Performance Assessment of Patch-based Bilateral Denoising

Arnaud de Decker[1], John Aldo Lee[2] and Michel Verleysen[1]

[1] Machine Learning Group, Université catholique de Louvain
pl. du Levant 3, 1348 Louvain-la-Neuve, Belgium
{arnaud.dedecker, michel.verleysen}@uclouvain.be
[2] Molecular Imaging and Experimental Radiotherapy StLuc University Hospital
Université Catholique de Louvain, av. Hippocrate 54, 1200 Bruxelles, Belgium
john.lee@uclouvain.be

Abstract. In the field of medical image analysis, denoising is one of the most important preprocessing steps before medical analysis. The design of an efficient, robust, and computationally effective edge-preserving denoising algorithm is a widely studied, and yet unsolved problem. One of the most efficient edge-preserving denoising algorithms is the bilateral filter, which is an intuitive generalization of the local M-smoother. In this paper, we propose to modify both the bilateral filter and the local M-smoother to use patches of the image instead of single voxels in the denoising process. Using patches instead of single voxels in the filtering process is a way to adapt the filter to the textures, ramps, and edges of the image, and make the filter more discriminant. The filtering performances of the patch-based algorithms are evaluated on a benchmark and a CT phantom image and compared to the bilateral filter and local M-smoother.

1 Introduction

Nowadays, medical images are essential tools for medical doctors. They are used in radiotherapy, nuclear medicine, radiology, oncology, and many other fields of medicine. However, they are often polluted by noise and blur. These problems can induce misinterpretations and lead to errors in diagnosis and treatment. For example, in radiotherapy, it is of crucial importance to have a precise identification of the volumes to be treated. For this reason, the first and more important preprocessing step after the acquisition of a medical image is to use a filtering algorithm in order to get rid of the noise before the actual analysis.

In denoising methods, the challenge is to obtain a filtering effect powerful enough to remove most of the noise generated during the acquisition process while preserving the edges and textures in the image. This problem becomes even more complicated for medical images as they often have a relatively low resolution. Furthermore, a filter which does not preserve edges accurately blurs the image, which reduces the resolution even more.

Several algorithms have been used for unsupervised edge-preserving denoising of medical images: wavelet transform [1], [2], partial differential equations [3], total variation [4], Bayesian denoising [5], kernel regression [6], gradient approximation [7] ,

local M-smoother [8], [9], bilateral [10], [11], or even trilateral filtering [12]. Kervrann also used patch based algorithms [13], [14] for denoising purposes.

Among these algorithms, the bilateral filter (BF) and the local M-smoother (LMS) have been quite popular lately because of their intuitive formulation, their low computational load, and their efficiency in terms of edge-preserving denoising performances. Still, while the BF and LMS algorithms are excellent for piecewise constant images denoising, their performances drop significantly when the image is blurred or contains ramps, which is the case in most realistic situations. Indeed, these algorithms are based on the principle of mode identification. When the image is not piecewise constant, there is no mode to identify and the filters become less efficient.

In this paper, we propose to generalize the BF algorithm to use subsets of adjacent voxels of the image, called *patches*, instead of single voxels. As these patches are composed of more than one voxel and gather information in 2 or 3 dimensions around their center, the filtering becomes more selective and the performance of this patch-based bilateral filtering (PBBF) improves.

The rest of this paper are organised as follows. In Section 2 we will define the image model, and briefly introduce the local M-smoother and the bilateral filter algorithms. Then, we will describe how patches are introduced in these classical algorithm. The benchmark image we used to test the filtering performances in this paper is introduced in Section 3 with a comparison of the performances and visual results of the algorithm with state-of-the art BF and LMS. Finally, Section 4 gathers the conclusions and future perspectives.

2 Image Model and Patch-based Bilateral Filter

2.1 Image Model

Let us define a D-dimensional image as a vector of voxels in which the i_{th} voxel position can be uniquely identified by vector $\mathbf{x}_i = (x_{i_1}, \ldots, x_{i_D})$. The voxel at coordinate $\mathbf{x}_i$ has an observed intensity f_i which can be decomposed in two part: the real image u_i and the noise ε_i:

$$f_i = u_i + \varepsilon_i. \tag{1}$$

In further developments, we will assume ε_i is white and Gaussian. This is not the case for most medical images. However it is usually possible to come back to a gaussian denoising problem by an appropriate transformation of the data. The derivation of the BF and LMS in this section follows [15], [8] where more details can be found.

2.2 Gaussian Filter

Local filtering can be achieved by averaging the values of the voxels in a neighborhood N_i of the voxel i. The filtering process can be described as the minimisation of the error function

$$E(\hat{\mathbf{u}}) = \frac{1}{2} \sum_{i=1}^{I} \sum_{j \in N_i} w_{ij} (\hat{u}_i - f_j)^2, \tag{2}$$

where $\mathbf{u}$ is the vector with the filtered voxels, I is the total number of voxels in the image, w_{ij} is the fixed weight of the contribution of f_j. The optimal solution is obtained by derivating this function with respect to $\hat{\mathbf{u}}$. Equating this derivative with zero allows us to identify the optimum $\hat{\mathbf{u}}$ as

$$\hat{\mathbf{u}}_i = \frac{\sum_{j \in N_i} w_{ij} f_j}{\sum_{j \in N_i} w_{ij}}. \tag{3}$$

The weights w_{ij} can be defined as a hard or soft window as long as they vanish when the distance to the i_{th} voxel increases. The most common choice for the weights is a soft window defined by a gaussian kernel:

$$w_{ij} = exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\rho^2}\right). \tag{4}$$

The width of the kernel is determined by parameter ρ. While this is an efficient filtering scheme for Gaussian noise, this filter smoothes the edges of the image. However, it can easily be modified into an edge-preserving filtering scheme: the local M-smoother.

2.3 Local M-Smoother

From the gaussian filter, it is possible to obtain an edge-preserving filtering algorithm if the error function is modified as follows

$$E(\hat{\mathbf{u}}) = \sum_{i=1}^{I} \sum_{j \in N_i} w_{ij} \Psi\left(\frac{1}{2}(\hat{u}_i - f_j)^2\right). \tag{5}$$

The key idea is to assume that there are two or more constant signals (or modes) mixed in the neighborhood N_i. The error function should separate the modes and thus prevent the image from being blurred during the filtering process. This can be done if Ψ is defined as

$$\Psi\left(\frac{1}{2}s^2\right) = 1 - exp\left(-\frac{s^2}{2\sigma^2}\right). \tag{6}$$

instead of a classical squared difference. This is a mode estimator, as typically used in robust statistic estimation.

It is now possible to minimize $E(\hat{\mathbf{u}})$ by gradient descent [15], which leads to the expression of the local M-smoother:

$$\hat{u}_i^{k+1} = \frac{\sum_{j \in N_i} w_{ij} \Psi'\left(\frac{1}{2}(\hat{u}_i^k - f_j)^2\right) f_j}{\sum_{j \in N_i} w_{ij} \Psi'\left(\frac{1}{2}(\hat{u}_i^k - f_j)^2\right)}. \tag{7}$$

In this equation, Ψ' is the derivative of equation (6) and is called the radiometric kernel. A common way to initialise this filter is to use $u_i^0 = f_i$ as a starting guess.

2.4 Bilateral Filter

The bilateral filter is an intuitive generalization of the local M-smoother. In the bilateral filter, u_i^k, the values of the voxels in the neighborhood of $\mathbf{x}_i$, are compared to u_i^k, the current estimated image, instead of f_i, the observed image. The algorithm becomes

$$\hat{u}_i^{k+1} = \frac{\sum_{i \in N_i} w_{ii} \Psi'\left(\frac{1}{2}(\hat{u}_i^k - \hat{u}_i^k)^2\right) \hat{u}_i^k}{\sum_{i \in N_i} w_{ii} \Psi'\left(\frac{1}{2}(\hat{u}_i^k - \hat{u}_i^k)^2\right)}. \tag{8}$$

This modification also means that the filter becomes non local as the intensities which were out of the neighborhood of $\hat{u}_i^k$ may have played a role in the determination of $\hat{u}_i^k$ at a precedent step. Because of this diffusion process, the BF can converge to a constant image and has to be stopped after a few iterations.

2.5 Patch-based Bilateral Filter

Although the bilateral filter produces excellent results for a piecewise-constant image, the filtering performances drop significantly on the edges and ramps. We propose to use a higher number of pixels in order to separate the modes on more information than a single voxel. Let us define patch P_i^k as the $p \times p \times p$ subset of voxels of the image at iteration k, centered on $\mathbf{x}_i$. The weights between the voxels in $\mathbf{x}_i$ and $\mathbf{x}_i$ can now be calculated using $d(P_i, P_i)$, the distance between patches P_i and P_i

$$\hat{u}_i^{k+1} = \frac{\sum_{i \in N_i} w_{ii} \Psi\left(d(P_i^k, P_i^k)\right) \hat{u}^k}{\sum_{i \in N_i} w_{ii} \Psi\left(d(P_i^k, P_i^k)\right)}. \tag{9}$$

The distance function can be chosen according to the noise model and the specificities of the image. The most obvious choice is to use the L_2 distance

$$d(P_i^k, P_i^k) = \sqrt{\sum_{n=1}^{N} (P_{i_n}^k - P_{i_n}^k)^2}. \tag{10}$$

where $P_{i_n}^k$ is the n_{th} voxel in patch P_i^k and N is the total number of voxels in a patch. Other types of distances can be considered depending of the properties desired from the filtering algorithm.

It is also possible to introduce the patch concept in the local M-smoother algorithm. In this case, the patch centered in $\mathbf{x}_i$ would be constructed from the original image f:

$$\hat{u}_i^{k+1} = \frac{\sum_{i \in N_i} w_{ii} \Psi\left(d(P_i^k, P_i^0)\right) f_i}{\sum_{i \in N_i} w_{ii} \Psi\left(d(P_i^k, P_i^0)\right)}. \tag{11}$$

If the patch size is set to 1 and the distance used is L_2, the PBBF and PBLMS algorithms are equivalent to the classical BF and LMS, respectively.

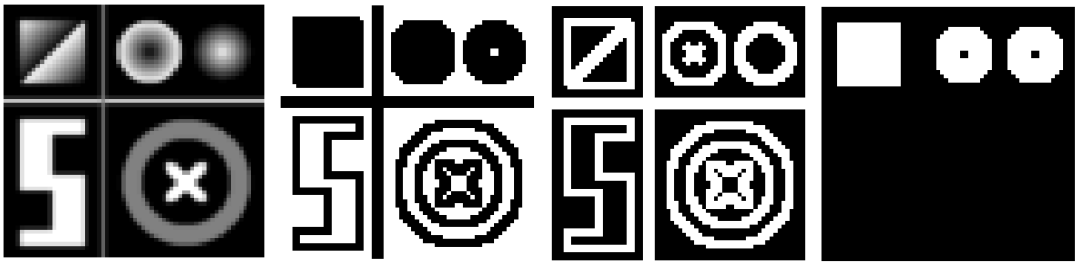

Fig. 1. From left to right: Blurred benchmark image without noise, plateaus mask, edges mask, ramps mask.

3 Experiments and Results

3.1 Benchmark Image Presentation

Assessing the filtering performances with real-life images such as those acquired on medical devices is difficult because the ground truth, that is, the noise free image remains unknown. For this reason, the PBBF algorithm has been tested on a 64×64 benchmark gray level image. This image consists in a pattern (see Fig. 1, left) which is a combination of constant plateaus, edges, and gradients. It combines all types of difficult situations a good filtering algorithm has to cope with. Most medical images present some or all of these types of difficulties. This pattern is also slightly blurred in order to avoid sharp edges that would be completely unrealistic. In the blurred image, the edges between the different areas are smoothed. Eventually, the two-dimensional pattern is repeated 100 times and stacked so that the final dimension is $64 \times 64 \times 100$. Gaussian noise is then added to the blurred image.

The denoising performances are evaluated by the RMSE:

$$RMSE = \sqrt{\frac{1}{M \sum_{i=1}^{N} \nu_i} \sum_{m=1}^{M} \sum_{i=1}^{N} \nu_i (\hat{u}_i^k - u_i)^2}. \quad (12)$$

where M is the number of trials and m then trial index. In this equation, the weights ν_i either have a value of $\{0,1\}$ and are used as masks. These masks are designed to evaluate the RMSE over the selected areas of the image (whole image, plateaus, edges or ramps (Fig. 1)). With these masks, the denoising effect can be evaluated specifically on these areas. More details about the benchmark image, its generation and the performances evaluation can be found in [15].

3.2 Experimental Setup and Results

In our implementation of the algorithm, the patches are circular and their size is fixed to the size of the spatial kernel. Indeed, a patch size smaller than the spatial kernel leads to a discontinuity in the weights evaluation as the patch would end when the spatial weights are still non negligible, on the opposite, a spatial kernel smaller than the patch

size would mean that the algorithm calculates pixels that would have a 0 weight in the filtering process. The patches size are then defined by ρ, the width of the spatial kernel.

In the experiment, each 64×64 slice of the image is denoised independently (2D denoising) patches and 2D kernels. The denoising performance on each slice is considered as the realization of a trial. The quality of the denoising process is measured by the RMSE for each slice of the image, using the different masks. The results are obtained on a test image independent from the one used for setting the parameters (it was generated using the same noise model). The PBBF and PBLMS results are compared to those of the BF and LMS. For both the PBBF and PBLMS, the patch to patch distance was defined as the L_2 distance.

The results of the denoising procedures are shown in Table 1. The PBBF outperforms both the classical BF and LMS approaches in terms of RMSE and visual appreciation (see Fig. 2). The mechanisms behind theses performances can be explained: optimal parameters for this filter use narrow radiometric (small σ in equation 6) and spatial kernels (small ρ in equation 4), but a larger (13) number of iterations, whereas the BF usually converges to a constant solution after more than 3 iterations.

As the PBBF is more selective than the BF for the computation of the weights, it can use a higher number of iterations and narrow radiometric and spatial kernels to converge slowly toward a better filtering result. The PBBF outperforms the other algorithms on all parts of the image.

The reason why the PBLMS is less performant than the PBBF can be explained too. After the first iteration, the algorithm computes the distance between the last estimated image P_i^k and the observed image P_i^0. As the filtering effect is applied on the estimated

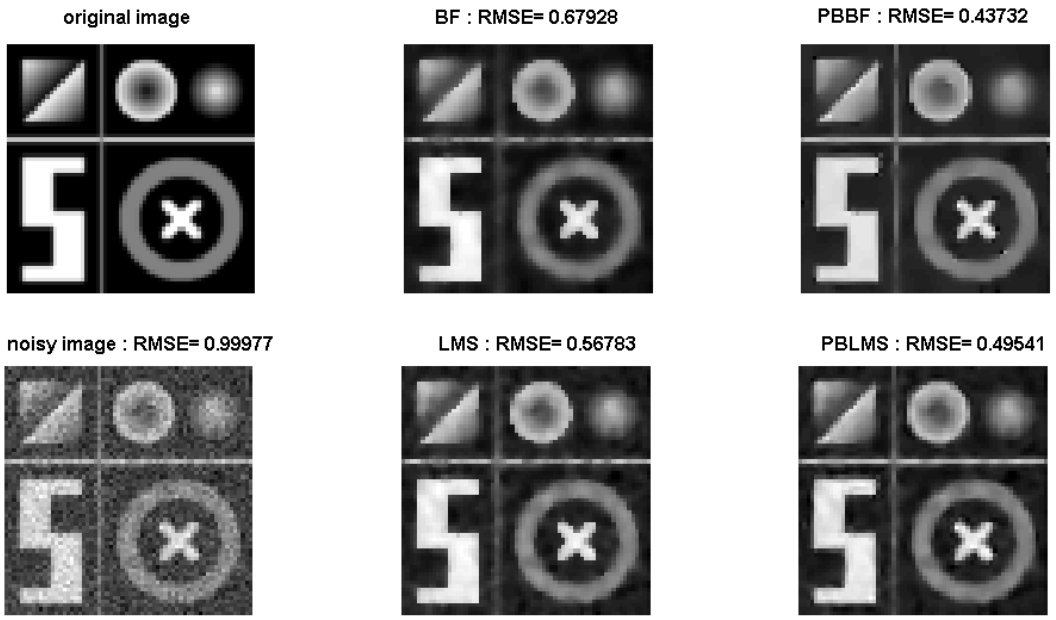

Fig. 2. Top row: left, original image. Middle, bilateral filtered image. Right, patch-based filtered image. Bottom row: left, noisy image. Middle, local M-smoother filtered image. Right, patch-based local M-smoother filtered image.

image, those images get more and more different, and the differences add up in the distance computation until the distance is large enough to eventually be out of the radiometric kernel range and thus reduces the weight to almost zero. The filtering process is then stopped too early and the image is not filtered anymore after a few iterations.

Fig. 3 shows the residuals of the filtered images. The residuals for the PBBF and PBLMS filtered image have almost no structure while the BF and LMS ones contain parts of the original image, it shows that in theses cases, the filtering effect was not optimal.

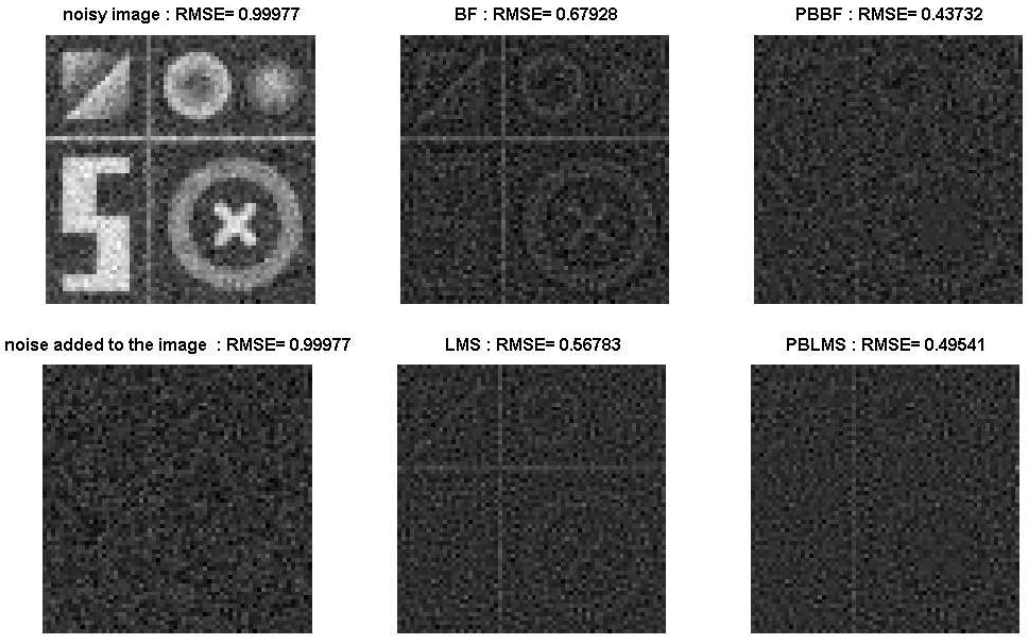

Fig. 3. Top row: left, noisy image. Middle, residuals of the bilateral filtered image. Right, residuals of the patch-based bilateral filtered image. Bottom row: left, noise added to the original image. Middle, residuals of the local M-smoother filtered image. Right, residuals of the patch-based local M-smoother filtered image.

Table 1. Mean RMSE over the 100 test images for the different algorithms and masks (2D). ρ, σ and n are respectively the width of the spatial kernel, radiometric kernel, and number of iterations of the filter.

	image	plateaus	edges	ramps	ρ	σ	n
PBBF	0.4373	0.2637	0.5941	0.5484	2.1	0.3	15
PBLMS	0.4954	0.3866	0.6272	0.4995	2.8	0.6	5
BF	0.5614	0.4033	0.7464	0.5695	1.9	2.6	2
LMS	0.5678	0.4051	0.7565	0.5491	2.5	2.6	3

These algorithms were tested on 3D images. In this case we generated 100 images of dimension $64 \times 64 \times 100$ with the same noise model, and used $p \times p \times p$ patches and 3D kernels. The RMSE is calculated over each one of these $64 \times 64 \times 100$ images for

all algorithms. In the 3D case, the performances of all algorithms improve (see Table 2). The PBBF filtering algorithm still gives the best RMSE, on all areas of the image. Obviously, the benchmark image that was used here is constituted of the same slices which help the denoising algorithms a lot. However, successive slices are usually highly correlated in real 3D images, therefore the PBBF may find some useful informations when a third dimension is added. Further experiments should be done on 3D benchmark images in order to confirm the filtering gain that we noticed on the 3D denoising.

Table 2. mean RMSE over the 100 test images for the different algorithms and masks (3D). ρ, σ and n are respectively the width of the spatial kernel, radiometric kernel, and number of iterations of the filter.

	image	plateaus	edges	ramps	ρ	σ	n
PBBF	0.2515	0.1432	0.3348	0.3941	2.1	0.3	13
PBLMS	0.3493	0.2380	0.4721	0.3765	2.8	0.6	5
BF	0.4045	0.2982	0.5290	0.4281	2.1	1.8	2
LMS	0.4228	0.3294	0.5366	0.4387	2.2	2	3

The PBBF algorithm was also compared to the BF on a phantom CT image. A phantom consists of several cylinders of different densities. As the absorption properties of the cylinders are known, the images acquired on these phantoms a used to determine the quantity and distribution of the noise generated in the acquisition process. This time, the parameters of the filters were fixed empirically as they would have been for a classical CT image for which no ground truth is available. The quality of the denoising effect was evaluated from the residuals: the difference between the original and the denoised image. Figure 4 shows the PBBF and BF denoised images ans their residuals. From visual evaluation, they seem very close in quality: for both algorithms, there is no visible blurring induced in the denoising process and the residual noise is located at the same areas. Still, the residuals from the PBBF algorithm show slightly less structure than those from the BF algorithm, which shows that the PBBF preserved the edges in a more efficient way. The PBBF seems to at least compare to the BF on images acquired with real scanners. More experiments on real medical images will be done in the future.

4 Conclusions

In this paper, we introduced the patch-based bilateral filter and patch-based local M-smoother. The introduction of patches in the classical bilateral filter and local M - smoother algorithms is based on the idea that an algorithm using more than one voxel at a time might be able to be more sensitive to the textures and ramps of the image. Our experiments shows that the patch-based bilateral filter outperforms the bilateral filter and local M-smoother algorithms both in terms of RMSE and visual appreciation. These results also seems promising in the 3D denoising tasks. In the future, we will adapt this method to more complex noise models (i.e. Poisson or multiplicative noise) and other patch-to-patch distances functions will be implemented and tested. More performances evaluations of these algorithms will be done on CT-scan or PET-scan images

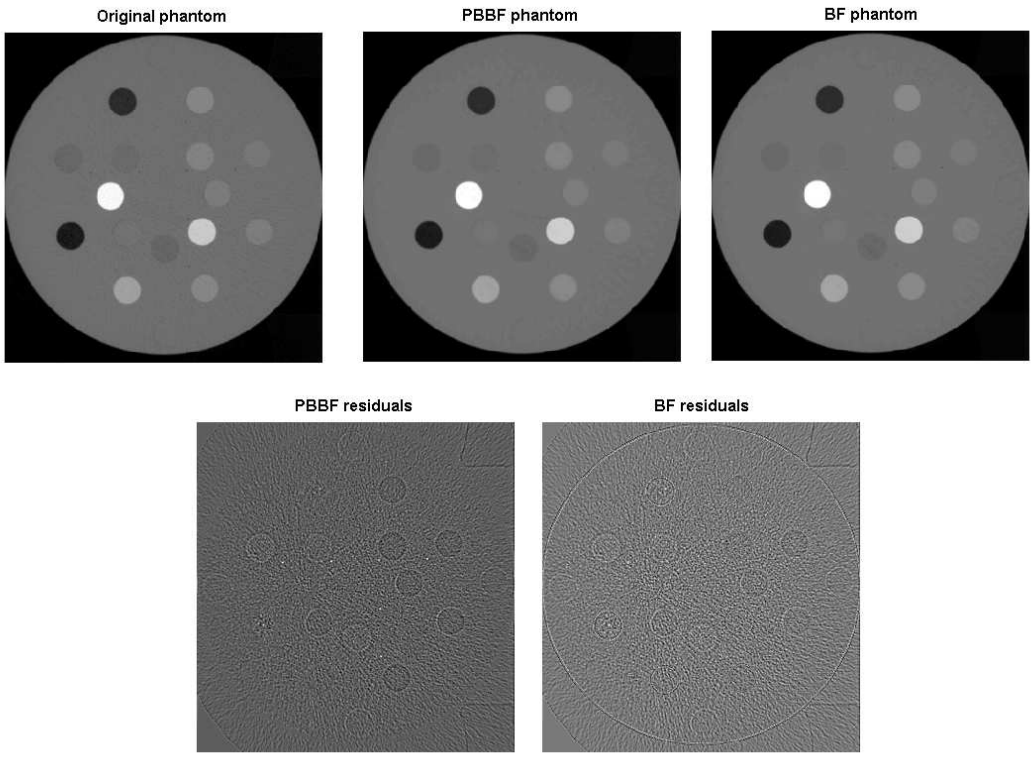

Fig. 4. Top row: left, original phantom image. Middle, PBBF denoised phantom. Right, LMS denoised image. Bottow row: left, residuals for the PBBF image. Right residuals for the BF denoised image.

and the results will be assessed by oncologists within the framework of tumor and organ delineation tasks.

Acknowledgements

A. de Decker is funded by a Belgian F.R.I.A. grant. J.A. Lee is a Postdoctoral Researcher funded by the Belgian National Fund of Scientific Research (FNRS).

References

1. Charnigo, R., Sun, J., Muzic, R.: A semi-local paradigm for wavelet denoising. IEEE Trans Image Process 15 (2006) 666–677
2. Lin, J.W., Laine, A.F., Bergmann, S.R.: Improving pet-based physiological quantification through methods of wavelet denoising. IEEE Trans Biomed Eng 48 (2001) 202–212
3. Bruni, V., Piccoliand, B., Vitulano, D.: Wavelets and partial differential equations for image denoising. Electronic Letters on Computer Vision and Image Analysis 6 (2008) 36–53
4. Osher, S., Burger, M., Donald, G., Jinjun, X., Yin, W.: An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. 4 (2005) 460–489

5. Sanches, J.M., Nascimento, J.C., Marques, J.S.: An unified framework for bayesian denoising for several medical and biological imaging modalities. Conf Proc IEEE Eng Med Biol Soc 1 (2007) 6267–6270
6. Takeda, H., Farsiu, S., Milanfar, P.: Kernel regression for image processing and reconstruction. IEEE Transactions on Image Processing 16 (2007) 349–365
7. Huang, R.H., Wang, X.H.: Image denoising via gradient approximation by upwind scheme. Signal Processing, 88 (2008) 69–74
8. Mrazek, P., Weickert, J., A., B.: On Robust Estimation and smoothing with Spatial and Tonal Kernels. In: Geometric Properties for Incomplete data. (2006) 335–352
9. Chu, C., Glad, I., Godtliebsen, F., J.S., M.: Edge-preserving smoothers for image processing. Journal of the American Statistical Association, 93,no.442 (1998) 526556
10. Elad, M.: On the origin of the bilateral filter and ways to improve it. IEEE Trans. Image Processing, 11 (2002) 11411151
11. Tomasi, C., Manduchi, R.: Bilateral ltering for gray and color images. In: 6th Int. Conf. Computer Vision, pp.839846. (1998)
12. Wong, W., Chung, A., Yu, S.: Trilateral filtering for biomedical images. In Chung, A., ed.: Proc. IEEE International Symposium on Biomedical Imaging: Nano to Macro. (2004) 820–823 Vol. 1
13. Kervrann, C., Trubuil, A.: An adaptive window approach for poisson noise reduction and structure preserving in confocal microscopy. In Trubuil, A., ed.: Proc. IEEE International Symposium on Biomedical Imaging: Nano to Macro. (2004) 788–791 Vol. 1
14. Kervrann, C., Boulanger, J.: Optimal spatial adaptation for patch-based image denoising. , 15 (2006) 2866–2878
15. Lee, J., Geets, X., Gregoire, V., Bol, A.: Edge-preserving filtering of images with low photon counts. IEEE Transactions on pattern analysis and machine intelligence, 30 (2008) 1014–1027