# A NEW DENOISING METHOD IN HIGH DIMENSIONAL PCA-SPACE

Quoc Bao DO, Azeddine BEGHDADI and Marie LUONG

L2TI, Université Paris 13
99 avenue Jean-Baptiste Clément, 93430 Villetaneuse, France

## ABSTRACT

Kernel-design based method such as Bilateral filter (BIL), non-local means (NLM) filter is known as one of the most attractive approaches for denoising. We propose in this paper a new noise filtering method inspired by BIL, NLM filters and principal component analysis (PCA). The main idea here is to perform the BIL in a multidimensional PCA-space using an anisotropic kernel. The filtered multidimensional signal is then transformed back onto the image spatial domain to yield the desired enhanced image. We will show that the proposed method is a generation of all kernel-design based methods. The obtained results are highly promising.

**Keywords:** Denoising; Bilateral filter; Non-local means; High dimensional space; PCA.

## 1. INTRODUCTION

Image denoising is an important problem in image and signal processing. Many methods share the same basic idea: denoising is carried out by averaging similar pixels. These methods are based on the observation that any image often contains self-similarity and some spatial redundancy. If the noise is considered as an independent and identically distributed (i.i.d.) random signal, it could be smoothed out by averaging similar pixels. The main issue of this approach is how to define similar pixels for a given one. The simplest one is Gaussian (GAU) filter which considers the neighbors as similar pixels. It is clear that this method fails at edge and texture regions. Another method called Unilateral filter (UNI)[1] uses intensity information to estimate the similarity between pixels. Bilateral filter (BIL)[2] combines GAU and UNI to define similar pixels for a given one. It takes into account both spatial and intensity information. The relation between BIL and anisotropic filtering has been investigated in [3, 4]. Another adaptive filtering approach, called Non-Local Means (NLM),[5,6] has been recently proposed. Unlike pixel-based similarity in GAU, UNI and BIL, NLM proposes to use patch-based similarity which makes the method more robust in textured and contrasted regions. Many methods for improving the performance of NLM have been proposed. The fast NLM (FNLM) is presented in [7]. NLM in the wavelet domain is introduced in [8]. In [9], the authors propose a transform which maps each patch in the image domain to a point in a high dimensional space called patch-space and show that NLM algorithm is a variant of an isotropic filter in this new space. In this paper, we propose to use principal component analysis (PCA) to reduce the dimensionality of the patch-space and then form another one called High Dimensional PCA-space (HDPCA) from the most significant components. Similar to the work in [9], FNLM can be drawn as a variant of an isotropic filter in the HDPCA-space. In order to improve the denoising performance, instead of using this isotropic filter, we propose to use BIL, i.e. an anisotropic filter, in the HDPCA-space.

The paper is organized as follows: section 2 is devoted for a review of related works, the proposed method is described in section 3 followed by experimental results in section 4. The conclusions are finally given in section 5.

Further author information: (Send correspondence to Quoc Bao DO)
Quoc Bao DO: E-mail: do.quocbao@univ-paris13.fr, Telephone: +33 (0)1 49 40 20 72
Azeddine BEGHDADI: E-mail: azeddine.beghdadi@univ-paris13.fr, Telephone: +33 (0)1 49 40 40 57
Marie LUONG: E-mail: marie.luong@univ-paris13.fr, Telephone: +33 (0)1 49 40 40 64

# 2. RELATED WORKS

## 2.1 Kernel design based method: from pixel-based to patch-based

Let us define a 2D noise-free image $u : R^2 \rightarrow R$. Its noisy version $v$ at pixel $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2)$ defined as $v(\mathbf{x}) = u(\mathbf{x}) + n(\mathbf{x})$ where $n$ is identical, independent Gaussian noise. The aim of denoising is to estimate $u$ from $v$. One of well-known approaches to filter out independent and identically distributed (i.i.d) Gaussian noise is to take the average of similar pixels. The basic idea behind the success of this approach is to exploit the spatial redundant information in the image. An unifying formula for these methods could be expressed as follows:

$$\widehat{u}(\mathbf{x}) = \frac{\sum\limits_{\mathbf{y} \in \Omega} w(\mathbf{x}, \mathbf{y}) v(\mathbf{y})}{\sum\limits_{\mathbf{y} \in \Omega} w(\mathbf{x}, \mathbf{y})} \tag{1}$$

where $\widehat{u}$ is restored pixel, $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2)$ and $\mathbf{y} = (\mathrm{y}_1, \mathrm{y}_2)$ are pixel's coordinates, $w(\mathbf{x}, \mathbf{y})$ is similarity measure between $v(\mathbf{x})$ and $v(\mathbf{y})$. The key issue is how to determine similar pixels for a given one, i.e. how to calculate $w(\mathbf{x}, \mathbf{y})$. Indeed, each method proposes a kernel to estimate this value. The simplest one is Gaussian filter (GAU) which define similar pixels in function of geometric distance. If closer a pixel is, more similar it is, then more weight it has:

$$w_{GAU}(\mathbf{x}, \mathbf{y}) = exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{h_s^2}\right) \tag{2}$$

where $h_s$ is the spatial parameter. However, it is well known that this weight fails at contour and texture regions. To overcome this drawback some methods have been proposed to adapt the filtering strength to the local content of the image. The first one is proposed by Yaroslavsky called unilateral filter (UNI):[1]

$$w_{UNI}(\mathbf{x}, \mathbf{y}) = exp\left(-\frac{\|v(\mathbf{x}) - v(\mathbf{y})\|^2}{h_r^2}\right) \tag{3}$$

where $h_r$ is the range parameter. Note that UNI filter is data dependent. Instead to consider the geometric distance between pixels, it takes into account the intensity. Unlike GAU filter, this filter can well distinguish pixels in contour regions. We can note that this type of filter average values that are similar and potentially can be far away in the image. Another method called Bilateral filter (BIL) is proposed Tomasi et al.[2] in 1998. This filter simply unifies GAU and UNI ones:

$$w_{BIL}(\mathbf{x}, \mathbf{y}) = exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{h_s^2}\right) exp\left(-\frac{\|v(\mathbf{x}) - v(\mathbf{y})\|^2}{h_r^2}\right) \tag{4}$$

This method is based on an average of pixels, where the weights are the product of two terms: the geometric difference and the photometric difference of two pixel. It is worth noticing that the above filters (GAU, UNI, BIL) are pixel-based to measure the similarity. Consequently they do not take into account the contextual and spatial information. To cope with this limitation, Non-Local Means filter (NLM) uses patch-based similarity approach.[5,6] Its kernel is defined as follows:

$$w_{NLM}(\mathbf{x}, \mathbf{y}) = exp\left(-\frac{G_a * \|\mathbf{N}(\mathbf{x}) - \mathbf{N}(\mathbf{y})\|^2}{h_r^2}\right) \tag{5}$$

where $\mathbf{N}(\mathbf{x})$ is small patch $r \times r$ around the pixel $\mathbf{x}$, $G_a$ is a Gaussian kernel with standard deviation $a$ of the same size as $\mathbf{N}(\mathbf{x})$. Note that, the patch-similarity measure is weighted by $G_a$ to give more weight to pixels close to the patch center. The equation(5) can be rewritten as follows:

$$w_{NLM}(\mathbf{x}, \mathbf{y}) = exp\left(\frac{-\|\boldsymbol{N}_a(\mathbf{x}) - \boldsymbol{N}_a(\mathbf{y})\|^2}{h_r^2}\right) \tag{6}$$

where the weighted patch $\boldsymbol{N}_a(\mathbf{x}) = \sqrt{G_a}\boldsymbol{N}(\mathbf{x})$. Note that NLM considers only intensity information. Table 1 gives a resume of the aforementioned filters

| Method | GAU | UNI | BIL | NLM |
|---|---|---|---|---|
| Spatial Neighbors | Yes |  | Yes |  |
| Range Neighbors |  | Yes | Yes | Yes |
| Pixel-based similarity | Yes | Yes | Yes |  |
| Patch-based similarity |  |  |  | Yes |

Table 1. Brief description of filter types

## 2.2 Closest space VS Closest structure

For NLM method, it has been shown that if the search range $\Omega$ is the whole image (non-local approach in strict sense), not only computational time increases but also the quality of the restored image reduces because many irrelevant patches are taken into account. The search range should be limited in a small window $\Omega_{\mathbf{x}} = S \times S$ around the pixel being processed $\mathbf{x}$. In the initial configuration, Buades et al. propose a window of size $21 \times 21$ which results of total 441 patches. Each approach has a weakness and an advantage. The non-local approach can get more relevant patch but many mismatching patches are considered. On the other hand, the semi non-local can discard mismatching patch but many relevant patches can not be used. In other work,[10] we verify the third approach which seeks in the whole image, rid of all irrelevant patches and takes into account only the best matching ones. This approach is called "*closest structure*" in the sense that only the best matching "structure" patches are considered. The limited search range approach (the original NLM) is called "*closest space*" in the sense that it uses only neighbour patches. Counter intuitively, the closest structure approach yields worse result in both term of subjective and objective measurement. An example is shown in Figure 1 where the noise still remains in the case of the closest structure approach. In flat regions, the noise pattern of a given-patch will match well with that of the best candidates. Averaging these similar noise patterns cannot effectively remove the noise. We refer to this as "best-worst paradox" in the sense that if we consider only the best candidates, the result is the worst. Based on these remarks, the semi non-local approach, i.e. restrained to a small window $\Omega_{\mathbf{x}} = S \times S$, is used in this work.

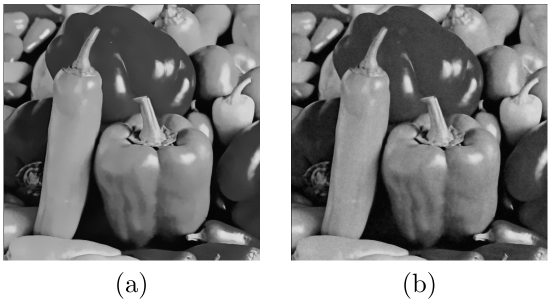


Figure 1. Best-worst paradox: if we consider only the best matching patches, the result is the worst (a) Result of the closest space approche with PSNR = 33.85 (b) Result of the closest structure approche with PSNR = 32.75

## 2.3 Fast Non-Local Means

A fast NLM (FNLM) filter[7] is proposed by approximating the distance $\|\boldsymbol{N}_a(\mathbf{x}) - \boldsymbol{N}_a(\mathbf{y})\|^2$ in (6) by another one estimated from projections of $\boldsymbol{N}_a$ onto a subspace defined by Principal Component Analysis (PCA). It is well known that the eigenvectors $\{\boldsymbol{e}_m\}_{m=1}^{r^2}$ (sorted in order of descending eigenvalues) of the covariance matrix $\mathbf{M}$ estimated from a set of all weighted patch $\boldsymbol{N}_a$ form an orthonormal basis. Note $\boldsymbol{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_{r^2}(\mathbf{x})]^T$ is projected vector of $\boldsymbol{N}_a(\mathbf{x})$ onto this orthonormal basis, i.e. $f_m(\mathbf{x}) =< \boldsymbol{N}_a(\mathbf{x}), \boldsymbol{e}_m >$ where $<,>$ stands for inner product. As the signal energy concentrates on a few the most significant $d$ components ($d \ll r^2$), Tasdizen[7] proposes to approximate the norm $\|\boldsymbol{N}_a(\mathbf{x}) - \boldsymbol{N}_a(\mathbf{y})\|^2$ by using only these $d$ components, i.e.

$$\|\boldsymbol{N}_a(\mathbf{x}) - \boldsymbol{N}_a(\mathbf{y})\|^2 \approx \|\boldsymbol{F}^d(\mathbf{x}) - \boldsymbol{F}^d(\mathbf{y})\|^2 = \sum_{m=1}^{d} \|f_m(\mathbf{x}) - f_m(\mathbf{y})\|^2 \qquad (7)$$

where $\boldsymbol{F}^d(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_d(\mathbf{x})]^T$. The new weight is now defined as follows:

$$w^d_{NLM}(\mathbf{x},\mathbf{y}) = exp\left(\frac{-\|\boldsymbol{F}^d(\mathbf{x}) - \boldsymbol{F}^d(\mathbf{y})\|^2}{h_r^2}\right) \tag{8}$$

Finally, the FNLM is given by:

$$\widehat{u}^d(\mathbf{x}) = \frac{\sum_{\mathbf{y}\in\Omega} w^d_{NLM}(\mathbf{x},\mathbf{y})v(\mathbf{y})}{\sum_{\mathbf{y}\in\Omega} w^d_{NLM}(\mathbf{x},\mathbf{y})} \tag{9}$$

where $d$ is a parameter of the algorithm. Recall that when $d = r^2$, FNLM tends to the classical NLM. Indeed, the use of PCA has twofold: (i) the computational complexity is highly reduced, (ii) patch similarity measure improves robustness to noise.

In the next sections, we present a new high dimensional space called HDPCA and show that FNLM and NLM are simply derived from an isotropic filter in this space.

## 3. PROPOSED METHOD

### 3.1 High Dimensional PCA-Space

**Mapping in the HDPCA-space:** First, each small patch is passed through the PCA system to obtain the corresponding projected vectors $\boldsymbol{F}^d$. We define $D$-dimensional HDPCA-space ($D = d+2$) noted $\Psi_D \in R^D$ where the coordinates $\mathbf{p}$ of each point in this space contains both spatial information $\mathbf{x} = (\mathrm{x}_1, \mathrm{x}_2)$ and all value of $\boldsymbol{F}^d$. Precisely, $\mathbf{p} = [\alpha\mathrm{x}_1, \alpha\mathrm{x}_2, f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_d(\mathbf{x})]$ where $\alpha \geq 0$ is factor to balance the importance between spatial and intensity features. By this definition, each patch in the image domain corresponds to a point in the new high dimensional space. From now, we note $\mathbf{p}$ and $\mathbf{q}$ as two points in the HDPCA which correspond to two patches $\mathbf{N}(\mathbf{x})$ and $\mathbf{N}(\mathbf{x})$, respcetively. Each value $\mathbf{V}$ of a point $\mathbf{p}$ in this space is defined as follows:

$$\mathbf{V}(\mathbf{p}) = (V_1(\mathbf{p}), V_2(\mathbf{p})) = (v(\mathbf{x}), 1) \tag{10}$$

Note that $\mathbf{V}(\mathbf{p})$ contains two components:

- The first one $V_1(\mathbf{p}) = v(\mathbf{x})$ (the gray level of the center pixel $\mathbf{x}$ of the patch $\mathbf{N}(\mathbf{x})$).
- The second one $V_2(\mathbf{p})$ is always set equal to 1.

**Back-projection on the image domain:** Instead of filtering directly the pixel value in the image domain, we alter the multi-values $\mathbf{V}(\mathbf{p})$ in the HDPCA-space to obtain $\widehat{\mathbf{U}}(\mathbf{p})$ (the filtering method in this space will be discussed in the next section). This filtered value is then transformed back onto the image domain as follows:

$$\widehat{u}(\mathbf{x}) = \frac{\widehat{U_1}(\mathbf{p})}{\widehat{U_2}(\mathbf{p})} \tag{11}$$

Note that HDPCA is a sparse space where only points corresponding to the patches of the image are defined.

### 3.2 Bilateral In High Dimensional PCA-Space

To restore the pixel $\mathbf{x}$, in order to avoid *"best-worst paradox"* phenomenon, instead of projecting all patches in the image domain onto the HDPCA-space, we project only the patches on the sub-domain $\Omega(\mathbf{x})$ (small windows of size 21 × 21 around the being processed pixel $\mathbf{x}$) and carry out the filtering on these projected values. Since all values in projected vector $\boldsymbol{F}^d(\mathbf{x})$ become spatial coordinates of point $\mathbf{p}$ in the HDPCA-space, we can rewrite equation (9) of FNLM as follows (in the case where $\alpha = 0$):

$$\widehat{u}^d(k,l) = \frac{\widehat{U}_1(\mathbf{p})}{\widehat{U}_2(\mathbf{p})} = \frac{\sum_{\mathbf{q}\in\Psi_D} exp\left(\frac{-\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) V_1(\mathbf{q})}{\sum_{\mathbf{q}\in\Psi_D} exp\left(\frac{-\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) V_2(\mathbf{q})} \tag{12}$$

Note that both nominator and denominator of this equation can be interpreted as Gaussian filter in the HDPCA-space. Therefore, we can summarize FNLM in the two following steps:

- Step 1: Gaussian filtering in the HDPCA-space
- Step 2: Projection back onto the image space by using the division of two components of the filtered values (equation (11))

It is worth to notice that the Gaussian filter in the first step is an isotropic filter. Here, we propose to replace it by an anisotropic one. In the literature, there are many anisotropic diffusion methods such as Total Variation,[11] Perona-Malik[12] which mimic physical processes by locally diffusing pixel values along the image structure. Since these methods are local-based, their adaptation to a such sparse HDPCA-space is rather a difficult task. However, as discussed above, BIL acts as an anisotropic filter and it works in non-local manner therefore it could be used. The proposed method (called BIL-HD) consists of the two following steps:

- Step 1: Bilateral filtering in the HDPCA-space
- Step 2: Projection back onto the image space by using the division of two components of the filtered values (equation (11))

In the first step, the filtered values $\widehat{\mathbf{U}}(\mathbf{p})$ are given by:

$$\widehat{U_\eta}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in\Psi_D} w_\eta(\mathbf{p},\mathbf{q}) V_\eta(\mathbf{q})}{\sum_{\mathbf{q}\in\Psi_D} w_\eta(\mathbf{p},\mathbf{q})} \tag{13}$$

where subscript $\eta = 1, 2$, and according to BIL's principle the weight $w_\eta(\mathbf{p},\mathbf{q})$ is estimated as follows:

$$w_\eta(\mathbf{p},\mathbf{q}) = exp\left(\frac{-\|V_\eta(\mathbf{p}) - V_\eta(\mathbf{q})\|^2}{h^2}\right) exp\left(\frac{-\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) \tag{14}$$

where the first term is an intensity proximity measure and the second one stands for a geometric proximity measure, $h$ is range parameter in the new HDPCA-space. The filtered values $\widehat{\mathbf{U}}(\mathbf{p})$ are finally projected back into the image domain using the division in equation (11). Since $V_2(\mathbf{p}) = 1$ for all the defined points in the HDPCA-space (see equation (10)), it is easy to see that the second filtered value $\widehat{U_2}(\mathbf{p}) = 1\ \forall \mathbf{p}$. Consequently, the restored value is as follows:

$$\widehat{u}(\mathbf{x}) = \widehat{U_1}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in\Psi_D} w_1(\mathbf{p},\mathbf{q}) V_1(\mathbf{q})}{\sum_{\mathbf{q}\in\Psi_D} w_1(\mathbf{p},\mathbf{q})} \tag{15}$$

From the definition of HDPCA in section 3.1, the weight $w_1(\mathbf{p},\mathbf{q})$ can be rewritten as follows:

$$w_1(\mathbf{p},\mathbf{q}) = exp\left(\frac{-\|v(\mathbf{x}) - v(\mathbf{y})\|^2}{h^2}\right) exp\left(\frac{-\|\alpha\mathbf{x} - \alpha\mathbf{y}\|^2 - \|\boldsymbol{F}^d(\mathbf{x}) - \boldsymbol{F}^d(\mathbf{y})\|^2}{h_r^2}\right) \tag{16}$$

By noting $h_s = h_r/\alpha$, equation (16) becomes:

$$w_1(\mathbf{p},\mathbf{q}) = exp\left(\frac{-\|v(\mathbf{x}) - v(\mathbf{y})\|^2}{h^2}\right) exp\left(\frac{-\|\boldsymbol{F}^d(\mathbf{x}) - \boldsymbol{F}^d(\mathbf{y})\|^2}{h_r^2}\right) exp\left(\frac{-\|\mathbf{x}-\mathbf{y}\|^2}{h_s^2}\right) \tag{17}$$

From the equation (17), it is easy to see that the proposed method is a generation of all aforementioned filters (FNLM, NLM, BIL, UNI, GAU). If $h = \infty$, $h_s = \infty$ and $d < r^2$ the proposed method becomes the FNLM filter. If $h = \infty$, $h_s = \infty$ and $d = r^2$ the proposed method tends to the NLM filter. If $h_r = \infty$, the proposed method becomes the BIL filter. If $h_r = \infty$ and $h_s = \infty$, it is the case of the UNI filter. Table 2 gives a resume about the parameters of all methods.

|  | $h$ | $h_r$ | $h_s$ | $d$ |
|---|---|---|---|---|
| GAU | $h = \infty$ | $h_r = \infty$ | $h_s < \infty$ |  |
| UNI | $h < \infty$ | $h_r = \infty$ | $h_s = \infty$ |  |
| BIL | $h < \infty$ | $h_r = \infty$ | $h_s < \infty$ |  |
| NLM | $h = \infty$ | $h_r < \infty$ | $h_s = \infty$ | $d = r^2$ |
| FNLM | $h = \infty$ | $h_r < \infty$ | $h_s < \infty$ | $d \leq r^2$ |
| BIL-HD | $h < \infty$ | $h_r < \infty$ | $h_s < \infty$ | $d \leq r^2$ |

Table 2. Brief description of the parameters of all methods

## 4. EXPERIMENTAL RESULTS

The experimental results are carried out on several natural images such as Barbara, Lena, Peppers and Fingerprint of size 512 × 512. The last one is typical of highly textured image whereas the third one contains mostly homogenous regions. The first and second images contain different types of features, texture, sharp edges and smooth regions. These images are perturbed by additive, independent Gaussian noise at two levels of standard deviation $\sigma = 10$ and $\sigma = 25$. The subspace $\Omega_{\mathbf{x}}$ is defined by small windows 21 × 21 around the being processed pixel $\mathbf{x}$. The patch size is equal to $7 \times 7$ which results of full dimension $r^2$ =49. The reduced dimension $d$ is tested with 11 values: 1, 3, 6, 8, 10, 15, 20, 25, 30, 40, 49. $h_r$ is set equal to $n_{h_r}\sigma$ where $n_{h_r} = [0.6 : 0.1 : 1.4]$ (here we use Matlab notation), $h = n_h\sigma$ with $n_h = [2 : 2 : 20, \infty]$ and $h_s = [1 : 1 : 10, \infty]$. A comparative evaluation using both objective and subjective measures has been performed to demonstrate the advantages of the proposed method over NLM and FNLM filters. To objectively evaluate the results, beside PSNR, we use also two other metrics namely MAD[13] and PSNR$_W$[14] which are based the human visual system (HVS). Note that while small value of MAD indicates high level of image quality, small value of PSNR$_W$, PSNR corresponds to a low level of image quality. Only *the best results* of these methods are reported in Tables 3-10 with the corresponding optimal parameters $d$, $n_h$, $n_{h_r}$ and $h_s$. Note that, for each method, each metric results in different optimal parameters. As can be seen, the proposed method outperforms the others methods for all cases and it is confirmed by all metrics. It is also worth to note that, in many cases, BIL-HD can achieve better quality with smaller dimension $d$ compared to FNLM (for example, for Fingerprint image, $\sigma = 10$, optimal $d$ for FNLM is 15 whereas in our case, this value is 6 which makes BIL-HD 2 times faster than FNLM - see table 12). In the case of $\sigma = 25$, except Fingerprint image, the proposed method achieves the best result at full dimension $d = 49$. Note that for these cases, if we reduce the $d$ to the that of FNLM, the quality is sightly decreased but still better than FNLM (see Table 11)

For the subjective comparison, an example of Lena image is presented in Figure 2 and 3 for the case $\sigma = 10$ and $\sigma = 25$, respectively. As can be seen in the differences between the restored images and the noisy one, the proposed method almost eliminates only noise whereas NLM and FNLM remove both noise and image details. (*please use your monitor to view all images in this paper*). More results can be found in `http://www-l2ti.univ-paris13.fr/~do/spie2011`

## 5. CONCLUSIONS

In this paper, a new anisotropic filtering method in High Dimensional PCA-space is proposed. Through this study, it has been shown that NLM and FNLM can be expressed as an isotropic filter in this space. A series of tests has been performed to assess the efficiency of the proposed method. The obtained results demonstrate the efficiency of the proposed filtering approach objectively and subjectively.

## REFERENCES

[1] Yaroslavsky, L., "Digital picture processing.," *Springer Series in Information Sciences. Springer-Verlag, Berlin* (1985).

[2] Tomasi, C. and Manduchi, R., "Bilateral filtering for gray and color images.," *IEEE International Conference on Computer Vision ICCV'98* , 839–846 (1998).

[3] Barash, D. and Comaniciu, D., "A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift.," *Image Vision Comput.* , 73–81 (2004).

[4] Elad, M., "On the origin of the bilateral filter and ways to improve it.," *IEEE Transactions on Image Processing* , 1141–1151 (2002).

[5] Buades, A., Coll, B., and Morel, J.-M., "Nonlocal image and movie denoising.," *International Journal of Computer Vision* , 123–139 (2008).

[6] Buades, A., Coll, B., and Morel, J., "A non-local algorithm for image denoising," *Conference on Computer Vision and Pattern Recognition CVPR'05* **2**, 60–65 (2005).

[7] Tasdizen, T., "Principal neighborhood dictionaries for nonlocal means image denoising.," *IEEE Transactions on Image Processing* , 2649–2660 (2009).

[8] Souidene, W., Beghdadi, A., and Abed-Meraim, K., "Image denoising in the transformed domain using non local neighborhood," *In Proc. ICASSP* (2006).

[9] Tschumperlé, D. and Brun, L., "Image denoising and registration by pde's on the space of patches," *International Workshop on Local and Non-Local Approximation in Image Processing (LNLA'08)* (2008).

[10] Do, Q. B., Beghdadi, A., and Luong, M., "Combination of closest space and closest structure to ameliorate non-local means method," *IEEE Symposium on Computational Intelligence for Multimedia, Signal and Vision Processing* (2011).

[11] Rudin, L. I., Osher, S., and Fatemi, E., "Nonlinear total variation based noise removal algorithms," *Physica D* **60**, 259–268 (1992).

[12] Perona, P. and Malik, J., "Scale-space and edge detection using anisotropic diffusion.," *IEEE Trans. Pattern Anal. Mach. Intell.* , 629–639 (1990).

[13] Larson, E. C. and Chandler, D. M., "Most apparent distortion: full-reference image quality assessment and the role of strategy," *Journal of Electronic Imaging* **19** (2010).

[14] Beghdadi, A. and Pesquet-Popescu, B., "A new image distortion measure based wavelet decomposition," *In Proc. ISSPA* , 485–488 (2003).

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.8 | $\infty$ | | 49 | 34.27 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 10 | 34.36 |
| | BIL-HD | 1.3 | 6 | 3 | 49 | **34.62** |
| PSNR$_W$ | NLM | 0.8 | $\infty$ | | 49 | 17.39 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 10 | 17.55 |
| | BIL-HD | 0.9 | 4 | 5 | **8** | **17.82** |
| MAD | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 1.66 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 10 | 1.59 |
| | BIL-HD | 0.9 | 6 | 6 | **8** | **1.40** |

Table 3. Objective measures of Lena image $\sigma = 10$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 29.77 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 15 | 29.79 |
| | BIL-HD | 1 | $\infty$ | 4 | 49 | **30.38** |
| PSNR$_W$ | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 12.19 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 15 | 12.23 |
| | BIL-HD | 0.8 | $\infty$ | 5 | 15 | **12.65** |
| MAD | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 7.06 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 15 | 7.00 |
| | BIL-HD | 1 | $\infty$ | 5 | 49 | **6.03** |

Table 4. Objective measures of Lena image $\sigma = 25$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.9 | $\infty$ | $\infty$ | 49 | 31.00 |
| | FNLM | 0.9 | $\infty$ | $\infty$ | 15 | 31.22 |
| | BIL-HD | 1.4 | 6 | 1 | **6** | **31.74** |
| PSNR$_W$ | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 19.14 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 15 | 19.38 |
| | BIL-HD | 1.4 | 6 | 1 | **6** | **19.95** |
| MAD | NLM | 0.6 | $\infty$ | $\infty$ | 49 | 0.22 |
| | FNLM | 0.6 | $\infty$ | $\infty$ | 8 | 0.15 |
| | BIL-HD | 1 | 8 | 2 | **6** | **0.09** |

Table 5. Objective measures of Fingerprint image $\sigma = 10$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 26.60 |
| | FNLM | 0.6 | $\infty$ | $\infty$ | 6 | 27.44 |
| | BIL-HD | 0.6 | 12 | $\infty$ | **6** | **27.45** |
| PSNR$_W$ | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 13.99 |
| | FNLM | 0.6 | $\infty$ | $\infty$ | 6 | 14.61 |
| | BIL-HD | 0.6 | 8 | 10 | **6** | **14.65** |
| MAD | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 2.31 |
| | FNLM | 0.6 | $\infty$ | $\infty$ | 6 | 1.59 |
| | BIL-HD | 0.6 | 10 | 10 | **6** | **1.52** |

Table 6. Objective measures of Fingerprint image $\sigma = 25$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 34.33 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 10 | 34.46 |
| | BIL-HD | 1.2 | 4 | 4 | **6** | **34.81** |
| PSNR$_W$ | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 17.77 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 10 | 18.00 |
| | BIL-HD | 1.1 | 4 | 4 | **6** | **18.41** |
| MAD | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 2.32 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 10 | 2.15 |
| | BIL-HD | 1.1 | 4 | 5 | **6** | **1.87** |

Table 7. Objective measures of Peppers image $\sigma = 10$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.9 | $\infty$ | $\infty$ | 49 | 30.23 |
| | FNLM | 0.9 | $\infty$ | $\infty$ | 15 | 30.25 |
| | BIL-HD | 1.1 | $\infty$ | 4 | 49 | **30.86** |
| PSNR$_W$ | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 12.78 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 15 | 12.89 |
| | BIL-HD | 0.9 | $\infty$ | 5 | 15 | **13.49** |
| MAD | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 8.15 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 15 | 8.03 |
| | BIL-HD | 1 | $\infty$ | 5 | 49 | **7.05** |

Table 8. Objective measures of Peppers image $\sigma = 25$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.9 | $\infty$ | $\infty$ | 49 | 33.76 |
| | FNLM | 0.9 | $\infty$ | $\infty$ | 20 | 33.86 |
| | BIL-HD | 1 | 6 | 7 | 20 | **34.00** |
| PSNR$_W$ | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 17.21 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 20 | 17.34 |
| | BIL-HD | 1 | 6 | 6 | **20** | **17.52** |
| MAD | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 1.54 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 15 | 1.37 |
| | BIL-HD | 0.9 | 8 | 6 | **15** | **1.27** |

Table 9. Objective measures of Barbara image $\sigma = 10$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| PSNR | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 28.70 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 49 | 28.70 |
| | BIL-HD | 0.8 | $\infty$ | 7 | 49 | **28.86** |
| PSNR$_W$ | NLM | 0.7 | $\infty$ | $\infty$ | 49 | 11.84 |
| | FNLM | 0.7 | $\infty$ | $\infty$ | 49 | 11.84 |
| | BIL-HD | 0.8 | $\infty$ | 6 | 49 | **12.05** |
| MAD | NLM | 0.8 | $\infty$ | $\infty$ | 49 | 7.24 |
| | FNLM | 0.8 | $\infty$ | $\infty$ | 49 | 7.24 |
| | BIL-HD | 0.9 | $\infty$ | 6 | 49 | **6.48** |

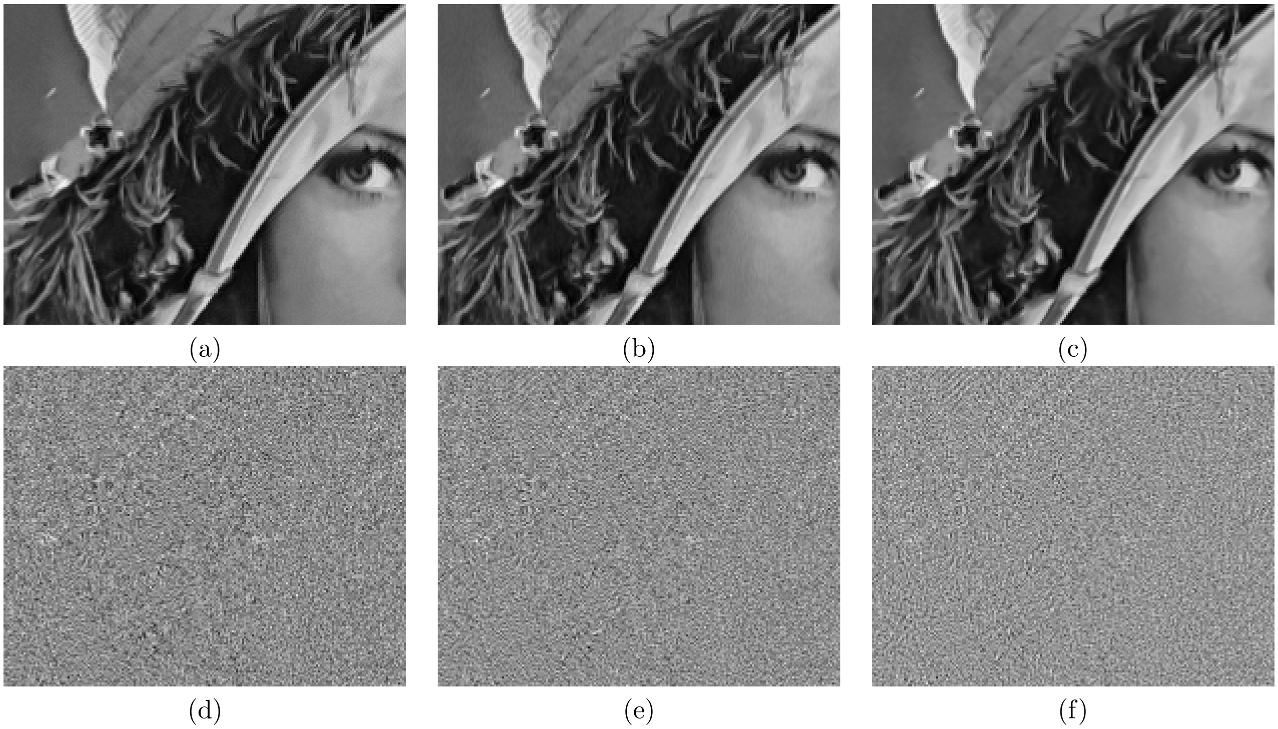
Table 10. Objective measures of Barbara image $\sigma = 25$

| Metric | Method | $n_{h_r}$ | $n_h$ | $h_s$ | $d$ | Result |
|---|---|---|---|---|---|---|
| Lena $\sigma = 25$ | | | | | | |
| PSNR | FNLM (the best) | 0.8 | $\infty$ | $\infty$ | 15 | 29.79 |
| | BIL-HD (the best) | 1 | $\infty$ | 4 | 49 | 30.38 |
| | BIL-HD | 1 | $\infty$ | 4 | **15** | *30.36* |
| MAD | FNLM (the best) | 0.8 | $\infty$ | $\infty$ | 15 | 7.00 |
| | BIL-HD (the best) | 1 | $\infty$ | 5 | 49 | 6.03 |
| | BIL-HD | 1 | $\infty$ | 5 | **15** | *6.05* |
| Peppers $\sigma = 25$ | | | | | | |
| PSNR | FNLM (the best) | 0.9 | $\infty$ | $\infty$ | 15 | 30.25 |
| | BIL-HD (the best) | 1.1 | $\infty$ | 4 | 49 | 30.86 |
| | BIL-HD | 1.1 | $\infty$ | 4 | **15** | *30.85* |
| MAD | FNLM (the best) | 0.8 | $\infty$ | $\infty$ | 15 | 8.03 |
| | BIL-HD (the best) | 1 | $\infty$ | 5 | 49 | 7.05 |
| | BIL-HD | 1 | $\infty$ | 5 | **15** | *7.08* |

Table 11. If we reduce the $d$ to the that of FNLM, the quality of the proposed method is slightly decreased but still better than FNLM

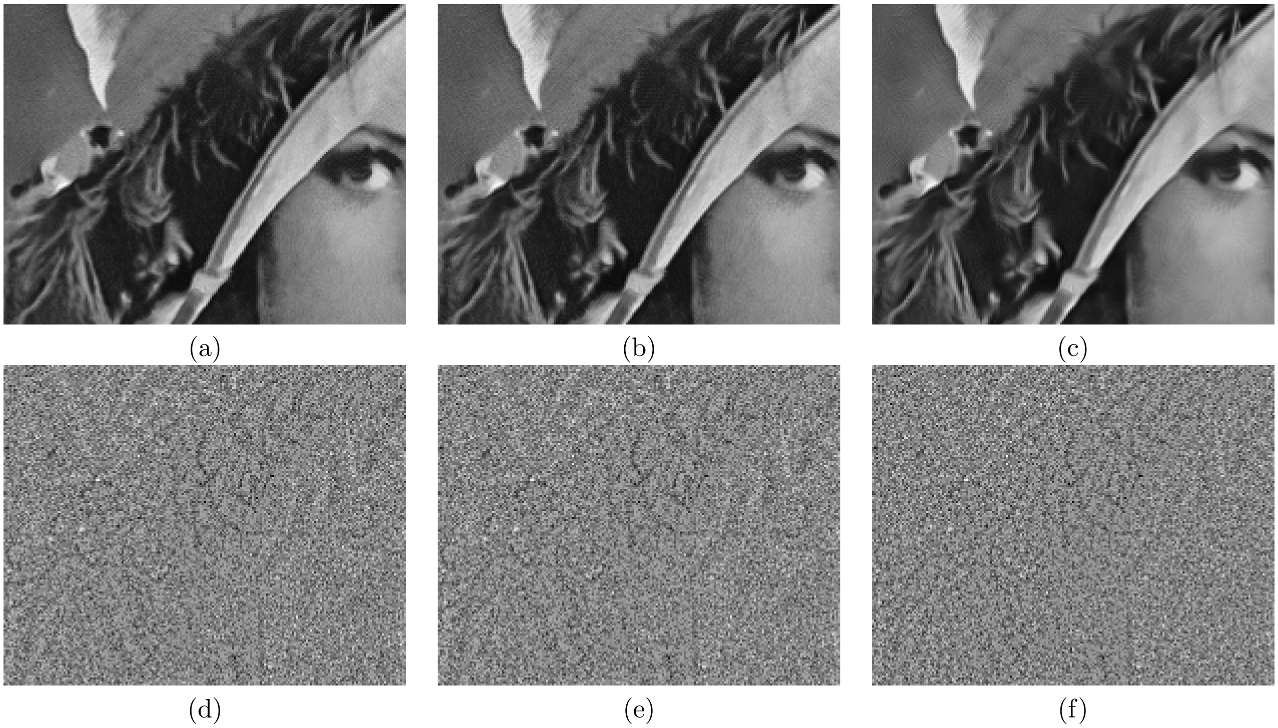
| $d$ | 1 | 3 | 6 | 8 | 10 | 15 | 20 | 25 | 30 | 40 | 49 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Time (in second) | 9.29 | 11.04 | 14.41 | 17.33 | 22.65 | 30.12 | 41.04 | 51.46 | 62.17 | 84.92 | 107.2 |

Table 12. Computational time in function of $d$ (The program is written by C, runs on PC of 2GHz and 2G Ram)


(a) (b) (c)

(d) (e) (f)

Figure 2. Result of Lena image in the case of $\sigma = 10$, first line: restored images, second line: difference between restored image with the noisy one, from left to right: NLM, FNLM and the proposed method



(a) (b) (c)

(d) (e) (f)

Figure 3. Result of Lena image in the case of $\sigma = 25$, first line: restored images, second line: difference between restored image with the noisy one, from left to right: NLM, FNLM and the proposed method