$ pixels. This can be performed in $\mathcal{O}(d(N/m + \kappa N)^d \log(N/m + \kappa N))$ arithmetic operations. Since there are m^d subregions, we get the announced result. □

Proof of ii). The proof is similar to the one in the continuous setting. The only additional requirement is that N and m are sufficiently large to partition I and J in at least 2^d subregions. □

Proof of iii). This is a simple consequence of complexity bound (13) and the bounds estimates provided in equations (14). In order to satisfy $\|\mathbf{H} - \tilde{\mathbf{H}}_m\|_{2 \rightarrow 2} \leq \epsilon$, it is sufficient to set $\frac{L}{2m} \leq \epsilon$. The smallest integer ensuring this condition is $m = \lceil \frac{L}{2\epsilon} \rceil$. Therefore, the numerical complexity of the method becomes

$$dN^d \left(1 + \frac{\kappa L}{\epsilon} \right)^d \log \left(\frac{N\epsilon}{L} + \kappa N \right).$$

The asymptotics of this complexity for small ϵ is $\frac{dL\kappa N^d \log(\kappa N)}{\epsilon^d}$. □

C Proof of Lemma 4

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

Lemma 8 below is a common result in numerical analysis [16] (see also Theorem 3.2.1 in [9]). 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 8 (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*

$$\inf_{g \in \Pi_M} \|f - g\|_{\mathbb{L}^p(\Omega)} \leq C \|f\|_{W^{M+1,p}(\Omega)}, \quad (35)$$

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

$$\inf_{g \in \Pi_M} \|f - g\|_{\mathbb{L}^p(I_h)} \leq Ch^{M+1} |f|_{W^{M+1,p}(I_h)}, \quad (36)$$

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

Let $m = (m_1, \dots, m_d)$. Let $I_{j,m} = \text{supp}(\psi_{j,m}^e)$ and $\lambda_{j,m}$ denote its center. From the wavelets definition, we get

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

therefore $|I_{j,m}| = c(M)^d \cdot 2^{-jd}$ and $\lambda_{j,m} = 2^{-j}m$. We will now prove Lemma 4.

Proof of Lemma 4. Since the mapping $(x, y) \mapsto K(x, y)\psi_{j,m}^e(y)\psi_{k,n}^{e'}(x)$ is bounded, it is also absolutely integrable on compact domains. Therefore $\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle$ is well-defined for all (j, m, k, n) . Moreover Fubini's theorem can be applied and we get

$$\begin{aligned} \langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle &= \int_{I_{k,n}} \int_{I_{j,m}} K(x, y)\psi_{j,m}^e(y)\psi_{k,n}^{e'}(x) dy dx \\ &= \int_{I_{j,m}} \int_{I_{k,n}} K(x, y)\psi_{j,m}^e(y)\psi_{k,n}^{e'}(x) dx dy. \end{aligned}$$

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 $\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle = \langle \psi_{j,m}^e, H^*\psi_{k,n}^{e'} \rangle$ 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 $g \in \Pi_{M-1}$, $\int_{I_{k,n}} g(x)\psi_{k,n}^{e'}(x) dx = 0$ since ψ has M vanishing moments. Therefore,

$$\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle = \int_{I_{j,m}} \inf_{g \in \Pi_{M-1}} \int_{I_{k,n}} (K(x, y) - g(x)) \psi_{j,m}^e(y) \psi_{k,n}^{e'}(x) dx dy,$$

and

$$\begin{aligned} \left| \langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle \right| &\leq \int_{I_{j,m}} \inf_{g \in \Pi_{M-1}} \int_{I_{k,n}} |K(x, y) - g(x)| |\psi_{j,m}^e(y)| |\psi_{k,n}^{e'}(x)| dx dy \\ &\leq \int_{I_{j,m}} \inf_{g \in \Pi_{M-1}} \|K(\cdot, y) - g\|_{\mathbb{L}^\infty(I_{k,n})} \|\psi_{k,n}^{e'}\|_{\mathbb{L}^1(I_{k,n})} |\psi_{j,m}^e(y)| dy. \end{aligned}$$

By Lemma 8, $\inf_{g \in \Pi_{M-1}} \|K(\cdot, y) - g\|_{\mathbb{L}^\infty(I_{k,n})} \lesssim 2^{-kM} |K(\cdot, y)|_{W^{M,\infty}(I_{k,n})}$ since $I_{k,n}$ is a cube of sidelength $c(M) \cdot 2^{-k}$. We thus obtain

$$\left| \langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \rangle \right| \lesssim 2^{-kM} \|\psi_{k,n}^{e'}\|_{\mathbb{L}^1(I_{k,n})} \|\psi_{j,m}^e\|_{\mathbb{L}^1(I_{j,m})} \text{ess sup}_{y \in I_{j,m}} |K(\cdot, y)|_{W^{M,\infty}(I_{k,n})}$$

$$\lesssim 2^{-kM} 2^{-\frac{dj}{2}} 2^{-\frac{dk}{2}} \operatorname{ess\,sup}_{y \in I_{j,m}} |K(\cdot, y)|_{W^{M,\infty}(I_{k,n})}$$

since $\|\psi_{j,m}^e\|_{\mathbb{L}^1} = 2^{-\frac{dj}{2}} \|\psi\|_{\mathbb{L}^1}$.

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

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

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

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

The case $k < j$ gives

$$\left| \left\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \right\rangle \right| \lesssim 2^{-(M+\frac{d}{2})|j-k|} 2^{-k(M+d)} f(\operatorname{dist}(I_{j,m}, I_{k,n})),$$

which allows to conclude that

$$\left| \left\langle H\psi_{j,m}^e, \psi_{k,n}^{e'} \right\rangle \right| \lesssim 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} f(\operatorname{dist}(I_{j,m}, I_{k,n})),$$

□

D Proof of Theorem 5

Let us begin with some preliminary results. Since f is compactly supported on $[0, \kappa]$ and bounded by c_f , we have $f(j, m, k, n) = f(\operatorname{dist}(I_{j,m}, I_{k,n})) \leq c_f \mathbb{1}_{\operatorname{dist}(I_{j,m}, I_{k,n}) \leq \kappa}$. By equation (20), $\operatorname{dist}(I_{j,m}, I_{k,n}) \leq \kappa$ 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 9. *Define*

$$\mathcal{G}_{j,k}^{e,e'} = \left\{ (m, n) \in \mathcal{T}_j \times \mathcal{T}_k \mid \mathbb{1}_{\operatorname{dist}(I_{j,m}, I_{k,n}) \leq \kappa} = 1 \right\}.$$

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

Proof. First note that

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

Now, define $\mathcal{G}_{j,k,m}^{e,e'} = \left\{ n \in \mathcal{T}_k \mid (m,n) \in \mathcal{G}_{j,k}^{e,e'} \right\}$. 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 $\left| \mathcal{G}_{j,k,m}^{e,e'} \right| \leq (2^{k+1}R_{j,k}^\kappa)^d$ 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). First note that a sufficient condition for $2^{-\min(j,k)(M+d)} f_{j,m,k,n} \leq \eta$ is that $\min(j,k) \geq J(\eta)$ with $J(\eta) = \frac{-\log_2(\eta/c_f)}{M+d}$. In the following, we let $\tilde{J}(\eta) = \min(J(\eta), \log_2(N))$ 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

$$\begin{aligned} \#\mathcal{G} &= \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \sum_{e,e' \in \{0,1\}^d} \#\mathcal{G}_{j,k}^{e,e'} \mathbf{1}_{\min(j,k) < J(\eta)} \\ &\lesssim (2^d - 1)^2 \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} 2^{kd} \left(\frac{c(M)}{2} (2^{-j} + 2^{-k}) + \kappa \right)^d \\ &\lesssim \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} 2^{kd} \left(\frac{c(M)^d}{2^d} 2^{-dj} + \frac{c(M)^d}{2^d} 2^{-dk} + \kappa^d \right) \\ &\lesssim \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} + \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{jd} \\ &\quad + \sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd} 2^{jd} \kappa^d. \end{aligned}$$

The first sum yields

$$\sum_{j=0}^{\log_2(N)-1} \sum_{k=0}^{\log_2(N)-1} \mathbf{1}_{\min(j,k) < J(\eta)} 2^{kd}$$

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

The second sum is handled similarly and the third sum gives

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

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

$$\left\| \Delta_\eta \right\|_{2 \rightarrow 2}^2 \leq \left\| \Delta_\eta \right\|_{1 \rightarrow 1} \left\| \Delta_\eta \right\|_{\infty \rightarrow \infty}. \quad (37)$$

Since the upper-bound (19) is symmetric,

$$\left\| \Delta_\eta \right\|_{\infty \rightarrow \infty} = \left\| \Delta_\eta \right\|_{1 \rightarrow 1} = \max_{j,m,e} \sum_{k,n} \left| \Delta_{j,m,k,n}^{e,e'} \right|$$

By definition of Θ_η we get that

$$\begin{aligned}
&\sum_{k,n,e'} \left| \Delta_{j,m,k,n}^{e,e'} \right| = \sum_{k=0}^{\log_2(N)-1} \sum_{e' \in \{0,1\}^d \setminus \{0\}} \sum_{n \in \mathcal{G}_{j,k,m}^{e,e'}} \left| \theta_{j,m,k,n}^{e,e'} \right| \mathbb{1}_{\min(j,k) > J(\eta)} \\
&\lesssim \sum_{k=0}^{\log_2(N)-1} \sum_{e' \in \{0,1\}^d \setminus \{0\}} \sum_{n \in \mathcal{G}_{j,k,m}^{e,e'}} 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} \mathbb{1}_{\min(j,k) > J(\eta)}.
\end{aligned}$$

Then

$$\begin{aligned}
\sum_{k,n,e'} \left| \Delta_{j,m,k,n}^{e,e'} \right| &\lesssim \sum_{k=0}^{\log_2(N)-1} 2^{-(M+\frac{d}{2})|j-k|} 2^{-\min(j,k)(M+d)} \mathbf{1}_{\min(j,k) > J(\eta)} \left| \mathcal{G}_{j,k}^{e,e'} \right| \\
&\lesssim \sum_{k=0}^{j-1} (2^k R_{j,k}^\kappa)^d 2^{(k-j)(M+d/2)} 2^{-k(M+d)} \mathbf{1}_{k > J(\eta)} \\
&\quad + \sum_{k=j}^{\log_2(N)-1} (2^k R_{j,k}^\kappa)^d 2^{(j-k)(M+d/2)} 2^{-j(M+d)} \mathbf{1}_{j > J(\eta)}.
\end{aligned}$$

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:

$$A_2 = \mathbf{1}_{j > J(\eta)} 2^{-jd/2} \sum_{k=j}^{\log_2(N)} (R_{j,k}^\kappa)^d 2^{-k(M-d/2)}.$$

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

$$\begin{aligned}
A_2 &\lesssim \mathbf{1}_{j > J(\eta)} 2^{-jd/2} \sum_{k=j}^{\log_2(N)} \left(2^{-jd} + 2^{-kd} + \kappa^d \right) 2^{-k(M-d/2)} \\
&\lesssim \mathbf{1}_{j > J(\eta)} 2^{-jd/2} \left(2^{-jd} 2^{-j(M-d/2)} + 2^{-j(M+d/2)} + \kappa^d 2^{-j(M-d/2)} \right) \\
&\lesssim \mathbf{1}_{j > J(\eta)} 2^{-jM} \left(2^{-jd} + \kappa^d \right).
\end{aligned}$$

Hence

$$\sum_{k,n,e'} |\Delta_{j,m,k,n}| \lesssim \mathbf{1}_{j>J(\eta)} 2^{-jM} \left(2^{-jd} + \kappa^d + 2^{-\frac{d}{2}(J(\eta)+j)} \right).$$

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 (37) to conclude. □

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