



HAL
open science

Denoising Very High Resolution Optical Remote Sensing Images: Application and Optimization of Non Local Bayes method

Antoine Masse, Sébastien Lefèvre, Renaud Binet, Stéphanie Artigues,
Gwendoline Blanchet, Simon Baillarin

► To cite this version:

Antoine Masse, Sébastien Lefèvre, Renaud Binet, Stéphanie Artigues, Gwendoline Blanchet, et al.. Denoising Very High Resolution Optical Remote Sensing Images: Application and Optimization of Non Local Bayes method. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2018, 11 (3), pp.691-700. 10.1109/JSTARS.2018.2793537 . hal-01672867

HAL Id: hal-01672867

<https://hal.science/hal-01672867>

Submitted on 13 Nov 2019

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.

Denoising Very High Resolution Optical Remote Sensing Images: Application and Optimization of Non Local Bayes method

Antoine Masse, Sébastien Lefèvre, Renaud Binet, Stéphanie Artigues, Gwendoline Blanchet, and Simon Baillarin

Abstract—Very High Resolution (VHR) optical Remote Sensing Images (RSI) are often corrupted by noise. Among popular denoising methods in the state-of-the-art, Non Local Bayes (NLB) has led to successful results on real datasets, with high quality and reasonable computation time. However, its computation time remains prohibitive with respect to requirements of operational RSI pipelines such as Pléiades one. In this paper, we tackle such an issue and introduce several optimizations aiming to significantly reduce the computation time required by NLB while keeping the best denoising quality (*i.e.* preserving edges, textures, homogeneous areas). More precisely, our improvements consist of reducing multiple estimations of a same pixel with a masking technique, and modifying the spatial extent of the similar patch search area (*i.e.* one of the main part of non local algorithms such as NLB). We report several experiments and discuss optimal settings for these parameters, allowing a gain in computation time of 50% (resp. 15%) with optimized masking strategy (resp. spatial extent of the search area). When both contributions are combined, we achieve the same denoising quality as standard NLB while doubling the computation efficiency, the latter being increased fivefold if we accept a very small (lower than 0.1%) loss in quality.

Index Terms—Image Denoising, Non Local Bayes, Remote Sensing, Very High Resolution, Pléiades

I. INTRODUCTION

REMOTE Sensing Images (RSI) are often corrupted by noise. In the case of Very High Resolution (VHR) optical images such as those acquired by the Pléiades satellite [1] [2], noise is mainly instrumental, well-known [3] and with a signal dependent variance. A simple variance stabilization transform [4] enables to transform the noise model to a zero-mean Gaussian noise (*i.e.* white noise) [5]. In this context, noise removal requires preserving important features like homogeneous areas, discontinuities, edges and textures [6]. While image denoising has been studied for decades in the image processing research community, and having led to efficient and effective methods to remove white noise in the state-of-the-art, denoising process needs to be improved in the context of RSI. Indeed, the nature of RSI makes their content particularly complex. To the best of the authors' knowledge, no successful results have been reported for denoising very high resolution optical images, and this issue remains challenging.

Image denoising methods can be divided into three families: spatial domain, transfer domain, and dictionary learning-based [7]; and into two categories: local and non-local [8]. We provide in Tab. I a representative list of state-of-the-art denoising methods. The three families differ on the image representation: *spatial domain* methods exploit correlations

TABLE I
TAXONOMY OF REPRESENTATIVE STATE-OF-THE-ART DENOISING METHODS. FAM. STANDS FOR FAMILY: S=SPATIAL DOMAIN, T=TRANSFORM DOMAIN, D=DICTIONARY LEARNING-BASED, AND CAT. FOR CATEGORY: L=LOCAL, NL=NON LOCAL.

Acronym	Method	Fam.	Cat.
TF	Trained Filter [11]	S	L
TV	Total Variation [12]	S	L
BF	Bilateral Filter [13]	S	L
NLM	Non Local Means [14]	S	NL
NLB	Non Local Bayes [9]	S	NL
BLS-GSM	Bayes Least Squares	T	L
	Gaussian Scale Mixture [15]	T	L
BM3D	Bloc Matching and 3D Filtering [10]	T	NL
KSVD	K-Singular Value Decomposition [16]	D	L
LSSC	Learned Simultaneous Sparse Coding [17]	D	NL

occurring within an image, *transfer domain* methods perform denoising through image projection to an orthonormal basis, and *dictionary learning-based* methods rely on emerging Machine Learning (ML) techniques to build denoising models with linear combinations learned from the image content. Each family can also be categorized as: *local* if it relies on a limited context around the pixel to be denoised and *non local* if it parses the whole image to denoise each single pixel. In a recent study [9], BM3D [10] and Non Local Bayes (NLB) [9] have been identified as the best denoising methods in terms of image quality, with a further superiority of NLB in terms of computation efficiency. We show that these generic results transfer very well to the specific case of RSI (see Sec. IV and Tab. IV).

However, in the operational context of VHR optical RSI production, computation time is a very important factor, and the prohibitive computation cost of denoising methods prevents them being included in the overall processing chain. In order to show that denoising could be included in an operational RSI production pipeline, we focus here on the promising NLB technique and introduce novel optimizations aiming to reduce both the amount of overestimations and the spatial extent of the search area. Such optimizations lead to significant reduction of the computation time while keeping the best denoising quality.

The paper is organized as follows. After an in-depth presentation of the NLB method (Sec. II), we introduce our optimizations (Sec. III). We then report and discuss experimental results achieved with our improvements over NLB on real RSI datasets (Sec. IV), before concluding this paper and providing future research directions.

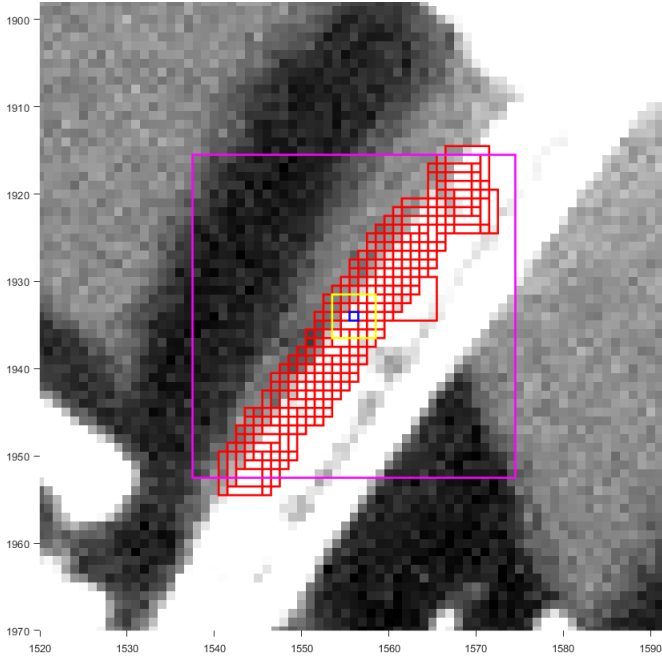


Fig. 1. Searching the N most similar patches (red) to a central patch (yellow) in a given search area (purple). Illustration on a Pléiades panchromatic image over the edge of Ponte de la Costituzione, Venezia, Italia.

II. NON LOCAL BAYES METHOD

A. Principle

Like BM3D, Non Local Bayes (NLB) is derived from the Non Local Means method (NLM) [14]. The seminal NLM method denoises each patch (*i.e.* group of pixels) through a weighted average of similar patches in a given neighborhood. Note that in theory, the term *Non Local* implies to look for similar patches in the whole image. However, all implementations constrain this search to a local window around the considered patch for computation efficiency issues. The NLB method improves NLM by involving a covariance matrix to estimate the variability in the group of similar patches. According to its authors [9], this improvement produces an optimal estimate in the sense of Bayesian minimal mean square error with a simple matrix inversion. Figure 1 illustrates the process of searching similar patches in a finite neighborhood. These similar patches are then used to build the 3D block of patches from which the Bayesian estimation is performed.

B. Algorithm

NLB is a two-step algorithm, and each step is composed of the same three parts (noted here a, b and c). For each patch within an image, NLB starts by (a) finding the most N similar patches and gathers them in a 3D block. Then it (b) computes 1st- and 2nd-order statistics of the 3D block and performs Bayesian estimation for every patch of the 3D block. Finally, since the patches of the 3D blocks may overlap and lead to a variable number of denoising estimations per pixel, (c) an aggregation and a weighting are performed.

The result of the first step is used to improve the search of similar patches in the second step. This improved search leads

Algorithm 1 Non Local Bayes algorithm: step 1.

Input: ImNoisy {Image corrupted with white noise}

Input: Set of parameters: w_1, k_1, N_1, β_1

Output: ImBasic {Denoised intermediate image}

for all pixel $x \in \text{ImNoisy}$ **do**

$p \leftarrow$ patch of size w_1^2 centered on x

$SA_p^{\text{noisy}} \leftarrow$ search area of size k_1^2 centered on p

for all patch $q \in SA_p$ **do**

 Compute $d(p, q)$ the similarity measure between patch p and q

end for

 Find the N_1 most similar patches and gather them into a 3D block BP_p^{noisy}

 Compute statistics of BP_p^{noisy} : $\bar{\mu}_p$ and Σ_p

for all patch q of BP_p **do**

$\hat{q} = \bar{\mu}_p + (\Sigma_p - \beta_1 \mathbb{I}) \Sigma_p^{-1} (q - \bar{\mu}_p)$ {Bayes estimation}

 Aggregate \hat{q} in ImBasic

 Increment weighting buffer

end for

 Weight ImBasic with weighting buffer

end for

Algorithm 2 Non Local Bayes algorithm: step 2.

Input: ImNoisy {Image with white noise}

Input: ImBasic {Denoised intermediate image from step 1}

Input: Set of parameters: $w_2, k_2, N_2, \beta_2, \tau_2$

Output: ImFinal {Denoised final Image}

for all pixel $x \in \text{ImNoisy}$ **do**

$p^{\text{noisy}} \leftarrow$ patch of size w_2^2 centered on x in ImNoisy

$p^{\text{basic}} \leftarrow$ patch of size w_2^2 centered on x in ImBasic

$SA_{p^{\text{basic}}} \leftarrow$ search area of size k_2^2 centered on p^{basic}

for all patch $q \in SA_{p^{\text{basic}}}$ **do**

 Compute $d(p^{\text{basic}}, q)$ the similarity measure between patch p^{basic} and q

end for

 Find at most N_2 most similar patches satisfying $d(p^{\text{basic}}, q) \leq \tau_2$ and gather them into a 3D block BP_p^{basic}

 Compute statistics of BP_p^{basic} : $\bar{\mu}_p^{\text{basic}}$ and Σ_p^{basic}

for all patch q of BP_p^{noisy} **do**

$\hat{q} = \bar{\mu}_p^{\text{basic}} + \Sigma_p^{\text{basic}} (\Sigma_p^{\text{basic}} + \beta_2 \mathbb{I})^{-1} (q - \bar{\mu}_p^{\text{basic}})$ {Bayes estimation}

 Aggregate \hat{q} in ImFinal

 Increment weighting buffer

end for

 Weight ImFinal with weighting buffer

end for

to better mean vector and covariance matrix estimations, and thus better denoising with application of the Bayes theory.

Our implementation is based on the patch-wise version of NLB with vectorization of patches [18]. A total of nine parameters are used for NLB parametrization (see Tab. III): four for the first step, and five for the second step. These parameters are used to tune the patch size (w_1 and w_2), the search area size (k_1 and k_2), the number of similar patches selected for building a 3D block (N_1 and N_2), the noise

TABLE II
COMPUTATION COMPLEXITY OF PATCH DENOISING.

Notation	Description	Estimation
\mathcal{O}_{dist}	Distance computation to find similar patches	$\mathcal{O}(k^2 w^2)$
\mathcal{O}_{sort}	Partial sorting, keeps N best similar patches	$\mathcal{O}(k^2 \log(N))$
\mathcal{O}_{mean}	Mean vector computation	$\mathcal{O}(N w^2)$
\mathcal{O}_{cov}	Covariance matrix computation	$\mathcal{O}(N w^4)$
\mathcal{O}_{inv}	Covariance matrix inversion	$\mathcal{O}(w^6)$
\mathcal{O}_{bayes}	Bayes estimation	$\mathcal{O}(N(2w^6))$
\mathcal{O}_{agg}	Denoised patch aggregation and weighting	$\mathcal{O}(N(3w^2))$

estimation or noise attenuation (β_1 and β_2), and (only for step 2) the similarity threshold for similar patch selection (τ_2). To ease understanding of our work and support its reproducibility, we provide in this paper complete pseudo-codes for both steps 1 and 2 (resp. through Algorithms 1 and 2).

C. Computation complexity

As already stated, a particular attention should be given to the computation complexity to embed a denoising stage in an operational RSI production pipeline. Both steps of the NLB algorithm have a similar complexity measured as follows:

$$\mathcal{O}_{NLB}^{step} = \#p(\mathcal{O}_{dist} + \mathcal{O}_{sort} + \mathcal{O}_{mean} + \mathcal{O}_{cov} + \mathcal{O}_{inv} + \mathcal{O}_{bayes} + \mathcal{O}_{agg}) \quad (1)$$

Complexity of the individual parts for each denoised patch are summarized in Tab. II. We can observe that four parameters actually affect the overall computation complexity of each step: the search area size k , the patch size w , the number N of similar patches, and the number of reference patches $\#p$ (i.e. the total number of patches within an image). Reducing computation time implies lowering one or more of these parameters, knowing that for both steps and in most applications: $\#p \gg N > k > w$.

In order to reduce the overall computation time without lowering the denoising quality, we propose and discuss in Sec. III some original optimizations aiming to reduce the reviewed terms. These improvements will then be experimentally assessed in Sec. IV.

III. CONTRIBUTIONS

From the complexity analysis reported in the previous section, we have identified four parameters that affect the NLB overall efficiency. We introduce and discuss here several original optimizations aiming to lower the computation time while preserving the denoising quality. Our contributions can be separated into two categories, *estimation domain* and *spatial domain*, aiming to reduce, respectively, the amount of overestimations and the amount of pixels (i.e. spatial extent) to be analyzed.

A. Reducing multiple estimations

A first strategy to lower the computation complexity focuses on reducing the amount of estimations conducted on the same pixel. To do so, we can limit either the overlapping between patch estimations ($\#p$) or the amount of similar patches (N).

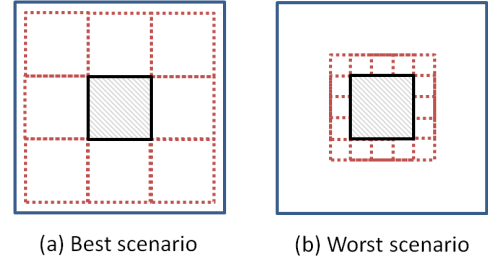


Fig. 2. Two extreme scenarios: (a) best case, minimal overlapping of similar patches; (b) worst case, maximal overlapping. The search area (blue) is defined around the central patch (black stripes) with the N similar patches (red dots).

1) *Masking patches*: Referring back to the NLB algorithm (see Algorithms 1 and 2), we can see that the Bayes estimation is applied to all patches within 3D blocks. These patches could also become reference patches and thus be denoised too many times. To reduce these overestimations, we propose a generalization of the PasteBoost masking method [9]. We recall Pasteboost creates a mask of patches that have been denoised at least once. As such, these patches cannot act as reference patches anymore. However, they can still be considered (and denoised) by another reference patch that has never been denoised before. This method drastically reduces the number of estimations of a given pixel. However, Let us note that the previous PasteBoost technique [9] only masks the 4-crossed pixels around the central pixel of each patch in the 3D block.

We propose to generalize this masking technique and formalize it as follows. Let us write w^{mask} the size of the patch to mask ($w^{\text{mask}} \leq w$), and H_{mask} the number of pixels masked in a 3D block. Due to unpredictable overlapping of patches, H_{mask} cannot be written explicitly but can only be framed between two extreme scenarios illustrated in Fig. 2 and expressed as follows:

$$H_{\text{mask}} \in \left[2w_{\text{mask}} \left(\sqrt{N} - 1 \right) + N - 1 ; w_{\text{mask}}^2 (N - 1) \right] \quad (2)$$

Finally, we write $\#p_{\text{unmasked}}$ the total number of reference patches used in the whole image (i.e. the number of unmasked patches). When incorporating the proposed masking technique into the original NLB algorithm, the overall computation complexity given in Eq. (1) becomes:

$$\mathcal{O}_{NLB}^{step} = \#p_{\text{unmasked}}(\mathcal{O}_{dist} + \mathcal{O}_{sort} + \mathcal{O}_{mean} + \mathcal{O}_{cov} + \mathcal{O}_{inv} + \mathcal{O}_{bayes} + \mathcal{O}_{agg}) \quad (3)$$

2) *Reducing the number of similar patches*: The in-depth complexity analysis conducted previously indicates that beyond the number of denoised patches, the number of similar patches (N) might also be lowered to decrease the computation complexity. However, we shall remind that a small value of N can lead to a badly conditioned and non-invertible covariance matrix. Furthermore, even if it is known from statistics theory that choosing $N > w^2$ should be enough to ensure good-conditioning, it is based on pixel independence hypothesis which can be wrong in very homogeneous images.

In a recent NLB sensibility analysis that we conducted on VHR RSIs [19], negligible effects from N_1 and N_2 have been

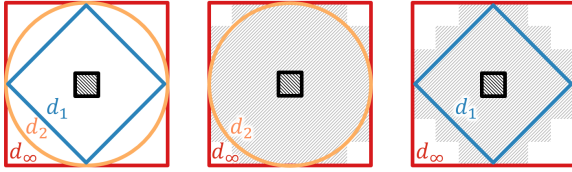


Fig. 3. Three different shapes of search area considering d_∞ (red), d_2 (orange), and d_1 (blue). Pixel (discretized) versions are also illustrated with striped blocks.

observed on both computation time and denoising quality, as long as N remains “large enough” to ensure covariance matrix inversion.

B. Reducing spatial domain

A second strategy consists in limiting the spatial extent of the analysis, by reducing either the search area or the patch size. In our recent study [19], we concluded that reducing search area size implied reducing computation complexity as denoising quality remains steady (except for very small values where denoising quality decreases). In the same study, reducing patch size from its optimal value led to similar computation times but brought a degradation of denoising quality. As reduction of patch size appears minor for reducing computation complexity, we focus on the reduction of the search area size by modifying its shape since we presume this parameter is already optimized (best quality).

1) *Reducing the search area:* As explained in Sec. II-C, the search area size k impacts the overall complexity. Let us recall the definition of the search area:

$$SA_p = \{x \in \text{ImNoisy} \subset \mathbb{R}^2 \mid \|x - p\|_a \leq k\} \quad (4)$$

with p the reference patch, k the search radius, ImNoisy the noisy image, and $\|\cdot\|_a$ the Minkowski norm of order a .

We consider here the three main distances derived from the Minkowski norm of order $a = 1, 2$ and ∞ :

$$d_1(x, p) = |x_1 - p_1| + |x_2 - p_2| \quad (5)$$

$$d_2(x, p) = \sqrt{(x_1 - p_1)^2 + (x_2 - p_2)^2} \quad (6)$$

$$d_\infty(x, p) = \max(|x_1 - p_1|, |x_2 - p_2|) \quad (7)$$

Traditional NL implementations assume $a = \infty$ which produces squared search area, but cases $a = 2$ (Euclidean distance) and $a = 1$ (Manhattan distance) can also be considered. All three search area shapes are illustrated in Fig. 3. The case $a = \infty$ is known to produce line artifacts in homogeneous regions [5]. Thus, reducing search area size while preventing artifacts can be done by modifying the search window shape instead of the search radius k , *i.e.* using different values for a .

The impact on computation complexity can be stated as follows: if the search area size is k^2 for $a = \infty$, it becomes $(\pi k^2)/4$ for $a = 2$, and $(k^2)/2$ for $a = 1$, which implies respectively a reduction of $\sim 22\%$ and 50% of the computation complexity for \mathcal{O}_{dist} and \mathcal{O}_{sort} . These reductions are theoretical since the search area needs to be discretized for image processing as illustrated in Fig. 3.

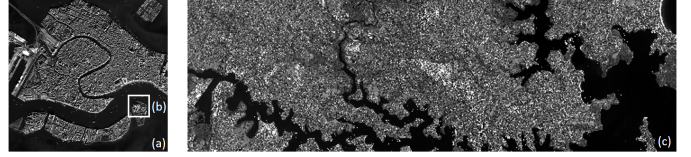


Fig. 4. RSI quick looks: (a) Venezia ($4,704 \times 4,704$ pixels) and (b) an extract of (a) (white square of 512×512 pixels in (a)), and (c) Sydney ($12,876 \times 40,000$ pixels, covering the third of a full PHR scene).

C. Summary of contributions

Among the different optimizations presented in this section, we have identified two techniques leading to some significant reductions of the computation time: (1) reducing over-estimations with a masking technique, and (2) modifying the search area shape. In the following section, we will report experimental assessments of these two contributions, taken either individually or together.

IV. EXPERIMENTS AND RESULTS

We report here the experiments conducted with some state-of-the-art denoising methods, and then our contributions to the NLB method, considering four datasets (one photograph and three RSI) and two architectures (desktop and High Performance Computing). We evaluate and discuss the results based on quality measures, computation times, and visual comparisons.

A. Datasets, architectures and evaluation criteria

1) *Datasets:* Four datasets are used in our experiments: (1) Lena, the most frequently used image in the image processing literature, (2)-(4) Pléiades (PHR) 1A satellite panchromatic band of resolution 0.7m (Very High Resolution). Image (2) is an extract of (3) over Venezia, Italia with half-urban, half-water; (4) has been acquired over Sydney, Australia with a majority of urban areas. Details are given in Tab. IV and quick looks in Fig. 4.

2) *Architectures:* Two architectures are used: (A) desktop platform with Intel i7-6700 HQ (2.6-3.5GHz), 8 GB RAM and Ubuntu 16.04 / Gdal 2.1.1 / Fftw 3.3.4 / GCC 4.8.2; (B) High Performance Computing (HPC) platform based on Lenovo NX360m5 with Intel Xeon E5-2650 v4 (2.2-2.9GHz), 128 GB RAM and CentOS 7.2 / Gdal 2.1.1 / Fftw 3.3.4 / GCC 4.8.2 and managed with PBS pro v13. Architecture (A) is only shown for comparison purposes with literature results since HPC (B) is specific to massive RSI production. We have restricted these two architectures to a single-core execution in order to evaluate the computation complexity and not the scalability. Note that NLB is multi-thread ready, the original image can be easily split before denoising due to NLB local search of similar patches.

3) *Evaluation criteria:* Denoising quality estimation usually relies on two quantitative measures: the Peak Signal-to-Noise Ratio (PSNR) and the Structural SIMilarity index (SSIM).

PSNR is a distortion measure that quantifies reconstruction quality between a denoised image and a reference one:

$$PSNR = 10 * \log_{10} \left(\frac{d^2}{MSE} \right) \quad (8)$$

with d the signal dynamics and MSE the Mean Squared Error between denoised and reference images.

SSIM [20] is a similarity measure between two images. Unlike the pixel-to-pixel comparison of the PSNR, SSIM performs a comparison between structures and is computed over patches from the two images. For two patches p and q respectively in denoised and reference images, SSIM is defined as:

$$SSIM(p, q) = \frac{(2\mu_p\mu_q + c_1)(2\sigma_{pq} + c_2)}{(\mu_p^2 + \mu_q^2 + c_1)(\sigma_p^2 + \sigma_q^2 + c_2)} \quad (9)$$

with μ_p and μ_q denoting respectively the mean of p and q , σ_p^2 and σ_q^2 the variance of p and q , σ_{pq} the covariance between p and q , and c_1 and c_2 two constants defined by

$$c_1 = (k_1 d)^2; c_2 = (k_2 d)^2 \quad (10)$$

with d the signal dynamics, $k_1 = 0.01$ and $k_2 = 0.03$ (default values).

The whole-image version of SSIM is called MSSIM and is defined as

$$MSSIM = \frac{1}{M} \sum_{m=1}^M SSIM(p_m, q_m) \quad (11)$$

with M the number of patches in the image.

Computation time is also used to compare denoising methods and our improvements to NLB. Times are presented in seconds and only reflect the cost of the main algorithm, without any I/O operations nor non-denoising operations.

B. Experimental protocol

Conversely to the Lena dataset which is already considered as noise-free, RSIs are corrupted with instrumental noise. This noise is converted to a white noise (zero mean and $\sigma^2 = 1$) thanks to Anscombe transform [4]. Since a noise-free reference remains unknown for VHR RSI, we have then adopted a specific experimental protocol summarized in Fig. 5. For each RSI denoising experimental evaluation, we consider a denoised image as the reference. This reference has been selected as the best result (*i.e.* method and optimal parameter settings) by a panel of photo-interpret experts who benefit from strong experience of those data in a professional environment. The reference image is then corrupted through artificial introduction of a white noise (zero mean and $\sigma^2 = 1$). Each experiment is repeated ten times to avoid negative influence from the random noise simulation. Finally, we evaluate each result and summarize all of them by computing mean and standard deviation for quality measures and computation time.

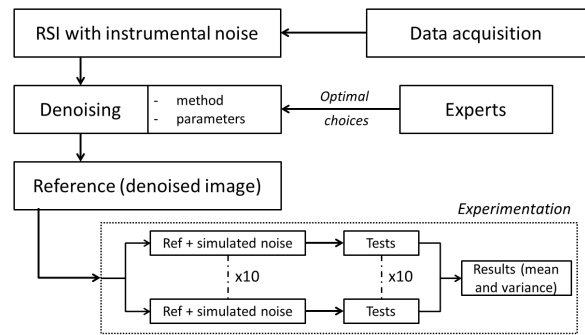


Fig. 5. Experimental protocol

TABLE III
LIST OF PARAMETERS FOR NLB ORIGINAL IMPLEMENTATION AND OPTIMAL VALUES SELECTED BY EXPERTS.

Parameter	Optimal value		Description	
	Step1	Step2		
w_1	w_2	5	5	Patch size
k_1	k_2	27	25	Search area size
N_1	N_2	74	30	Number of similar patches
β_1	β_2	1.0	1.6	Noise attenuation
	τ_2		2.5	Similarity threshold

C. Comparison of state-of-the-art denoising methods

We first provide an experimental comparison of state-of-the-art denoising methods considering the four datasets. Implementations and optimal parameter settings have been extracted from the IPOL database and are noted NLM [21], BLS-GSM [22], BM3D [23] and K-SVD [24]. For NLB, we based our implementation on the IPOL implementation [18]. ‘‘Lena’’ denoising parameters have been extracted from IPOL [18] and RSI denoising parameters have been optimally selected from a panel of experts as explained previously and are constant for all PHR datasets. These optimal NLB parameters are given in Tab. III and will be kept for all further experiments.

PSNR and computation time are given in Tab. IV, while denoised images are shown in Fig. 6. For all the results presented in this paper, MSSIM is close to 1 *i.e.* the highest possible score. Thus, the MSSIM index used in the literature is not adapted to small noise differences, consequently making it not suitable for RSI denoising assessment.

As far as PSNR is concerned, we can observe that NLB and BM3D achieve the highest scores, with an advantage of NLB for RSI. These two methods are followed by K-SVD, NLM and BLS-GSM. We can also notice that some implementations cannot deal with large images (BM3D and especially BLS-GSM and K-SVD), even with use of image splitting techniques.

Assessing computation time led us to observe that dictionary learning-based method is the slowest method, while spatial domain methods are the fastest ones. We can further notice that NLB is faster than NLM.

Figure 6 provides some visual comparisons of denoising results, as well as difference images computed between these images and the reference one. These results have been ex-

tracted from the PHR “Venezia Church” dataset (200×50 pixels *i.e.* $140m \times 35m$) and have been contrast-stretched to accentuate differences. This extracted area contains three bell tower shadows on water, two boats (white dots) and water textures. The denoising results look similar (Fig. 6 (c), (e), (g), (i) and (k)), but difference images highlight the method differences. We can observe that (d) BLS-GSM produces artifacts (points and vertical lines) due to domain transform of homogeneous area, NLM (f) and K-SVD (h) are still noisy, BM3D (j) and NLB (l) have similar denoising results with an advantage to NLB on edges and textures. NLB result (k) is the closest to the reference (b). These first experiments have shown that denoising qualities are different (improvement from noisy PSNR) between “Venezia” and “Sydney” datasets due to a majority of water around Venezia.

D. Experimental assessment of contributions

For the assessment of our contributions, we add new parameters and the associated modules (mask and shape) to the standard implementation.

1) *Reducing the number of estimations with masking technique:* To evaluate the impact of masking on denoising quality and computation time, we compare the results obtained with various masking parameters: $w_1^{\text{mask}} = \{1, 3, 5\}$, and $w_2^{\text{mask}} = \{1, 3, 5\}$. Let us recall that the case $w_1^{\text{mask}} = w_2^{\text{mask}} = 1$ is equivalent to original NLB. Results are reported in Tab. V for a total of 9 experiments on 3 datasets. Pasteboost [9] results are also provided for the sake of comparison. First, we can notice that masking at step 1 has no impact on these denoising quality (same PSNR values) but drastically reduces the computation time. Masking at step 2 led to different conclusions, since increasing masking size at step 2 slightly decreases denoising quality, while computation time behaves similarly to step 1. Thus, if the goal is to keep the same denoising quality while decreasing computation time, maximum masking at step 1 and minimal masking at step 2 is the optimal choice: it reduces the overall computation time by a factor of 2.

2) *Reducing the search area by modifying its shape:* Similarly, we evaluate our contribution related to the search area shape. To do so, we consider several settings for shape parameters: $k_1^a = k_1^{\text{opt}}$ and $k_2^a = k_2^{\text{opt}}$ with $a = \infty, 2$, and 1, the three types of distances and shapes introduced before. We recall that the case (k_1^∞, k_2^∞) is equivalent to original NLB. Results in terms of denoising quality and computation time are presented in Tab. VI for a total of 9 experiments on 3 datasets. Conclusions are similar to those driven in the previous experiment: shape modifications during step 1 have no influence on the final denoising quality but slightly decrease the computation time. For step 2, best results are obtained with a standard squared search area k^∞ . Let us remark that the analysis of computation time is less conclusive. Thus, if the goal is to keep the same denoising quality while decreasing the overall computation time, smaller areas for step 1, k_1^1 , and largest areas for step 2, k_2^∞ , is the optimal choice. It reduces the computation time by about 15%.

For the sake of comparison, Tab. VI also includes results from reduction of search radius k as a comparison point since

k is supposed to be optimal. Given a similar search area sizes, *i.e.* $(k_1^2 = 27; k_2^2 = 25)$ vs. $(k_1^\infty = 25; k_2^\infty = 23)$, and $(k_1^1 = 27; k_2^1 = 25)$ vs. $(k_1^\infty = 19; k_2^\infty = 17)$, new shapes fasten the process with similar quality results. Finally, we can note that for the “Sydney” dataset, all results have similar PSNR values and differences are too small to be noticed.

3) *Combine both mask and shape contributions:* We finally evaluate both contributions together. To do so, we compare $3^2 \times 3^2 = 81$ denoising results. For the “PHR Venezia” dataset, we illustrate the results with four scatter plots. The Figure 7.a shows the impact of step 1 parameters on step 1 result (“imBasic”). Masking and shape impact both denoising quality and computation time. We can notice that higher the masking size is, smaller is the impact of shape on computation time and denoising quality. Figure 7.b illustrates the impact of step 1 parameters on the final result (“imFinal”). Choices on step 1 parameters mainly influence computation time but not denoising quality. Figure 7.c confirms the non-influence of step 1 parameters on denoising quality, since the best final result is not obtained with best step 1 result. Finally, step 2 parameters impact is presented in Fig. 7.d. Final results are now mainly influenced by masking and shapes: minimum mask size and maximum search area (shape ∞) give best results for denoising quality. Four main profiles can be extracted from these results and are illustrated in Fig. 7.b and d: (A) the original result, without masking techniques and with traditional squared shape search area, (B) the best result that maximizes PSNR score, (D) the most efficient configuration that minimizes computation time, and (C) our view of the best compromise between efficiency and denoising quality. These four scenarios have been tested on the other datasets and are presented and detailed in Tab. VII. Compared to original results (A), best results (B) are about 2 times quicker with better or equivalent denoising quality. If efficiency becomes more important than denoising quality, most efficient configuration (D) is about 5 times faster than original settings (A) with a little but significant loss in dB. Finally, best compromise provides results (C) about 4 times faster than the original method (A) with a limited loss in quality ($< 0.02\text{dB}$).

V. CONCLUSION AND PERSPECTIVES

Non Local Bayes (NLB) is a powerful denoising method with multiple advantages including denoising quality, edge preservation, and computation efficiency. In this paper, we show the relevance of NLB for denoising VHR RSIs and improve the original NLB by introducing some computation optimizations. The new masking techniques keeps the best denoising quality while drastically reducing computation time by 50%. Furthermore, modifying the shape for the search area reduces computation time but with a little loss of quality. When both contributions are combined, we divide by two the computation time of standard NLB while keeping the same denoising quality. We have reported results from various experiments conducted in one-core conditions. NLB being multi-thread ready due to its local search of similar patches, let us note that the original image can be easily split to

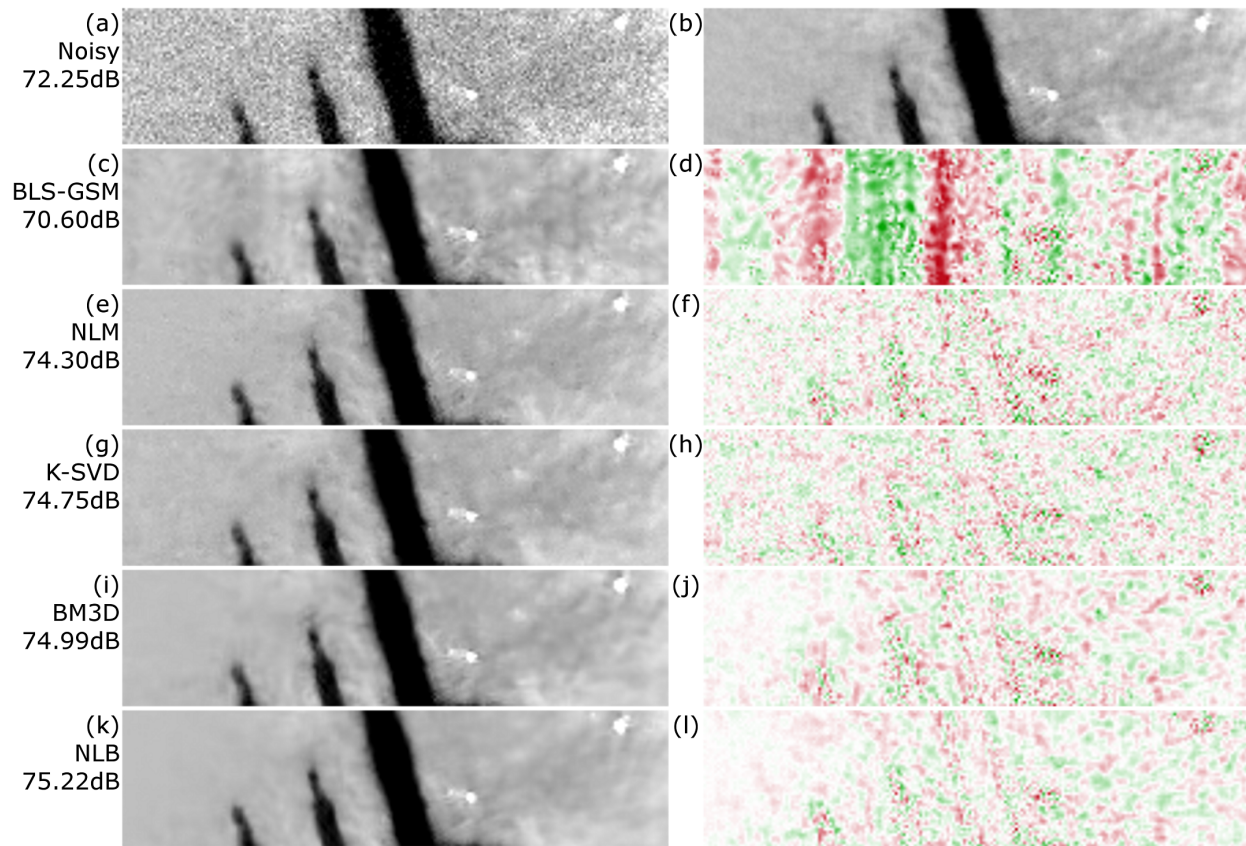


Fig. 6. Extracted results (200×50 pixels) of state-of-the-art denoising methods: (a) reference + noise, (b) reference, (c) BLS-GSM, (e) NLM, (g) K-SVD, (i) BM3D, (k) NLB, and respective difference images from reference (d), (f), (h), (j) and (l). Positive and negative differences are respectively given in green and red. PSNR values (result vs. reference) are indicated in dB for each method.

ensure scalability. Nevertheless, and to the best of the authors' knowledge, the contributions introduced in this paper establish new state-of-the-art results for VHR optical image denoising both on denoising quality and computation efficiency.

We have identified several research perspectives. The static, empirical choice of parameters remains a problem for large images with various landscape. Automating NLB parameter settings will require to elaborate new evaluation techniques that would further allow optimizing each parameter depending on the local context. Indeed, we can not rely on expert reference (since it is not universal and may be not adapted to all landscapes) nor on MSSIM (not significant for small variations). Evaluation of the denoising process can also be achieved by evaluating its impact on downstream methods like correlation and segmentation, or final applications like 3D reconstruction and land cover mapping.

ACKNOWLEDGMENT

We would like to thanks Airbus Defense and Space (ADS) and the French Spatial Agency (CNES) for image providing and computation facilities. This work is partially founded by CNES postdoctoral grant.

REFERENCES

- [1] S. Baillarin, C. Panem, L. Lebegue, and F. Bignalet-Cazalet, "Pleiades-HR imaging system: ground processing and products performances, few months before launch," in *ISPRS Advancing Remote Sensing Science*, vol. XXXVIII. Vienna, Austria: W. Wagner, B. Szkel, 2010, pp. 51–55.
- [2] L. Lebegue, D. Greslou, F. de Lussy, S. Fourest, G. Blanchet, C. Latry, L. Lacherade, J. M. Delvit, P. Kubik, C. Dchoz, V. Amberg, and F. Porez-Nadal, "Pleiades-HR image quality commissioning," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIX-B1, pp. 561–566, 2012. [Online]. Available: <http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XXXIX-B1/561/2012/>
- [3] C. Latry, S. Fourest, and C. Thiebaut, "Restoration technique for Pleiades-HR panchromatic images," in *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 39. Melbourne, Australia: M. Shortis, N. El-Sheimy, 2012, pp. 555–560. [Online]. Available: <http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XXXIX-B1/555/2012/>
- [4] F. J. Anscombe, "The transformation of Poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, no. 3/4, pp. 246–254, 1948. [Online]. Available: <http://www.jstor.org/stable/2332343>
- [5] M. Lebrun, M. Colom, and J.-M. Morel, "The Noise Clinic: a Blind Image Denoising Algorithm," *Image Processing On Line*, vol. 5, pp. 1–54, 2015.
- [6] C. Kervrann and J. Boulanger, "Optimal Spatial Adaptation for Patch-Based Image Denoising," *IEEE Transactions on Image Processing*, vol. 15, no. 10, pp. 2866–2878, Oct. 2006.
- [7] L. Shao, R. Yan, X. Li, and Y. Liu, "From Heuristic Optimization to Dictionary Learning: A Review and Comprehensive Comparison of Image Denoising Algorithms," *IEEE Transactions on Cybernetics*, vol. 44, no. 7, pp. 1001–1013, Jul. 2014.
- [8] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From Local Kernel to Nonlocal Multiple-Model Image Denoising," *International Journal of Computer Vision*, vol. 86, no. 1, p. 1, 2009. [Online]. Available: <http://dx.doi.org/10.1007/s11263-009-0272-7>
- [9] M. Lebrun, A. Buades, and J. M. Morel, "A Nonlocal Bayesian Image Denoising Algorithm," *SIAM Journal on Imaging*

TABLE IV

COMPARISON OF REPRESENTATIVE STATE-OF-THE-ART DENOISING METHODS, BEST RESULTS ARE SHOWN IN **BOLD**, AND N/A WHEN NOT APPLICABLE. EACH EXPERIMENT RESULT IS EXPRESSED AS THE MEAN AND THE STANDARD DEVIATION OF THE DENOISING RESULTS FOR 10 NOISE SIMULATIONS.

Dataset		Lena	Venezia Church	Venezia Church	Venezia	Sydney
Type		Photograph	PHR	PHR	PHR	PHR
Noise variance added		1	1	1	1	1
Architecture		A	A	B	B	B
# Pixels		~ 260K	~ 260K	~ 260K	~ 22M	~ 515M
PSNR Noisy/Ref (dB)		48.16	72.25	72.25	72.25	72.25
NLM	PSNR(dB)	47.84 ± 0.01	74.30 ± 0.01	74.30 ± 0.02	74.47 ± 0.00	72.81 ± 0.00
	Time(s)	4.17 ± 0.19	4.82 ± 0.19	5.80 ± 0.03	500.57 ± 0.77	11498 ± 190
NLB	PSNR(dB)	48.70 ± 0.02	75.21 ± 0.02	75.22 ± 0.01	75.31 ± 0.00	73.07 ± 0.00
	Time(s)	4.90 ± 0.03	3.54 ± 0.04	4.96 ± 0.01	434.95 ± 5.52	9138 ± 64
BLS-GSM	PSNR(dB)	47.82 ± 0.01	70.60 ± 0.01	70.60 ± 0.00	38.00 ± 0.00	N/A
	Time(s)	6.71 ± 0.06	12.62 ± 0.21	16.88 ± 0.18	1169.74 ± 3.11	N/A
BM3D	PSNR(dB)	48.68 ± 0.02	75.00 ± 0.02	74.99 ± 0.01	75.13 ± 0.00	N/A
	Time(s)	7.44 ± 0.05	10.01 ± 0.11	13.17 ± 0.03	771.87 ± 1.41	N/A
K-SVD	PSNR(dB)	48.47 ± 0.01	74.76 ± 0.02	74.75 ± 0.02	N/A	N/A
	Time(s)	16.59 ± 0.22	15.14 ± 0.24	17.51 ± 0.14	N/A	N/A

TABLE V

IMPACT OF MASKING PARAMETERS FOR PATCH SIZE $w_1^{\text{OPT}} = w_2^{\text{OPT}} = 5$, BEST RESULTS ARE SHOWN IN **BOLD**. EACH EXPERIMENT RESULT IS EXPRESSED AS THE MEAN AND THE STANDARD DEVIATION OF THE DENOISING RESULTS FOR 10 NOISE SIMULATIONS.

Dataset		Venezia Church		Venezia		Sydney	
Type	Noise Arch. #pixels	PHR 1 B 260K	PHR 1 B 260K	PHR 1 B 22M	PHR 1 B 22M	PHR 1 B 515M	PHR 1 B 515M
w_1^{mask}	w_2^{mask}	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)
1	1	75.22 ± 0.02	4.96 ± 0.08	75.31 ± 0.00	434.95 ± 5.52	73.07 ± 0.00	9138 ± 64
	3	75.19 ± 0.02	4.04 ± 0.02	75.29 ± 0.00	333.85 ± 5.42	73.05 ± 0.00	6578 ± 40
	5	75.15 ± 0.02	3.77 ± 0.01	75.25 ± 0.00	298.05 ± 2.77	73.03 ± 0.00	5991 ± 39
3	1	75.22 ± 0.02	2.80 ± 0.01	75.31 ± 0.00	229.51 ± 8.77	73.07 ± 0.00	5792 ± 82
	3	75.20 ± 0.01	1.88 ± 0.01	75.29 ± 0.00	164.14 ± 0.56	73.06 ± 0.00	3295 ± 191
	5	75.15 ± 0.02	1.62 ± 0.01	75.25 ± 0.00	140.53 ± 0.27	73.03 ± 0.00	2593 ± 47
5	1	75.22 ± 0.02	2.43 ± 0.01	75.31 ± 0.00	213.18 ± 1.59	73.07 ± 0.00	5090 ± 100
	3	75.20 ± 0.02	1.50 ± 0.02	75.29 ± 0.00	128.83 ± 2.76	73.06 ± 0.00	2566 ± 112
	5	75.14 ± 0.02	1.23 ± 0.02	75.25 ± 0.00	106.25 ± 0.33	73.03 ± 0.00	2077 ± 20
NLB PasteBoost [9]		75.21 ± 0.01	2.46 ± 0.01	75.30 ± 0.00	198.98 ± 8.28	73.06 ± 0.00	4032 ± 5

Sciences, vol. 6, no. 3, pp. 1665–1688, 2013. [Online]. Available: <http://dx.doi.org/10.1137/120874989>

- [10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering,” *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [11] L. Shao, H. Zhang, and G. d. Haan, “An Overview and Performance Evaluation of Classification-Based Least Squares Trained Filters,” *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1772–1782, Oct. 2008.
- [12] C. Louchet and L. Moisan, “Total Variation as a local filter,” *SIAM Journal on Imaging Sciences*, vol. 4, no. 2, pp. 651–694, 2011.
- [13] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Sixth International Conference on Computer Vision*, Jan. 1998, pp. 839–846.
- [14] A. Buades, B. Coll, and J. M. Morel, “A Review of Image Denoising Algorithms, with a New One,” *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005. [Online]. Available: <http://dx.doi.org/10.1137/040616024>
- [15] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, “Image denoising using scale mixtures of Gaussians in the wavelet domain,” *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.
- [16] M. Elad and M. Aharon, “Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries,” *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [17] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Non-local sparse models for image restoration,” in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 2272–2279.
- [18] M. Lebrun, A. Buades, and J.-M. Morel, “Implementation of the “Non-Local Bayes” (NL-Bayes) Image Denoising Algorithm,” *Image Processing On Line*, vol. 3, pp. 1–42, 2013.
- [19] A. Masse, S. Lefevre, R. Binet, S. Artigues, P. Lassalle, G. Blanchet, and S. Baillarin, “Fast and accurate denoising method applied to very high resolution optical remote sensing images,” p. 20, 2017. [Online]. Available: <http://dx.doi.org/10.1117/12.2277705>
- [20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [21] A. Buades, B. Coll, and J.-M. Morel, “Non-Local Means Denoising,” *Image Processing On Line*, vol. 1, 2011.
- [22] B. Rajaei, “An Analysis and Improvement of the BLS-GSM Denoising Method,” *Image Processing On Line*, vol. 4, pp. 44–70, 2014.
- [23] M. Lebrun, “An Analysis and Implementation of the BM3d Image Denoising Method,” *Image Processing On Line*, vol. 2, pp. 175–213, 2012.
- [24] M. Lebrun and A. Leclaire, “An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm,” *Image Processing On Line*, vol. 2, pp. 96–133, 2012.

TABLE VI
IMPACT OF SHAPE PARAMETERS FOR SEARCH AREA SIZE, BEST RESULTS ARE SHOWN IN **BOLD**. EACH EXPERIMENT RESULT IS EXPRESSED AS THE MEAN AND THE STANDARD DEVIATION OF THE DENOISING RESULTS FOR 10 NOISE SIMULATIONS.

Dataset		Venezia Church		Venezia		Sydney			
Type	Noise	Arch.	#pixels	PHR	1 B 260K	PHR	1 B 22M	PHR	1 B 515M
$k_1^a = 27$	$k_2^b = 25$			PSNR(dB)	Time(s)	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)
	$b = \infty$			75.22 \pm 0.02	4.96 \pm 0.01	75.31 \pm 0.00	407.12 \pm 12.74	73.07 \pm 0.00	9119 \pm 35
$a = \infty$	$b = 2$			75.21 \pm 0.02	4.67 \pm 0.02	75.30 \pm 0.00	393.42 \pm 11.74	73.07 \pm 0.00	8532 \pm 33
	$b = 1$			75.20 \pm 0.01	4.57 \pm 0.01	75.30 \pm 0.00	385.67 \pm 11.89	73.07 \pm 0.00	8146 \pm 30
	$b = \infty$			75.22 \pm 0.02	4.55 \pm 0.01	75.31 \pm 0.00	375.07 \pm 15.68	73.07 \pm 0.00	8620 \pm 178
$a = 2$	$b = 2$			75.21 \pm 0.01	4.27 \pm 0.02	75.30 \pm 0.00	360.69 \pm 9.38	73.07 \pm 0.00	7893 \pm 33
	$b = 1$			75.19 \pm 0.02	4.25 \pm 0.07	75.29 \pm 0.00	345.62 \pm 8.48	73.07 \pm 0.00	7563 \pm 70
	$b = \infty$			75.22 \pm 0.01	4.16 \pm 0.02	75.31 \pm 0.00	344.38 \pm 7.52	73.07 \pm 0.00	7874 \pm 48
$a = 1$	$b = 2$			75.21 \pm 0.01	4.03 \pm 0.13	75.30 \pm 0.00	322.52 \pm 6.91	73.07 \pm 0.00	8161 \pm 7
	$b = 1$			75.19 \pm 0.01	3.76 \pm 0.02	75.29 \pm 0.00	288.64 \pm 5.94	73.07 \pm 0.00	6876 \pm 34
$k_1^\infty = 27$	$k_2^\infty = 25$			75.22 \pm 0.01	4.96 \pm 0.01	75.31 \pm 0.00	407.12 \pm 12.74	73.07 \pm 0.00	9119 \pm 35
$k_1^\infty = 25$	$k_2^\infty = 23$			75.20 \pm 0.01	4.74 \pm 0.03	75.31 \pm 0.00	389.24 \pm 11.75	73.07 \pm 0.00	9072 \pm 467
$k_1^\infty = 19$	$k_2^\infty = 17$			75.19 \pm 0.01	4.01 \pm 0.01	75.29 \pm 0.00	331.69 \pm 10.58	73.07 \pm 0.00	7891 \pm 473

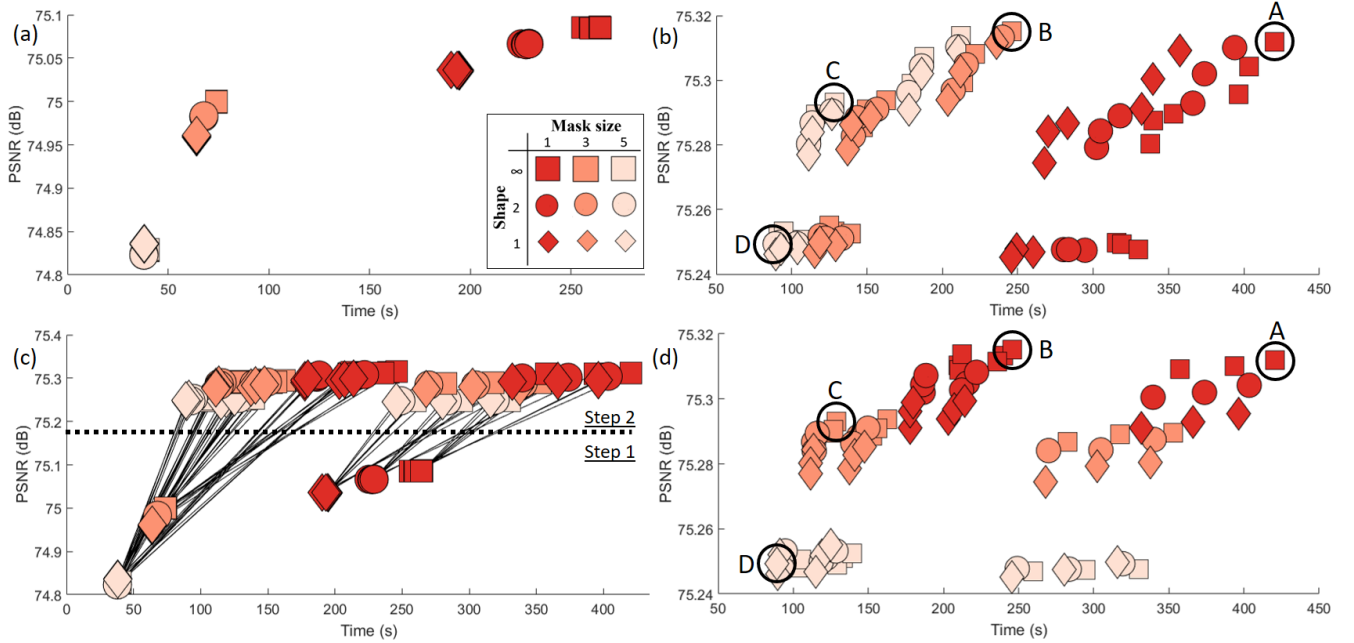


Fig. 7. Scatter plots for variation of masking sizes and shapes for step 1 and 2 and for “Venezia” dataset: (a) impacts of step 1 parameters on step 1 results, (b) impacts of step 1 parameters on final results, (c) evolution between step 1 and step 2, and (d) impacts of step 2 parameters on final results.

TABLE VII
QUANTITATIVE EVALUATION OF RESULTS OBTAINED WITH FOUR PARAMETER SETS: (A) ORIGINAL, (B) BEST PSNR, (C) COMPROMISE BETWEEN BEST PSNR AND LOWEST COMPUTATION TIME, AND (D) FASTEST. BEST RESULTS ARE SHOWN IN **BOLD**. EACH EXPERIMENT RESULT IS EXPRESSED AS THE MEAN AND THE STANDARD DEVIATION OF THE DENOISING RESULTS FOR 10 NOISE SIMULATIONS.

Dataset		Venezia Church		Venezia		Sydney				
Type	Noise	Arch.	#pixels	PHR	1 B 260K	PHR	1 B 22M	PHR	1 B 515M	
	w_1^m	w_2^m	k_1^a	k_2^b	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)
A	1	1	∞	∞	75.22 \pm 0.01	4.96 \pm 0.01	75.31 \pm 0.00	407.12 \pm 12.74	73.07 \pm 0.00	9119 \pm 35
B	3	1	∞	∞	75.22 \pm 0.02	2.80 \pm 0.01	75.31 \pm 0.00	229.51 \pm 8.77	73.07 \pm 0.00	5792 \pm 82
C	5	3	∞	∞	75.20 \pm 0.02	1.50 \pm 0.02	75.29 \pm 0.00	128.83 \pm 2.76	73.06 \pm 0.00	2566 \pm 112
D	5	5	1	1	75.14 \pm 0.01	1.04 \pm 0.04	75.25 \pm 0.00	86.18 \pm 2.47	73.05 \pm 0.00	1724 \pm 87