



**HAL**  
open science

# Non-local means and optimal weights for noise removal

Qiyu Jin, Ion Grama, Charles Kervrann, Quansheng Liu

► **To cite this version:**

Qiyu Jin, Ion Grama, Charles Kervrann, Quansheng Liu. Non-local means and optimal weights for noise removal. SIAM Journal on Imaging Sciences, 2017. hal-01575918

**HAL Id: hal-01575918**

**<https://inria.hal.science/hal-01575918>**

Submitted on 21 Aug 2017

**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.

---

# Non-local means and optimal weights for noise removal

Qiyu Jin<sup>1</sup>, Ion Grama<sup>2</sup>, Charles Kervrann<sup>3</sup>, Quansheng Liu<sup>2,4</sup>

<sup>1</sup> School of mathematical science, Inner Mongolia University,  
No. 235 Daxue West Road, Hohhot, China, 010021

<sup>2</sup> Université de Bretagne-Sud, Campus de Tohaninic, BP 573 56017 Vannes, France

<sup>3</sup>Centre Inria Rennes - Bretagne Atlantique, Serpico Project-Team, Campus Universitaire de Beaulieu,  
35042 Rennes Cedex, France

<sup>4</sup>School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing  
210044, China

---

## Abstract

In this paper, a new denoising algorithm to deal with the additive white Gaussian noise model is described. In the line of work of the Non-Local means approach, we propose an adaptive estimator based on the weighted average of observations taken in a neighborhood with weights depending on the similarity of local patches. The idea is to compute adaptive weights that best minimize an upper bound of the pointwise  $L_2$  risk. In the framework of adaptive estimation, we show that the “oracle” weights are optimal if we consider triangular kernels instead of the commonly-used Gaussian kernel. Furthermore, we propose a way to automatically choose the spatially varying smoothing parameter for adaptive denoising. Under conventional minimal regularity conditions, the obtained estimator converges at the usual optimal rate. The implementation of the proposed algorithm is also straightforward and the simulations show that our algorithm improves significantly the classical NL-means and is competitive when compared to the more sophisticated NL-means filters both in terms of PSNR values and visual quality.

**Keywords :** image denoising, image patches, nonparametric estimation, pointwise  $L_2$  risk, optimization.

## 1 Introduction

Several mathematical frameworks have been proposed to solve the image denoising problem including partial derivative equations [1], frequency-based methods [2], Markov Random Fields methods [3], locally adaptive kernel-based methods [4] and sparse representation [5, 6]. The goal is to estimate an image assumed to be corrupted with additive white Gaussian (WGN) noise. The white Gaussian noise is generally considered as a good approximation of source of corruptions in many real situations [7]. Formally, we consider

$$v(x) = f(x) + \varepsilon(x), \quad x \in \Omega \tag{1}$$

where  $\Omega$  is a uniform  $N \times N$  grid of pixels on the unit square,  $v = (v(x))_{x \in \Omega}$  is the observed image brightness,  $f : \Omega \subset \mathbf{R}^2 \rightarrow [0, f_{\max}]$  is an unknown target regression function,  $f_{\max}$  is the maximum

intensity value in the image  $f$ , and  $\varepsilon = (\varepsilon(x))_{x \in \Omega}$  are independent and identically distributed Gaussian random variables with mean 0 and standard deviation  $\sigma > 0$ .

In the last decade, the state-of-the-art results have been considerably improved by considering patches in the case of Gaussian noise removal. The theoretical limits of denoising algorithms are at present more frequently discussed in the literature [8, 9, 10] mainly because the most competitive algorithms including BM3D [11], LSSC [6], EPLL [12], NL-Bayes [13], PEWA [14], PLOW [15], S-PLE [16], produce surprisingly similar PSNR values. The most famous BM3D algorithm typically combines clustering of noisy patches, DCT-based transform and shrinkage operation to achieve the current state-of-the-art results [11]. It is worth noting that the aforementioned algorithms are mostly inspired from the N(on) L(ocal)-means filter [7]. The basic idea of NL-means is to estimate the unknown image  $f(x_0)$  at location  $x_0 \in \Omega$  by a weighted average of observations in the noisy input image defined as :

$$\widehat{f}(x_0) = \sum_{x \in U_h(x_0)} w(x, x_0) v(x) \quad \text{subject to} \quad w(x, x_0) \geq 0 \quad \text{and} \quad \sum_{x \in \Omega} w(x, x_0) = 1, \quad (2)$$

where  $U_h(x_0) \subset \Omega$  is the *search window* which can be the whole image [7]. The choice of the non-negative weights  $w = (w(x, x_0))_{x \in \Omega}$  is based essentially on two criteria : a local criterion so that the weight  $w(x, x_0)$  is a decreasing function of the spatial distance of the pixel  $x$  to the current pixel  $x_0$ , and a non-local criterion which gives more importance to pixel  $x$  when the brightness  $v(x)$  is close to  $v(x_0)$  at pixel  $x_0$ . Instead of considering pointwise brightness [17, 18, 19], an important feature of NL-means consists in attaching small square regions (or patches) to each pixel and comparing patches. In general, image patches are viewed as pictorial elements able to take into account semi-local spatial contexts in the 2D image. The weights are chosen in the family of Gaussian kernels since they usually produce the best denoising results in experiments.

In the continuity of [7], several filters with data-dependent weights have been studied in the statistical framework, among which we mention [20, 21, 7, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 19, 36, 37, 38]. In [23, 26], the authors analysed the bias-variance trade-off in the NL-means framework to select an optimal neighborhood ; a sequence of increasing neighborhoods and a discrete setting has been proposed to minimize the mean square error (MSE) of the estimator. In [24] [27], [30], [34], [39], the authors focused on the selection of the global bandwidth parameter. Besides, Stein's Unbiased Risk Estimator (SURE) has recently been investigated for selecting the global [30] and local [40] smoothing parameters of the NL-means. In most of the aforementioned papers, it is established that the bandwidth is a function of the noise variance. Note that, contrary to [26], it is recommended in [40] to adapt the smoothing parameter rather than the search window size. In all these papers, the starting point is the definition of NL-means (exponential form). Our starting point is different since our objective is to derive a filter (2) by minimizing a tight bound of the MSE of the estimator  $\widehat{f}(x_0)$  at each location  $x_0 \in \Omega$  defined as :

$$R(\widehat{f}(x_0)) = \mathbb{E}(\widehat{f}(x_0) - f(x_0))^2, \quad (3)$$

with respect to weights  $w$  under the constraints  $w(x, x_0) \geq 0$  and  $\sum_{x \in \Omega} w(x, x_0) = 1$ . As a result we obtain an explicit formulation of the optimal weights  $w^*$  depending on the unknown function  $f$ . In order to get a computable estimator, we approximate  $w^*$  by some adaptive weights  $\widehat{w}$  based on patches from the noisy image  $v$ . We carefully investigated the calibration of the distance between patches and a convergence analysis is provided. We also establish a formal equation that relates the smoothing parameter to the noise variance. Finally, we obtain the so-called *Optimal Weights Filter* (OWF) for which the smoothing parameter is locally computed. The corresponding algorithm is

straightforward to implement and produces competitive results when compared to several variants of NL-means [23, 26, 41]. With the proposed weights optimization approach the smoothing parameter is defined adaptively for every search window. The search window can be very large since the weights selects the relevant data points for averaging. In practical imaging, we only need to adjust the patch size. Finally, the proposed OWF procedure is shown to be more robust if we consider large patch sizes (reduction of mottling effects) combined with an adaptive choice of the smoothing parameter depending on the image contents and weights with compact support. In that sense, we confirm the previous analysis presented in [40].

In the last part of the paper, we address the problem of convergence of the estimator to the true image in the line of work of [7, 23, 26] and [42]. Here, we mention that the NL-means was interpreted as a special case of the Exponential Weighted Aggregation (EWA) procedure that achieves the optimal average risk [43, 35, 14]. Related optimization problems for nonparametric signal and density recovering have been proposed earlier in [44, 45, 46, 47, 48]. In these papers the weights are optimized over a given class of regular functions and thus depend only on some parameters of the class. In [10, 49], the authors analysed the performance for a specific class of functions (images with sharp edges). Our second contribution is then to find the optimal weights depending on the image  $f$  at hand, which implies that OWF automatically attains the optimal rate of convergence for each particular image  $f$  under some regularity conditions on  $f$ . Results of this type are related to the “oracle” concept developed by Donoho & Johnstone [50].

The remainder of the paper is organized as follows. Our new estimator based on the optimization of weights is presented in Section 2. We give an implementation of the estimator and a procedure to select automatically the spatially varying bandwidth. The statistical and theoretical properties are presented in Section 3.1 where we give the rate of convergence of the constructed estimators. In Section 4, we present our simulation results and demonstrate the potential of the algorithm on artificially noisy images and real images.

**Notations.** The Euclidean norm of a vector  $\mathbf{z} = (z_1, \dots, z_d)^T \in \mathbf{R}^d$  is denoted by  $\|\mathbf{z}\|_2 = \sqrt{\sum_{i=1}^d z_i^2}$  and the supremum norm of  $\mathbf{z}$  is denoted by  $\|\mathbf{z}\|_\infty = \sup_{1 \leq i \leq d} |z_i|$ . The positive part of a real number  $z$  is denoted by  $[z]_+ : [z]_+ = z$  if  $z \geq 0$  and 0 otherwise.

Define a uniform  $N \times N$ -grid  $\Omega$  of pixels on the unit square as

$$\Omega = \left\{ \frac{1}{N}, \frac{2}{N}, \dots, \frac{N-1}{N}, 1 \right\}^2. \quad (4)$$

Each element  $x$  of the grid  $\Omega$  is a pixel and  $\text{card}(\Omega) = N^2 = n$ . To deal with the boundary of the image we mirror the image and the set  $\Omega$  outside the image limits, that is we extend the image symmetrically with respect to the border and the corners. The extended domain of the image is denoted by  $\bar{\Omega}$ . For any pixel  $x_0 \in \Omega$  and a given  $h > 0$  assumed to be fixed in the following, the square window of pixels

$$U_h(x_0) = \{x \in \bar{\Omega} : \|x - x_0\|_\infty \leq h\} \quad (5)$$

is the *search window*  $U_h(x_0) \subset \bar{\Omega}$  centered at pixel  $x_0$ . We choose  $h$  as a multiple of  $\frac{1}{N}$  (i.e.  $h = \frac{k}{N}$  for some  $k \in \{1, 2, \dots, N\}$ ). We denote  $M$  the number of pixels falling in the square search window  $U_h(x_0)$  :

$$M = \text{card}(U_h(x_0)) = (2Nh + 1)^2. \quad (6)$$

The width  $h$  of the search window  $U_h(x_0)$  plays an important role in the NL-means.

For any pixel  $x \in U_h(x_0)$  a square patch  $\mathbf{v}(x) = (v(y))_{y \in V(x)}$  of size

$$m = \text{card}(V(x)) = (2r + 1)^2 \quad (7)$$

( $r > 0$  is an integer) and width  $\eta = \frac{r}{N}$ , is formed by the noisy data  $v(y)$  observed at locations  $y$  in the square window  $V(x) \subset \bar{\Omega}$  centered at pixel  $x$  :

$$\mathbf{v}(x) := (v(y), y \in V(x)), \quad y = x + \tau, \tau \in \mathcal{T} = \left\{ -\frac{r}{N}, \dots, +\frac{r}{N} \right\}^2. \quad (8)$$

Without loss of generality, isotropic circular patches could be considered as well.

## 2 Statistical framework and optimal weights filter

For any pixel  $x_0 \in \Omega$ , consider a family of weighted estimates  $\hat{f}(x_0)$  of the form

$$\hat{f}(x_0) = \sum_{x \in U_h(x_0)} w(x, x_0) v(x) \quad \text{subject to} \quad w(x, x_0) \geq 0, \quad \sum_{x \in U_h(x_0)} w(x, x_0) = 1, \quad (9)$$

where the weights  $w(x, x_0)$  are non-random. The usual ‘‘bias-variance’’ decomposition of the point-wise  $L_2$  risk of this family of estimators has the following form

$$\mathbb{E} \left( \hat{f}(x_0) - f(x_0) \right)^2 = \text{Bias}^2(\hat{f}(x_0)) + \text{Var}(\hat{f}(x_0)), \quad (10)$$

with

$$\begin{aligned} \text{Bias}(\hat{f}(x_0)) &:= \mathbb{E} \left( \hat{f}(x_0) - f(x_0) \right) = \sum_{x \in U_h(x_0)} w(x, x_0) \mathbb{E}(v(x) - f(x_0)) \\ &= \sum_{x \in U_h(x_0)} w(x, x_0) (f(x) - f(x_0)), \end{aligned} \quad (11)$$

$$\begin{aligned} \text{Var}(\hat{f}(x_0)) &:= \mathbb{E} \left( \hat{f}(x_0) - \mathbb{E}(\hat{f}(x_0)) \right)^2 = \mathbb{E} \left( \sum_{x \in U_h(x_0)} w(x, x_0) (v(x) - f(x)) \right)^2 \\ &= \sigma^2 \sum_{x \in U_h(x_0)} w(x, x_0)^2. \end{aligned} \quad (12)$$

The decomposition (10) is commonly used to construct asymptotically minimax estimators over some given classes of functions in the nonparametric function estimation (see [44]). With our approach the bias term is bounded in terms of the unknown function  $f$  itself. As a result we obtain some ‘‘oracle’’ weights  $w$  adapted to the unknown function  $f$  at hand, which will be estimated further using noisy patches from the input image  $v$ .

### 2.1 Oracle weights

First, we shall address the problem of determining the oracle weights.

From the decomposition (10), we easily obtain the following upper bound :

$$\mathbb{E} \left( \hat{f}(x_0) - f(x_0) \right)^2 \leq \left( \sum_{x \in U_h(x_0)} w(x, x_0) |f(x) - f(x_0)| \right)^2 + \sigma^2 \sum_{x \in U_h(x_0)} w(x, x_0)^2. \quad (13)$$

In what follows, we shall minimize the upper bound in (13) instead of the expectation in the left hand side. There are mainly two reasons for this. First, the right hand side of (13) is much easier to deal with and the solution is surprisingly explicit and simple as demonstrated in this section (triangular kernel with adaptive bandwidth). Second, the simulation results that we present in Table 1 show that the “oracle” filter coming from the optimization of the right hand side of (13) is able to produce excellent results with very high PSNR values.

Replacing the distance  $|f(x) - f(x_0)|$  by an arbitrary non-negative function  $\phi$  on  $U_h(x_0) \times U_h(x_0)$  with  $\phi(x_0, x_0) = 0$ , we consider the more general objective functional

$$\mathcal{J}_\phi(w) := \left( \sum_{x \in U_h(x_0)} w(x, x_0) \phi(x, x_0) \right)^2 + \sigma^2 \sum_{x \in U_h(x_0)} w(x, x_0)^2, \quad (14)$$

where  $w$  is the vector whose components are  $w(x, x_0)$ ,  $x \in U_h(x_0)$ .

**Theorem 2.1.** *Assume that  $\phi$  is a non-negative function on  $U_h(x_0) \times U_h(x_0)$  satisfying  $\phi(x_0, x_0) = 0$ , not identically 0. The unique closed-form solution of*

$$\min_w \mathcal{J}_\phi(w) \quad \text{subject to} \quad w(x, x_0) \geq 0 \quad \text{and} \quad \sum_{x \in U_h(x_0)} w(x, x_0) = 1 \quad (15)$$

is given by

$$w(x, x_0) = \frac{K_\Delta \left( \frac{\phi(x, x_0)}{a(x_0)} \right)}{\sum_{y \in U_h(x_0)} K_\Delta \left( \frac{\phi(y, x_0)}{a(x_0)} \right)}, \quad x \in U_h(x_0), \quad (16)$$

where  $K_\Delta(z) := [1 - |z|]_+$ ,  $z \in \mathbb{R}$ , is the triangular kernel and the spatially varying bandwidth  $a(x_0) = a_h(x_0) > 0$  is the unique solution on  $(0, \infty)$  of the equation

$$H_\phi(a(x_0)) := \sum_{x \in U_h(x_0)} \phi(x, x_0) [a(x_0) - \phi(x, x_0)]_+ = \sigma^2. \quad (17)$$

*Proof* : see Appendix A.1 ■

Theorem 2.1 (see (17)) establishes a formal relationship between the noise variance  $\sigma^2$  and the variable bandwidth  $a(x_0)$  which plays a central role in most of smoothing kernel methods for image denoising [20, 7, 23, 26, 24, 25, 30, 40, 29, 34, 35, 36]. A computational procedure will be presented in Section 2.3 to determine the variable bandwidth at each spatial location.

## 2.2 Patch-based representation and optimal weights filter

Using the weights (16) which are obtained as optimal for the problem (15), we define the following oracle filter :

$$f_h^*(x_0) := \frac{\sum_{x \in U_h(x_0)} K_\Delta \left( \frac{\phi(x, x_0)}{a(x_0)} \right) v(x)}{\sum_{y \in U_h(x_0)} K_\Delta \left( \frac{\phi(y, x_0)}{a(x_0)} \right)}, \quad x \in U_h(x_0), \quad (18)$$

where  $\phi(x, x_0) = |f(x) - f(x_0)|$ ,  $a(x_0) = a_h(x_0)$  is the solution of (17) and  $h > 0$ .

Let us briefly explain an important effect of adaptivity of the oracle filter, which also holds true for its computable version. Note that  $f_h^*(x_0)$  is actually a bilateral filter, which has two bandwidths parameters  $h$  and  $a(x_0) = a_h(x_0)$ . When the bandwidth  $h$  is fixed the bandwidth  $a(x_0)$  is determined automatically as the solution of the equation (17). Accordingly, in what follows, we omit the subscript  $h$  for simplicity but we write  $a(x_0) = a_h(x_0)$  when it is necessary to indicate the dependence on  $h$ . Note also that the number of pixels used in the estimation may be considerably less than the cardinality of the set  $U_h(x_0)$ , which is explained by the fact that some weights can be equal to zero : this happens when  $\phi(x, x_0) > a(x_0)$ . The adaptive neighborhood corresponding to positive weights is inside the search window and its geometry is entirely determined by the values  $\phi(x, x_0)$  for  $x \in U_h(x_0)$ . In this sense our oracle estimator has an adaptive window of varying geometry. In simulations we shall choose  $h$  large so that the oracle filter  $f_h^*(x_0)$  has practically an automatic bandwidth.

The oracle filter (18) is non computable from the noisy image, since the variation  $|f(x) - f(x_0)|$  depends on the true image  $f$ . To derive a computable estimator, we have to replace the unknown variation  $|f(x) - f(x_0)|$  by a quantity which can be computed from the noisy image  $v$ . To this end, let us consider the Gaussian noise model (1) and write :

$$v(x) - v(x_0) = f(x) - f(x_0) + \epsilon(x) - \epsilon(x_0). \quad (19)$$

It follows that

$$|f(x) - f(x_0)|^2 = \mathbb{E}(v(x) - v(x_0))^2 - 2\sigma^2, \quad \forall x \neq x_0. \quad (20)$$

Consider first the following approximation :

$$|f(x) - f(x_0)|^2 \approx \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}^2 - 2\sigma^2 \quad (21)$$

with

$$\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}^2 = \sum_{\tau \in \mathcal{T}} K_0(\tau)(v(x + \tau) - v(x_0 + \tau))^2, \quad (22)$$

where  $K_0(\cdot)$  is a normalized kernel such that  $\sum_{y \in \mathcal{T}} K_0(y) = 1$ , which is generally used to take into account the distance between the central pixel and other pixels in the patch. If we choose  $K_0$  to be the rectangular kernel, we obtain

$$|f(x) - f(x_0)|^2 \approx \frac{1}{m} \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_2^2 - 2\sigma^2. \quad (23)$$

This approximation based on the distance between two patches  $\mathbf{v}(x)$  and  $\mathbf{v}(x_0)$  fulfills the *patch regularity* assumption (“similar patches have similar central pixels”) discussed in [40]. Nevertheless, while the approximation (23) appears natural, it seems unsuitable for use with the triangular kernel  $K_\Delta$  under consideration. Our data driven substitute is not based on a direct approximation of the variation  $|f(x) - f(x_0)|$ , but rather on a quantity which is basically bounded by  $|f(x) - f(x_0)|$ . We define a substitute for  $\phi(x, x_0)$  as follows :

$$\phi_{\mathbf{v}}(x, x_0) = \left[ \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2} - \sqrt{2}\sigma \right]_+. \quad (24)$$

With this choice,  $\phi_{\mathbf{v}}(x, x_0) = 0$  if the distance  $\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}$  is smaller than  $\sqrt{2}\sigma$ . Under suitable conditions, in Theorem 3.3 we show that  $\phi_{\mathbf{v}}(x, x_0)$  satisfies

$$0 \leq \phi_{\mathbf{v}}(x, x_0) \leq \alpha_0 |f(x) - f(x_0)| + \delta_n, \quad (25)$$

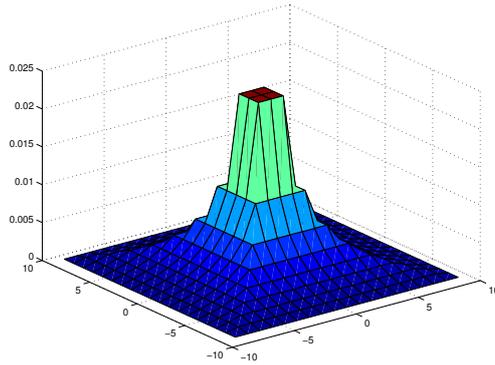


FIGURE 1 – Plot of the kernel  $\kappa_0$ .

where  $\alpha_0 > 0$  is a constant and  $\delta_n$  is a small error. A formal ground for using the approximation (24) is given by Theorem 3.2 in Section 3.1, where we show that when  $\phi(x, x_0)$  admits error-in measurements of type (25) the rate of convergence of the oracle estimator remains the same. In [51] the authors have proposed the following weight  $w(x, x_0) = \exp(-\max(\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}^2 - 2\sigma^2, 0)/h^2)$  which are similar to ours.

Using the approximation (24) we can now define the optimal weights filter (OWF) as follows :

$$\hat{f}(x_0) = \hat{f}_{\eta,h}(x_0) := \frac{\sum_{x \in U_h(x_0)} K_{\Delta} \left( \frac{\phi_{\mathbf{v}}(x, x_0)}{a(x_0)} \right) v(x)}{\sum_{y \in U_h(x_0)} K_{\Delta} \left( \frac{\phi_{\mathbf{v}}(y, x_0)}{a(x_0)} \right)}, \quad x \in U_h(x_0), \quad (26)$$

where  $a(x_0)$  is defined as the solution of the equation (17) with  $\phi(x, x_0) = \phi_{\mathbf{v}}(x, x_0)$ .

Obviously the patch size  $m = (2N\eta + 1)^2$  and the search window size  $M = (2Nh + 1)^2$  are the parameters of  $\hat{f}(x_0)$ . In the sequel, to compute  $\phi_{\mathbf{v}}(x, x_0)$ , we use the kernel  $K_0(\tau) = \kappa_0(\tau) / \sum_{\tau \in \mathcal{T}} \kappa_0(\tau)$  for  $\tau \in \mathcal{T}$ , with  $\kappa_0$  defined as follows :

$$\kappa_0(\tau) = \sum_{k=r_{\tau}}^r \frac{1}{(2k+1)^2}, \quad \tau \in \mathcal{T}, \quad (27)$$

where  $r_{\tau} = \max\{1, \|\tau\|_{\infty} N\}$  and  $r$  is such that  $m = (2r + 1)^2$  (see (7)). The kernel  $\kappa_0$  has the advantage that it decreases not too rapidly as  $\|\tau\|_{\infty}$  increases (see Figure 1), so that the contribution of the difference  $|\mathbf{v}(x_0 + \tau) - \mathbf{v}(x_0 - \tau)|$  in the distance  $\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{\kappa_0,2}$  also decreases slowly. This kernel was used in the original implementation (“Matlab” version) of the NL-means algorithm [7]. Note that there is no theory to select a particular kernel. Consequently, we evaluated several compact support kernels and it turns out that the proposed  $\kappa_0$  produced the best PSNR values.

We have seen that the weights in (26) are determined by the triangular kernel  $K_{\Delta}$  appearing in Theorem 2.1. Due to the form of  $\phi_{\mathbf{v}}$  in (24), as function of the distance  $d(x, x_0) = \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}$ , the weights in (26) can also be expressed using the trapezoidal kernel defined as  $K_T(d(x, x_0)) = K_{\Delta} \left( \frac{[d(x, x_0) - \sqrt{2}\sigma]_+}{a(x_0)} \right) = K_{\Delta} \left( \frac{\phi_{\mathbf{v}}(x, x_0)}{a(x_0)} \right)$ .

Another important issue in the non-local means filter is the selection of the global bandwidth parameter. This problem has already been addressed in [23], [27], [30], [34], [39]. In [40], the authors focused also on the original definition of NL-means but they proposed a SURE method to automatically select a spatially-varying smoothing parameter. In general, the starting point of all the aforementioned papers is the definition of NL-means with exponential weights.

In our model the bandwidth  $a(x_0) > 0$  is estimated by the solution  $\widehat{a}(x_0)$  of the equation

$$H_{\phi_{\mathbf{v}}}(\widehat{a}(x_0)) = \sum_{x \in U_h(x_0)} \phi_{\mathbf{v}}(x, x_0) [\widehat{a}(x_0) - \phi_{\mathbf{v}}(x, x_0)]_+ = \sigma^2. \quad (28)$$

As in [40], by adapting the smoothing parameter or bandwidth  $a(x_0)$  to the spatial location, the choice of the search window  $U_h(x_0)$  becomes not crucial. Actually, we set the size of  $U_h(x_0)$  to a large value in practice and the unique free parameter of the algorithm is the patch size  $m$ . A fast procedure for computing  $\widehat{a}(x_0)$  is presented in Section 2.3.1 where we replace  $\phi_{\mathbf{v}}(x, x_0)$  and  $\widehat{a}(x_0)$  by  $\phi(x, x_0)$  and  $a(x_0)$  respectively. The corresponding algorithm is given in Section 2.3.2.

## 2.3 Implementation of OWF and computational issues

In this section we propose an iterative procedure to estimate the spatially varying smoothing parameter  $a$  from (28) (see Subsection 2.3.1) and a practical algorithm to compute OWF (see Subsection 2.3.2).

### 2.3.1 Adaptive estimation of smoothing parameter

According to Theorem 2.1 we choose the bandwidth  $a(x_0)$  as the solution of the equation to (17). Below we propose a practical rule to detect the value of  $a(x_0)$ .

First, we sort the set  $\{\phi(x_i, x_0) \mid x_i \in U_h(x_0)\}$  in the ascending order, that is  $0 = \phi(x_0, x_0) \leq \phi(x_1, x_0) \leq \dots \leq \phi(x_{M-1}, x_0)$ , where  $M = \text{Card}(U_h(x_0))$ . Expression (17) can be then rewritten as

$$H_{\phi}(a(x_0)) := \sum_{i=0}^{M-1} \phi(x_i, x_0) [a(x_0) - \phi(x_i, x_0)]_+ = \sigma^2. \quad (29)$$

Let

$$a_j(x_0) = \frac{\sigma^2 + \sum_{i=0}^j \phi^2(x_i, x_0)}{\sum_{i=0}^j \phi(x_i, x_0)}, \quad j = 0, \dots, M-1,$$

with the convention  $a_0(x_0) = +\infty$ . Then we have :

**Theorem 2.2.** *The solution  $a(x_0) > 0$  of (29) can be expressed as  $a(x_0) = a_{j^*}(x_0)$  where  $j^*$  is the unique integer  $j \in \{0, \dots, M-1\}$  such that :*

$$j^* = \max\{0 \leq j \leq M-1 : a_j(x_0) \geq \phi(x_j, x_0)\}. \quad (30)$$

*Proof :* Denote  $\phi(x_M, x_0) = +\infty$ . Since function  $H_{\phi}(a(x_0))$  is strictly increasing on  $(0, +\infty)$  with  $H_{\phi}(0) = 0$  and  $H_{\phi}(+\infty) = +\infty$ , equation (29) admits a unique solution  $a(x_0)$  on  $(0, +\infty)$ , which must be located in some interval  $[\phi(x_{\bar{j}}, x_0), \phi(x_{\bar{j}+1}, x_0))$ , where  $0 \leq \bar{j} \leq M-1$ . Hence (29) becomes

$$\sum_{i=0}^{\bar{j}} \phi(x_i, x_0) (a(x_0) - \phi(x_i, x_0)) = \sigma^2, \quad (31)$$

where  $\phi(x_{\bar{j}}, x_0) \leq a(x_0) < \phi(x_{\bar{j}+1}, x_0)$ . From (31), it follows that

$$a(x_0) = \frac{\sigma^2 + \sum_{i=0}^{\bar{j}} \phi^2(x_i, x_0)}{\sum_{i=0}^{\bar{j}} \phi(x_i, x_0)}, \quad \phi(x_{\bar{j}}, x_0) \leq a(x_0) < \phi(x_{\bar{j}+1}, x_0). \quad (32)$$

We now show that  $\bar{j} = j^*$  with  $j^*$  defined by (30), so that  $a(x_0) = a_{\bar{j}}(x_0) = a_{j^*}(x_0)$ . Accordingly, it is sufficient to verify that  $a_{\bar{j}}(x_0) \geq \phi(x_{\bar{j}}, x_0)$  and  $a_j(x_0) < \phi(x_j, x_0)$  if  $\bar{j} < j \leq M - 1$ . We have already seen that  $a_{\bar{j}}(x_0) \geq \phi(x_{\bar{j}}, x_0)$ ; if  $\bar{j} < j \leq M - 1$ , then  $a_{\bar{j}}(x_0) < \phi(x_{\bar{j}+1}, x_0) \leq \phi(x_j, x_0)$ , so that

$$\begin{aligned} a_j(x_0) &= \frac{\sigma^2 + \sum_{i=0}^{\bar{j}} \phi^2(x_i, x_0) + \sum_{i=\bar{j}+1}^j \phi^2(x_i, x_0)}{\sum_{i=0}^j \phi(x_i, x_0)} = \frac{a_{\bar{j}}(x_0) \sum_{i=0}^{\bar{j}} \phi(x_i, x_0) + \sum_{i=\bar{j}+1}^j \phi^2(x_i, x_0)}{\sum_{i=0}^j \phi(x_i, x_0)} \\ &< \frac{\phi(x_j, x_0) \sum_{i=0}^{\bar{j}} \phi(x_i, x_0) + \sum_{i=\bar{j}+1}^j \phi(x_j, x_0) \phi(x_i, x_0)}{\sum_{i=0}^j \phi(x_i, x_0)} = \phi(x_j, x_0). \end{aligned}$$

We finally prove that if  $0 \leq j < M - 1$  and  $a_j(x_0) < \phi(x_j, x_0)$ , then  $a_{j+1}(x_0) < \phi(x_{j+1}, x_0)$ , so that  $j^* = \max_{0 \leq j \leq M-1} \{a_j(x_0) \geq \phi(x_j, x_0)\}$  is the unique integer  $j \in \{0, \dots, M - 1\}$  such that  $a_j(x_0) \geq \phi(x_j, x_0)$  and  $a_{j+1}(x_0) < \phi(x_{j+1}, x_0)$  if  $0 \leq j < M - 1$ . For  $0 \leq j < M - 1$ , the inequality  $a_j(x_0) < \phi(x_j, x_0)$  implies that

$$\sigma^2 + \sum_{i=0}^j \phi^2(x_i, x_0) < \phi(x_j, x_0) \sum_{i=0}^j \phi(x_i, x_0).$$

This, in turn, implies that

$$a_{j+1}(x_0) = \frac{\sigma^2 + \sum_{i=0}^j \phi^2(x_i, x_0) + \phi^2(x_{j+1}, x_0)}{\sum_{i=0}^{j+1} \phi(x_i, x_0)} < \frac{\phi(x_j, x_0) \sum_{i=0}^j \phi(x_i, x_0) + \phi^2(x_{j+1}, x_0)}{\sum_{i=0}^{j+1} \phi(x_i, x_0)} \leq \phi(x_{j+1}, x_0).$$

■

### 2.3.2 OWF algorithm for denoising

In this section, we give an algorithm for computing the estimator (26). The parameters of OWF are the variance  $\sigma^2$  of the white Gaussian noise, the patch size  $m$  and the search window size  $M$  respectively. The kernel  $K_0$  is taken to be  $\kappa_0$  defined by (27). In the definition of  $a_j(x_0)$  (see Algorithm), the denominator can be 0, in which case  $a_j(x_0) = +\infty$ . In practice, the maximal value of  $a_j(x_0)$  is set to a large value  $a_{\max}$  (e.g.  $a_{\max} = 10^5 \times f_{\max}$ ).

---

**Algorithm OWF** : Optimal Weights Estimator

---

**Initialization** : A noisy input image  $v$  with maximal brightness  $f_{\max}$  and the noise variance  $\sigma^2$ .

**Parameters** : Size  $M$  of spatial neighborhoods, size  $m$  of patches.

**for each pixel**  $x_0 \in \Omega$  **do**

1. **Estimation of adaptive bandwidth** :

– **for**  $i = 0$  **to**  $M - 1$  **do**

    compute  $\phi_{\mathbf{v}}(x_i, x_0) = [\|\mathbf{v}(x_i) - \mathbf{v}(x_0)\|_{K_{0,2}} - \sqrt{2}\sigma]_+$ .

**end for**

– Sort  $\{\phi_{\mathbf{v}}(x_i, x_0)\}$  such that  $\phi_{\mathbf{v}}(x_0, x_0) \leq \dots \leq \phi_{\mathbf{v}}(x_{M-1}, x_0)$ .

– **for**  $j = 0$  **to**  $M - 1$  **do**

    compute  $a_j(x_0) = \min \left\{ \frac{\sigma^2 + \sum_{i=0}^j \phi_{\mathbf{v}}^2(x_i, x_0)}{\sum_{i=0}^j \phi_{\mathbf{v}}(x_i, x_0)}, a_{\max} \right\}$ .

**end for**

– Select  $j^* = \max\{0 \leq j \leq M - 1 : a_j(x_0) \geq \phi(x_j, x_0)\}$ .

– Compute the optimal bandwidth :  $\hat{a}(x_0) = a_{j^*}(x_0)$ .

2. **Computation of the estimator** : 
$$\hat{f}(x_0) = \frac{\sum_{x_i \in U_h(x_0)} K_{\Delta} \left( \frac{\phi_{\mathbf{v}}(x_i, x_0)}{\hat{a}(x_0)} \right) v(x_i)}{\sum_{x_i \in U_h(x_0)} K_{\Delta} \left( \frac{\phi_{\mathbf{v}}(x_i, x_0)}{\hat{a}(x_0)} \right)}$$

**end for**

---

Our implementation in “C++” and “Matlab” of the algorithm can be found on the web site : <http://serpico.rennes.inria.fr/doku.php?id=software:owf>

In our experiments (4), we focused on real images corrupted by white Gaussian noise with a known variance  $\sigma^2$ . The numerical simulations show that our filter outperforms the classical NL-means filter under the same conditions. On real noisy images, the value  $\sigma^2$  is estimated according to the robust method presented in [23, 52] (see also [20]). Other methods could be used to estimate  $\sigma^2$  as described in [53, 54, 55]). Note that OWF individually processes each pixel and then can exploit a spatially-varying noise variance  $\sigma^2(x)$  (e.g. signal-dependent noise) estimated locally as described in [56, 38]. To complete the description of OWF, we give a justification of the oracle filter in the next section.

### 3 Properties of the oracle filter

#### 3.1 Main convergence properties and theoretical results

In this section, we show that the oracle estimator  $f_h^*(x_0)$  attains the optimal rate of convergence. Set  $\phi(x) = |f(x) - f(x_0)|$ . Let  $w(x, x_0)$  be the optimal weights given by (16) and  $a(x_0)$  be the

bandwidth given by (17). Recall that the oracle filter  $f_h^*(x_0)$  is defined by

$$f_h^*(x_0) = \sum_{x \in U_h(x_0)} w(x, x_0) v(x) \quad \text{with} \quad w(x, x_0) = \frac{K_\Delta \left( \frac{\phi(x, x_0)}{a(x_0)} \right)}{\sum_{y \in U_h(x_0)} K_\Delta \left( \frac{\phi(y, x_0)}{a(x_0)} \right)}, \quad x \in U_h(x_0). \quad (33)$$

First we prove the convergence of  $f_h^*(x_0)$ . This will be done under the condition that the image  $f$  satisfies the local Hölder condition :

$$|f(x) - f(y)| \leq L \|x - y\|_\infty^\beta, \quad x, y \in U_h(x_0), \quad (34)$$

where  $0 < \beta \leq 1$  is a constant,  $h > 0$ , and  $x_0 \in \Omega$ .

In the following  $c, c_i$  denote positive constants depending only on  $L, \beta$  and  $\sigma$ ; their values can be different from line to line.  $O(a_n)$  denotes a sequence bounded by  $ca_n$  for all  $n \geq 1$ .

**Theorem 3.1.** *Assume that  $f$  satisfies the local Hölder's condition (34). Let  $h \geq c_1 n^{-\alpha}$ , where  $0 \leq \alpha \leq \frac{1}{2\beta+2}$  and  $c_1 > 0$  if  $0 < \alpha < \frac{1}{2\beta+2}$ ,  $c_1 > c_0 = \left( \frac{\sigma^2(\beta+2)(2\beta+2)}{8L^2\beta} \right)^{\frac{1}{2\beta+2}}$  if  $\alpha = \frac{1}{2\beta+2}$ . Let  $f_h^*(x_0)$  be the oracle filter given by (33) with  $\phi(x, x_0) = |f(x) - f(x_0)|$ . Then*

$$\mathbb{E} (f_h^*(x_0) - f(x_0))^2 = O \left( n^{-\frac{2\beta}{2+2\beta}} \right). \quad (35)$$

*Proof :* See Appendix A.2. ■

The next result shows that the rate of convergence of the oracle filter  $f_h^*$  remains the same when instead of the precise relation  $\phi(x, x_0) = |f(x) - f(x_0)|$  we have the following bound

$$\phi(x, x_0) \leq \alpha |f(x) - f(x_0)| + \delta_n, \quad x \in U_h(x_0), \quad x_0 \in \Omega, \quad (36)$$

where  $\alpha > 0$  is a constant and  $\delta_n \geq 0$  is a small deterministic error.

**Theorem 3.2.** *Assume that  $f$  satisfies the local Hölder's condition (34). Let  $h \geq c_1 n^{-\alpha}$ , where  $0 \leq \alpha \leq \frac{1}{2\beta+2}$  and  $c_1 > 0$  if  $0 < \alpha < \frac{1}{2\beta+2}$ ,  $c_1 > c_0 = \left( \frac{\sigma^2(\beta+2)(2\beta+2)}{8L^2\beta} \right)^{\frac{1}{2\beta+2}}$  if  $\alpha = \frac{1}{2\beta+2}$ . Let  $f_h^*(x_0)$  be the oracle filter given by (33) with  $\phi(x, x_0)$  satisfying (36), and suppose that the error term in (36) satisfies  $\delta_n = O \left( n^{-\frac{\beta}{2+2\beta}} \right)$ . Then*

$$\mathbb{E} (f_h^*(x_0) - f(x_0))^2 = O \left( n^{-\frac{2\beta}{2+2\beta}} \right). \quad (37)$$

*Proof :* See Appendix A.2. ■

Now we present a result which shows that the choice  $\phi_v(x, x_0)$  satisfies (36) with high probability for  $\delta_n = O \left( n^{-\frac{\beta}{2+2\beta}} \right)$ , provided that the patch size  $\eta$  and the search window size  $h$  are properly chosen.

**Theorem 3.3.** *Assume that  $f$  satisfies the local Hölder's condition (34) and that  $h = c_1 n^{-\frac{1}{2\beta+2}}$  with  $c_1 > c_0 = \left( \frac{\sigma^2(\beta+2)(2\beta+2)}{8L^2\beta} \right)^{\frac{1}{2\beta+2}}$ . Assume also that  $\eta = c_2 n^{-\alpha}$  for some  $\frac{1}{2(\beta+1)^2} < \alpha < \frac{1}{2\beta+2}$  and  $c_2 > 0$ . Then, there are constants  $\alpha_0 > 0$  and  $c_3 > 0$ , such that*

$$\lim_{n \rightarrow \infty} \max_{x_0 \in \Omega} \max_{x \in U_h(x_0)} \mathbb{P} \left\{ \phi_v(x, x_0) \leq \alpha_0 |f(x) - f(x_0)| + c_3 n^{-\frac{\beta}{2+2\beta}} \right\} = 1. \quad (38)$$

The proof will be given in the Appendix A.3. We mention that the summands in the norm  $\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_{0,2}}$  in the definition (24) of  $\phi_{\mathbf{v}}(x, x_0)$  are not necessarily independent when the windows  $V(x)$  and  $V(x_0)$  overlap, as it will be seen from the proof.

Theorem 3.1 ensures that the oracle filter converges at the optimal rate when the radius of the search window  $h$  is chosen appropriately. Indeed, when  $h$  is of the usual optimal order  $n^{-\frac{1}{2\beta+2}}$ , the convergence of the oracle filter rate is  $n^{-\frac{2\beta}{2\beta+2}}$  which is the usual optimal convergence rate for a given Hölder smoothness  $\beta$  (see for example [31]). When the radius  $h$  is chosen larger, for instance of order  $n^{-\alpha}$  with  $0 \leq \alpha < \frac{1}{2\beta+2}$ , we still get the optimal rate of convergence of the oracle filter. This is explained by the fact that the order of the bandwidth  $a(x_0)$  in both cases remains the same, as shown in (35) and (37)

Theorem 3.1 also shows that the oracle filter  $f_h^*$  has an adaptivity property. Actually, at each pixel  $x_0$ , the filter adapts to the best rate of convergence. To better understand this, assume that the image  $f$  has Hölder smoothness  $\beta = \beta(x_0)$  at  $x_0$  and that  $h \geq c_0 n^{-\alpha}$  with  $0 \leq \alpha < \frac{1}{2\beta+2}$ , which means that the radius  $h > 0$  of the search window  $U_h(x_0)$  has been chosen larger than the standard  $n^{-\frac{1}{2\beta+2}}$ . In particular, the larger  $\beta$  is, the better the rate of convergence, which ensures good estimation of the flat regions. Since Theorem 3.2 is valid for arbitrary  $\beta$ , it applies for the maximal local Hölder smoothness  $\beta(x_0)$  at  $x_0$ , the oracle  $f_h^*(x_0)$  will exhibit the best rate of convergence of order  $n^{-\frac{\beta(x_0)}{2+2\beta(x_0)}}$  at  $x_0$ . Therefore, the procedure adapts to the best rate of convergence at each point  $x_0$  of the image.

We justify by simulation results that the difference between the oracle  $f_h^*$  computed with  $\phi_f(x, x_0) = |f(x) - f(x_0)|$ , and the true image  $f$ , is extremely small (see Table 1). This shows that, at least from the practical point of view, it is justified to optimize the upper bound  $\mathcal{J}_\phi(w)$  instead of optimizing the pointwise  $L_2$  risk (or Mean Square Error)  $\mathbb{E}(f_h^*(x_0) - f(x_0))^2$  itself.

From Theorem 3.3 it follows that the local Hölder's condition (34) implies the regularity of the function  $\phi(x, x_0)$  in  $x$ . Indeed, Theorem 3.3 states that, under appropriate assumptions,  $\phi_{\mathbf{v}}(x, x_0)$  has the same decay rate as  $|f(x) - f(x_0)|$  with high probability. We mention that another type of condition related to patch regularity in the patch space has been considered in [40].

Below, we mention the differences between our approach and the theory of adaptive choice of the bandwidth in nonparametric estimation. The well known standard minimax results for nonparametric kernel estimators of functions under Hölder's condition have been established when the weights  $w(x_i, x_0) = K(\frac{x_i - x_0}{a})$  depend on the spatial position of the pixels  $x_i$  and  $x_0$  (but not on the values  $f(x_i)$  and  $f(x_0)$  of the function  $f$ ). In the NL-means setting the weights  $w(x_i, x_0) = K(\frac{f(x_i) - f(x_0)}{a})$  depend on the unknown function  $f$  so that the situation is quite different from that in the nonparametric estimation. The usual theory of adaptive choice of the bandwidth also does not apply here since the corresponding results have not been established for the situation that we encounter here. In particular the adaptive theory of Lepski [57, 58] does not apply in this case. In practice it may encounter some difficulties related to the fact that the method is based on consecutive testing of a sequence of estimators  $\hat{f}_{a_1}, \dots, \hat{f}_{a_m}$  with increasing bandwidths  $a_1 < \dots < a_m$ , i.e. it selects the bandwidth  $a_j$  when  $|\hat{f}_{a_j} - \hat{f}_{a_i}|$  does not exceed a certain critical value for all  $i < j$  for the last time  $j$ . The practical difficulty is that this procedure select  $a_j$  too late since the estimator  $\hat{f}_{a_j}$  has already accumulated some bias to attain the critical value.

Actually, our method is very different for the following reasons : we first compute the oracle bandwidth  $a(x_0)$  and the oracle weights by Theorem 2.1 which depends on the unknown function  $f$  (based on Theorem 2.2), then we plug-in an estimator instead of the unknown function to compute the adaptive bandwidth. To the best of our knowledge our result is the first theoretical work on

the minimax properties for  $f_h^*(x_0)$ . The plug-in method for the choice of the bandwidth that we propose here is also new in the image denoising.

### 3.2 Numerical performance of the oracle filter

In this section, given the true image  $f$ , we evaluated the performance of the oracle estimator (18) defined in Section 2.2 where  $\phi(x, x_0) := |f(x) - f(x_0)|$ . The computation of the oracle estimator is very fast and is controlled by a unique free parameter, that is the neighborhood size  $M$ . The oracle bandwidth  $a(x_0)$  is computed as described in Section 2.3.1. by setting  $\phi(x, x_0) := |f(x) - f(x_0)|$ .

Each original image was corrupted with white Gaussian noise (WGN) with zero mean and variance  $\sigma^2$ . The performance of OWF is measured by the Peak Signal-to-Noise Ratio (PSNR) in decibels (db) :

$$PSNR = 10 \log_{10} \frac{255^2}{MSE}, \quad \text{with} \quad MSE = \frac{1}{\text{card}(\Omega)} \sum_{x \in \Omega} (f(x) - \hat{f}(x))^2,$$

where  $f$  is the original image, and  $\hat{f}$  the estimator. In Table 1, we observe first that the performance (PSNR values) of the oracle estimator increases with  $M$  for all the tested images and for any  $\sigma$  values. We also computed two oracle estimators based on the usual Gaussian and the triangle kernels (see Theorem 2.1) and controlled by a unique bandwidth  $b$ . In our experiments, we exploited the set of oracle bandwidth values  $\{a(x)\}$  obtained previously to set the global bandwidth  $b$  as follows :  $b = |\Omega|^{-1} \sum_{x \in \Omega} a(x)$ . The average of oracle bandwidths, depending on  $\sigma$  and  $M$ , appeared to be a satisfying choice as confirmed by the experimental results. Finally, the two kernels have been scaled to get a fair comparison presented in Table 2. It turns out that the adaptive oracle filter provides the best PSNR values in all cases. In the case of non-adaptive filters, the Gaussian kernel gave PSNR results slightly superior (0.1 to 0.2 db) to the ones obtained with the triangular kernel.

## 4 Simulations and experimental results

We evaluated OWF on 25 natural images showing natural, man-made, indoor and outdoor scenes (see Figure 2). To avoid the undesirable border effects in our simulations, we mirror the image outside the image limits, that is we extend the image outside the image limits symmetrically with respect to the border. At the corners, the image is extended symmetrically with respect to the corner pixels. In our procedure (see Section 2.3.2), the central patch does not get a particular status and is processed as any patch unlike the usual NL-means methods. This is possible thanks to the triangle kernel where the positive part of  $a(x_0) - \phi(x, x_0)$  saturates the lower values of  $\phi(x, x_0)$ . For similar conclusions, we refer to [59, 60]. This is a clear advantage of OWF. In the experiments, we found that the parameters  $m$  and  $M$  can be fixed to  $m = 27 \times 27$  and  $M = 13 \times 13$  to get the highest PSNR values (smaller patch sizes  $m$  can be considered for processing piecewise smooth images). This means that the method provides better results for a high rate of overlapping patches. In [7, 23], the authors recommended to use small patches and large search windows :  $m < M$  :  $m = 7 \times 7$  and  $M = 21 \times 21$ .

### 4.1 Analysis of algorithm results

In this section, the potential of OWF is mainly illustrated with the  $512 \times 512$  ‘‘lena’’ and ‘‘barbara’’ images. First, in Figure 3(c), we can see that the noise is reduced in a natural manner and significant



FIGURE 2 – Plot Set of 25 tested images. Top left : images from the websites (<http://www.cs.tut.fi/~foi/GCF-BM3D/>); Bottom left : images from IPOL (<http://www.ipol.im>); Right : images from the Berkeley segmentation database (<https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>).

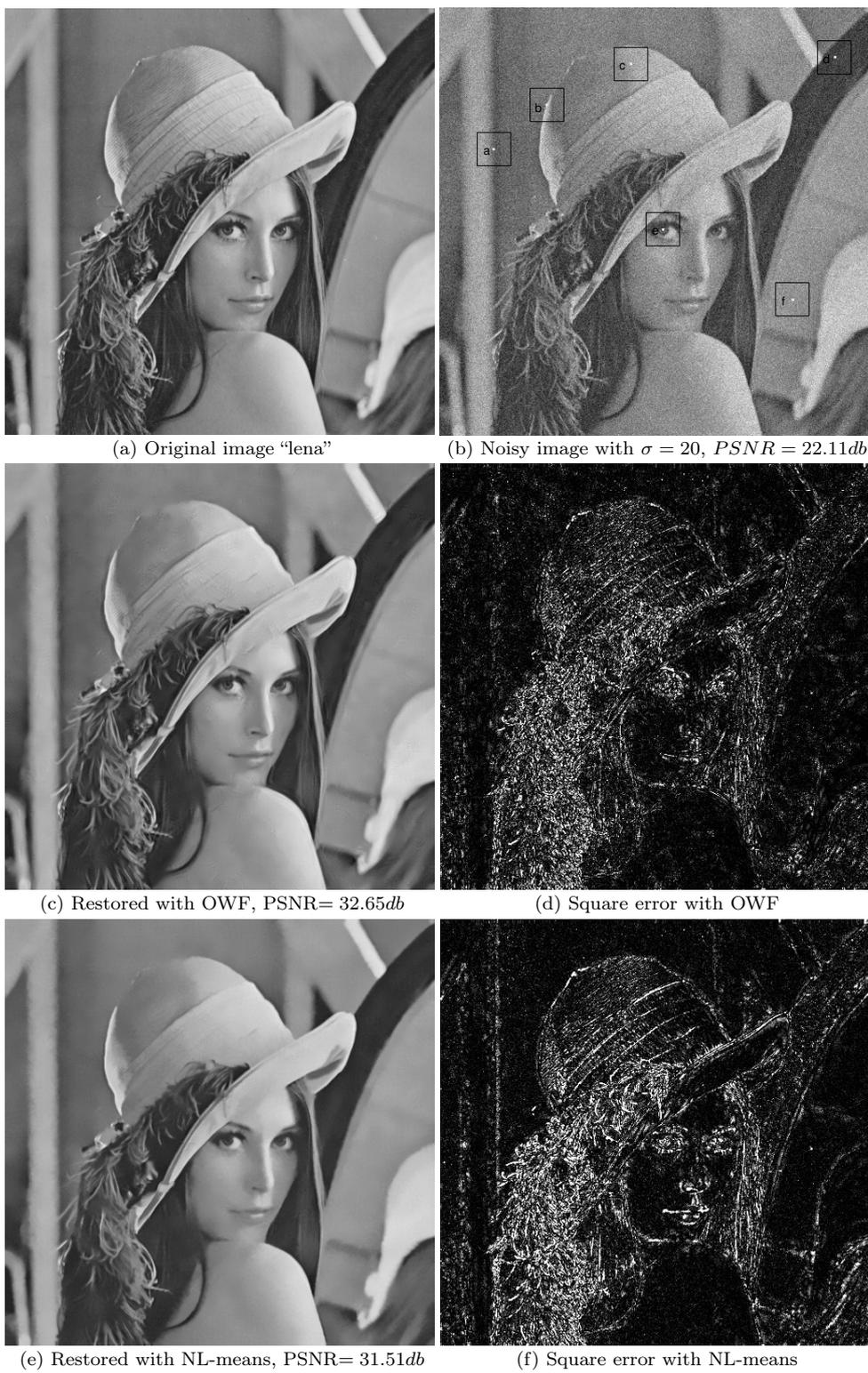


FIGURE 3 – Results of denoising “lena”  $512 \times 512$  image. The noisy “lena” image ( $\sigma = 20$ ) with six selected search windows centered at pixels a, b, c, d, e, f is shown in (b). Comparing (d) and (f) we see that the Optimal Weights Filter (OWF) captures more details than NL-means [7].

geometric features, fine textures, details and original contrasts are visually well recovered with no undesirable artifacts (PSNR = 32.65db, “lena” image). To better appreciate the accuracy of the restoration process, the square of the difference between the original image and the recovered image is shown in Figure 3(d), where the dark values correspond to a high-confidence estimate. As expected, pixels with a low level of confidence are located in the neighborhood of image discontinuities. For comparison, we present the image denoised by NL-means [7] in Figures 3(e) and (f), which show that the overall visual impression and the numerical results are improved using our algorithm.

OWF provides a feasible and rational method to detect automatically the details of images and take the proper weights for every possible geometric configuration of the image. For illustration purposes, we have chosen a series of search windows  $U_h(x_0)$  with centers at some testing pixels  $x_0$  on the noisy image (see Figure 3(b)). The distribution of the weights inside the search window  $U_h(x_0)$  depends on the estimated brightness variation function  $\phi_v(x, x_0), x \in U_h(x_0)$ . If the estimated brightness variation  $\phi_v(x, x_0)$  is less than  $\hat{a}(x_0)$ , the similarity between pixels is measured by a linear decreasing function of  $\phi_v(x, x_0)$ , otherwise it is zero. Thus  $\hat{a}(x_0)$  acts as an automatic threshold. In Figure 4, it is shown how OWF selects a proper weight configuration in each case. On the second row of Figure 4, we observe a ringing effect produced by OWF but this artifact is not so visible at other locations in Figure 4 and on images presented in Figs. 7-9. For comparison, we display in Figure 5 the corresponding weights for the standard NL-means filter. From Figs. 4-5, we can see that the distribution of the weights are rather different for the two filters.

In Figure 6, the distribution of bandwidth values is displayed for the “lena” and “barbara” images corrupted by WGN ( $\sigma = 5$  and  $\sigma = 20$ ). The values of  $a(x_0)$  are low in flat zones and higher along image discontinuities. In homogeneous regions, small values are expected since all local patches are similar and the neighborhood contains many repeated patterns. The histograms of bandwidth values  $\{a(x_0)\}$  vary a lot from one image to another and depend on signal-to-noise ratios.

Finally, to demonstrate the performance of adaptive bandwidth, we compared the PSNR values with those obtained with the OWF algorithm by imposing a constant bandwidth  $a = \sigma^2/A$  with  $A = \{2., 2.5, 5., 7.5, 10., 25., 30., 35., 50., 60., 100.\}$ . The bandwidth depends here on the noise variance as generally recommended in the literature. From the results in Table 3 and Table 4, we found an optimal value  $A$  corresponding to a maximum PSNR value for each tested image and each tested noise variance. Clearly, this value is not the same for the nine tested images and the maximum PSNR values is generally lower (about 0.7 db in average) than the the PSNR value obtained with the (adaptive) OWF. In addition the optimal value  $A$  is not the same from one  $\sigma^2$  value to another. This experiment shows clearly that OWF unequivocally outperforms the non-adaptive OWF which needs a specific adjustment of the bandwidth depending on the noise variance.

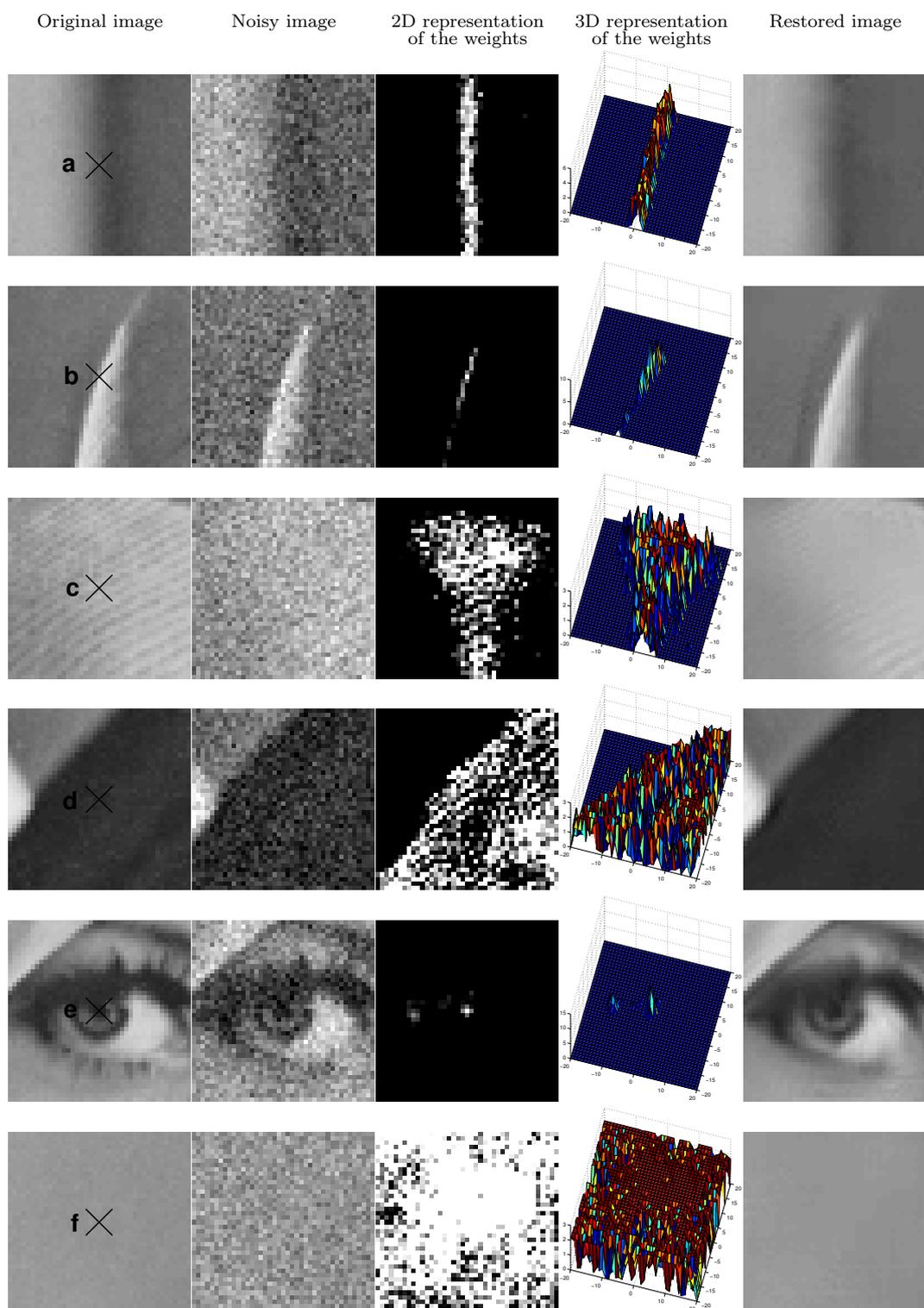


FIGURE 4 – These pictures show how OWF detects the features of the image by choosing appropriate weights. The first column displays six selected search windows used to estimate the image at the corresponding central pixels  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $e$  and  $f$  (see Figure 3(b)). The second column displays the corresponding search windows corrupted by a Gaussian noise with standard deviation  $\sigma = 20$ . The third column displays the two-dimensional representation of the weights used to estimate central pixels. The fourth column gives the three-dimensional representation of the weights. The fifth column gives the restored images.

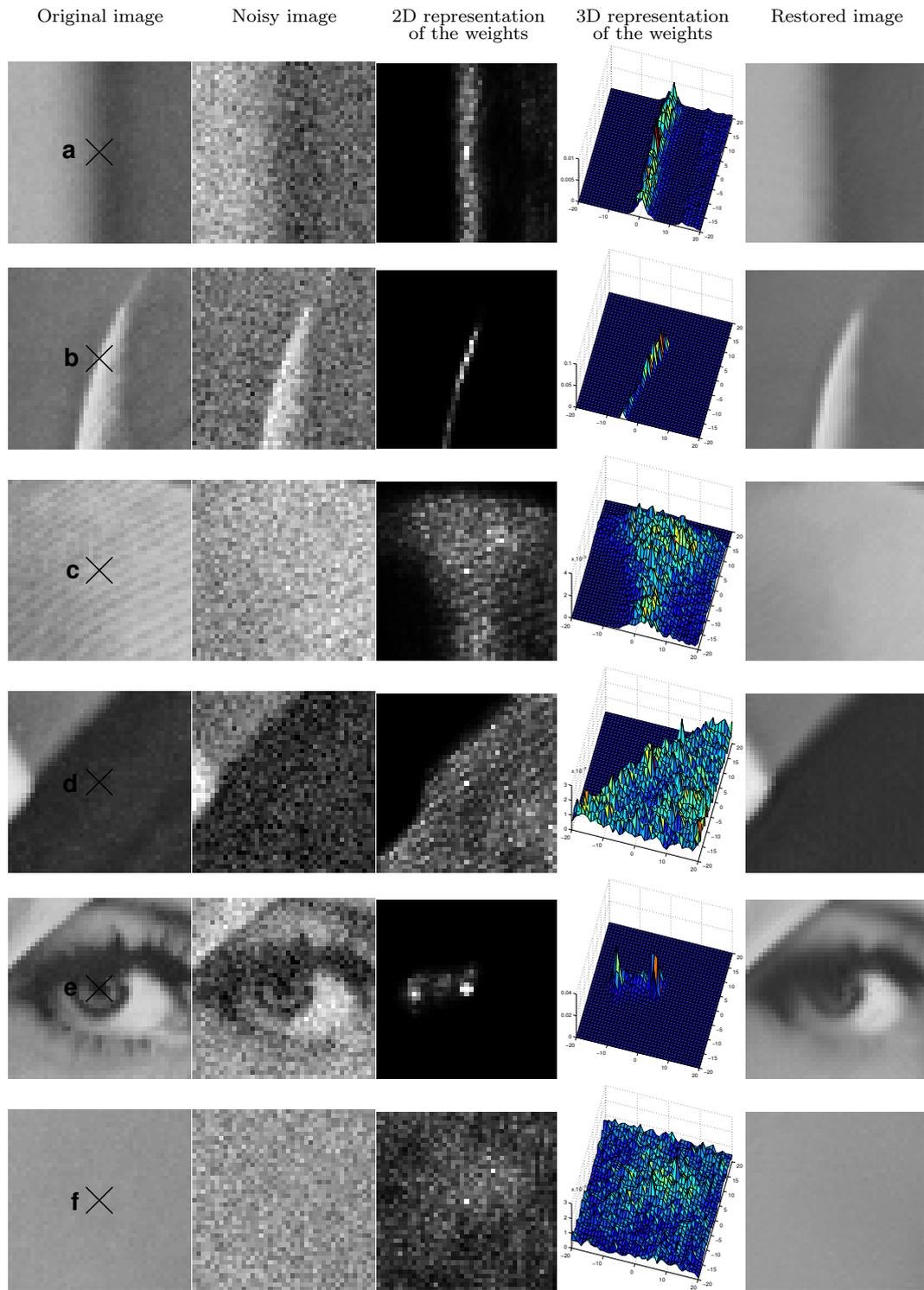


FIGURE 5 – These pictures show how the Non-Local Means detects the features of the image by choosing appropriate weights. The first column displays six selected search windows used to estimate the image at the corresponding central pixels a, b, c, d, e and f (see Figure 3(b)). The second column displays the corresponding search windows corrupted by a Gaussian noise with standard deviation  $\sigma = 20$ . The third column displays the two-dimensional representation of the weights used to estimate central pixels. The fourth column gives the three-dimensional representation of the weights. The fifth column gives the restored images.

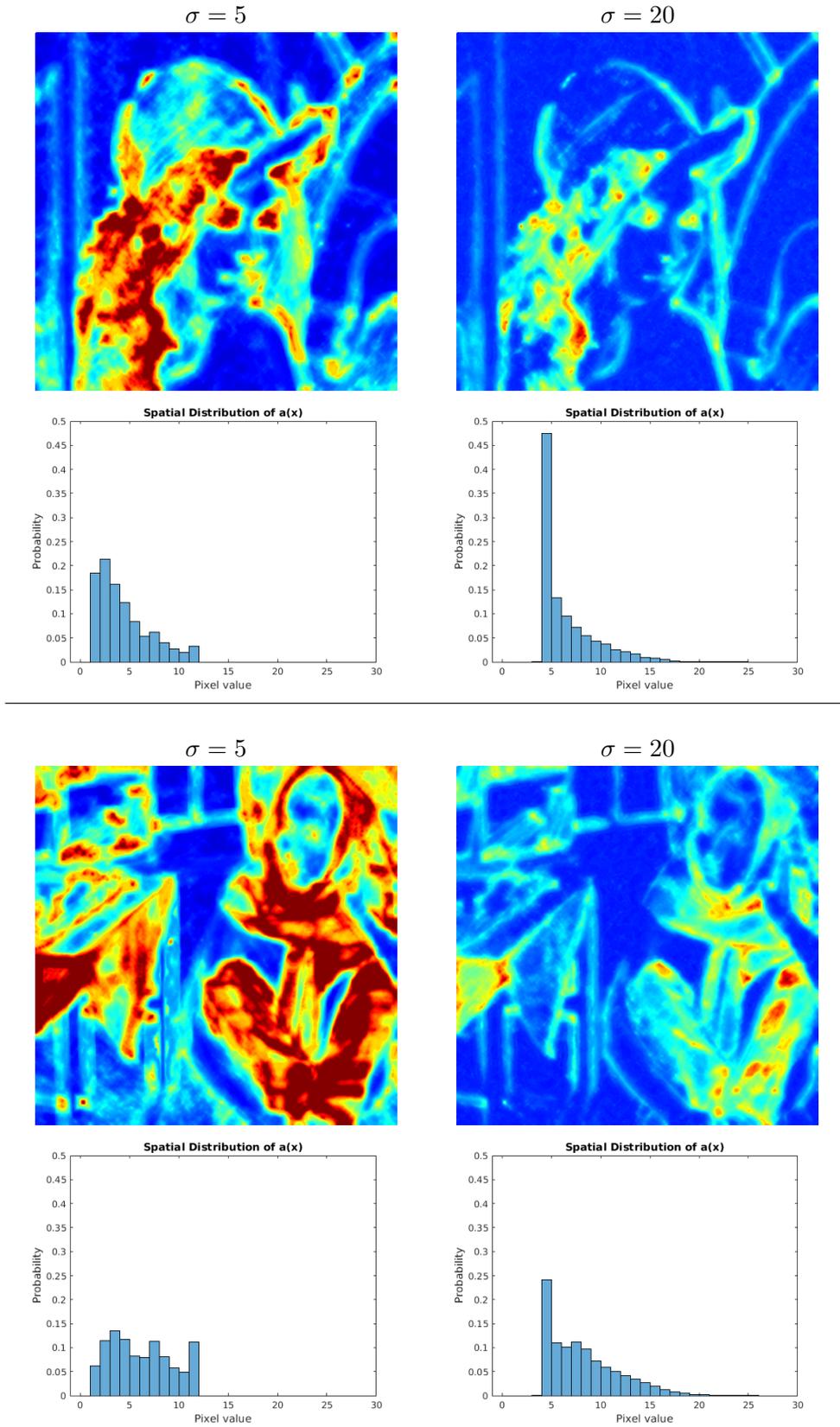


FIGURE 6 – Images and histograms of bandwidth values for two images (“lena” (two first rows), “barbara” (two last rows)) corrupted by white Gaussian noise ( $\sigma = 5$  and  $\sigma = 20$ ). The “hot” colors correspond to large values of  $a(x)$  and “cold” colors to small values of  $a(x)$ .

TABLE 1 – Denoising results using the oracle estimator (18) with  $\phi(x, x_0) = |f(x) - f(x_0)|$  on nine tested images. The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations.

$M$	$\sigma = 5$			$\sigma = 15$			$\sigma = 25$			$\sigma = 50$		
	13×13	25×25	49×49	13×13	25×25	49×49	13×13	25×25	49×49	13×13	25×25	49×49
barbara	45.01	47.86	50.82	38.50	41.35	44.30	35.47	38.34	41.30	31.27	34.25	37.24
cameraman	45.85	48.65	51.41	39.29	42.11	44.84	36.24	39.09	41.81	31.96	35.02	37.81
lena	46.17	49.01	51.92	39.70	42.55	45.39	36.67	39.58	42.42	32.42	35.48	38.37
girl	50.03	52.93	55.62	43.16	46.28	48.89	39.76	43.18	45.81	34.79	38.79	41.75
traffic	45.49	48.36	51.34	38.87	41.86	44.79	35.76	38.81	41.80	31.45	34.66	37.73
valldemossa	44.51	47.56	50.73	37.94	41.10	44.20	34.86	38.13	41.23	30.51	34.03	37.22
aircraft	50.53	53.71	56.84	43.81	47.98	50.26	40.62	43.97	47.25	36.22	39.67	43.07
castle	46.01	38.95	51.98	39.42	42.42	45.44	36.31	39.39	42.46	31.98	35.22	38.40
maya	46.75	50.03	53.42	40.29	43.58	46.98	37.22	40.54	43.95	32.97	36.33	39.76

TABLE 2 – Table 2 : Comparison (PSNR values) between the “oracle” filter with spatial adaptive kernel ( $A-K_\Delta$ ) and with non-adaptive triangular ( $K_\Delta$ ) and Gaussian ( $K_G$ ) kernels for different values of search window sizes ( $M \times M$ ) on several images corrupted by white Gaussian noise ( $\sigma = 20$ ).

		11 × 11	13 × 13	15 × 15	17 × 17	19 × 19	21 × 21
barbara	OWF	36.64 db	37.46 db	38.17 db	38.79 db	39.32 db	39.79 db
	$K_\Delta$	36.35 db	37.16 db	37.86 db	38.44 db	38.95 db	39.40 db
	$K_G$	36.47 db	37.28 db	37.98 db	38.56 db	39.07 db	39.53 db
boat	OWF	36.74 db	37.55 db	38.22 db	38.82 db	39.36 db	39.84 db
	$K_\Delta$	36.31 db	37.12 db	37.80 db	38.37 db	38.87 db	39.33 db
	$K_G$	36.39 db	37.19 db	37.88 db	38.46 db	38.97 db	39.43 db
house	OWF	37.38 db	38.16 db	38.82 db	39.37 db	39.84 db	39.94 db
	$K_\Delta$	37.15 db	37.89 db	38.51 db	38.97 db	38.46 db	39.80 db
	$K_G$	37.29 db	37.99 db	38.61 db	39.12 db	39.60 db	39.91 db
lena	OWF	37.73 db	38.52 db	39.20 db	39.78 db	40.28 db	40.73 db
	$K_\Delta$	37.27 db	38.06 db	38.73 db	39.32 db	39.82 db	40.28 db
	$K_G$	37.37 db	38.17 db	38.83 db	39.41 db	39.92 db	40.39 db
peppers	OWF	36.54 db	37.28 db	37.83 db	38.33 db	38.75 db	39.14 db
	$K_\Delta$	36.39 db	37.04 db	37.62 db	38.10 db	38.54 db	38.91 db
	$K_G$	36.49 db	37.14 db	37.71 db	38.19 db	38.63 db	38.99 db
<b>Average</b>	OWF	37.00 db	37.79 db	38.45 db	39.02 db	39.51 db	39.89 db
	$K_\Delta$	36.69 db	37.45 db	38.10 db	38.64 db	38.93 db	39.34 db
	$K_G$	36.80 db	37.55 db	38.20 db	38.75 db	39.24 db	39.65 db

## 4.2 Comparison with state-of-the-art algorithms

The results on the 25 images (Figure 2) demonstrate that OWF is competitive when compared to state-of-the-art algorithms and significantly improves the NL-means results. In Table 5, we provide the PSNR values for all the tested images and different signal-to-noise ratios.

Tables 6-8 compares the PSNR values on these 25 images obtained by OWF and three more recent state-of-the-art denoising methods [11, 13, 14]; PEWA [14] can be viewed as a generalization of NL-means. Unlike [11, 13, 14], OWF is a pointwise approach, does not include an aggregation procedure (central projection [60]) and only one stage is required with OWF. In our experiments, we used the implementations provided by the authors: BM3D (<http://www.cs.tut.fi/~foi/GCF-BM3D/>) and NL-Bayes (<http://www.ipol.im>). The best PSNR values are in bold. We also compared OWF to the baseline NL-means [7] and DCT [2] (using the implementation of IPOL – <http://www.ipol.im>). Finally, we compared the results to the S-PLF method which uses SURE to guide the probabilistic patch-based filtering described in [16]. From Tables 6-8, the performance of OWF, computed from the 25 test images, is close to S-PLF [16]. For very high  $\sigma$  values ( $\sigma \geq 50$ ), OWF produces results similar to those obtained with NL-means [7] and DCT [2]. Figures (7)-(9) provide a visual compari-

TABLE 3 – Performance (PSNR) of OWF (adaptive bandwidth) and the OWF algorithm with a constant bandwidth  $a$  when applied to nine selected images corrupted by white Gaussian noise ( $\sigma = 5$ ). We give the results for several values of bandwidth  $a = \sigma^2/A$  where  $A \in \{2.0, 2.5, 5.0, 7.5, 10, 25, 30\}$  and  $m = 27 \times 27, M = 13 \times 13$ . The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations. We denote  $K_{\Delta,A}$  the OWF with constant bandwidth  $a$  and the best PSNR values are in bold.

$\sigma = 5$	OWF	$K_{\Delta,2.0}$	$K_{\Delta,2.5}$	$K_{\Delta,5.0}$	$K_{\Delta,7.5}$	$K_{\Delta,10}$	$K_{\Delta,25}$	$K_{\Delta,30}$
barbara	37.34	36.02	<b>36.49</b>	36.31	35.97	35.68	34.81	34.70
cameraman	37.17	36.14	<b>36.54</b>	36.40	36.24	36.13	35.75	35.64
lena	38.35	36.34	<b>36.86</b>	<b>37.50</b>	37.25	36.86	35.52	35.29
girl	43.72	40.15	41.00	43.01	<b>43.28</b>	42.88	40.42	40.04
traffic	37.10	35.37	35.99	<b>36.20</b>	35.97	35.81	35.25	35.15
valldemossa	36.22	35.57	<b>35.74</b>	35.64	35.51	35.43	35.20	35.17
aircraft	37.01	35.00	35.49	<b>35.85</b>	35.74	35.55	34.51	34.42
castle	37.29	35.60	36.11	<b>36.70</b>	36.58	36.31	35.80	35.71
maya	34.29	34.22	<b>34.28</b>	34.18	34.16	34.16	34.16	34.16
<b>Average</b>	37.61	36.04	36.50	<b>36.87</b>	36.74	36.53	35.71	35.60

TABLE 4 – Performance (PSNR) of OWF (adaptive bandwidth) and the basic OWF algorithm with a constant bandwidth  $a$  when applied to nine selected images corrupted by white Gaussian noise ( $\sigma = 25$ ). We give the results for several values of bandwidth  $a = \sigma^2/A$  where  $A \in \{10, 25, 30, 35, 50, 60, 100\}$  and  $m = 27 \times 27, M = 13 \times 13$ . The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations. We denote  $K_{\Delta,A}$  the OWF with constant bandwidth  $a$  and the best PSNR values are in bold.

$\sigma = 25$	OWF	$K_{\Delta,10}$	$K_{\Delta,25}$	$K_{\Delta,30}$	$K_{\Delta,35}$	$K_{\Delta,50}$	$K_{\Delta,60}$	$K_{\Delta,100}$
barbara	29.96	23.42	27.45	28.21	28.75	<b>29.36</b>	29.23	27.53
cameraman	27.92	23.75	27.93	<b>27.99</b>	27.92	27.40	26.94	25.53
lena	31.57	26.40	29.43	30.08	30.56	31.29	<b>31.37</b>	30.26
girl	35.73	33.63	34.84	35.12	35.36	35.85	<b>36.02</b>	35.87
traffic	28.10	23.23	25.98	26.53	26.93	<b>27.36</b>	27.15	25.55
valldemossa	26.14	20.83	24.65	25.08	<b>25.24</b>	24.83	24.34	23.17
aircraft	30.29	25.69	28.11	28.64	29.02	29.58	<b>29.62</b>	28.60
castle	28.76	23.74	27.34	27.62	<b>27.67</b>	27.35	27.08	26.21
maya	24.22	18.43	23.33	<b>23.66</b>	23.52	22.48	21.93	20.99
<b>Average</b>	29.19	24.35	27.67	28.10	28.33	<b>28.39</b>	28.19	27.08

son of the different methods applied to the “aircraft” (WGN,  $\sigma = 5$ ), “valldemossa” (WGN,  $\sigma = 20$ ) and “traffic” (WGN,  $\sigma = 15$ ) images. We observe that the results in detail preservation and smoothing are improved with OWF when compared to NL-means. We also observe very small differences between OWF and S-PLF (confirmed by PSNR values in Table 8). It is worth noting that OWF does not introduce mottling effect in the restored images unlike most sophisticated patch-based methods.

Finally, we visually compared the results obtained by the most related methods [7, 23, 34, 16]. In Figures 10-11, OWF gives similar results to the NL-means (<http://www.ipol.im>) if we consider the usual parameters :  $m = 7 \times 7$  patches and  $M = 21 \times 21$ . If we set  $m = 27 \times 27$  patches and  $M = 13 \times 13$ , for  $\sigma \leq 20$ , OWF outperforms the NL-means and the performance is actually very close and sometimes higher than [23] (SAFIR), [34] (TV-means) and [16] (S-PLF, <http://www.ipol.im>). In Figure 10, the PSNR values are the same with OWF and the considered algorithms but our results appear more natural visually. Note that we did not observe high visual patch adherence effects or jittering effects on the images denoised with OWF (a small ringing effect is only visible in Figure 4) as it is the case with variants of NL-means filters. The jittering effect can be reduced by applying adaptive regularizing nonlocal methods [38]. Post-processing the images denoised with OWF seems not useful at first glance from our experiments.

TABLE 5 – Denoising results using OWF ( $m = 27 \times 27$ ,  $M = 13 \times 13$ ) on the 25 tested images for several values of  $\sigma$ . The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations.

	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 50$
barbara	37.34	34.04	32.34	31.01	29.96	26.05
boat	36.73	33.44	31.45	30.27	29.20	25.74
cameraman	37.17	33.11	31.05	29.69	28.66	25.57
couple	36.97	33.46	31.54	30.09	28.93	25.20
hill	36.80	33.37	31.49	30.18	29.20	26.00
house	38.71	35.77	34.01	32.90	31.73	27.71
lena	38.35	35.59	33.95	32.65	31.57	27.76
man	37.31	33.67	31.65	30.26	29.17	25.92
peppers	37.11	33.74	31.89	30.55	29.45	25.64
Average	37.39	34.02	32.15	30.84	29.76	26.17
alley	35.84	31.67	29.60	28.25	27.21	23.78
computer	37.93	34.26	32.23	30.77	29.60	25.79
dice	45.99	42.73	40.48	38.70	37.21	32.12
flowers	42.45	38.91	36.62	34.93	33.63	29.64
girl	43.72	40.66	38.60	37.01	35.73	31.60
traffic	37.10	32.91	30.70	29.22	28.10	24.86
trees	34.80	29.81	27.33	25.78	24.70	21.87
valldemossa	36.22	31.40	28.89	27.28	26.14	22.91
Average	39.25	35.29	33.05	31.39	30.29	26.57
aircraft	37.01	34.14	32.54	31.32	30.29	26.90
asia	37.77	33.95	31.85	30.38	29.25	26.01
castle	37.29	33.26	31.27	29.87	28.76	25.44
man picture	36.17	32.35	30.25	28.76	27.61	23.77
maya	34.29	29.27	27.00	25.41	24.22	20.86
panther	37.74	33.49	31.38	29.86	28.70	25.24
tiger	36.41	32.57	30.45	28.95	27.78	24.23
young man	39.10	35.96	34.37	33.16	32.16	28.41
Average	36.97	33.12	31.14	29.71	28.59	25.10
<b>Average</b>	<b>37.85</b>	<b>34.14</b>	<b>32.11</b>	<b>30.65</b>	<b>29.55</b>	<b>25.96</b>

### 4.3 Computings and algorithm implementation

In terms of computational complexity, denoising a  $512 \times 512$  grayscale image with an unoptimized implementation of our method in C++ take about 50s (Intel Core i7 64-bit CPU 2.4 Ghz). The implementation of OWF is straightforward compared to more sophisticated algorithms developed in recent years. Nevertheless, the values  $\{\phi_v(x, x_0)\}$  in the search window centered at pixel  $x_0$  are sorted (from the smallest one to the largest one). This step is mandatory in OWF and takes time to compute. A faster sort algorithm can be potentially used to speed OWF and to get timings closer to those obtained with BM3D [11]. Moreover, we did not evaluate all the possible implementations as described for instance in [61, 62]. Note that OWF has been recently implemented in parallel since every patch can be processed independently. The computational times become a few seconds in that case<sup>1</sup>.

Additionally, we have also evaluated the performance of the algorithm with a block implementation. The block (or patchwise) implementation is very popular in patch-based approaches and our implementation is similar to the description given in Section 5.5.2 in [7] (see also [51, 52]). The idea consists in computing a block/patch estimator given an estimation of the bandwidth attached to

---

1. Supporting software and an online demo will be submitted to IPOL - Image Processing On Line - website (<http://www.ipol.im>)

TABLE 6 – Performance (PSNR) of denoising algorithms when applied to the 25 noisy (WGN) images ( $\sigma = 5$ ). The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations. The best PSNR values are in bold.

$\sigma = 5$	OWF	PEWA [14]	BM3D [11]	NL-BAYES [13]	S-PLF [16]	NL-MEANS [7]	DCT [2]
barbara	37.34	38.09	38.31	<b>38.38</b>	37.46	37.08	37.98
boat	36.73	37.12	<b>37.28</b>	<b>37.26</b>	36.49	36.35	36.63
cameraman	37.17	38.20	38.29	<b>38.35</b>	37.68	37.04	37.24
couple	36.97	37.35	<b>37.52</b>	37.40	37.21	36.05	36.75
hill	36.80	37.01	<b>37.13</b>	37.04	36.83	36.56	36.38
house	38.71	39.56	<b>39.83</b>	39.58	38.97	38.64	38.92
lena	38.35	38.57	<b>38.72</b>	<b>38.72</b>	38.28	36.67	38.17
man	37.31	37.68	<b>37.82</b>	<b>37.86</b>	37.46	36.81	36.80
peppers	37.11	38.00	<b>38.12</b>	<b>38.10</b>	37.64	36.02	36.80
alley	35.84	36.29	36.37	<b>36.50</b>	36.02	35.45	35.69
computer	37.93	39.04	39.11	<b>39.29</b>	38.49	38.12	37.63
dice	45.99	<b>46.82</b>	46.64	46.40	46.09	45.58	45.45
flowers	42.45	43.48	<b>43.61</b>	43.18	42.87	40.87	42.68
girl	43.72	<b>43.95</b>	<b>43.92</b>	43.89	43.47	42.28	43.06
traffic	37.10	<b>37.85</b>	37.81	<b>37.90</b>	37.53	36.86	36.77
trees	34.80	34.88	34.89	<b>34.97</b>	34.79	33.86	34.59
valldemossa	36.22	36.65	<b>36.70</b>	<b>36.75</b>	36.29	35.74	36.03
aircraft	37.01	37.59	37.57	<b>37.70</b>	36.75	37.39	36.72
asia	37.77	38.67	<b>38.76</b>	<b>38.72</b>	38.39	37.65	37.80
castle	37.29	38.06	38.11	<b>38.22</b>	37.69	37.35	37.28
man picture	36.17	37.78	<b>37.90</b>	37.75	37.35	36.86	36.79
maya	34.29	34.72	34.82	<b>34.88</b>	34.75	33.75	34.48
panther	37.74	38.53	38.62	<b>38.71</b>	38.48	37.21	38.10
tiger	36.41	36.92	<b>37.22</b>	<b>37.22</b>	36.94	36.13	36.61
young man	39.10	40.79	40.94	<b>41.00</b>	40.56	39.60	39.92
<b>Average</b>	37.85	38.54	<b>38.64</b>	<b>38.60</b>	38.17	37.44	37.81

the central pixel of each block/patch. Several estimators are collected at the same pixel location because of patch overlapping and are aggregated by computing the uniform average. Surprisingly, the quality increases by 0.1 db at most with the patchwise OWF, which is visually imperceptible.

Finally, the algorithm can be potentially iterated; in that case, the denoised image after the first iteration is processed by OWF given an estimation of the noise variance. The PSNR values increases in the range [0, 0.4] db, depending on the images and the signal-to-noise ratios.

In all our experiments, the PSNR values correspond to the application of the straightforward pointwise OWF procedure with no iteration.

## 5 Conclusion

A new image denoising filter to deal with the additive Gaussian white noise model based on a weights optimization problem is proposed. The proposed algorithm is computationally fast and its implementation is straightforward. Although the Gaussian kernel performs reasonably well in most papers, we showed that it is preferable to choose the triangular kernel with adaptive bandwidth. The obtained estimator uses an appropriate calibrated patch distance and is shown to converge at the usual optimal rate, under some regularity conditions on the target function. To the best of our knowledge such convergence results have not been established so far. The special case of images

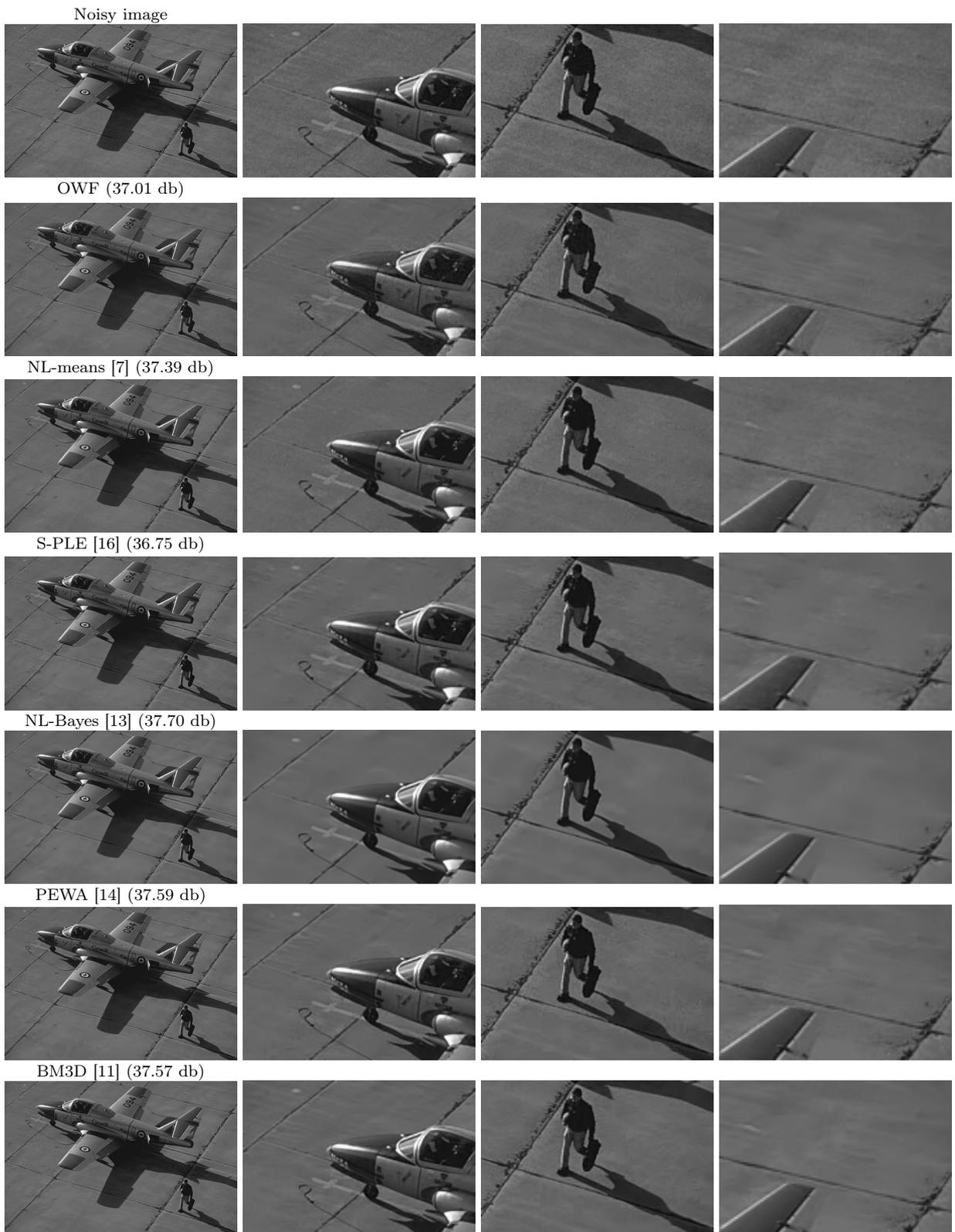


FIGURE 7 – Comparative results on the “aircraft” image corrupted with WGN ( $\sigma = 5$ ). From top to bottom : noisy image, OWF, NL-means [7], S-PLE [16], NL-Bayes [13], PEWA [14], BM3D [11].



FIGURE 8 – Comparative results on the “valldemossa” image corrupted with WGN ( $\sigma = 15$ ). From top to bottom : noisy image, OWF, NL-means [7], S-PLE [16], NL-Bayes [13], PEWA [14], BM3D [11].



FIGURE 9 – Comparative results on the “traffic” image corrupted with WGN ( $\sigma = 20$ ). From top to bottom : noisy image, OWF, NL-means [7], S-PLF [16], NL-Bayes [13], PEWA [14], BM3D [11].

TABLE 7 – Performance (PSNR) of denoising algorithms when applied to the 25 noisy (WGN) images ( $\sigma = 25$ ). The PSNR values are averaged over 3 experiments corresponding to 3 different noise realizations. The best PSNR values are in bold.

$\sigma = 25$	OWF	PEWA [14]	BM3D [11]	NL-BAYES [13]	S-PLF [16]	NL-MEANS [7]	DCT [2]
barbara	29.96	30.28	<b>30.72</b>	30.39	29.38	28.90	29.22
boat	29.20	29.65	<b>29.91</b>	29.67	29.38	28.60	28.69
cameraman	28.66	<b>29.48</b>	<b>29.45</b>	29.33	28.88	28.77	27.60
couple	28.93	29.48	<b>29.72</b>	29.19	29.16	28.28	28.57
hill	29.20	29.56	<b>29.85</b>	29.40	29.33	28.63	28.97
house	31.73	32.77	<b>32.86</b>	32.45	32.22	31.25	31.10
lena	31.57	31.89	<b>32.08</b>	31.86	31.57	30.51	31.02
man	29.17	29.50	<b>29.62</b>	29.33	29.26	28.52	28.49
peppers	29.45	<b>30.30</b>	30.16	30.07	29.69	28.90	28.36
alley	27.21	27.46	27.49	<b>27.59</b>	27.07	26.72	26.04
computer	29.60	30.01	<b>30.16</b>	<b>30.12</b>	29.61	28.99	28.33
dice	37.21	<b>39.36</b>	38.85	39.17	39.02	35.73	37.58
flowers	33.63	34.55	<b>34.62</b>	34.42	33.81	32.26	33.47
girl	35.73	<b>37.33</b>	37.27	37.11	36.86	34.69	36.21
traffic	28.10	28.48	<b>28.53</b>	<b>28.58</b>	28.34	27.74	27.50
trees	24.70	24.69	24.66	<b>24.90</b>	24.82	24.56	25.49
valldemossa	26.14	26.37	26.28	<b>26.50</b>	26.15	25.96	25.34
aircraft	30.29	30.72	30.78	<b>30.84</b>	30.37	29.89	28.93
asia	29.25	29.60	29.55	<b>29.67</b>	29.43	28.84	28.43
castle	28.76	<b>29.49</b>	29.36	<b>29.51</b>	28.89	28.66	28.18
man picture	27.61	28.44	<b>28.54</b>	28.24	27.82	27.33	26.93
maya	24.22	24.28	24.18	<b>24.43</b>	24.21	23.85	23.35
panther	28.70	28.83	28.81	<b>29.07</b>	28.85	28.08	28.06
tiger	27.78	27.99	28.03	<b>28.12</b>	27.81	27.12	27.11
young man	32.16	<b>33.25</b>	33.20	32.90	32.48	31.30	31.32
<b>Average</b>	29.55	<b>30.15</b>	<b>30.19</b>	30.12	29.77	28.96	28.97

with sharp edges needs to be investigated further as in [10, 49].

OWF automatically adjusts the bandwidth parameter and make the filter attractive in many applications. We need to adjust the patch size in most cases. Our numerical results confirm that optimal choice of the kernel improves the performance of the non-local means filter, under the same conditions. OWF is a very competitive filter when compared to recent patch-based filters.<sup>2</sup>

## A Proofs of the main results

### A.1 Proof of Theorem 2.1

We begin with some preliminary results. The following lemma is similar to Theorem 1 of Sacks and Ylvisaker [44] where, however, the inequality constraints are absent.

*Lemma A.1* : Let  $\mathcal{J}_\phi(w)$  be defined by (14). The unique weights  $w(x, x_0)$  which minimize  $\mathcal{J}_\phi(w)$

---

2. **Acknowledgments** : The authors would like to thank the referees for comments and remarks which helped us to improve considerably the manuscript. They would also like to thank B. Coll for discussions and fruitful comments and suggestions. Jin and Liu have been partially supported by the National Natural Science Foundation of China (grants No. 61661039, No. 11571052 and No. 11401590), the Natural Science Fund of Inner Mongolia Autonomous Region (grant No. 2016MS0107) and the Scientific Research Projection of Higher Schools of Inner Mongolia (grant No. NJZY16017).



NL-means (PSNR = 31.50 db)



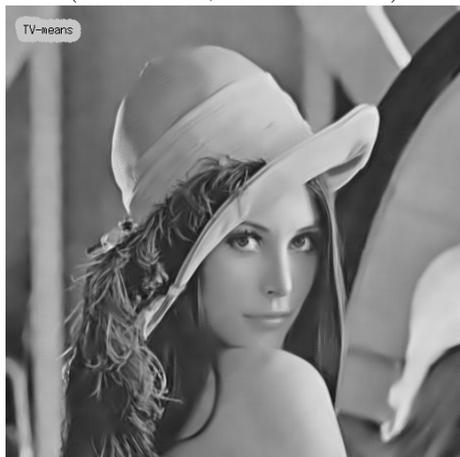
SAFIR (PSNR = 32.64 db)



OWF (PSNR = 31.39 db)  
( $m = 7 \times 7$ ,  $M = 21 \times 21$ )



OWF (PSNR = 32.64 db)  
( $m = 27 \times 27$ ,  $M = 13 \times 13$ )



TV-means (PSNR = 32.48 db)



S-PLC (PSNR = 32.64 db)

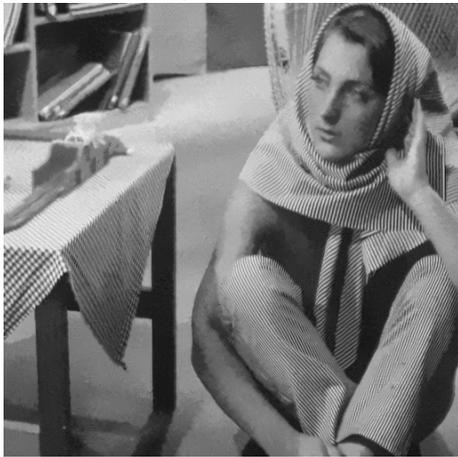
FIGURE 10 – Comparisons of related methods applied to the “lena” image (PSNR = 22.10 db,  $\sigma = 20$ ).



NL-means (PSNR = 30.22 db)



SAFIR (PSNR = 30.37 db)



OWF (PSNR = 30.24 db)  
( $m = 7 \times 7$ ,  $M = 21 \times 21$ )



OWF (PSNR = 31.01 db)  
( $m = 27 \times 27$ ,  $M = 13 \times 13$ )



TV-means (PSNR = 30.93 db)



S-PLS (PSNR = 30.82 db)

FIGURE 11 – Comparisons of related methods applied to the “barbara” image (PSNR = 22.10 db,  $\sigma = 20$ ).

TABLE 8 – Average of denoising results over the 25 tested images for several values of  $\sigma$ . The experiments with NL-Bayes [13], S-PLE [16], NL-means [7] and DCT [2] have been performed using the implementations of IPOL (*ipol.im*). The best PSNR values are in bold.

	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$	$\sigma = 50$
BM3D [11]	<b>38.64</b>	<b>34.78</b>	<b>32.68</b>	<b>31.25</b>	<b>30.19</b>	<b>26.97</b>
PEWA [14]	38.54	<b>34.75</b>	<b>32.67</b>	<b>31.26</b>	<b>30.15</b>	<b>26.95</b>
NL-Bayes [13]	<b>38.60</b>	<b>34.75</b>	32.48	<b>31.22</b>	30.12	26.90
S-PLE [16]	38.17	34.38	32.35	30.67	29.77	26.46
OWF	37.85	34.14	32.11	30.65	29.55	25.96
NL-means [7]	37.44	33.35	31.00	30.16	28.96	25.53
DCT [2]	37.81	33.57	31.87	29.95	28.97	25.91

subject to

$$w(x, x_0) \geq 0 \quad \text{and} \quad \sum_{x \in U_h(x_0)} w(x, x_0) = 1 \quad (39)$$

are given by

$$w(x, x_0) = \frac{1}{\sigma^2} [\lambda - S(x_0)\phi(x, x_0)]_+, \quad (40)$$

where  $\lambda$  and  $S(x_0)$  are uniquely determined by

$$\sum_{x \in U_h(x_0)} \frac{1}{\sigma^2} [\lambda - S(x_0)\phi(x, x_0)]_+ = 1, \quad (41)$$

and

$$\sum_{x \in U_h(x_0)} \frac{1}{\sigma^2} [\lambda - S(x_0)\phi(x, x_0)]_+ \phi(x, x_0) = S(x_0). \quad (42)$$

*Proof* : Since the objective functional  $\mathcal{J}_\phi(w)$  is continuous, it attains its minimum on the compact set determined by the constraints (39). Therefore the minimization problem has at least one solution. Recall that  $M = \text{card}(U_h(x_0))$ . Consider the Lagrange function

$$\bar{\mathcal{J}}(w) = \mathcal{J}_\phi(w) - 2\lambda \left( \sum_{x \in U_h(x_0)} w(x, x_0) - 1 \right) - 2 \sum_{x \in U_h(x_0)} g(x)w(x, x_0),$$

where  $w, g \in \mathbb{R}^M$  are vectors whose components are respectively  $w(x, x_0)$ ,  $x \in U_h(x_0)$  and  $g(x) \geq 0$ ,  $x \in U_h(x_0)$ , and  $\lambda \in \mathbb{R}$ . Let  $w^*$  be a minimizer of  $\mathcal{J}_\phi(w)$  under the constraints (39).

We first give an expression for  $w^*(x, x_0)$ . By standard results (cf. Theorem 2.2 of Rockafellar (1993 [63]) ; see also Theorem 3.9 of Whittle (1971 [64])), there are Lagrange multipliers  $\lambda \in \mathbb{R}$  and  $g(x) \geq 0$ ,  $x \in U_h(x_0)$  such that the following Karush-Kuhn-Tucker conditions hold : writing

$$S(x_0) = \sum_{x \in U_h(x_0)} w^*(x, x_0)\phi(x, x_0) \quad \text{for} \quad x \in U_h(x_0),$$

we have

$$\frac{\partial}{\partial w(x, x_0)} \bar{\mathcal{J}}(w) \Big|_{w=w^*} = 2S(x_0)\phi(x, x_0) + 2\sigma^2 w^*(x, x_0) - 2\lambda - 2g(x) = 0, \quad (43)$$

$$\frac{\partial}{\partial \lambda} \bar{\mathcal{J}}(w) \Big|_{w=w^*} = -2 \left( \sum_{x \in U_h(x_0)} w^*(x, x_0) - 1 \right) = 0, \quad (44)$$

$$\frac{\partial}{\partial g(x)} \bar{\mathcal{J}}(w) \Big|_{w=w^*} = -2w^*(x, x_0) \begin{cases} = 0, & \text{if } g(x) > 0, \\ \leq 0, & \text{if } g(x) = 0. \end{cases} \quad (45)$$

(The equality (45) holds since  $w^*(x, x_0)g(x) = 0$  for any  $x \in U_h(x_0)$  by the Karush-Kuhn-Tucker condition.) Notice that the gradients of the equality constraint function

$$G(w) = \sum_{x \in U_h(x_0)} w(x, x_0) - 1$$

and of the active inequality constraint functions (which attain 0 at  $w^*$ )

$$H_x(w) = w(x, x_0), \quad x \in U_h(x_0),$$

are linearly independent (for details we refer to Theorem 2.2 of Rockafellar [63]). Indeed they form a subset of the set of the following vectors in  $\mathbb{R}^M$ :  $(1, \dots, 1)$ ,  $(1, 0, \dots, 0)$ ,  $\dots$ ,  $(0, \dots, 0, 1)$ . The vectors in this subset are always linearly independent, since the number of active inequality constraints (which coincides with the number of weights  $w^*(x, x_0)$  that are equal to 0) is strictly less than  $M = \text{card}(U_h(x_0))$  due to the fact that  $\sum_{x \in U_h(x_0)} w^*(x, x_0) = 1$ .

We shall prove that

$$w^*(x, x_0) = \frac{[\lambda - S(x_0)\phi(x, x_0)]_+}{\sigma^2}, \quad (46)$$

with  $\lambda$  and  $S(x_0)$  satisfying (41) and (42). If  $g(x) = 0$ , then, since  $w^*(x, x_0) \geq 0$ , from (43) we obtain

$$\lambda - S(x_0)\phi(x, x_0) = \sigma^2 w^*(x, x_0) \geq 0.$$

Taking positive parts in the previous equality we see that (46) holds.

If  $g(x) > 0$ , then, by (45),  $w^*(x, x_0) = 0$ . Taking into account (43) we obtain

$$\lambda - S(x_0)\phi(x, x_0) = -g(x) \leq 0, \quad (47)$$

so that the relation (46) is again satisfied with

$$w^*(x, x_0) = 0 = \frac{[\lambda - S(x_0)\phi(x, x_0)]_+}{\sigma^2}.$$

Next we prove that  $\lambda$  and  $S(x_0)$  satisfy (41) and (42). The relation (41) follows immediately from the constraint (44) and the relation (46). The relation (42), is a consequence of the definition  $S(x_0)$  and (46).

Finally we prove the unicity of the weights  $w^*(x, x_0)$  as well as the unicity of the numbers  $\lambda$  and  $S(x_0)$  satisfying (41) and (42). Let  $a(x_0) = \frac{\lambda}{S(x_0)}$ . Substituting  $\lambda = a(x_0)S(x_0)$  in (42) we obtain

$$\sum_{x \in U_h(x_0)} [a(x_0) - \phi(x, x_0)]_+ \phi(x, x_0) = \sigma^2. \quad (48)$$

Since the function

$$H_\phi(z) = \sum_{x \in U_h(x_0)} [z - \phi(x, x_0)]_+ \phi(x, x_0)$$

is strictly increasing and continuous on  $[0, +\infty)$  with  $H_\phi(0) = 0$  and  $\lim_{z \rightarrow \infty} H_\phi(z) = +\infty$ , we see that  $a(x_0)$  is the unique solution in  $(0, +\infty)$  of the equation

$$H_\phi(z) = \sigma^2.$$

Therefore  $a(x_0)$  is uniquely determined by  $\phi(x, x_0)$ ,  $x \in U_h(x_0)$ .

In the same way, substituting  $\lambda = a(x_0)S(x_0)$  in (41) we obtain

$$S(x_0) \sum_{x \in U_h(x_0)} [a(x_0) - \phi(x, x_0)]_+ = \sigma^2. \quad (49)$$

From this equation we see that  $S(x_0)$  is also uniquely determined by  $\phi(x, x_0)$ ,  $x \in U_h(x_0)$ . As both  $a(x_0)$  and  $S(x_0)$  are uniquely determined by  $\phi(x, x_0)$ ,  $x \in U_h(x_0)$  we conclude that the same is true for  $\lambda$  and then for the weights  $w(x, x_0)$ ,  $x \in U_h(x_0)$ . ■

**Proof of Theorem 2.1.** We use Lemma A.1. As in the proof of this lemma writing  $a(x_0) = \frac{\lambda}{S(x_0)}$  and substituting  $\lambda = a(x_0)S(x_0)$  in (41) we see that the unique optimal weights  $w$  minimizing  $\mathcal{J}_\phi(w)$  subject to (39) are given by

$$w^*(x, x_0) = \frac{S(x_0)}{\sigma^2} [a(x_0) - \phi(x, x_0)]_+, \quad (50)$$

where  $a(x_0)$  and  $S(x_0)$  are uniquely determined by (48) and (49). With (49) this implies that

$$w^*(x, x_0) = \frac{[a(x_0) - \phi(x, x_0)]_+}{\sum_{x \in U_h(x_0)} [a(x_0) - \phi(x, x_0)]_+}. \quad (51)$$

Dividing the numerator and the denominator by  $a(x_0)$  gives the desired result.

## A.2 Proof of Theorem 3.1

Set  $\phi(x, x_0) = |f(x) - f(x_0)|$ . Recall that for the function  $\mathcal{J}_\phi(w)$  defined by (14) the optimal weights  $w$  are defined by (16). Using Hölder's condition (34) we have, for any  $w$ ,

$$\mathcal{J}_\phi(w) \leq \overline{\mathcal{J}}(w), \quad (52)$$

where

$$\overline{\mathcal{J}}(w) = \left( \sum_{x \in U_h(x_0)} w(x, x_0) L \|x - x_0\|_\infty^\beta \right)^2 + \sigma^2 \sum_{x \in U_h(x_0)} w^2(x, x_0). \quad (53)$$

By Theorem 2.1,

$$\overline{w}(x, x_0) = \frac{\left[ \overline{a}_h(x_0) - L \|x - x_0\|_\infty^\beta \right]_+}{\sum_{y \in U_h(x_0)} \left[ \overline{a}_h(x_0) - L \|y - x_0\|_\infty^\beta \right]_+}, \quad (54)$$

where  $\bar{a}_h(x_0) > 0$  is the unique solution on  $(0, \infty)$  of the equation

$$\bar{H}_h(\bar{a}_h(x_0)) = \sigma^2, \quad (55)$$

with

$$\bar{H}_h(z) = \sum_{x \in U_h(x_0)} L \|x - x_0\|_\infty^\beta \left[ z - L \|x - x_0\|_\infty^\beta \right]_+, \quad z \geq 0. \quad (56)$$

The assertion of Theorem 3.2 follows from (13), (14), (52) and the following lemma.

*Lemma A.2* : Assume that  $\phi(x, x_0) = L \|x - x_0\|_\infty^\beta$  and that  $h \geq c_1 n^{-\alpha}$  with  $0 \leq \alpha < \frac{1}{2\beta+2}$  and  $c_1 > 0$ , or  $h = c_1 n^{-\frac{1}{2\beta+2}}$  with  $c_1 > c_0 = \left( \frac{\sigma^2(2\beta+2)(\beta+2)}{8L^2} \right)^{\frac{1}{2\beta+2}}$ . Then

$$\bar{a}_h(x_0) = c_3 n^{-\beta/(2\beta+2)} (1 + o(1)) \quad (57)$$

and

$$\bar{\mathcal{J}}(\bar{w}) \leq c_4 n^{-\frac{2\beta}{2+2\beta}}, \quad (58)$$

where  $c_3$  and  $c_4$  are positive constants depending only on  $\beta$ ,  $L$  and  $\sigma$ .

*Proof* : We first prove (57) in the case when  $h = 1$ , i.e. when  $U_h(x_0) = \Omega$ . By the definition of  $\bar{a}_1(x_0)$  (see (55)), we have

$$\bar{H}_1(\bar{a}_1(x_0)) = \sum_{x \in \Omega} \left[ \bar{a}_1(x_0) - L \|x - x_0\|_\infty^\beta \right]_+ L \|x - x_0\|_\infty^\beta = \sigma^2. \quad (59)$$

Let  $\bar{h}_1 = (\bar{a}_1(x_0)/L)^{1/\beta}$  which implies  $\bar{a}_1(x_0) = L \bar{h}_1^\beta$ . The crucial point is the simple observation that

$$\left[ \bar{a}_1(x_0) - L \|x - x_0\|_\infty^\beta \right]_+ = 0 \quad \text{iff} \quad \|x - x_0\|_\infty > \bar{h}_1. \quad (60)$$

Using this equivalence, from (59), we obtain the following equation

$$\sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left( L^2 \bar{h}_1^\beta \|x - x_0\|_\infty^\beta - L^2 \|x - x_0\|_\infty^{2\beta} \right) = \sigma^2, \quad (61)$$

from which we will determine  $\bar{h}_1$  and  $\bar{a}_1(x_0)$ . By the definition of the neighborhood  $U_{\bar{h}_1}(x_0)$  it is easy to see that

$$\sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \|x - x_0\|_\infty^\beta = 8N^{-\beta} \sum_{k=1}^{N\bar{h}_1} k^{\beta+1} = 8N^2 \frac{\bar{h}_1^{\beta+2}}{\beta+2} (1 + o(1)).$$

This formula also applies with  $\beta$  replaced by  $2\beta$ , so that

$$\sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \|x - x_0\|_\infty^{2\beta} = 8N^2 \frac{\bar{h}_1^{2\beta+2}}{2\beta+2} (1 + o(1)).$$

Therefore, (61) implies

$$\frac{8L^2\beta}{(\beta+2)(2\beta+2)} N^2 \bar{h}_1^{2\beta+2} (1 + o(1)) = \sigma^2,$$

from which we infer that

$$\bar{h}_1 = c_0 n^{-\frac{1}{2\beta+2}} (1 + o(1)) \quad (62)$$

with  $c_0 = \left( \frac{\sigma^2(\beta+2)(2\beta+2)}{8L^2\beta} \right)^{\frac{1}{2\beta+2}}$ . From (62) and the definition of  $\bar{h}_1$ , we obtain

$$\bar{a}_1(x_0) = L\bar{h}_1^\beta = Lc_0^\beta n^{-\frac{\beta}{2\beta+2}} (1 + o(1)),$$

which proves (57) in the case when  $h = 1$ .

We next prove (57) for  $c_1 n^{-\alpha} \leq h < 1$ , where  $0 \leq \alpha \leq \frac{1}{2\beta+2}$  and  $c_1 > 0$  if  $\alpha < \frac{1}{2\beta+2}$ ,  $c_1 > c_0$  if  $\alpha = \frac{1}{2\beta+2}$ . It is clear that  $h \geq \bar{h}_1 = c_0 n^{-\frac{1}{2\beta+2}} (1 + o(1))$  for  $n$  sufficiently large. Using again the equivalence (60) we see that

$$\bar{H}_1(\bar{a}_1(x_0)) = \bar{H}_h(\bar{a}_1(x_0)) = \bar{H}_{\bar{h}_1}(\bar{a}_1(x_0)) = \sigma^2,$$

which proves that  $\bar{a}_1(x_0)$  is also the solution of (55) for  $c_1 n^{-\alpha} \leq h < 1$ . From this equality we deduce (62) as in the case  $h = 1$ .

We finally prove (58). Assume that  $h$  satisfies  $c_1 n^{-\alpha} \leq h \leq 1$ , where, as before,  $0 \leq \alpha \leq \frac{1}{2\beta+2}$  and  $c_1 > 0$  if  $\alpha < \frac{1}{2\beta+2}$ ,  $c_1 > c_0$  if  $\alpha = \frac{1}{2\beta+2}$ . Denote for brevity

$$G_h = \sum_{\|x-x_0\|_\infty \leq h} \left[ \bar{h}_1^\beta - \|x-x_0\|_\infty^\beta \right]_+.$$

Since  $h \geq \bar{h}_1$  for  $n$  sufficiently large, using the equivalence (60), we have  $\bar{H}_h(\bar{a}_1(x_0)) = \bar{H}_{\bar{h}_1}(\bar{a}_1(x_0)) = \sigma^2$  and  $G_h = G_{\bar{h}_1}$ . Thus, from (53) and (54), it follows that

$$\begin{aligned} \bar{\mathcal{J}}(\bar{w}) &= \frac{\bar{H}_{\bar{h}_1}(\bar{a}_1(x_0))^2 + \sigma^2 \sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left[ \bar{a}_1(x_0) - L\|x-x_0\|_\infty^\beta \right]_+^2}{L^2 G_{\bar{h}_1}^2} \\ &= \frac{\sigma^4 + \sigma^2 \sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left( \bar{a}_1(x_0) - L\|x-x_0\|_\infty^\beta \right)^2}{L^2 G_{\bar{h}_1}^2}, \end{aligned}$$

where to obtain the last line we use the fact that  $\bar{a}_1(x_0) = L\bar{h}_1^\beta$ . Developing the square and using  $\bar{a}_1(x_0) = L\bar{h}_1^\beta$  we obtain

$$\begin{aligned} &\sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left( \bar{a}_1(x_0) - L\|x-x_0\|_\infty^\beta \right)^2 \\ &= \bar{a}_1(x_0)L \sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left( \bar{h}_1^\beta - \|x-x_0\|_\infty^\beta \right) - \sum_{\|x-x_0\|_\infty \leq \bar{h}_1} \left( L^2 \bar{h}_1^\beta \|x-x_0\|_\infty^\beta - L^2 \|x-x_0\|_\infty^{2\beta} \right) \\ &= \bar{a}_1(x_0)LG_{\bar{h}_1} - \sigma^2, \end{aligned}$$

where the last step holds from the definition of  $G_h$  and (61). Therefore

$$\bar{\mathcal{J}}(\bar{w}) = \frac{\sigma^2 \bar{a}_1(x_0)}{L G_{\bar{h}_1}}.$$

Since

$$\begin{aligned} G_{\bar{h}_1} &= \sum_{\|x-x_0\|_\infty \leq \bar{h}_1} (\bar{h}_1^\beta - \|x-x_0\|_\infty^\beta) = \bar{h}_1^\beta \sum_{1 \leq k \leq N\bar{h}_1} 8k - \frac{8}{N^\beta} \sum_{1 \leq k \leq N\bar{h}_1} k^{\beta+1} \\ &= \frac{4\beta}{\beta+2} N^2 \bar{h}_1^{\beta+2} (1+o(1)) = \frac{4\beta}{(\beta+2)L^{1/\beta}} N^2 \bar{a}_1(x_0)^{(\beta+2)/\beta} (1+o(1)), \end{aligned}$$

taking into account that  $N^2 = n$  (by definition of  $n$ ), we obtain

$$\begin{aligned} \bar{\mathcal{J}}(\bar{w}) &= \sigma^2 \frac{(\beta+2)}{4\beta} L^{1/\beta-1} \bar{a}_1(x_0)^{-\frac{2}{\beta}} N^2 (1+o(1)) \\ &= c \frac{n^{\frac{2}{2\beta+2}}}{n} (1+o(1)) = cn^{-\frac{2\beta}{2\beta+2}} (1+o(1)), \end{aligned}$$

where the constant  $c$  depends only on  $\beta$ ,  $L$  and  $\sigma$ . ■

**Proof of Theorem 3.2.** Assume that  $\phi(x, x_0) \leq \alpha |f(x) - f(x_0)| + \delta_n$ . As  $(a+b)^2 \leq 2a^2 + 2b^2$ , we have

$$\begin{aligned} \left( \sum_{x \in U_h(x_0)} w(x, x_0) \phi(x, x_0) \right)^2 &\leq \left( \sum_{x \in U_h(x_0)} w(x, x_0) (\alpha |f(x) - f(x_0)| + \delta_n) \right)^2 \\ &\leq 2\alpha^2 \left( \sum_{x \in U_h(x_0)} w(x, x_0) |f(x) - f(x_0)| \right)^2 + 2\delta_n^2. \end{aligned}$$

Hence  $\mathcal{J}_\phi(w) \leq \max\{1, 2\alpha^2\} \bar{\mathcal{J}}(w) + 2\delta_n^2$ . So using (13) and setting  $\bar{w} = \arg \min_w \bar{\mathcal{J}}(w)$ , we get

$$\mathbb{E} (f_h^*(x_0) - f(x_0))^2 \leq \min_w \mathcal{J}_\phi(w) \leq \mathcal{J}_\phi(\bar{w}) \leq \max\{1, 2\alpha^2\} \bar{\mathcal{J}}(\bar{w}) + 2\delta_n^2. \quad (63)$$

Therefore, by Lemma A.2 and the condition that  $\delta_n = O\left(n^{-\frac{\beta}{2\beta+2}}\right)$ , we obtain

$$\mathbb{E} (f_h^*(x_0) - f(x_0))^2 = O\left(n^{-\frac{2\beta}{2\beta+2}}\right),$$

which is just (37).

### A.3 Proof of Theorem 3.3

We first prove the following result which shows that, with a proper choice of the search window size  $h$  and of the patch size  $\eta$ , there are two constants  $0 < \underline{\alpha}_1 < \alpha_1 < \infty$  such that, the quantity  $\phi_{\mathbf{v}}(x, x_0)$  defined by (24) lies between  $\underline{\alpha}_1 |f(x) - f(x_0)|^2$  and  $\alpha_1 |f(x) - f(x_0)|^2$  with high probability and with an error term  $\delta_n = O\left(n^{-\frac{\beta}{2+2\beta}}\right)$ , where  $n = \text{card}(\Omega) = N^2$ .

*Proposition A.3 :* Assume that  $f$  satisfies the local Hölder's condition (34) and that  $h = c_1 n^{-\frac{1}{2\beta+2}}$  with  $c_1$  sufficiently large :  $c_1 > c_0 = \left(\frac{\sigma^2(\beta+2)(2\beta+2)}{8L^2\beta}\right)^{\frac{1}{2\beta+2}}$ . Assume also that  $\eta = c_2 n^{-\alpha}$  for some  $\frac{1}{2(\beta+1)^2} < \alpha < \frac{1}{2\beta+2}$  and  $c_2 > 0$ . Then there exist constants  $\underline{\alpha}_1, \alpha_1 > 0$  and  $c_3 > 0$  depending only on  $\beta, L, \sigma$  and  $c_1, c_2$ , such that

$$\lim_{n \rightarrow \infty} \max_{x_0 \in \Omega} \max_{x \in U_h(x_0)} \mathbb{P} \left\{ \phi_{\mathbf{v}}(x, x_0) - \alpha_1 |f(x) - f(x_0)|^2 \leq c_3 n^{-\frac{\beta}{2+2\beta}} \right\} = 1 \quad (64)$$

and

$$\lim_{n \rightarrow \infty} \max_{x_0 \in \Omega} \max_{x \in U_h(x_0)} \mathbb{P} \left\{ \alpha_1 |f(x) - f(x_0)|^2 - \phi_{\mathbf{v}}(x, x_0) \leq c_3 n^{-\frac{\beta}{2+2\beta}} \right\} = 1. \quad (65)$$

Using this proposition we easily prove Theorem 3.3. Indeed, from (64) and the bound

$$\phi(x, x_0) = |f(x) - f(x_0)| \leq L \|x - x_0\|_{\infty}^{\beta} \leq L,$$

we obtain

$$\lim_{n \rightarrow \infty} \max_{x_0 \in \Omega} \max_{x \in U_h(x_0)} \mathbb{P} \left\{ \phi_{\mathbf{v}}(x, x_0) \leq \alpha_0 |f(x) - f(x_0)| + c_3 n^{-\frac{\beta}{2+2\beta}} \right\} = 1, \quad (66)$$

where  $\alpha_0 = \alpha L$ , which is the assertion of Theorem 3.3.

**Proof of Proposition A.3.** For the simplicity we shall assume that the kernel  $K_0$  is rectangular, the proof in the general case can be carried in the same way. Note that for the rectangular kernel we have  $\|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_0,2}^2 = \frac{1}{m} \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_2^2$ . For convenience, for  $\tau \in \mathcal{T}$ , let

$$\Lambda_{x_0,x}(\tau) = f(x_0 + \tau) - f(x + \tau) \quad (67)$$

and

$$\zeta_{x_0,x}(\tau) = \varepsilon(x_0 + \tau) - \varepsilon(x + \tau). \quad (68)$$

Recall that the size of the set  $\mathcal{T}$  is the same as the size of the patch  $\mathbf{v}(x)$  which is equal to  $m = (2r + 1)^2$ . For brevity set  $\Lambda_{x_0,x} = \Lambda_{x_0,x}(0) = f(x_0) - f(x)$ . Consider the estimator of  $\Lambda_{x_0,x}^2 = |f(x_0) - f(x)|^2$  defined by

$$\widehat{\Lambda}_{x_0,x}^2 = \frac{1}{m} \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_2^2 - 2\sigma^2. \quad (69)$$

With these notations, using (1), we get a decomposition of the estimation error  $\widehat{\Lambda}_{x_0,x}^2 - \Lambda_{x_0,x}^2$  into a bias term and a stochastic error :

$$\begin{aligned} \widehat{\Lambda}_{x_0,x}^2 - \Lambda_{x_0,x}^2 &= \frac{1}{m} \sum_{\tau \in \mathcal{T}} (v(x + \tau) - v(x_0 + \tau))^2 - 2\sigma^2 - \Lambda_{x_0,x}^2 \\ &= \frac{1}{m} \sum_{\tau \in \mathcal{T}} (\Lambda_{x_0,x}(\tau) + \zeta_{x_0,x}(\tau))^2 - 2\sigma^2 - \Lambda_{x_0,x}^2 \\ &= \overline{\Lambda}^2(x_0, x) - \Lambda_{x_0,x}^2 + \overline{S}(x_0, x), \end{aligned} \quad (70)$$

where  $\overline{\Lambda}^2(x_0, x) - \Lambda_{x_0,x}^2$  is the bias term with

$$\overline{\Lambda}^2(x_0, x) = \frac{1}{m} \sum_{\tau \in \mathcal{T}} \Lambda_{x_0,x}^2(\tau) \quad (71)$$

and  $\overline{S}(x_0, x) = \frac{1}{m} S(x_0, x)$  is the stochastic term with

$$S(x_0, x) = \sum_{\tau \in \mathcal{T}} (\zeta_{x_0,x}^2(\tau) - 2\sigma^2 + 2\Lambda_{x_0,x}(\tau) \zeta_{x_0,x}(\tau)). \quad (72)$$

To handle the both terms in (70) we make use of the following two lemmas. The first one give an estimate of the bias term  $\overline{\Lambda}^2(x_0, x) - \Lambda_{x_0,x}^2$ .

*Lemma A.4* : Assume the local Hölder condition (34). Then, with  $h = \frac{k}{N}$  and  $\eta = \frac{r}{N}$  being the widths of the search window  $U_h(x_0)$  and of the patch  $\mathbf{v}(x)$  respectively, we have, for any  $x_0 \in \Omega$  and  $x \in U_h(x_0)$ ,

$$\left| \overline{\Lambda}^2(x_0, x) - |f(x) - f(x_0)|^2 \right| \leq 4L^2 h^\beta \eta^\beta.$$

*Proof* : Clearly

$$\overline{\Lambda}^2(x_0, x) - |f(x) - f(x_0)|^2 = \frac{1}{m} \sum_{\tau \in \mathcal{T}} (\Lambda_{x_0, x}^2(\tau) - |f(x) - f(x_0)|^2). \quad (73)$$

For each  $\tau \in \mathcal{T}$ , we have

$$\begin{aligned} \Lambda_{x_0, x}^2(\tau) - |f(x) - f(x_0)|^2 &= ((f(x_0 + \tau) - f(x + \tau))^2 - (f(x_0) - f(x))^2) \\ &= [f(x_0 + \tau) - f(x + \tau) + f(x_0) - f(x)] \\ &\quad \times [f(x_0 + \tau) - f(x_0) + f(x) - f(x + \tau)]. \end{aligned} \quad (74)$$

Since  $\|x - x_0\|_\infty \leq h$  and  $\|\tau\|_\infty \leq \eta$ , from the local Hölder condition (34) we have

$$|f(x_0 + \tau) - f(x + \tau) + f(x_0) - f(x)| \leq 2L\|x - x_0\|_\infty^\beta \leq 2Lh^\beta \quad (75)$$

and

$$|f(x_0 + \tau) - f(x_0) + f(x) - f(x + \tau)| \leq 2L\|\tau\|_\infty^\beta \leq 2L\eta^\beta. \quad (76)$$

From (73)-(76) we obtain the conclusion of the lemma. ■

The second lemma will help us to give an estimate of the stochastic term  $\frac{1}{m}S(x_0, x)$ , where we recall that  $m$  is the size of the patch  $\mathbf{v}(x)$ .

*Lemma A.5* : There are two positive constants  $c_2$  and  $c_3$ , depending only on  $L$  and  $\sigma$ , such that, for any  $x_0 \in \Omega$ ,  $x \in U_h(x_0)$  and any  $0 \leq z \leq c_2 m^{1/2}$ ,

$$\mathbb{P}(|S(x_0, x)| \geq z\sqrt{m}) \leq c_3 z^{-2}.$$

*Proof* : By Chebyshev's inequality,

$$\mathbb{P}(|S(x_0, x)| \geq z\sqrt{m}) \leq \frac{\text{Var}(S(x_0, x))}{z^2 m}. \quad (77)$$

To complete the proof it is enough to have a control of  $\text{Var}(S(x_0, x))$ . For this, note that the variables  $\zeta_{x_0, x}(\tau)$ ,  $\tau \in \mathcal{T}$ , are Gaussian with  $\mathbb{E}\zeta_{x_0, x}(\tau) = 0$  and finite variance  $\mathbb{E}\zeta_{x_0, x}^2(\tau) = 2\sigma^2$ . Set for brevity

$$X_\tau = Y_\tau + 2\Lambda_{x_0, x}(\tau) \zeta_{x_0, x}(\tau), \quad \text{and} \quad Y_\tau = \zeta_{x_0, x}^2(\tau) - 2\sigma^2, \quad \tau \in \mathcal{T},$$

where the variables  $Y_\tau$ ,  $\tau \in \mathcal{T}$  are identically distributed with mean  $\mathbb{E}Y_\tau = 0$  and finite variance  $\mathbb{E}Y_\tau^2 \leq c\sigma^4$ , where  $c$  is an absolute constant. With these notations we have  $S(x_0, x) = \sum_{\tau \in \mathcal{T}} X_\tau$ . Moreover, the expectation of  $X_\tau$  is zero,  $\mathbb{E}(X_\tau) = 0$ , and using the obvious bound  $(u + v)^2 \leq 2u^2 + 2v^2$ , for the variance of  $X_\tau$  we have

$$\mathbb{E}X_\tau^2 \leq 2\mathbb{E}Y_\tau^2 + 8\Lambda_{x_0, x}^2(\tau) \mathbb{E}\zeta_{x_0, x}^2(\tau) \leq 2c\sigma^4 + 8L^2\sigma^2 \leq b^2, \quad (78)$$

where  $\Lambda_{x_0, x}^2(\tau) = |f(x + \tau) - f(x_0 + \tau)|^2 \leq L^2 \|x - x_0\|_\infty^{2\beta} \leq L^2$ , and  $b$  is a constant depending only on  $L$  and  $\sigma$ .

The point in handling the sum  $S(x_0, x) = \sum_{\tau \in \mathcal{T}} X_\tau$  is that the variables  $X_\tau$ ,  $\tau \in \mathcal{T}$ , are not necessarily independent. Remark that  $\zeta_{x_0, x}(\tau)$  and  $\zeta_{x_0, x}(\tau')$  are correlated if and only if  $\tau - \tau' = \pm(x_0 - x)$ . Indeed, it can be easily checked that

$$\mathbb{E}(\zeta_{x_0, x}(\tau)\zeta_{x_0, x}(\tau')) = \begin{cases} -\sigma^2, & \text{if } \tau - \tau' = x_0 - x, \\ \sigma^2, & \text{if } \tau - \tau' = x - x_0, \\ 0, & \text{otherwise.} \end{cases}$$

Since  $\zeta_{x_0, x}(\tau)$  and  $\zeta_{x_0, x}(\tau')$  are Gaussian random variables, this implies that if  $\tau - \tau' \neq \pm(x - x_0)$ , then  $\zeta_{x_0, x}(\tau)$  and  $\zeta_{x_0, x}(\tau')$  are independent. This implies that  $X_\tau$  and  $X_{\tau'}$  are also independent if  $\tau - \tau' \neq \pm(x - x_0)$ , and that, in particular, that  $\mathbb{E}(X_\tau X_{\tau'}) = 0$  if  $\tau - \tau' \neq \pm(x - x_0)$ . Consequently

$$\begin{aligned} \text{Var}(S(x_0, x)) &= \mathbb{E}(S(x_0, x)^2) = \sum_{\tau, \tau' \in \mathcal{T}} \mathbb{E}(X_\tau X_{\tau'}) \\ &= \sum_{\tau \in \mathcal{T}} \mathbb{E}(X_\tau^2) + \sum_{\tau \in \mathcal{T}} \sum_{\tau' \in \mathcal{T}: \tau' = \tau \pm (x - x_0)} \mathbb{E}(X_\tau X_{\tau'}). \end{aligned} \quad (79)$$

By the Cauchy-Schwartz inequality and (78)

$$\mathbb{E}(X_\tau X_{\tau'}) \leq E^{1/2}(X_\tau^2)E^{1/2}(X_{\tau'}^2) \leq b^2.$$

Implementing this into (79),

$$\text{Var}(S(x_0, x)) \leq mb^2 + 2mb^2 = 3mb^2. \quad (80)$$

The assertion of the lemma follows now from (77) and (80). ■

Now we proceed to prove the assertion (64) of Proposition A.3. From (69), (70) and the inequality  $[a - b]_+ \leq |a - b| = \frac{|a^2 - b^2|}{a + b}$  for  $a, b > 0$ , we have

$$\begin{aligned} \phi_{\mathbf{v}}(x, x_0) &= \left( \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_{0,2}} - \sqrt{2}\sigma \right)^+ \\ &\leq \frac{\left| \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_{K_{0,2}}^2 - 2\sigma^2 \right|}{\sqrt{2}\sigma} \\ &\leq \frac{\bar{\Lambda}^2(x_0, x)}{\sqrt{2}\sigma} + \frac{|\bar{S}(x_0, x)|}{\sqrt{2}\sigma}. \end{aligned}$$

By Lemma A.4 we have  $\bar{\Lambda}^2(x_0, x) \leq |f(x) - f(x_0)|^2 + 4L^2 h^\beta \eta^\beta$ , so that

$$\phi_{\mathbf{v}}(x, x_0) \leq \frac{|f(x) - f(x_0)|^2 + 4L^2 h^\beta \eta^\beta}{\sqrt{2}\sigma} + \frac{|\bar{S}(x_0, x)|}{\sqrt{2}\sigma}. \quad (81)$$

Now we shall control the stochastic term. Since  $\eta = c_2 n^{-\alpha}$  (where  $\frac{1}{4\beta+2} \leq \alpha \leq \frac{1}{2\beta+2}$ ), it follows that the size  $m = (2N\eta + 1)^2 = (2\sqrt{n}\eta + 1)^2$  of the patch  $\mathbf{v}(x)$  satisfies  $c_4^{-1} n^{1-2\alpha} \leq m \leq c_4 n^{1-2\alpha}$ . Applying Lemma A.5, we see that uniformly in  $x_0 \in \Omega$  and  $x \in U_h(x_0)$ ,

$$\mathbb{P} \left( |\bar{S}(x_0, x)| \geq \sqrt{\frac{\ln n}{m}} \right) = \mathbb{P} \left( |S(x_0, x)| \geq \sqrt{m \ln n} \right) \leq \frac{c_3}{\ln n}. \quad (82)$$

Therefore, taking into account (81) and (82), we obtain that uniformly in  $x_0 \in \Omega$  and  $x \in U_h(x_0)$ , with probability  $1 - O\left(\frac{1}{\ln n}\right)$ ,

$$\phi_{\mathbf{v}}(x, x_0) \leq \frac{|f(x) - f(x_0)|^2 + 4L^2 h^\beta \eta^\beta}{\sqrt{2}\sigma} + \frac{c_5 \sqrt{\ln n}}{n^{1/2-\alpha}}. \quad (83)$$

Since  $h = c_1 n^{-\alpha}$  and  $\eta = c_2 n^{-\alpha}$ , we obtain uniformly in  $x_0 \in \Omega$  and  $x \in U_h(x_0)$ , with probability  $1 - O\left(\frac{1}{\ln n}\right)$ ,

$$\begin{aligned} \phi_{\mathbf{v}}(x, x_0) - \frac{|f(x) - f(x_0)|^2}{\sqrt{2}\sigma} &\leq \frac{4L^2 h^\beta \eta^\beta}{\sqrt{2}\sigma} + \frac{c_5 \sqrt{\ln n}}{n^{1/2-\alpha}} \\ &\leq \frac{c_6}{n^{2\alpha\beta}} + \frac{c_5 \sqrt{\ln n}}{n^{1/2-\alpha}}. \end{aligned}$$

Taking into account that for  $\alpha$  satisfying  $\frac{1}{4\beta+2} \leq \alpha \leq \frac{1}{2\beta+2}$ , it holds  $2\alpha\beta \geq 1/2 - \alpha$ , we have uniformly in  $x_0 \in \Omega$  and  $x \in U_h(x_0)$ , with probability  $1 - O\left(\frac{1}{\ln n}\right)$ ,

$$\phi_{\mathbf{v}}(x, x_0) - \frac{|f(x) - f(x_0)|^2}{\sqrt{2}\sigma} \leq c_7 n^{\alpha - \frac{1}{2}} \sqrt{\ln n},$$

which proves (64).

For the second assertion (65) of the proposition we use a similar argument. From (70) and the equality  $[a - b]_+ = \frac{[a^2 - b^2]_+}{a + b}$  for  $a, b > 0$ , we have

$$\begin{aligned} \phi_{\mathbf{v}}(x, x_0) &= \left[ \frac{1}{\sqrt{m}} \|\mathbf{v}(x) - \mathbf{v}(x_0)\|_2 - \sqrt{2}\sigma \right]_+ \\ &\geq \frac{\left[ \bar{\Lambda}^2(x_0, x) + \bar{S}(x_0, x) \right]_+}{\sqrt{\bar{\Lambda}^2(x_0, x) + \bar{S}(x_0, x) + 2\sigma^2 + \sqrt{2}\sigma}} \\ &\geq \frac{|\bar{\Lambda}^2(x_0, x) - \bar{S}(x_0, x)|}{\sqrt{\bar{\Lambda}^2(x_0, x) + \bar{S}(x_0, x) + 2\sigma^2 + \sqrt{2}\sigma}}. \end{aligned}$$

By Lemma A.4 we have

$$|f(x) - f(x_0)|^2 - 4L^2 h^\beta \eta^\beta \leq \bar{\Lambda}^2(x_0, x) \geq |f(x) - f(x_0)|^2 + 4L^2 h^\beta \eta^\beta,$$

so that

$$\phi_{\mathbf{v}}(x, x_0) \geq \frac{|f(x) - f(x_0)|^2 - 4L^2 h^\beta \eta^\beta - |\bar{S}(x_0, x)|}{\sqrt{|f(x) - f(x_0)|^2 + 4L^2 h^\beta \eta^\beta + \bar{S}(x_0, x) + 2\sigma^2 + \sqrt{2}\sigma}}.$$

Recall that  $|f(x) - f(x_0)| \leq L \|x - x_0\|_\infty^\beta \leq L$ . Using (82) and the above inequality, we see that, with probability  $1 - O\left(\frac{1}{\ln n}\right)$ ,

$$\phi_{\mathbf{v}}(x, x_0) \geq c_8 |f(x) - f(x_0)|^2 - c_8 L^2 h^\beta \eta^\beta - c_8 n^{\alpha - \frac{1}{2}} \sqrt{\ln n}.$$

As in the case of the upper bound, this implies that, with probability  $1 - O\left(\frac{1}{\ln n}\right)$ ,

$$\phi_{\mathbf{v}}(x, x_0) - c_8 |f(x) - f(x_0)|^2 \geq c_9 n^{\alpha - \frac{1}{2}} \sqrt{\ln n},$$

which proves (65). This ends the proof of Proposition A.3.

## Références

- [1] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D : Nonlinear Phenomena*, 60 :259–268, 1992.
- [2] G. Yu and G. Sapiro. DCT image denoising : a simple and effective image denoising algorithm. *Image Processing On Line* (<http://dx.doi.org/10.5201/ipol.2011.ys-dct>), 2011.
- [3] S. Roth and M.J. Black. Fields of experts. *Int. J. Computer Vision*, 82(2) :205–229, 2009.
- [4] P. Milanfar. A tour of modern image filtering. *IEEE Signal Processing Magazine*, 30 :106–128, 2013.
- [5] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Processing*, 15(12) :3736–3745, 2006.
- [6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *IEEE Int. Conf. Comp. Vis. (ICCV'09)*, pages 2272–2279, Tokyo, Japan, 2009.
- [7] A. Buades, B. Coll, and J.-M. Morel. A review of image denoising algorithms, with a new one. *SIAM J. Multiscale Modeling & Simulation*, 4(2) :490–530, 2005.
- [8] P. Chatterjee and P. Milanfar. Is denoising dead? *IEEE Trans. Image Processing*, 19 :895–911, 2010.
- [9] A. Levin, B. Nadler, F. Durand, and W.T. Freeman. Patch complexity, finite pixel correlations and optimal denoising. In *Eur. Conf. Comp. Vis. (ECCV'12)*, pages 73–86, Firenze, Italy, 2012.
- [10] A. Maleki, M. Narayan, and R.G. Baraniuk. Suboptimality of nonlocal means for images with sharp edges. *App. Comput. Harmon. Anal.*, 33(3) :370–387, 2012.
- [11] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Processing*, 16(8) :2080–2095, 2007.
- [12] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *IEEE Int. Conf. Comp. Vis. (ICCV'11)*, pages 479–486, Barcelona, Spain, 2011.
- [13] M. Lebrun, A. Buades, and J.-M. Morel. Implementation of the “Non-Local Bayes” (NL-Bayes) image denoising algorithm. *Image Processing On Line 3 (2013)* (<http://dx.doi.org/10.5201/ipol.2013.16>), pages 1–42, 2013.
- [14] C. Kervrann. PEWA : Patch-based exponentially weighted aggregation for image denoising. In *Neural Information Processing Systems (NIPS'14)*, pages 2150–2158, Montreal, Canada, 2014.
- [15] P. Chatterjee and P. Milanfar. Patch-based near-optimal image denoising. *IEEE Trans. Image Processing*, 21 :1635–1649, 2012.
- [16] Y.-Q. Wang and J.-M. Morel. SURE guided Gaussian mixture image denoising. *SIAM J. Imaging Sciences*, 6 :999–1034, 2013.
- [17] L. P. Yaroslavsky. Digital picture processing. An introduction. In *Springer-Verlag, Berlin*, 1985.
- [18] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *IEEE Int. Conf. Comp. Vis. (ICCV'98)*, pages 839–846, Bombay, India, 1998.
- [19] J. Salmon, R. Willett, and E. Arias-Castro. A two-stage denoising filter : the preprocessed Yaroslavsky filter. In *IEEE Statistical Signal Processing Workshop (SSP'12)*, pages 464–467, Ann Arbor, Michigan, USA, 2012.

- [20] J. Polzehl and V.G. Spokoiny. Adaptive weights smoothing with applications to image restoration. *J. Royal Statist. Soc. B*, 62(2) :335–354, 2000.
- [21] J. Polzehl and V. Spokoiny. Image denoising : pointwise adaptive approach. *Ann. Statistics*, 31(1) :30–57, 2003.
- [22] J. Polzehl and V. Spokoiny. Propagation-separation approach for local likelihood estimation. *Probab. Theory Rel. Fields*, 135(3) :335–362, 2006.
- [23] C. Kervrann and J. Boulanger. Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Processing*, 15(10) :2866–2878, 2006.
- [24] C. Kervrann, J. Boulanger, and P. Coupé. Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal. In *Int. Conf. on Scale Space and Variational Methods (SSVM'07)*, pages 520–532, 2007.
- [25] M. Raphan and E.P. Simoncelli. Empirical Bayes least squares estimation without an explicit prior. In *Courant Institute of Mathematical Sciences, New York University ; New York : 2007a. (Tech. Rep. no. TR2007-900)*, 2007.
- [26] C. Kervrann and J. Boulanger. Local adaptivity to variable smoothness for exemplar-based image regularization and representation. *Int. J. Computer Vision*, 79(1) :45–69, 2008.
- [27] B. Goossens, Q. Luong, A. Pizurica, and W. Philips. An improved non-local denoising algorithm. In *Int. Workshop Local and Non-Local Approximation in Image Processing*, pages 1–14, Lausanne, Switzerland, 2008.
- [28] V. Doré and M. Cheriet. Robust NL-means filter with optimal pixel-wise smoothing parameter for statistical image denoising. *IEEE T. Signal Processing*, 57(5) :1703–1716, 2009.
- [29] C.-A. Deledalle, L. Denis, and F. Tupin. Iterative weighted maximum likelihood denoising with probabilistic patch-based weights. *IEEE Trans. Image Processing*, 18 :2661–2672, 2009.
- [30] D. Van de Ville and M. Kocher. SURE based non-local means. *IEEE Signal Processing Letters*, 16 :973–976, 2009.
- [31] L. Pizarro, P. Mrázek, S. Didas, S. Grewenig, and J. Weickert. Generalised nonlocal image smoothing. *Int. J. Computer Vision*, 90(1) :62–87, 2010.
- [32] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola. From local kernel to nonlocal multiple-model image denoising. *Int. J. Computer Vision*, 86(1) :1–32, 2010.
- [33] Y. Lou, X. Zhang, S. Osher, and A. Bertozzi. Image recovery via nonlocal operators. *J. Sci. Comput.*, 42(2) :185–197, 2010.
- [34] C. Louchet and L. Moisan. Total variation as a local filter. *SIAM J. Imaging Sciences*, 4 :651–694, 2011.
- [35] C.-A. Deledalle, V. Duval, and J. Salmon. Non-local methods with shape-adaptive patches (NLM-SAP). *J. Math. Imaging Vision*, 43(2) :103–120, 2012.
- [36] C. Louchet and L. Moisan. Posterior expectation of the total variation model : properties and experiments. *SIAM J. Imaging Sciences*, 6 :2640–2684, 2013.
- [37] I. Ram, M. Elad, and I. Cohen. Image processing using smooth ordering of its patches. *IEEE Trans. Image Processing*, 22 :2764–2774, 2013.
- [38] C. Sutour, C.-A. Deledalle, and J.-F. Aujol. Adaptive regularization of the nl-means : Application to image and video denoising. *IEEE Trans. Image Processing*, 23(8) :3506–3521, 2014.
- [39] J. Salmon and Y. Strozeccki. Patch reprojections for non-local methods. *Signal Processing*, 92(2) :477–489, 2012.

- [40] V. Duval, J.F. Aujol, and Y. Gousseau. A bias-variance approach for the nonlocal means. *SIAM J. Imaging Sciences*, 4 :760–788, 2011.
- [41] T. Buades, Y. Lou, J.-M. Morel, and Z. Tang. A note on multi-image denoising. In *Int. Workshop on Local and Non-Local Approximation in Image Processing*, pages 1–15, Tuusula, Finland, 2009.
- [42] B. Li, Q.S. Liu, J.W. Xu, and X.J. Luo. A new method for removing mixed noises. *Sci. China Ser. F (Information sciences)*, 54 :1–9, 2010.
- [43] J. Salmon and E. Le Pennec. NL-means and aggregation procedures. In *IEEE Int. Conf. Image Processing (ICIP'09)*, pages 2977–2980, Cairo, Egypt, 2009.
- [44] J. Sacks and D. Ylvisaker. Linear estimation for approximately linear models. *Ann. Statistics*, 6(5) :1122–1137, 1978.
- [45] J. Roll. Local and piecewise affine approaches to system identification. *Ph.D. dissertation, Dept. Elect. Eng., Linköping University, Linköping, Sweden*, 2003.
- [46] J. Roll and L. Ljung. Extending the direct weight optimization approach. In *Technical Report LiTH-ISY-R-2601. Dept. of EE, Linköping Univ., Sweden*, 2004.
- [47] J. Roll, A. Nazin, and L. Ljung. Nonlinear system identification via direct weight optimization. *Automatica*, 41(3) :475–490, 2005.
- [48] A.V. Nazin, J. Roll, L. Ljung, and I. Grama. Direct weight optimization in statistical estimation and system identification. In *System Identification and Control Problems (SICPRO08)*, pages 27–67, Moscow, Russia, 2008.
- [49] A. Maleki, M. Narayan, and R.G. Baraniuk. Anisotropic nonlocal means. *App. Comput. Harmon. Anal.*, 35(3) :452–482, 2013.
- [50] D.L. Donoho and J.M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3) :425–455, 1994.
- [51] A. Buades, B. Coll, and J.-M. Morel. Non-local means denoising. *Image Processing On Line*, 1, 2011.
- [52] P. Coupé, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot. An optimized blockwise Non-Local means denoising filter for 3D Magnetic Resonance Images. *IEEE Trans. Medical Imaging*, 27(4) :325–441, 2008.
- [53] J. Immerkaer. Fast noise variance estimation. *Comput. Vis. Image Understand.*, 64(9) :300–302, 1996.
- [54] M. Colom and A. Buades. Analysis and extension of the percentile method, estimating a noise curve from a single image. *Image Processing On Line*, 3 :332–359, 2013.
- [55] C. Sutour, C.-A. Deledalle, and J.-F. Aujol. Estimation of the noise level function based on a non-parametric detection of homogeneous image regions. *SIAM J. Imaging Sciences*, 8(4) :2622–2661, 2015.
- [56] Ch. Kervrann and J. Boulanger. Unsupervised patch-based image regularization and representation. In *Eur. Conf. Comp. Vision (ECCV'06)*, pages 555–567, Graz, Austria, 2006.
- [57] O.V. Lepski. On a problem of adaptive estimation in Gaussian white noise. *Theory Prob. Appl.*, 35 :454–466, 1990.
- [58] O.V. Lepski, E. Mammen, and V.G. Spokoiny. Optimal spatial adaptation to inhomogeneous smoothness : an approach based on kernel estimates with variable bandwidth selectors. *Annals of Statistics*, 25(3) :929–947, 1997.

- [59] S. Kindermann, S. Osher, and P. Jones. Deblurring and denoising of images by nonlocal functionals. *SIAM J. Multiscale Modeling & Simulation*, 4(4) :1091–1115, 2005.
- [60] J. Salmon. On two parameters for denoising with Non-Local means. *IEEE Signal Processing Letters*, 17 :269–272, 2010.
- [61] J. Darbon, A. Cunha, and T.F. Chan. Fast nonlocal filtering applied to electron cryomicroscopy. In *IEEE Int. Symp. Biomedical Imaging (ISBI)*, pages 1331–1334, Paris, France, 2008.
- [62] J. Froment. Parameter-free fast pixelwise non-local means denoising. *Image Processing On Line*, 4 :300–326, 2014.
- [63] R.T. Rockafellar. Lagrange multipliers and optimality. *SIAM review*, 35(2) :183–238, 1993.
- [64] P. Whittle. Optimization under constraints : theory and applications of nonlinear programming. In *Wiley-Interscience, New York*, 1971.