

# Anisotropic Non-Local Means with Spatially Adaptive Patch Shapes

Charles-Alban Deledalle<sup>1</sup>, Vincent Duval<sup>1</sup>, and Joseph Salmon<sup>2</sup>

<sup>1</sup> Institut Telecom – Telecom ParisTech – CNRS LTCI  
46, rue Barrault 75634 Paris cedex 13, France  
[perso.telecom-paristech.fr/~deledall/](mailto:perso.telecom-paristech.fr/~deledall/),  
[perso.telecom-paristech.fr/~vduval/](mailto:perso.telecom-paristech.fr/~vduval/)

<sup>2</sup> Université Paris 7 – Diderot– LPMA – CNRS-UMR 7599  
175 rue du Chevaleret 75013 Paris, France  
[www.math.jussieu.fr/~salmon/](http://www.math.jussieu.fr/~salmon/)

**Abstract.** This paper is about extending the classical Non-Local Means (NLM) denoising algorithm using general shapes instead of square patches. The use of various shapes enables to adapt to the local geometry of the image while looking for pattern redundancies. A fast FFT-based algorithm is proposed to compute the NLM with arbitrary shapes. The local combination of the different shapes relies on Stein’s Unbiased Risk Estimate (SURE). To improve the robustness of this local aggregation, we perform an anisotropic diffusion of the risk estimate using a properly modified Perona-Malik equation. Experimental results show that this algorithm improves the NLM performance and it removes some visual artifacts usually observed with the NLM.

**Keywords:** Image denoising, non-local means, spatial adaptivity, aggregation, risk estimation, SURE.

## 1 Introduction

During the last decades, the problem of image denoising in the presence of additive white Gaussian noise has drawn a lot of efforts. A wide variety of strategies were proposed, from partial differential equations (PDE) to transform-domain methods (e.g., wavelets), approximation theory or stochastic analysis.

A major difficulty in image denoising is to handle efficiently regular parts while preventing edges from being blurred, thus one needs spatial adaptive methods to deal with images. In PDE-driven image processing, this is often achieved using anisotropic diffusion [1–3]. Spatial adaptivity can also be reached by considering adaptive neighborhood filters, as the Yaroslavsky [4] or Bilateral [5] filters, or by applying Lepski’s method [6] (cf. [7, 8]). Though efficient at dealing with edges and smooth regions, such methods cannot proceed efficiently in textured regions.

To overcome this drawback, many authors have proposed to work with small sub-images, called patches, to take into account the redundancy in natural images, especially in textured parts. The interest of using patches lies in their

robustness to noise. The Non-Local Means algorithm (NLM) [9] and its variants [10, 11] are typical examples consisting in averaging similar pixels, measuring their similarity with patches. Dictionary learning on patches achieves *state-of-the-art* performance for denoising [12–14]. The key point of this method is to get a good representation for each patch of the image by using  $\ell_1$  regularization or greedy algorithms. Another *state-of-the-art* method in denoising is BM3D [15]. It also relies on patches and combines classical filtering techniques, such as wavelet thresholding and Wiener’s Filter, applied in the space of patches.

The NLM is quite efficient at dealing with smooth regions and textures. However, since it uses patches with fixed (square) shape and scale over the whole image, the performance is limited when dealing with edges with high contrast. Such edges can appear in natural images and in high dynamic range images (HDR) since these images present high contrasted features. They present few redundancies in term of patches, and their denoising versions suffer from a persistence of residual noise: this is called the *noise halo*. A way to overcome this drawback is to use locally chosen scales and orientations of shapes. As far as we know, few attempts have taken advantage of several patch sizes [13, 16] and only one handle variable shapes rather than squares ([17], to improve the BM3D algorithm).

In the NLM framework, spatial-adaptivity may be reached by locally selecting the parameters according to a local estimate of the risk [18]. This relies on Stein’s Unbiased Risk Estimate (SURE) [19] which was first used with NLM to globally select the bandwidth [20]. SURE-based methods were widely used in image processing [21, 22] after their introduction for wavelet thresholding [23].

*Our contributions* — We investigate the potential benefit of replacing the simple square patches with more general shapes, in the classical NLM filter. We give in Section 2 a general overview of the NLM method. We propose in Section 3 a fast algorithm, Non-Local Means with Shape-Adaptive Patches (NLM-SAP), based on the FFT, which allows to compute the solution of the NLM for arbitrary shapes. In Section 4, we locally select or combine the shape-based estimates by measuring the performance of their associated denoisers with SURE. As in [18], one has to regularize SURE to make a local decision. Since the choice of shape is an anisotropic decision, a specificity of our approach is that it uses an anisotropic diffusion scheme in the spirit of Perona and Malik [1]. In Section 5, we illustrate numerically, and above all visually, the gain in aggregating various shape-based estimates: using adaptive patch shapes in the context of NLM reduces the *noise halo* produced around edges.

## 2 An Overview of the NLM

We focus on the problem of denoising: an observed image  $\mathbf{Y}$  is assumed to be a noisy version of an unobserved image  $\mathbf{f}$  corrupted by a white Gaussian noise. Let  $\Omega \subset \mathbb{Z}^2$  be the indexing set of the pixels. For any pixel  $x \in \Omega$ :

$$\mathbf{Y}(x) = \mathbf{f}(x) + \varepsilon(x), \quad (1)$$

where  $\varepsilon$  is a centered Gaussian random variable with known variance  $\sigma^2$  and the noise components  $\varepsilon(x)$  are independent. First, let us present the definition of the NLM as introduced in [9]. For each pixel the output of the procedure is a weighted average of the whole image. The weights used are selected using a “metric” which determines whether two pixels are similar or not. The core idea of the NLM is to create a metric governed by patches surrounding each pixel, regardless of their position, i.e., non-local in the image space. For a fixed (odd) width  $p$ , a patch  $P_x$  is a subimage of width  $p$ , centered around the pixel  $x$ , and the NLM estimator of  $\mathbf{f}(x)$  is then:

$$\hat{\mathbf{f}}(x) = \frac{\sum_{x' \in \Omega} \omega(x, x') \mathbf{Y}(x')}{\sum_{x' \in \Omega} \omega(x, x')}, \text{ where } \omega(x, x') = \exp\left(-\frac{\|P_x - P_{x'}\|_{2,a}^2}{2h^2}\right), \quad (2)$$

where  $h > 0$  is the bandwidth,  $\|\cdot\|_{2,a}$  is a weighted Euclidean norm in  $\mathbb{R}^{|P|}$  ( $|P| = p^2$ ) using a Gaussian kernel,  $a$  controlling the concentration of the norm around the central pixel. The denominator is a normalizing factor ensuring the weights sum to one. Let us briefly recall the influence of each parameter.

**The bandwidth  $h$**  plays the same role as the bandwidth for kernel methods: the larger the bandwidth, the smoother the image. In [11], the authors set its value according to the quantile of a  $\chi^2$  distribution, due to the metric they consider to compare patches. We adapt this method for our more general shapes.

**The search window size  $\ell$**  determines the pixels to be averaged in Eq. (2). The summation is restricted to an  $\ell \times \ell$  search window  $W$  around the pixel of interest. This was proposed in [9] for computational acceleration. However, some authors have noticed that choosing locally the best search window [11] or using small ones [18, 24] could benefit to the NLM.

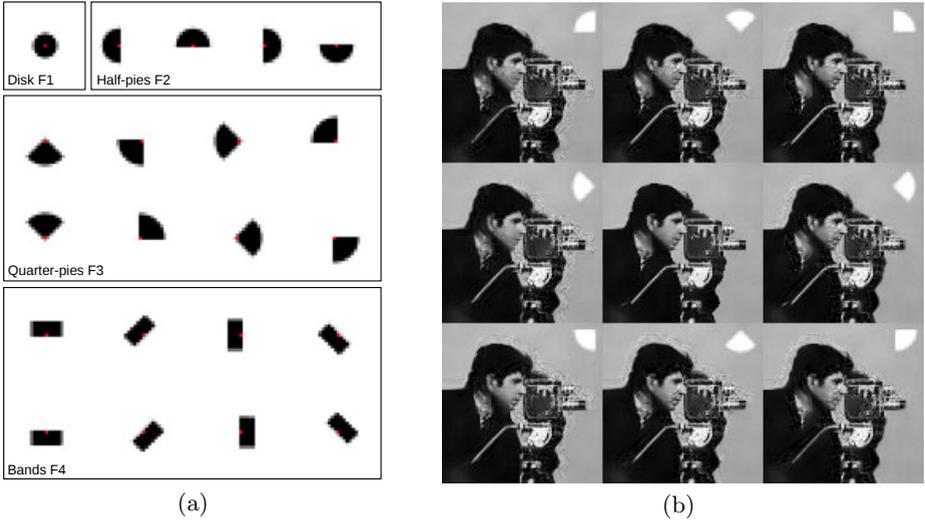
**The patch size  $p$**  is usually set globally (between 5 and 9). Choosing  $p = 1$  would lead to a method close to the Bilateral Filter [5] or Yaroslavsky Filter [4].

### 3 From Patches to Shapes: Beyond the *Rare Patch Effect*

The NLM algorithm suffers from a *noise halo* around edges, due to an abrupt lack of redundancy of the image, sometimes referred to as the *rare patch effect*. It occurs because the NLM has large variance around edges. Several solutions have already been proposed to handle this drawback [16, 18, 25]. We extend the latter two approaches by considering general shapes instead of simple square patches. To deal with arbitrary shapes, we reformulate the way the distance between pixels is measured. We generalize the distance  $\|\cdot\|_{2,a}$  used in Eq. (2) by:

$$d_{\mathbf{S}}^2(x, x') = \sum_{\tau \in \Omega} \mathbf{S}(\tau) (\mathbf{Y}(x + \tau) - \mathbf{Y}(x' + \tau))^2, \quad (3)$$

where  $\mathbf{S}$  encodes the shape we aim at. We can use several shapes, so we need to choose the collection of shapes and a way to take the most of each proposed one. We provide an efficient algorithm to compute the distances in Eq. (3). It relies on



**Fig. 1.** (a) Examples of shapes with the “central” pixel shown in red. Shapes are grouped in four categories: F1. the *disk* family, F2. the *half-pies* family, F3. the *quarter-pies* family and F4. the *bands* family. (b) Eight denoised images obtained for different oriented pie slices. Each denoiser provides good performance in a specific target direction but suffers from *noise halos* in the other directions. The final aggregate (center) takes advantage of every oriented-denoiser to provide high quality restored edges.

the FFT and is independent of the shape  $\mathbf{S}$ . We extend to general shapes, works initiated to speed up the NLM [26, 27] by computing the distances between patches with “Summed Area Tables” (also referred to as “Integral Images”). We modify the original algorithm by swapping the two loops: instead of considering all the shifts for each pixel, we consider all the pixels for each shift (see Fig. 2 for details). This reduces the computational cost from  $O(|W| \cdot |\Omega| \cdot |P|)$  to  $O(|W| \cdot |\Omega| \cdot \log(|\Omega|))$ , where  $|W| = \ell^2$ ,  $|\Omega|$  is the image size and  $|P| = p^2$ .

The main purpose of this paper is to show that the use of different shapes allows to reduce the *rare patch effect*. Another alternative consists in properly handling overlapping square patches. Indeed, we get  $|P|$  estimates for each pixel. In [9, 11], those  $|P|$  estimates are uniformly averaged while a weighted average is performed in [16]. In our framework, these blockwise approaches are equivalent to combine  $|P|$  (possibly) decentered square shapes. Now, the challenge is to find shapes with enough similar candidates in the search window. We have considered new shapes: *disks*, *bands* and *pies* (see Fig 1).

## 4 Aggregation of Shape-Based Estimates

For any pixel  $x$ , we can build a collection of  $K$  pixel estimators  $\hat{\mathbf{f}}_1(x), \dots, \hat{\mathbf{f}}_K(x)$  based on different shapes, as estimates of their corresponding performance. We can now focus on different aggregation procedures.

**Algorithm** 2D-FFT NLM for an arbitrary shape**Inputs:** noisy image  $\mathbf{Y}$ , 2D-FFT of the shape  $\mathcal{F}(\mathbf{S})$ **Parameters:** search window  $W$ , bandwidth  $h$ **Output:** estimated image  $\hat{\mathbf{f}}$ Initialize accumulator images  $\mathbf{A}$  and  $\mathbf{B}$  to zero**for all** shift vector  $\delta$  in the search window  $W$  **do**    Compute  $\Delta_\delta(x) := (\mathbf{Y}(x) - \mathbf{Y}(x + \delta))^2$  for all pixels  $x$     Compute the 2D-FFT  $\mathcal{F}(\Delta_\delta)$     Perform the convolution of  $\Delta_\delta$  by the shape  $\mathbf{S}$ 

$$d_{\mathbf{S}}^2(\cdot, \cdot + \delta) \leftarrow \left( \mathcal{F}^{-1} \left( \overline{\mathcal{F}(\mathbf{S})} \mathcal{F}(\Delta_\delta) \right) \right) (\cdot)$$

**for all** pixels  $x$  in  $\Omega$  **do**

$$\text{Compute } \omega(x, x + \delta) = \exp \left( -\frac{d_{\mathbf{S}}^2(x, x + \delta)}{2h^2} \right)$$

$$\text{Update the accumulators } \mathbf{A}(x) \leftarrow \mathbf{A}(x) + \omega(x, x + \delta) \mathbf{Y}(x + \delta)$$

$$\mathbf{B}(x) \leftarrow \mathbf{B}(x) + \omega(x, x + \delta)$$

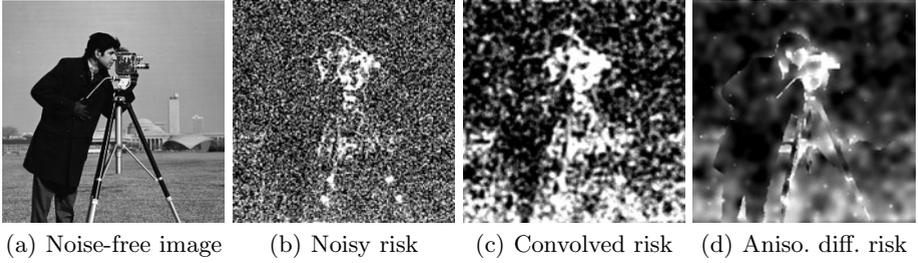
**end for****end for**Final (normalized) estimator  $\hat{\mathbf{f}}(x) = \frac{\mathbf{A}(x)}{\mathbf{B}(x)}$  for all pixel  $x$ 

**Fig. 2.** NLM pseudo-code for an arbitrary patch shape  $\mathbf{S}$ . Pre-computations (2D-FFT) of distances between shapes from the noisy image and shapes from its shifted version leads to a complexity of  $O(|W| \cdot |\Omega| \cdot \log |\Omega|)$ , independent of the shape  $\mathbf{S}$ .

#### 4.1 Classical Methods

**Uniformly weighted aggregation (UWA).** The idea to give the same weight to any shape-based estimator was already proposed for (possibly decentered) square patches in [9, 11], leading to the pixel-estimate  $\hat{\mathbf{f}}_{\text{UWA}}(x)$ . With few shapes it is already an improvement in practice (see Table 2), but as the number of shapes increases, we can take into account irrelevant positions. Moreover, such a procedure still suffers from the *rare patch effect*.

**Variance-based decision, Weighted Average (WAV).** A possible way to limit the *noise halo* is to adapt WAV-reprojection [16] to general shapes. The idea, also proposed by Dabov *et al.* [15] in a different context, is to perform a weighted average of the estimates  $\hat{\mathbf{f}}_1(x), \dots, \hat{\mathbf{f}}_K(x)$ , where each weight is chosen inversely proportional to the (estimated) variance of the corresponding estimator. However, this method tends to over-smooth edges and thin details since it does not consider the bias of each estimator.



**Fig. 3.** Maps of the estimated risk associated with disk shape. From left to right, the noise-free image, the map of the risk without regularization, with convolution and with regularization based on anisotropic diffusion. Low risks are black, high ones are white.

### 4.2 SURE-Based Methods

In [20], a closed-form expression of SURE for the NLM allows to select the best bandwidth  $h$  for the whole image. Our approach is different and closer to the one in [18] (where SURE locally determines the parameter  $h$  and  $p$ ), since we use SURE to locally combine the shape-based estimators. Stein’s Lemma [19] still holds when considering shapes: for the pixel  $x$  and the  $k$ -th shape-based estimate

$$\mathbf{r}_k(x) = (\hat{\mathbf{f}}_k(x) - \mathbf{Y}(x))^2 + 2\sigma^2 \frac{\partial \hat{\mathbf{f}}_k(x)}{\partial \boldsymbol{\varepsilon}(x)} - \sigma^2, \tag{4}$$

is an unbiased estimate of the risk. Thanks to Eq. (2), the derivative is:

$$\frac{\partial \hat{\mathbf{f}}_k(x)}{\partial \boldsymbol{\varepsilon}(x)} = \left( 1 + \sum_{x'} \mathbf{Y}(x') \frac{\partial \omega(x, x')}{\partial \boldsymbol{\varepsilon}(x')} - \left( \frac{\sum_{x'} \mathbf{Y}(x') \omega(x, x')}{C_x} \right) \sum_{x''} \frac{\partial \omega(x, x'')}{\partial \boldsymbol{\varepsilon}(x)} \right) / C_x.$$

where  $C_x = \sum_{x'} \omega(x, x')$ . Our shape-based norm defined in Eq. (3) leads to the following expression of the derivative of the weights  $\omega(x, x')$ :

$$\frac{\partial \omega(x, x')}{\partial \boldsymbol{\varepsilon}(x')} = \frac{\mathbf{S}(0) [\mathbf{Y}(x) - \mathbf{Y}(x')] + \mathbf{S}(x - x') [\mathbf{Y}(x) - \mathbf{Y}(2x - x')]}{h^2}. \tag{5}$$

where  $\mathbf{S}$  encodes the shape of our  $k$ -th shape-based estimator. Combining the last equations leads to unbiased risk estimates  $\mathbf{r}_1(x), \dots, \mathbf{r}_K(x)$  for our  $K$  denoisers.

**Minimizer of the risk estimates (MRE).** A simple proposition is to select the shape that minimizes the local risk estimates we have at hand:

$$\hat{\mathbf{f}}_{\text{MRE}}(x) := \hat{\mathbf{f}}_{k^*}(x) \quad \text{where} \quad k^* = \underset{k}{\operatorname{arg\,min}} \mathbf{r}_k(x). \tag{6}$$

This rule is all the more relevant as the estimators are different. Selecting the locally optimal shape yields satisfying results, but combining some of the best performing estimators may improve the results.



**Fig. 4.** Chosen  $256 \times 256$  noise-free images for our experiments

**Exponentially Weighted Aggregation (EWA).** It might be better to combine several estimators rather than just selecting one. This happens if the best estimators are diversified enough or if the risk of the MRE was wrongly underestimated. Thus, we have used the statistical method of Exponentially Weighted Aggregation, studied for instance in [28] and adapted for patch-based denoising in [29]. It consists in aggregating the estimators by performing a weighted average, with higher weights for estimators with low risks:

$$\hat{\mathbf{f}}_{\text{EWA}}(x) := \sum_{k=1}^K \alpha_k \hat{\mathbf{f}}_k(x), \quad \text{with} \quad \alpha_k = \frac{\exp(-\mathbf{r}_k(x)/T)}{\sum_{k'=1}^K \exp(-\mathbf{r}_{k'}(x)/T)}.$$

The temperature  $T > 0$  is a smoothing parameter that controls the confidence attributed to the risk estimates. If  $T \rightarrow \infty$ , the EWA is simply the uniform aggregate  $\hat{\mathbf{f}}_{\text{UWA}}$  defined before. Conversely, if  $T \rightarrow 0$ , then  $\hat{\mathbf{f}}_{\text{EWA}} \rightarrow \hat{\mathbf{f}}_{\text{MRE}}$ .

The problem of using SURE to take a local decision for each pixel  $x$  is difficult since this estimator has large oscillations (see Fig. 3), so that regularizing the risk maps  $\mathbf{r}_1, \dots, \mathbf{r}_K$  is required.

### 4.3 Regularizing the Risk Maps with Anisotropic Diffusion

To make the risk estimates more robust, it is necessary to regularize it. The convolution of the risk map is an efficient way to estimate the local risk in view of setting  $h$  since on both sides of an edge a large value of  $h$  should be used [18]. Here, the anisotropy of the shapes implies that on one side of an edge the risk may be low whereas it may be high on the other side.

Since convolutions diffuse the risks across the edge, the risk maps become blurred and their comparison becomes difficult. To diffuse the risks on each side of edges, we have adopted a heat equation with spatially and timely dependent coefficients (inspired by the Perona-Malik equation [1]).

More precisely, we let the risk maps  $\mathbf{r}_1, \dots, \mathbf{r}_K$  evolve according to:

$$\begin{cases} \frac{\partial \mathbf{r}_k}{\partial t}(x, t) = \text{div}(g(|\nabla u(x, t)|) \nabla \mathbf{r}_k(x, t)), \\ \mathbf{r}_k(x, 0) = (\hat{\mathbf{f}}_k(x) - \mathbf{Y}(x))^2 + 2\sigma^2 \frac{\partial \hat{\mathbf{f}}_k(x)}{\partial \varepsilon(x)} - \sigma^2, \end{cases} \quad (7)$$

**Table 1.** Gain in using anisotropic or mixture of isotropic and anisotropic shapes in terms of PSNR/SSIM. The studied patch shapes are the isotropic *disks*, the *half-pies*, the *quarter-pies*, the *bands* and some combination of them (see Fig. 1.a).

$\sigma = 20$		Cameraman	City	Windmill	Lake
<i>Disk</i> shapes (F1)		29.45/0.832	28.16/0.885	30.97/0.904	28.68/0.863
<i>Half-pie</i> shapes (F2)		29.43/0.832	28.08/0.886	30.97/0.906	28.60/0.863
<i>Quarter-pie</i> shapes (F3)		29.31/0.831	27.87/0.883	30.95/0.909	28.49/0.862
<i>band</i> shapes (F4)		29.46/0.832	28.05/0.885	31.05/0.906	28.61/0.862
Combination: F1, F2		<b>29.50/0.833</b>	<b>28.21/0.887</b>	31.11/0.907	<b>28.73/0.865</b>
Combination: F1, F2, F3, F4		<b>29.50/0.833</b>	28.20/0.887	<b>31.19/0.909</b>	28.72/0.865

where  $g(x) = \exp(-x^2/\kappa^2)$ , the parameter  $\kappa$  controls the anisotropy of the diffusion (the larger  $\kappa$ , the more isotropic the diffusion), and  $u$  is the smoothed noisy image which jointly evolves using the Perona-Malik equation:

$$\begin{cases} \frac{\partial u}{\partial t}(x, t) = \operatorname{div}(g(|\nabla u(x, t)|)\nabla u(x, t)) , \\ u(x, 0) = \mathbf{Y}(x) . \end{cases} \quad (8)$$

Curiously, we have noticed that we obtain better risk maps by diffusing  $\sqrt{r_k}$  instead of  $r_k$  itself. Figure 3 shows that this regularization procedure provides smooth risk maps, following edges of the underlying noise-free image, and finer than without regularization or with convolution.

## 5 Numerical and Visual Results

The corrupted images are obtained from  $256 \times 256$  images: *cameraman*, *city*, *windmill* and *lake*<sup>1</sup> (Fig. 4). These images are interesting to study since they present highly contrasted edges for which the classical NLM suffers from the *rare patch effect*. In all the experiments, unless otherwise specified, the NLM-SAP is used with the following default parameters: the search window width  $\ell = 11$  px, the shape family combines 15 shapes from families F1 and F2 (Fig. 1.a) with shape areas of 12.5, 25 and 50 px<sup>2</sup>, we use EWA with  $T = 0.02\sigma^2$  and anisotropic risk regularization with 50 iterations, time-step  $dt = 1/8$  and  $\kappa = 30$ . The parameter  $h$  is adapted to the size of the shapes using the rule given by [11]. For the central pixel, we set its central weight as recommended in [24].

Table 1 gives numerical results for different families. The compared families are (see Fig. 1.a): the *disks*, the *half-pies*, the *quarter-pies* and the *bands* and combinations of these families. Our experiments show that suitable families should contain isotropic shapes, directional shapes and various scales of shapes. Increasing the number of shapes does not necessarily improve the quality.

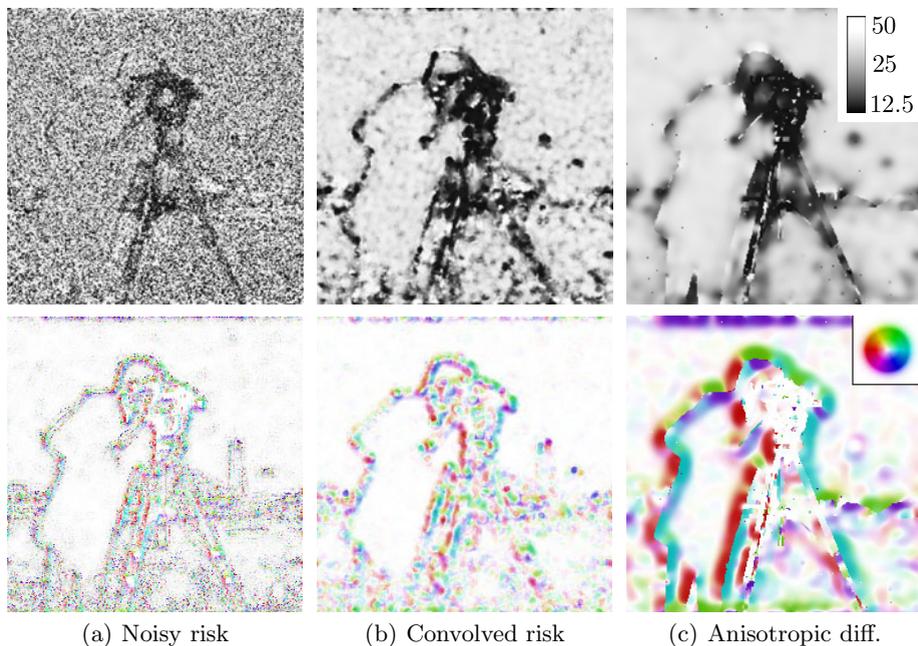
Table 2 presents the numerical performance for the four aggregation procedures: UWA, WAV, MRE and EWA. MRE suffers from brutal transitions, since it selects only one shape per pixel, while EWA evolves in a smoother way due to the weighted combination of shapes for each pixel and provides best results.

<sup>1</sup> Images from L. Condat's database: <http://www.greyc.ensicaen.fr/~lcondat>

**Table 2.** Comparisons of different aggregation procedures in terms of PSNR/SSIM: UWA, WAV, MRE and EWA

$\sigma = 20$	Cameraman	City	Windmill	Lake
UWA	29.40/0.830	27.99/0.880	30.76/0.897	28.53/0.858
WAV	29.46/0.830	27.98/0.879	30.82/0.898	28.48/0.856
MRE	29.33/0.829	28.02/0.885	30.88/0.905	28.58/0.862
EWA	<b>29.50/0.833</b>	<b>28.21/0.887</b>	<b>31.11/0.907</b>	<b>28.73/0.865</b>

We have studied the influence of the regularization of the risk maps on the aggregation results. Three methodologies are compared: aggregation using the noisy risk maps (i.e., SURE maps), the convolved risk maps (using a disk kernel of radius 4) and the risk maps obtained by anisotropic diffusions (Fig. 3). The choices of the local sizes and orientations of the patch shapes are more relevant with the maps obtained by anisotropic diffusions, in terms of scale adaptivity, feature directions and spatial coherency (Fig. 5). Using anisotropic diffusion, the NLM-SAP acts as expected, selecting big sizes of shapes, even around edges, since the shape orientations have been chosen properly to reduce the *rare patch effect*.



**Fig. 5.** (top) Average areas and (bottom) average orientations of selected shapes for different risk maps. From left to right, results using the noisy risk maps, the convolved risk maps and the risk maps obtained by anisotropic diffusions. The average areas and the average orientations are represented using gray level colors.

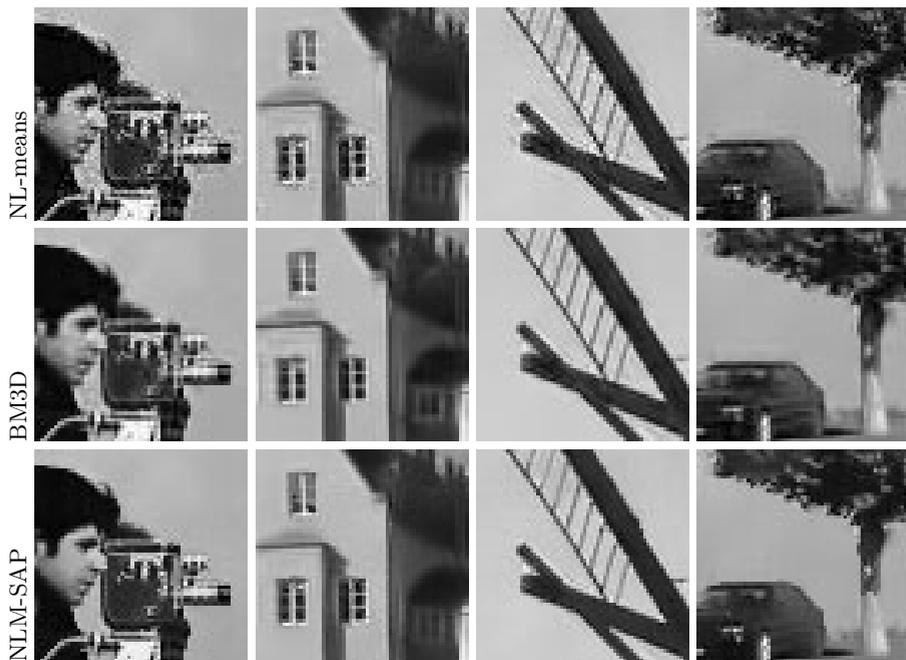
**Table 3.** Comparisons of denoising approaches for various noise levels in terms of PSNR/SSIM: pixelwise NLM [9], blockwise NLM using UWA reprojection [9], blockwise NLM using WAV reprojection [16], pixelwise NL-means using SURE-based adaptive bandwidth selection [18], BM3D denoiser [15], and our proposed NLM-SAP.

	Camerman	City	Windmill	Lake
$\sigma = 5$				
NLM [9]	36.92/0.951	35.87/0.965	38.10/0.972	36.76/0.964
UWA Blockwise NLM [9]	36.99/0.953	35.94/0.966	38.18/0.973	36.77/0.963
WAV Blockwise NLM [16]	37.31/0.956	36.34/0.972	38.79/0.978	37.10/0.970
SURE adaptive NLM [18]	37.46/0.956	36.76/0.975	39.14/0.978	37.28/0.970
BM3D [15]	<b>38.17/0.962</b>	<b>37.48/0.978</b>	<b>39.91/0.983</b>	<b>38.15/0.977</b>
NLM-SAP	37.80/0.957	37.26/0.975	39.60/0.979	37.92/0.974
$\sigma = 10$				
NLM [9]	32.46/0.905	31.11/0.932	33.62/0.945	32.07/0.926
UWA Blockwise NLM [9]	32.43/0.913	30.99/0.926	33.49/0.942	32.04/0.924
WAV Blockwise NLM [16]	32.84/0.922	31.48/0.941	34.07/0.953	32.37/0.936
SURE adaptive NLM [18]	33.11/0.918	32.11/0.948	34.78/0.954	32.61/0.935
BM3D [15]	<b>34.06/0.931</b>	<b>33.15/0.956</b>	<b>35.84/0.966</b>	<b>33.63/0.950</b>
NLM-SAP	33.44/0.914	32.84/0.950	35.28/0.955	33.27/0.940
$\sigma = 20$				
NLM [9]	28.72/0.820	27.11/0.870	30.04/0.897	28.12/0.855
UWA Blockwise NLM [9]	28.88/0.830	27.02/0.868	29.92/0.890	28.14/0.860
WAV Blockwise NLM [16]	29.16/0.838	27.27/0.877	30.17/0.901	28.12/0.865
SURE adaptive NLM [18]	29.49/0.845	27.85/0.889	30.96/0.906	28.46/0.867
BM3D [15]	<b>30.35/0.871</b>	<b>29.07/0.912</b>	<b>32.07/0.936</b>	<b>29.38/0.895</b>
NLM-SAP	29.50/0.833	28.21/0.887	31.11/0.907	28.73/0.865

Comparisons have been performed with the classical (pixelwise) NLM [9], the blockwise NLM using UWA reprojection [9], the blockwise NLM using WAV reprojection [16], the pixelwise NL-means using SURE-based adaptive bandwidth selection [18], BM3D [15], and our proposed NLM-SAP approach. Table 3 shows that NLM-SAP outperforms all other NLM improvements. NLM-SAP brings a gain of PSNR of about 1 dB compared to the classical NLM. The BM3D approach leads to better numerical results than all NLM variants. While the presence of the *rare patch effect* is well illustrated by the *noise halos* for NLM, BM3D and NLM-SAP have reduced a lot this phenomenon. Our NLM-SAP provides smooth results with accurate details: the quality of the images we obtained challenges those by BM3D.

## 6 Conclusion

We have addressed the problem of the *rare patch effect* arising in the NLM and responsible of the *noise halos* around edges. Our method consists in substituting the square patches of fixed size by spatially adaptive shapes. A fast implementation based on the FFT has been proposed to handle arbitrary shapes. Several estimates are obtained by using different patch shapes, and we have extended



**Fig. 6.** Comparisons of the NLM [9], the BM3D [15] and the proposed NLM-SAP on images damaged by additive white Gaussian noise with standard deviation  $\sigma = 20$

SURE-based approaches to aggregate them. The SURE-based risk maps require regularization, and diffusions can be satisfactorily used. Future work is to reduce computation time and treat other regularization strategies.

## References

1. Perona, P., Malik, J.: Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* 12, 629–639 (1990)
2. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations of image processing. *Arch. Rational Mech. Anal.* 123(3), 199–257 (1993)
3. Weickert, J.: Anisotropic diffusion in image processing. European Consortium for Mathematics in Industry. B. G. Teubner, Stuttgart (1998)
4. Yaroslavsky, L.P.: Digital picture processing. Springer Series in Information Sciences, vol. 9. Springer, Berlin (1985)
5. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: *ICCV*, pp. 839–846 (1998)
6. Lepski, O.V., Mammen, E., Spokoiny, V.G.: Optimal spatial adaptation to inhomogeneous smoothness: an approach based on kernel estimates with variable bandwidth selectors. *Ann. Statist.* 25(3), 929–947 (1997)
7. Polzehl, J., Spokoiny, V.G.: Adaptive weights smoothing with applications to image restoration. *J. R. Stat. Soc. Ser. B Stat. Methodol.* 62(2), 335–354 (2000)

8. Katkovnik, V., Foi, A., Egiazarian, K.O., Astola, J.T.: Directional varying scale approximations for anisotropic signal processing. In: EUSIPCO, pp. 101–104 (2004)
9. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.* 4(2), 490–530 (2005)
10. Awate, S.P., Whitaker, R.T.: Unsupervised, information-theoretic, adaptive image filtering for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* 28(3), 364–376 (2006)
11. Kervrann, C., Boulanger, J.: Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Process.* 15(10), 2866–2878 (2006)
12. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing over-complete dictionaries for sparse representation. *IEEE Trans. Signal Process.* 54(11), 4311–4322 (2006)
13. Mairal, J., Sapiro, G., Elad, M.: Learning multiscale sparse representations for image and video restoration. *Multiscale Model. Simul.* 7(1), 214–241 (2008)
14. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Non-local sparse models for image restoration. In: ICCV (2009)
15. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.O.: Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.* 16(8), 2080–2095 (2007)
16. Salmon, J., Strozeccki, Y.: From patches to pixels in non-local methods: Weighted-Average reprojection. In: ICIP (2010)
17. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.O.: BM3D image denoising with shape-adaptive principal component analysis. In: Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations, SPARS 2009 (2009)
18. Duval, V., Aujol, J.F., Gousseau, Y.: On the parameter choice for the non-local means. Technical Report hal-00468856, HAL (2010)
19. Stein, C.M.: Estimation of the mean of a multivariate normal distribution. *Ann. Statist.* 9(6), 1135–1151 (1981)
20. Van De Ville, D., Kocher, M.: SURE-based Non-Local Means. *IEEE Signal Process. Lett.* 16, 973–976 (2009)
21. Blu, T., Luisier, F.: The SURE-LET approach to image denoising. *IEEE JIP* 16(11), 2778–2786 (2007)
22. Ramani, S., Blu, T., Unser, M.: Monte-Carlo SURE: a black-box optimization of regularization parameters for general denoising algorithms. *IEEE Trans. Image Process.* 17(9), 1540–1554 (2008)
23. Donoho, D.L., Johnstone, I.M.: Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Statist. Assoc.* 90(432), 1200–1224 (1995)
24. Salmon, J.: On two parameters for denoising with Non-Local Means. *IEEE Signal Process. Lett.* 17, 269–272 (2010)
25. Zimmer, S., Didas, S., Weickert, J.: A rotationally invariant block matching strategy improving image denoising with non-local means. In: LNLA (2008)
26. Wang, J., Guo, Y.W., Ying, Y., Liu, Y.L., Peng, Q.S.: Fast non-local algorithm for image denoising. In: ICIP, pp. 1429–1432 (2006)
27. Darbon, J., Cunha, A., Chan, T.F., Osher, S., Jensen, G.J.: Fast nonlocal filtering applied to electron cryomicroscopy. In: ISBI, pp. 1331–1334 (2008)
28. Dalalyan, A.S., Tsybakov, A.B.: Aggregation by exponential weighting, sharp pac-bayesian bounds and sparsity. *Mach. Learn.* 72(1-2), 39–61 (2008)
29. Salmon, J., Le Pennec, E.: NL-Means and aggregation procedures. In: ICIP, pp. 2977–2980 (2009)