On two parameters for denoising with Non-Local Means

Joseph Salmon

***Abstract*—Non-Local Means (NLM) provides a very efficient procedure to denoise digital images. We study the influence of two important parameters on this algorithm: the size of the searching window and the weight given to the central patch. We verify numerically the common knowledge that the searching zone can be advantageously limited and we propose an efficient modification of the central weight based on the Stein's Unbiased Risk Estimate principle.**

***Index Terms*—Non-Local Means, denoising, aggregation, patches.**

I. INTRODUCTION

THE problem of image denoising has attracted a huge amount of work during the last decades. This problem consists in finding a good estimate of an image corrupted by a random noise. Many directions were successfully visited, though becoming more and more complex. Efficient denoising methods are mainly divided into two categories : transforms domain methods (among which are wavelet based methods [1] and second generation wavelets going from curvelets [2] to bandlets [3]) and pixeldomain methods.

The later are inspired by statistical kernel smoothing. A major step in this direction was initiated with the Bilateral Filter [4]. The idea is to smooth the image not only in the spatial domain, but also in the photometric domain. This procedure is shown to be close to a discretized version of anisotropic diffusion [5]. Other approaches investigate iterative versions of this kind of procedure [6], linking it to M-estimation approximation procedure.

Trying to improve the impact of the photometric closeness between pixels, Buades et al. [7] introduce the Non-Local Means (NLM). Their idea is to measure the similarity between two pixels by evaluating the distance between small patches centered on these two pixels. This extends the pixelwise photometric proximity to a patch based proximity. The authors interpret the NLM procedure as a kernel method on a bigger space, a space of images patches. To limit the computation time they restrict the search of patches to a narrower searching window. Doing so, they obtained a surprisingly efficient method though very simple to explain and to implement.

J. Salmon is with Laboratoire de Probabilités et Modèles Aléatoires,UMR 7599 , Université Paris Diderot, Paris, Case courier 7012, 75251 PARIS Cedex 05 FRANCE e-mail: (see http://people.math.jussieu.fr/~salmon), Telephone: +33 1 44 27 68 67, Fax: +33 1 44 27 72 23, EDICS:IMD-ANAL,SAS-STAT.
This work has been funded in part by the ANR Parcimonie.

Manuscript received October 2009; revised November, 2009.

Following the initial direction, improvement of the original NLM is quickly proposed [8]. It consists in applying the Lepskii's method [9] to select both a good window parameter and a good local size for the searching zone. The performance is also improved by iterating this refined procedure, as confirmed in variational approaches of NLM [10].

Other patch based methods have led to state-of-the-art denoising algorithms [11], [12] and encourage to extend the patches approaches for image denoising.

Here, we study some aspects of the original NLM : the little impact of the size of the searching window on natural images and the crucial role the weight of the central patch.

II. DEFINITION OF THE NL-MEANS

Let us define the NLM procedure [7], for a model of gray image corrupted by an additive Gaussian white noise. Let $f(i) = f(i_1, i_2) \in \mathbb{R}$, with $1 \leq i_1 \leq N_1$ and $1 \leq i_2 \leq N_2$, be a gray image with $N = N_1 \times N_2$ pixels. Assume we observe only a noisy version Y obtained with an additive random error W:

$$Y(i) = f(i) + \sigma W(i)\,.$$

Suppose that the $W(i)$ are i.i.d standard normal random variables and that the variance σ^2 is known. Our objective is to find a good estimate of f based on patches using only the noisy observed Y. We define now more precisely those patches. Let S be an odd integer, a patch (or neighborhood) $P(f)(i)$ is a subimage of f of size $S \times S$ centered on the pixel i, so that for $j = (j_1, j_2)$ with $j_1 \in [0; S-1]$ and $j_2 \in [0, S-1]$ one has:

$$P(f)(i)(j) = f\left(i_1 + j_1 - \frac{S-1}{2}, i_2 + j_2 - \frac{S-1}{2}\right)\,.$$

In patch based methods, one is interested by an estimate of the patch $P(f)(i_0)$ obtained from the collection of noisy patches $P(Y)(i)$. More precisely, the estimator $\widehat{P(f)}(i_0)$ is a weighted average of patches on a square window $\Omega(i_0)$ centered on i_0 of size $R \times R$. The weights used depend on the proximity between patches.

$$\widehat{P(f)}(i_0) = \sum_{i \in \Omega(i_0)} \lambda_{i_0,i}(Y) P(Y)(i)\,.$$

From the patch estimator, it is possible to recover a pixel estimator by projection. The easiest way, but not the only one, is to take into account only the centers of the patches. This leads to the estimator:

$$\hat{f}(i_0) = \sum_{i \in \Omega(i_0)} \lambda_{i_0,i}(Y) Y(i)\,.$$

In order to define the NLM estimator $\hat{f}^{NLM}(i_0)$, we must define the corresponding weights $\lambda^{NLM}_{i_0,i}(Y)$. First, denote $\alpha^{NLM}_{i_0,i}(Y) = e^{-\frac{1}{\beta}\|P(Y)(i_0)-P(Y)(i)\|^2}$, a coefficient measuring the proximity between patches. Now, we can define the raw weights $\lambda^{NLM}_{i_0,i}(Y)$ by normalizing, giving

$$\lambda^{NLM}_{i_0,i}(Y) = \frac{\alpha^{NLM}_{i_0,i}(Y)}{\sum_{i'\in\Omega(i_0)} \alpha^{NLM}_{i_0,i'}(Y)} \,. \quad (1)$$

Notice that the denominator is a normalizing factor, guaranteeing that the weights add to one. The NLM procedure consists in applying formula (1) to every pixel in the image.

The importance of the temperature parameter β has already been underlined by [7], [8] and we do not focus on it in this paper. The patch width parameter S is known to be less important for natural images (common choices are between 5 and 9) and we do not investigate its influence either. R, the size of the searching window is an important parameter that we study more carefully. A less obvious but crucial issue is how to compute the weight of the central patch. This problem emerges because for this weight the formula always yields $\alpha^{NLM}_{i_0,i_0}(Y) = 1$, so the importance of the central patch is always overestimated. In practice, a modified version should be used for better results.

III. Influence of the central patch

In their seminal work [7], the authors proposed to handle the central patch differently from the others. Indeed, the role of this patch is different in nature as it plays two roles at the same time. On the one hand it is the reference patch to be compared with the others, and on the other hand it is also an estimator patch to be averaged with the others.

Instead of using the original weight, the authors chose to assign the same value as the maximum of the other weights observed in the searching windows $\Omega(i_0)$, to the central patch weight. Then, they normalized the weights so that they would add to one.

Though this choice is not validated by theory, they obtained better results in practice. Denote $\alpha^{Max}_{i_0,i}(Y) = \max_{k\neq i_0} \alpha^{NLM}_{i_0,k}(Y)$ for $i = i_0$ and $\alpha^{Max}_{i_0,i}(Y) = \alpha^{NLM}_{i_0,i}(Y)$ for $i \neq i_0$. Then, normalizing the weights, they used in experiments

$$\lambda^{Max}_{i_0,i}(Y) = \frac{\alpha^{Max}_{i_0,i}(Y)}{\sum_{i'\in\Omega(i_0)} \alpha^{Max}_{i_0,i'}(Y)} \,.$$

Another approach was proposed by [13] but this leads to introducing an extra parameter to deal with the central patch.

In order to understand the influence of the central patch, we also define the weights $\lambda^{Zero}_{i_0,i}(Y)$ where we do not take into account the central patch. So, $\alpha^{Zero}_{i_0,i}(Y) = 0$ for $i = i_0$ and $\alpha^{Zero}_{i_0,i}(Y) = \alpha^{NLM}_{i_0,i}(Y)$ for $i \neq i_0$. We get $\lambda^{Zero}_{i_0,i}(Y)$ by normalizing as above.

We propose to change the way we compare pixels. The weights used should depend on the Euclidean distance between the true patches $\|P(f)(i_0)-P(f)(i)\|^2$. But unfortunately this information is not available so we use instead an unbiased estimator of the L^2 error between a patch $P(f)(i_0)$ and its translated versions $P(f)(i)$. This is the philosophy on which the Stein Unbiased Risk Estimator relies [14], [15]. Using only the properties that the $W(i)$ are independent and centered, the following equality holds for $i \neq i_0$

$$\mathbb{E}(\|P(Y)(i_0)-P(Y)(i)\|^2 - 2\sigma^2 S^2) = \|P(f)(i_0)-P(f)(i)\|^2 \,.$$

For $i = i_0$, $\|P(f)(i_0)-P(f)(i_0)\|^2 = 0$, and 0 is an unbiased estimator of the last quantity. Remind that σ is known, so defining

$$\hat{r}_{i_0,i} = \|P(Y)(i_0) - P(Y)(i)\|^2 - 2\sigma^2 S^2, \text{ for } i \neq i_0$$

and $\hat{r}_{i_0,i} = 0$ for $i = i_0$, we can define $\alpha^{Stein}_{i_0,i}(Y) = e^{-\frac{1}{\beta}\hat{r}_{i_0,i}}$ and we eventually get our new weights after normalizing

$$\lambda^{Stein}_{i_0,i}(Y) = \frac{\alpha^{Stein}_{i_0,i}(Y)}{\sum_{i'\in\Omega(i_0)} \alpha^{Stein}_{i_0,i'}(Y)} \,. \quad (2)$$

Notice that multiplying all the unmodified weights by $e^{-2\sigma^2 S^2/\beta}$ and normalizing them do not change the final weights. So, our modification of the NLM is simply equivalent to replacing the central weight in the NLM procedure by $e^{-2\sigma^2 S^2/\beta}$ (without modifying the other weights), before normalization.

We conducted experiments based on grayscale images[1] of size 512×512 for Barbara, Boat, Fingerprints, Flinstones and Lena. House and Peppers are 256×256, Barco is 302×231 and Chessboard is 200×200. The noise we used is normal, with standard deviation $\sigma = 5, 10, 20$ and 50.

In the experiments we present in Fig.1, we compare the performance of the different way to treat the central patch using (in this order) the weights $\lambda^{Stein}_{i_0,i}(Y), \lambda^{Max}_{i_0,i}(Y), \lambda^{NLM}_{i_0,i}(Y)$ and $\lambda^{Zero}_{i_0,i}(Y)$. We have optimized the temperature β for each image (meaning that we compare the best performance achievable for the four kind of weights considered). More precisely, for each image, we have selected the best temperature β with regards to Peak Signal to Noise Ratio (PSNR). We limit the tested values of β to a discrete grid ranging from $3\sigma^2$ to $90\sigma^2$ by step of $3\sigma^2$. In Fig.1 we used $S = 5$, but the same general behavior occurs for other experiments we conducted with $S = 3, 7, 9, 11$.

Our proposed weights $\lambda^{Stein}_{i_0,i}(Y)$ always outperformed the modified weights proposed by Buades et al. [7] (except for Fingerprint with $\sigma = 20$). The choice $\lambda^{Max}_{i_0,i}(Y)$ is interesting for strong noise ($\sigma \geq 20$) as the performance of these weights is on par with the results obtained with our modifications. Anyway, below this level it is better to use our modified weight $\lambda^{Stein}_{i_0,i}(Y)$. The loss for using $\lambda^{Max}_{i_0,i}(Y)$ instead of $\lambda^{Stein}_{i_0,i}(Y)$ or $\lambda^{NLM}_{i_0,i}(Y)$ can be reach 2dB (see for instance Flinstones and Fingerprint with $\sigma = 5$).

Moreover, the experiments also illustrate the need for modifying the raw $\lambda^{NLM}_{i_0,i}(Y)$ weights, as one can gain several dB doing it. For instance for a noise level $\sigma = 10$, the gain is bigger than 1dB for all our test images, and is even more than 2 dB for Barco, Fingerprints, Flinstones and Peppers!

As one could expect when the noise is strong, it is as relevant to set the central weight to zero as to use more refined weight (Max or Stein). In this case, the central patch is not

[1]Images available at http://people.math.jussieu.fr/~salmon/software.html

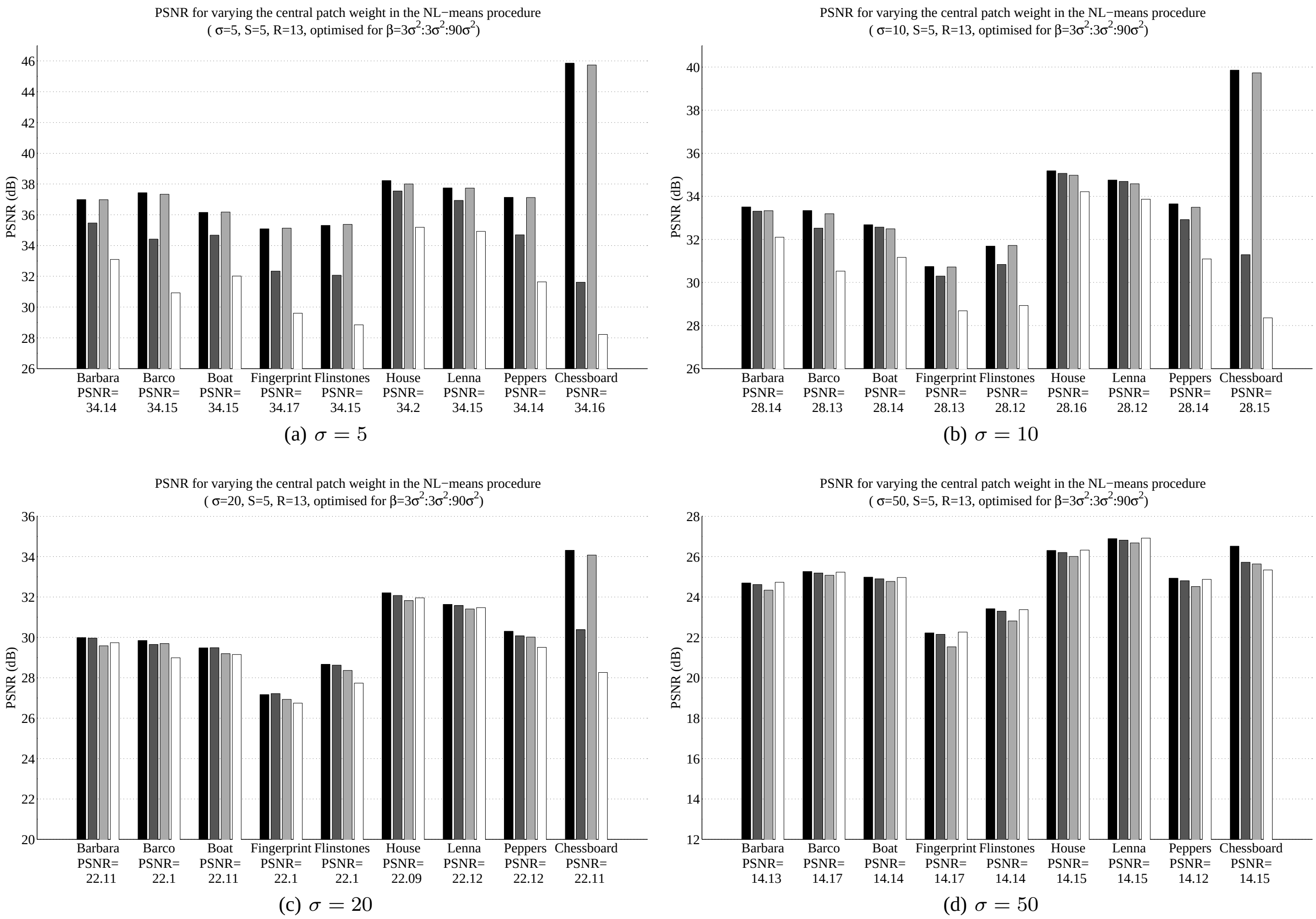

Fig. 1. Comparing performance of NLM changing the weight of the central patch (in order from black to white : Stein, Max, Normal, Zero) with four level of noise (a) $\sigma = 5$, (b) $\sigma = 10$, (c) $\sigma = 20$ and (d) $\sigma = 50$, with $S = 5$ and $R = 13$. The PSNR given below the name of the image is the one obtained with the noisy version of each image.

as important as for low noise, where it can represent a close version of the true patch. So its weight should (could) be lower to catch the others similar patches.

IV. Influence of the size of the searching window

In this section we illustrate the impact of the parameter R, the size of the searching window, on real images. Intuitively R should be as big as possible to have as many copies of the patch as we can. Also in the proof of convergence of the NLM procedure [16], R needs to tend to infinity. However, it is all the more interesting to pick R as small as possible, since the computation time depends crucially on R (it is proportional to $S^2R^2N_1N_2$).

In [8] it is proposed to automatically and locally select this parameter. Our simulations (cf. Fig.2) show how of little importance it is in practice to globally select R. For most standard images, the gain is insignificant for a parameter R greater than 15, with a fixed choice of S. The only observed exceptions to this phenomenon is for periodic (or quasi-periodic) images such as Chessboard and Fingerprint, where it is obvious that the larger R, the better the PSNR. We also choose β among values on a finite grid, this time from $3\sigma^2$ to $72\sigma^2$ by step of $3\sigma^2$ in order to maximize the performance (in term of PSNR) for each image. The weight used in Fig.2 are the common $\lambda_{i_0,i}^{Max}(Y)$, with $S = 5$, but the same behavior occurs when choosing other weights and other parameters S.

The performance decreases for most images (except for the textured Fingerprint) if R becomes bigger than 15. The loss can even overcome 2 dB (see Barbara or Peppers, with $\sigma = 50$) just changing R from 13 to 45. This phenomenon is due to the accumulation of small weights, leading to average non-similar patches, and so biasing the estimation.

V. Conclusion

In this work, we have illustrated the fact that the Non-Local Means are semi-local rather than non local. For natural images, one should not use the whole image as a searching zone to get better numerical results. Moreover, the way the central weight is defined is a crucial issue. Though for high noise level the correction given by Buades et al. is efficient, our Stein Unbiased Risk Estimator based correction is a better choice for more moderate noise level. Moreover, it yields a different point of view on the weights and we are investigating how to exploit this theoretical framework to design novel weighting schemes.

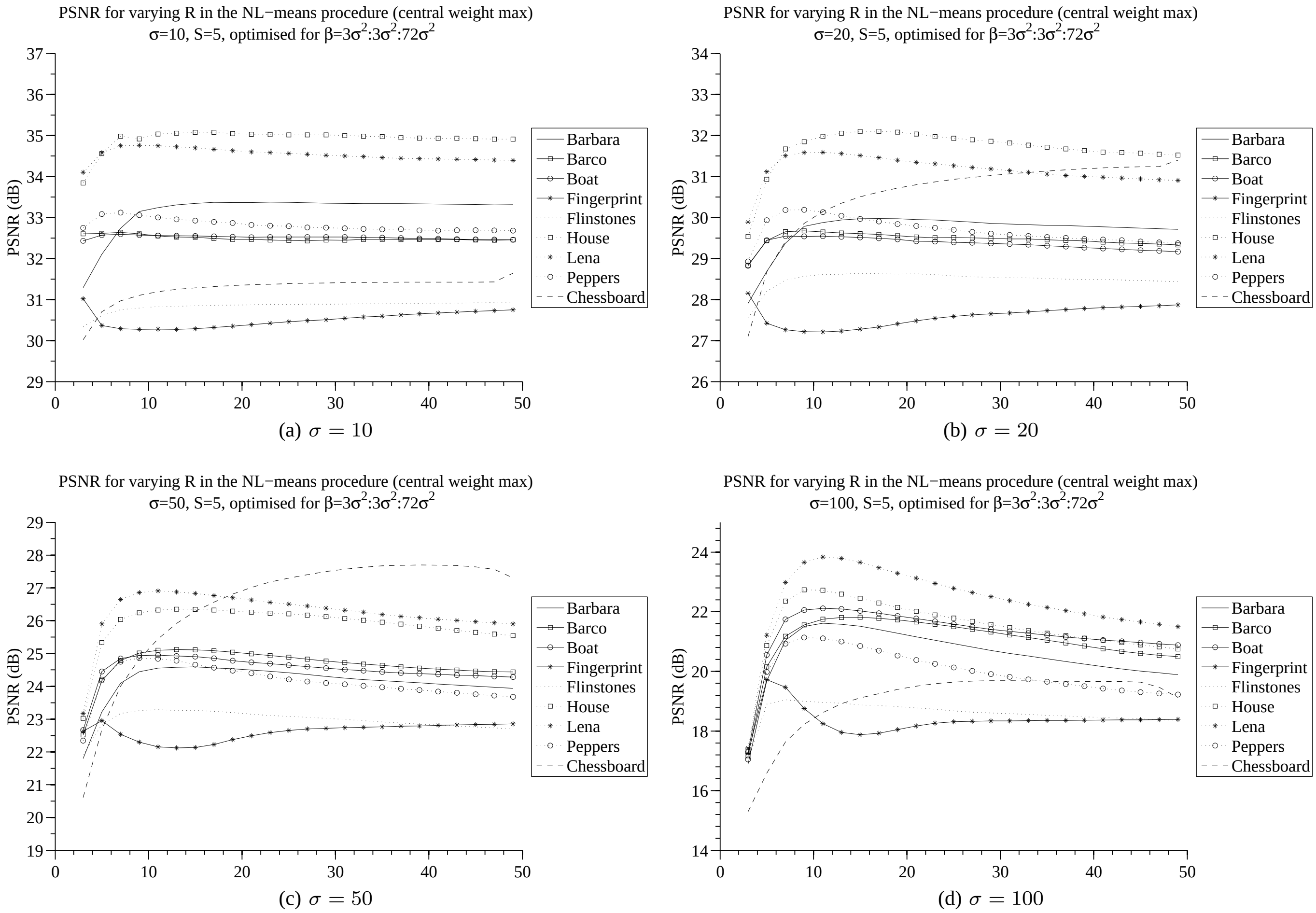

Fig. 2. Influence of R (size of the searching zone) on the PSNR of the NLM procedure, with coefficient $\lambda_{i_0,i}^{Max}(Y)$, $S = 5$, and optimizing the temperature from $\beta = 3\sigma^2 : 3\sigma^2 : 72\sigma^2$ in order to have the best PSNR for each image.

ACKNOWLEDGMENT

The authors would like to thank E. Le Pennec for fruitful discussions and computing tips. We also thanks G. Peyré for all the Matlab material he provides for free to the community.

REFERENCES

[1] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, 2003.

[2] J.-L. Starck, E. Candès, and D. Donoho, "The curvelet transform for image denoising," *IEEE Trans. Image Process.*, vol. 11, no. 6, pp. 670–684, 2002.

[3] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, 2005.

[4] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *ICCV*, p. 839, 1998.

[5] D. Barash and D. Comaniciu, "A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift," *Image Vision Comput.*, vol. 22, no. 1, pp. 73–81, 2004.

[6] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1141–1151, 2002.

[7] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530 (electronic), 2005.

[8] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. '2866–2878, 2006.

[9] O. V. Lepskiĭ, "On problems of adaptive estimation in white Gaussian noise," in *Topics in nonparametric estimation*, ser. Adv. Soviet Math. Providence, RI: Amer. Math. Soc., 1992, vol. 12, pp. 87–106.

[10] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Model. Simul.*, vol. 7, no. 3, pp. 1005–1028, 2008.

[11] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *Multiscale Model. Simul.*, vol. 7, no. 1, pp. 214–241, 2008.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, 2007.

[13] S. Zimmer, S. Didas, and J. Weickert, "A rotationally invariant block matching strategy improving image denoising with non-local means," in *LNLA*, 2008.

[14] J. Salmon and E. Le Pennec, "An aggregator point of view on NL-Means," in *Proceedings of the SPIE Conference on Mathematical Imaging: Wavelet XIII*, vol. 7446. SPIE, 2009, p. 74461E.

[15] D. van de Ville and M. Kocher, "SURE-based Non-Local Means," *IEEE Signal Process. Lett.*, 2009.

[16] A. Buades, "Image and movie denoising by non local means," Ph.D. dissertation, Universitat de les Illes Balears, 2006.