IMAGE DENOISING USING BILATERAL FILTER IN HIGH DIMENSIONAL PCA-SPACE

Quoc Bao DO, Azeddine BEGHDADI, Marie LUONG

L2TI,University of Paris 13
do.quocbao,azeddine.beghdadi,marie.luong@univ-paris13.fr

Abstract. This paper proposes a new noise filtering method inspired by Bilateral filter (BF), non-local means (NLM) filter and principal component analysis (PCA). The main idea here is to perform the BF 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. The proposed method is compared to the state-of-art. 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 each pixel is carried out by averaging other ones which are similar to it. 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. In [13], the authors propose Bilateral filter (BF) which takes into account both spatial and intensity information to define similar pixels for a given one. The relation between BF and anisotropic filtering has been inversigated in [1, 6]. Another adaptive filtering approach, called non-local means (NLM) [3, 4], has been recently proposed. Instead of using pixel-based similarity as in BF, 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 [12]. NLM in the wavelet domain is introduced in [11]. In [14], 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 [14], 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 BF, i.e. an anisotropic filter, in the HDPCA-space.

The paper is organized as follows: section 2 is devoted for a short review of NLM and related works, the HDPCA-space is presented in section 3. The proposed method is described in section 4 followed by experimental results in section 5. The conclusions are finally given in section 6.

2 Related works

Let us define a 2D noise-free image $u : R^2 \rightarrow R$. Its noisy version v at pixel (k, l) defined as $v(k, l) = u(k, l) + n(k, l)$ where n is identical, independent Gaussian noise. The aim of denoising is to estimate u from v. Both BF and NLM restore noisy pixel by averaging the neighboring pixels. An unifying formula for these methods could be expressed as follows:

$$\widehat{u}(k,l) = \frac{\sum_{(i,j)\in\Omega} w(k,l,i,j)v(i,j)}{\sum_{(i,j)\in\Omega} w(k,l,i,j)} \tag{1}$$

where Ω is the image domain and $w(k, l, i, j)$ stands for the weight which corresponds to the similarity between pixel $v(k, l)$ and $v(i, j)$. Indeed, each method proposes a kernel to estimate this weight. According to BF [13], the kernel is defined as follows:

$$w_{BF}(k,l,i,j) = exp\left(\frac{-\left(\|k-i\|^2 + \|l-j\|^2\right)}{h_s^2}\right) exp\left(\frac{-\|v(k,l) - v(i,j)\|^2}{h_r^2}\right) \tag{2}$$

where h_s and h_r are space and range parameters, respectively. Note that this filter takes into account both spatial and intensity information. In [1, 6], it has been proven that BF is an anisotropic filter with a special regularization term. In the case of NLM, its weight is given by:

$$w_{NLM}(k,l,i,j) = exp\left(\frac{-G_a * \|\boldsymbol{N}(k,l) - \boldsymbol{N}(i,j)\|^2}{h_r^2}\right) \tag{3}$$

where $\boldsymbol{N}(k, l), \boldsymbol{N}(i, j)$ are two small patches of size $r \times r$ around the pixel (k, l) and (i, j), respectively (r is usually equal to 7), G_a is a Gaussian kernel with standard deviation a of the same size as $\boldsymbol{N}(k, l), \boldsymbol{N}(i, j)$. Note that, the patch-similarity measure is weighted G_a to give more weight to pixels close to the patch center. The equation(3) can be rewritten as follows:

$$w_{NLM}(k,l,i,j) = exp\left(\frac{-\|\boldsymbol{N}_a(k,l) - \boldsymbol{N}_a(i,j)\|^2}{h_r^2}\right) \tag{4}$$

where the weighted patch $\boldsymbol{N}_a(k, l) = \sqrt{G_a}\boldsymbol{N}(k, l)$. Note that NLM considers only intensity information.

Due to computational burden, to denoise pixel (k,l), instead of considering all pixel (i,j) in the noisy image, Buades et al. propose to restraint a search window $\Omega(k,l)$ which is usually set equal to 21x21, i.e. there are 441 patch candidates. Indeed, further analyses [7] have shown that if the size of $\Omega(k,l)$ increase, more bias is introduced due to several mismatching patches which are taken into account. In [5], we used a strategy where only the best candidates are selected by exploring the entire image plane. Counter intuitively, this approach yields worse result in both term of subjective and objective measurement. 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-worse paradox*" in the sense that if we consider only the *best* candidates, the result is *worse*. Based on these remarks, the semi-non local approach, i.e. restrained to a small window $\Omega(k,l)$, is used in this work.

In [12], fast NLM (FNLM) filter is proposed by approximating the distance $\|\boldsymbol{N}_a(k,l) - \boldsymbol{N}_a(i,j)\|^2$ in (4) by another one estimated from projections of $\boldsymbol{N}_a$ onto a subspace defined by 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 $\boldsymbol{F}(k,l) = [f_1(k,l), f_2(k,l), ..., f_{r^2}(k,l)]^T$ is projected vector of $\boldsymbol{N}_a(k,l)$ onto this orthonormal basis, i.e. $f_m(k,l) =< \boldsymbol{N}_a(k,l), \boldsymbol{e}_m >$ where $<,>$ stands for inner product. As the signal energy concentrates on a few the most significant d components ($d \ll r^2$), Tasdizen[12] proposes to approximate the norm $\|\boldsymbol{N}_a(k,l) - \boldsymbol{N}_a(i,j)\|^2$ by using only these d components, i.e.

$$\|\boldsymbol{N}_a(k,l) - \boldsymbol{N}_a(i,j)\|^2 \approx \|\boldsymbol{F}^d(k,l) - \boldsymbol{F}^d(i,j)\|^2 = \sum_{m=1}^{d} \|f_m(k,l) - f_m(i,j)\|^2 \quad (5)$$

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

$$w_{NLM}^d(k,l,i,j) = exp\left(\frac{-\|\boldsymbol{F}^d(k,l) - \boldsymbol{F}^d(i,j)\|^2}{h_r^2}\right) \quad (6)$$

Finally, the FNLM is given by:

$$\widehat{u}^d(k,l) = \frac{\sum_{(i,j)\in\Omega} w_{NLM}^d(k,l,i,j) v(i,j)}{\sum_{(i,j)\in\Omega} w_{NLM}^d(k,l,i,j)} \quad (7)$$

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 in which spatial coordinates of each point corresponds to values of projected vector $\boldsymbol{F}^d$. We will show that FNLM and NLM are simply derived from an isotropic filter in this space.

3 High dimensional PCA-Space

Mapping in the HDPCA-space: First, each small patch is passed through the PCA system to obtain the corresponding projected vector $\boldsymbol{F}^d$. We define d dimensional HDPCA-space noted $\Psi_d \in R^d$ where the coordinates $\mathbf{p}$ of each point in this space are the values of $\boldsymbol{F}^d$. In other words, the intensity values of $\boldsymbol{F}^d$ become spatial coordinates in the new HDPCA-space. Each value $\mathbf{V}$ of a point $\mathbf{p}$ in this space is defined as follows:

$$\mathbf{V}(\mathbf{p}) = (V_1(\mathbf{p}), V_2(\mathbf{p})) = (v(k,l), 1)\ if\ \mathbf{p} = \mathbf{F}^d$$

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

- The first one $V_1(\mathbf{p}) = v(k,l)$ (the gray level of the center pixel (k,l) of the patch $\boldsymbol{N}(k,l)$).
- 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 Ω, 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 Ω as follows:

$$\widehat{u}(k,l) = \frac{\widehat{U_1}(\mathbf{p})}{\widehat{U_2}(\mathbf{p})}\ if\ \mathbf{p} = \mathbf{F}^d \tag{8}$$

Note that HDPCA is a sparse space where only points corresponding to the projected vectors $\boldsymbol{F}^d$ are defined.

4 Proposed method

To restore the pixel (k,l), in order to avoid *"best-worse paradox"* phenomenon, instead of projecting all patches in the image domain Ω onto the HDPCA-space, we project only the patches on the sub-domain $\Omega(k,l)$ (small windows of size 21×21 around the being processed pixel (k,l)) and carry out the filtering on these projected values. Since all values in projected vector $\boldsymbol{F}^d(k,l)$ become spatial coordinates $\mathbf{p}$ in the HDPCA-space, we can rewrite equation (7) of FNLM as follows:

$$\widehat{u}^d(k,l) = \frac{\widehat{U}_1(\mathbf{p})}{\widehat{U}_2(\mathbf{p})} = \frac{\sum_{\mathbf{q}\in\Psi_d} exp\left(\frac{\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) V_1(\mathbf{q})}{\sum_{\mathbf{q}\in\Psi_d} exp\left(\frac{\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) V_2(\mathbf{q})} \tag{9}$$

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 (8))

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[9], Perona-Malik[10] 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, BF acts as an anisotropic filter and it works in non-local manner therefore it could be used. The proposed method (called BF-HDPCA) 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 (8))

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

$$\widehat{U_\eta}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in\Psi_d} w_\eta(\mathbf{p},\mathbf{q}) V_\eta(\mathbf{q})}{\sum_{\mathbf{q}\in\Psi_d} w_\eta(\mathbf{p},\mathbf{q})} \tag{10}$$

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

$$w_\eta(\mathbf{p},\mathbf{q}) = exp\left(\frac{-\|V_\eta(\mathbf{p}) - V_\eta(\mathbf{q})\|^2}{h^2}\right) exp\left(\frac{-\|\mathbf{p}-\mathbf{q}\|^2}{h_r^2}\right) \tag{11}$$

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. Recall that this second term is the patch similarity measure defined in the image domain. The filtered values $\widehat{\mathbf{U}}(\mathbf{p})$ are finally projected back into the image domain Ω using the division in equation (8). Note that when $h = \infty$ the proposed method tends to FNLM.

5 Experimental results

The experimental results are carried out on three natural images: Lena, Peppers and Fingerprint of size 512×512. The last one is typical of highly textured image whereas the second one contains mostly homogenous regions. The first image contains 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 $\Omega(k,l)$ is defined by small windows 21×21 around the being processed pixel (k,l). The patch size is equal to 7×7 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) and $h = n_h\sigma$ with $n_h = [1 : 1 : 10, \infty]$. Recall that when $n_h = \infty$ and $d = 49$, the proposed method tends to the classical NLM. 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 [8] and PSNR_W [2] 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. *The best results* of three methods are reported in Tables 1, 2, 3 with the corresponding optimal parameters d, n_h, n_{h_r}. Note that, for each method, each metric results in different optimal parameters. For $\sigma = 10$, BF-HDPCA clearly outperforms the others methods for all images and it is confirmed by all metrics. For higher noise level $\sigma = 25$, this advantage is no longer true for Lena image, but still valid for images which contain many redundant structures such as Fingerprint and Peppers. It is also worth to note that, in many cases, BF-HDPCA can achieve better quality with smaller dimension d compared to FNLM (for example, for Pepper image, $\sigma = 25$, optimal d for FNLM is 15 whereas in our case, this value is 3 which makes BF-HDPCA 3 times faster than FNLM - see table 4). For all images and all metrics, the optimal h of the proposed method can be found from 2σ to 4σ. For the subjective comparison, due to limit of space, only Lena images are presented in Fig.1. As can be seen, the small details on the hat are well preserved with the proposed method whereas they are oversmoothed in NLM and FNLM. There is still some kind of natural grain in the forehead and cheek regions in our case, what seems more comfortable for our eyes than too flat region obtained with NLM and FNLM (*please use your monitor to view all images in this paper*).

6 Conclusions

A new nonlinear anisotropic filtering method based on BF and the HDPCA-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. Barash, D., Comaniciu, D.: A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. Image Vision Comput. pp. 73–81 (2004)
2. Beghdadi, A., Pesquet-Popescu, B.: A new image distortion measure based wavelet decomposition. In Proc. ISSPA pp. 485–488 (2003)
3. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Simul **4**, 490–530 (2005)
4. Buades, A., Coll, B., Morel, J.M.: Nonlocal image and movie denoising. International Journal of Computer Vision pp. 123–139 (2008)
5. Do, Q., Beghdadi, A., 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)

6. Elad, M.: On the origin of the bilateral filter and ways to improve it. IEEE Transactions on Image Processing pp. 1141–1151 (2002)
7. Kervrann, C., Boulanger, J.: Optimal spatial adaptation for patch-based image denoising. IEEE Trans. on Image Processing **15**(10), 2866–2878 (2006)
8. Larson, E.C., Chandler, D.M.: Most apparent distortion: full-reference image quality assessment and the role of strategy. Journal of Electronic Imaging **19** (2010)
9. Osher, Fatemi, E.: Nonlinear Total Variation Based Noise Removal Algorithms. Physica D **60**, 259–268 (1992)
10. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. pp. 629–639 (1990)
11. Souidene, W., Beghdadi, A., Abed-Meraim, K.: Image denoising in the transformed domain using non local neighborhood. In Proc. ICASSP (2006)
12. Tasdizen, T.: Principal neighborhood dictionaries for nonlocal means image denoising. IEEE Transactions on Image Processing pp. 2649–2660 (2009)
13. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. pp. 839–846 (1998)
14. Tschumperlé, D., Brun, L.: Image denoising and registration by pde's on the space of patches (2008)

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	0.9	∞	49	33.55
	FNLM	0.9	∞	15	33.63
	BF-HDPCA	0.9	4	**8**	**34.04**
PSNR_W	NLM	0.8	∞	49	16.61
	FNLM	0.8	∞	10	16.78
	BF-HDPCA	0.8	4	**8**	**17.18**
MAD	NLM	0.8	∞	49	1.88
	FNLM	0.7	∞	10	1.75
	BF-HDPCA	0.8	4	**8**	**1.66**

$\sigma = 10$

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	1	∞	49	29.81
	FNLM	1	∞	30	29.81
	BF-HDPCA	1	20	**15**	29.80
PSNR_W	NLM	0.7	∞	49	12.19
	FNLM	0.7	∞	15	12.21
	BF-HDPCA	0.7	20	15	12.21
MAD	NLM	1	∞	49	7.00
	FNLM	1	∞	15	6.97
	BF-HDPCA	0.6	4	**3**	**6.71**

$\sigma = 25$

Table 1. Objective measures of Lena image

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	1	∞	49	29.93
	FNLM	1	∞	15	30.09
	BF-HDPCA	1.1	4	15	**31.02**
PSNR_W	NLM	0.9	∞	49	17.93
	FNLM	0.9	∞	15	18.19
	BF-HDPCA	1.1	2	**6**	**19.54**
MAD	NLM	0.8	∞	49	0.42
	FNLM	0.6	∞	6	0.29
	BF-HDPCA	0.7	4	6	**0.21**

$\sigma = 10$

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	0.7	∞	49	26.64
	FNLM	0.6	∞	6	27.27
	BF-HDPCA	0.6	6	6	**27.40**
PSNR_W	NLM	0.6	∞	49	13.91
	FNLM	0.6	∞	6	14.48
	BF-HDPCA	0.6	4	6	**14.63**
MAD	NLM	0.6	∞	49	2.34
	FNLM	0.6	∞	6	1.70
	BF-HDPCA	0.6	4	6	**1.57**

$\sigma = 25$

Table 2. Objective measures of Fingerprint image

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	0.9	∞	49	33.71
	FNLM	0.8	∞	10	33.89
	BF-HDPCA	0.9	4	**6**	**34.35**
PSNR_W	NLM	0.8	∞	49	17.09
	FNLM	0.7	∞	10	17.39
	BF-HDPCA	1	4	**6**	**17.86**
MAD	NLM	0.8	∞	49	3.07
	FNLM	0.7	∞	10	2.49
	BF-HDPCA	0.8	4	**6**	**2.20**

$\sigma = 10$

Metric	Method	n_{h_r}	n_h	d	Result
PSNR	NLM	1	∞	49	30.24
	FNLM	1	∞	15	30.28
	BF-HDPCA	0.6	4	**3**	**30.41**
PSNR_W	NLM	0.7	∞	49	12.81
	FNLM	0.7	∞	15	12.90
	BF-HDPCA	0.6	4	**3**	**13.05**
MAD	NLM	1	∞	49	8.14
	FNLM	1	∞	15	8.04
	BF-HDPCA	0.6	4	**3**	**7.75**

$\sigma = 25$

Table 3. Objective measures of Peppers image

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

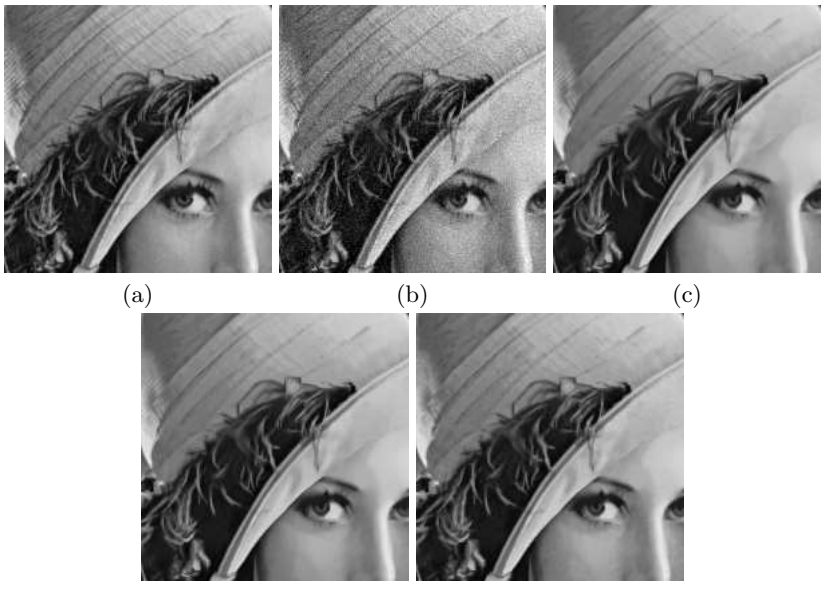

Fig. 1. Lena image: from left to right, top to bottom, (a) The original image, (b) The noisy image for $\sigma = 10$, (c) NLM's result (the best PSNR=33.55 for $h_r = 0.9\sigma$,$h = \infty$,$d = 49$), (d) FNLM's result (the best PSNR=33.63 for $h_r = 0.9\sigma$,$h = \infty$,$d = 15$), (e) The proposed method (the best PSNR=34.04 for $h_r = 0.9\sigma$,$h = 4\sigma$,$d = 8$)