EFFICIENT ALGORITHM FOR IMPULSIVE NOISE REDUCTION

Bogdan Smolka *,1 **Cristian Cantón-Ferrer** **

* *Department of Automatics,*
Electronics and Computer Science
Silesian University of Technology
Akademicka 16 Str, 44-101 Gliwice, Poland
bsmolka@ia.polsl.gliwice.pl

** *Polytechnical University of Catalonia, Spain*

Abstract: In this work a novel approach to the problem of impulsive noise reduction for color and gray scale images is presented. The new image filtering technique is based on the maximization of the similarities between pixels in the filtering window. The new method removes the noise component, while adapting itself to the local image structures. In this way, the proposed algorithm eliminates impulsive noise, while preserving edges and fine image details. Since the algorithm can be considered as a modification of the standard vector median filter driven by fuzzy membership functions, it is fast, computationally efficient and very easy to implement. Experimental results indicate that the new method is superior to the commonly used algorithms for impulsive noise reduction.

Keywords: biomedical systems, medical applications, filtering techniques, comet assay, biomedical preprocessing, computer vision

1. BRIEF OVERVIEW OF STANDARD COLOR NOISE REDUCTION FILTERS

A number of nonlinear, multichannel filters, which utilize correlation among multivariate vectors using various distance measures, have been proposed to date [1-5]. The most popular nonlinear, multichannel filters are based on the ordering of vectors in a predefined moving window. The output of these filters is defined as the lowest ranked vector according to a specific vector ordering technique.

Let $\mathbf{F}(x)$ represent a multichannel image and let W be a window of finite size n (filter length). The noisy image vectors inside the filtering window W are denoted as $\mathbf{F}_j,\ j = 0, 1, ..., n-1$. If the distance between two vectors $\mathbf{F}_i, \mathbf{F}_j$ is denoted as $\rho\{\mathbf{F}_i, \mathbf{F}_j\}$ then the scalar quantity $R_i = \sum_{j=0}^{n-1} \rho\{\mathbf{F}_i, \mathbf{F}_j\}$, is the total distance associated with the noisy vector $\mathbf{F}_i$.

The ordering of the R_i's : $R_{(0)} \leq R_{(1)} \leq ... \leq R_{(n-1)}$, implies the same ordering to the corresponding vectors $\mathbf{F}_i : \mathbf{F}_{(0)} \leq \mathbf{F}_{(1)} \leq ... \leq \mathbf{F}_{(n-1)}$. Nonlinear ranked type multichannel estimators define the vector $\mathbf{F}_{(0)}$ as the filter output. However, the concept of input ordering, initially applied to scalar quantities is not easily extended to multichannel data, since there is no universal way to define ordering in vector spaces.

To overcome this problem, distance functions are often utilized to order vectors. As an example,

[1] This work has been partially supported by the KBN

norms to order vectors according to their relative magnitude differences [1, 3, 6].

The orientation difference between two vectors can also be used as their distance measure. This so-called *vector angle criterion* is used by the *Vector Directional Filters* (VDF) to remove vectors with atypical directions [4, 7].

The *Basic Vector Directional Filter* (BVDF) is a ranked-order, nonlinear filter which parallelizes the VMF operation. However, a distance criterion, different from the L_1, L_2 norms used in VMF is utilized to rank the input vectors. The output of the BVDF is that vector from the input set, which minimizes the sum of the angles with the other vectors. In other words, the BVDF chooses the vector most centrally located without considering the magnitudes of the input vectors.

To improve the efficiency of the directional filters, a new method called *Directional-Distance Filter* (DDF) was proposed [4]. This filter retains the structure of the BVDF but utilizes a new distance criterion to order the vectors inside the processing window.

Another efficient rank-ordered technique called Hybrid Directional Filter was presented in [8]. This filter operates on the direction and magnitude of the color vectors independently and then combines them to produce a unique final output.

All standard filters detect and replace well noisy pixels, but their property of preserving pixels which were not corrupted by the noise process is far from the ideal. In this paper we show the construction of a simple, efficient and fast filter which removes noisy pixels, but has the ability of preserving original image pixel values.

2. NEW FILTERING TECHNIQUE

Let us start from a gray scale image in order to better explain how the new algorithm is constructed. Let the gray scale image be represented by a matrix F of size $N_1 \times N_2$, $F = \{F(i,j) \in \{0,\dots,255\},\ i = 1,2,\dots N_1,\ j = 1,2,\dots,N_2\}$.

Our construction starts with the introduction of the similarity function $\mu : [0;\infty) \to \mathbf{R}$. We will need the following assumptions for μ:

1. μ is decreasing in $[0;\infty)$,

2. μ is convex in $[0;\infty)$,

3. $\mu(0) = 1, \quad \mu(\infty) = 0$.

In the construction of our filter, the central pixel in the window W is replaced by that one, which maximizes the sum of similarities between all its neighbours. Our basic assumption is that a (introducing pixels which do not occur in the image is prohibited like in the VMF and BVDF).

For this purpose μ must be convex, which can be easily shown. For the gray scale images we define the following fuzzy measure of similarity between two pixels F_k and F_l [11] :

$$\rho\{F_k, F_l\} = \mu(|F_k - F_l|)\,. \tag{1}$$

Let us now assume that F_0 is the center pixel in the window W and the pixels F_1, F_2, … , F_{n-1} are surrounding F_0, (Fig. 1).

The filter works as follows:
In the first step the total sum R_0 of the similarities between the central pixel F_0 (suspected to be noisy) and its neighbours $F_i, i = 1,\dots,n$ is calculated. In the second step each of the neighbours of the central pixel F_0 is moved to the center of the filtering window and the central pixel is removed from W. For each pixel F_i of the neighbourhood, which is being placed in the center of W, the total sum of similarities R_i is calculated and then compared with R_0. It has to be stressed that in the second step the total sum of similarities is calculated without taking into account the original central pixel, which is rejected from the filter window.

In this way, the central pixel F_0 is replaced by that F_i from the neighbourhood, for which the total similarity function R_i, which is a sum of all values of similarities between the central pixel and its neighbours reaches its maximum. In other words if for some i

$$R_i = \sum_{j=1}^{n-1}(1-\delta_{i,j})\,\rho\{F_i, F_j\}\,, i = 1,\cdots,n-1\,, \tag{2}$$

is larger than

$$R_0 = \sum_{j=1}^{n-1}\rho\{F_0, F_j\}\,. \tag{3}$$

then the center pixel is replaced by F_i.

Generally the pixel F_0 is given the value F_{i_*} where $i_* = \arg\max_i R_i$

$$R_i = \delta_{i,0}\sum_{j=1}^{n-1}\rho\{F_i, F_j\} + (1-\delta_{i,0})\sum_{j=1}^{n-1}(1-\delta_{i,j})\,\rho\{F_i, F_j\} \tag{4}$$

This approach can be in a easily applied to color images. In this case, we use the similarity function defined by $\rho\{\mathbf{F}_k, \mathbf{F}_l\} = \mu(||\mathbf{F}_k - \mathbf{F}_l)||$, where $||\cdot||$ denotes the specific vector norm.

Now in exactly the same way we maximize the

In finding the maximum in (4), we obtain $(n-1)$ nonzero components in R_0. If we replace the central pixel by one of its neighbourhood (by F_2 in Fig.1 a), then we obtain only $(n-2)$ nonzero components in R, as the pixel which has been put into the center disappears from the filter window (Fig. 1 b). In this way the filter replaces the central pixel only when it is really noisy and preserves the image structures.

The *BASIC* code, which can be used for the fast computer implementation (L_1 vector norm) is presented in the APPENDIX.

3. RESULTS

The performance of the new algorithm was compared with the standard procedures of noise reduction used in color image processing.

The color image *LENA* has been contaminated by 4% of impulsive "salt & pepper" noise added independently to each RGB channel.

The root of the mean squared error (RMSE), peak signal to noise ratio (PSNR), normalized mean square error (NMSE) have been used as quantitative measures of quality for evaluation purposes.

We investigated the behaviour of the proposed filter using various convex functions in order to compare the new approach with the standard filters presented in Tab. 1, and obtained the best results when applying the following :

$$\mu_1(x) = e^{-\beta_1 x}\ , \qquad \beta_1 \in (0;\infty), \qquad (5)$$

$$\mu_2(x) = \frac{1}{1+\beta_2 x}\ , \qquad \beta_2 \in (0;\infty), \qquad (6)$$

$$\mu_3(x) = \frac{1}{(1+x)^{\beta_3}}\ , \qquad \beta_3 \in (0;\infty), \qquad (7)$$

$$\mu_4(x) = 1 - \frac{2}{\pi}\arctan(\beta_4 x)\ ,\ \beta_4 \in (0;\infty), \quad (8)$$

$$\mu_5(x) = \frac{2}{1+e^{\beta_5 x}}\ , \qquad \beta_5 \in (0;\infty), \qquad (9)$$

$$\mu_6(x) = \frac{1}{1+x^{\beta_6}}\ , \qquad \beta_6 \in (0;1), \qquad (10)$$

$$\mu_7(x) = \begin{cases} 1-\beta_7 x & \text{if } x < 1/\beta_7, \\ 0 & \text{if } x \geq 1/\beta_7, \end{cases}\ \beta_7 \in (0;\infty)\,. \ (11)$$

There are no special reasons to choose exactly these forms of the similarity function. One can easily find other μ fuctions, which meet the required conditions and also yield good filter results. We expect that there exists something like an "optimal shape" of the similarity function, but statistical properties of the noise and the image structure.

Table 2 gives the best values of parameters β_i for functions μ_i and test images distorted by impulsive "salt & pepper" noise up to 10 % on each RGB channel. Figure 6 shows the graphs of these functions. According to the results depicted in Tab. 3 and extensive simulations with other noise intensities and color test images we suppose that an optimal shape of the similarity function is somewhere between μ_5 and μ_7.

Obviously, the presented functions $\mu_1, \ldots, \mu_7$ are rather plain and it is easy to propose procedures which can give us function closer to the optimal one. We did not do it for two reasons. Firstly, more complicated form of the similarity function makes the filter significantly slower. Secondly, we do not think that significant improvement of the filter efficiency is possible.

Table 3 summarizes the results obtained for the test image LENA distorted by 4 % impulsive noise. We have used the L_2 norm and the values of β_i from Tab. 2 in order to obtain results shown in Tab. 3. All proposed functions μ give very good results, although especially worth attention are μ_1, μ_5, μ_7. Table 4 shows *RMSE* values obtained using the proposed filter for four different norms. As can be seen, the best choice is as expected L_2.

The efficiency of the new filtering technique as compared with the vector median and other related filters is shown in Fig. 3.

Figure 4 depicts the result of noise reduction using the new method applied to a gray scale image *LENA* in comparison with the standard median filter. The test image was contaminated by 4% 'salt & pepper' noise and a 3×3 filtering mask was used. As can be seen the new class of filters eliminates efficiently impulsive noise, while preserving important image structures like edges, corners, lines and fine texture.

Another interesting property of the presented method of noise attenuation is shown in Fig. 5. Iterating the filtration process improves the image quality, whis is not the case when using the standard VMF. Additionally the output image converges much faster to its root than the VMF, when we repeat the filtration process.

4. CONCLUSIONS

In this letter, a new class of filters has been presented. Experimental results included in this letter, indicate that the new method of noise reduction significantly outperforms standard procedures used to restore gray scale and color images nique is fast and very easy to implement. The BASIC code is given in the APPENDIX, so that the filter can be easily evaluated by the image processing community.

5. REFERENCES

[1] I. Pitas, A. N. Venetsanopoulos, 'Nonlinear Digital Filters : Principles and Applications', Kluwer Academic Publishers, Boston, MA, (1990)

[2] K.N. Plataniotis, A.N. Venetsanopoulos, 'Color Image Processing and Applications', Springer Verlag, (June 2000)

[3] I. Pitas, P. Tsakalides, Multivariate ordering in color image processing, IEEE Trans. on Circuits and Systems for Video Technology, 1, 3, 247-256, (1991)

[4] D.G. Karakos, P.E. Trahanias, Generalized multichannel image filtering structures, IEEE Trans. on Image Processing, 6, 7, 879-881, (1996)

[5] I. Pitas, A.N. Venetsanopoulos, Order statistics in digital image processing, Proc. of IEEE, 80, 12, 1893-1923, (1992)

[6] J. Astola, P. Haavisto, Y. Neuovo, Vector median filters, IEEE Proc., 78, 678-689, (1990)

[7] P.E. Trahanias, A.N. Venetsanopoulos, Vector directional filters: A new class of multichannel image processing filters, IEEE Trans. on Image Processing, 2, 4, 528-534, (1993)

[8] M. Gabbouj, F.A. Cheickh, Vector Median-Vector Directional hybrid filter for color image restoration, Proc. of EUSIPCO-96, 879-881, (1996)

[9] K.N. Plataniotis, D. Androutsos, A.N. Venetsanopoulos, Color image processing using adaptive vector directional filters, IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing, 45, 10, 1414-1419, (1998)

[10] K.N. Plataniotis, D. Androutsos, V. Sri, A.N. Venetsanopoulos, A nearest neighbour multichannel filter, Electronics Letters, 1910-1911, (1995)

[11] B. Smolka, K. Wojciechowski, Random walk approach to image enhancement, Signal Processing, Vol. 81, 465-482, (2001)

[12] B. Smolka, K.N. Plataniotis, A. Chydzinski, A. N. Venetsanopoulos, K. W. Wojciechowski, On the Reduction of Impulsive Noise in Multichannel Image Processing, Optical Engineering, Vol. 40, No. 6, 902-908, (2001).

6. APPENDIX

```
BASIC CODE OF THE NEW ALGORITHM
'cr(N1,N2), cg(N1,N2), cb(N1,N2) - input color image,
'wr(N1,N2), wg(N1,N2), wb(N1,N2) - output color image
'beta - similarity function coefficient
'sim - total similarity between pixels in 3x3 window
For i=0 To 255
For j=0 To 255
expo(i,j)=Exp(-beta*Abs(i-j))
Next
Next
For i=2 To N1-1
For j=2 To N2-1
max=-1
For g=-1 To 1
For h=-1 To 1
w=i+g
z=j+h
sim=0
For r=-1 To 1
For s=-1 To 1
x=i+r
y=j+s
If Not w=x Or Not z=y Then
If Not r=0 Or Not s=0 Then
simr=expo(cr(x,y),cr(w,z))
simg=expo(cg(x,y),cg(w,z))
simb=expo(cb(x,y),cb(w,z))
sim=sim+simr+simg+simb
End If
End If
Next
Next
If sim>max Then
max=sim
pixr=cr(w,z)
pixg=cg(w,z)
pixb=cb(w,z)
End If
Next
Next
wr(i,j)=pixr
wg(i,j)=pixg
wb(i,j)=pixb
Next
Next
```

Notation	Filter	Reference
AMF	Arithmetic Mean Filter	[1]
VMF	Vector Median Filter	[6]
ANNF	Adaptive Nearest Neighbor Filter	[10]
BVDF	Basic Vector Directional Filter	[7]
HDF	Hybrid Directional Filter	[8]
AHDF	Adaptive Hybrid Directional Filter	[8]
DDF	Directional-Distance Filter	[4]
FVDF	Fuzzy Vector Directional Filter	[9]

Table 1. Filters compared

β_1	β_2	β_3	β_4	β_5	β_6	β_7
5,04	6,62	192	6,97	7,90	266	3,72

Table 2. Optimal values of constans β_i $[10^{-3}]$.

METHOD	**NMSE** $[10^{-4}]$	**RMSE**	**PSNR** [dB]
NONE	514,95	32,165	17,983
AMF	82,863	12,903	25,917
VMF	23,304	6,842	31,427
ANNF	31,271	7,926	30,149
BVDF	29,074	7,643	30,466
HDF	22,845	6,775	31,513
AHDF	22,603	6,739	31,559
DDF	24,003	6,944	31,288
FVDF	26,755	7,331	30,827
PROPOSED			
$\mu_1(x)$	**4,959**	**3,157**	**38,145**
$\mu_2(x)$	5,398	3,294	37,776
$\mu_3(x)$	9,574	4,387	35,288
$\mu_4(x)$	5,064	3,190	38,054
$\mu_5(x)$	**4,777**	**3,099**	**38,307**
$\mu_6(x)$	11,024	4,707	34,675
$\mu_7(x)$	**4,693**	**3,072**	**38,384**

Table 3. Comparison of the new filter with the standard techniques (*LENA* color image contaminated with 4% "salt & pepper" noise added independently to each RGB channel).

	L_1	L_2	L_3	L_∞
$\beta_1(x)$	3,615	**3,157**	3,172	3,462
$\beta_5(x)$	3,579	**3,099**	3,167	3,694
$\beta_7(x)$	3,838	**3,072**	3,138	3,752

Table 4. Comparison of the new filter results (RMSE) using different norms (*LENA*).

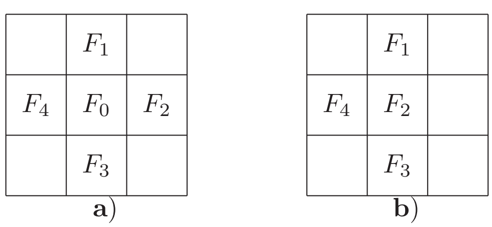
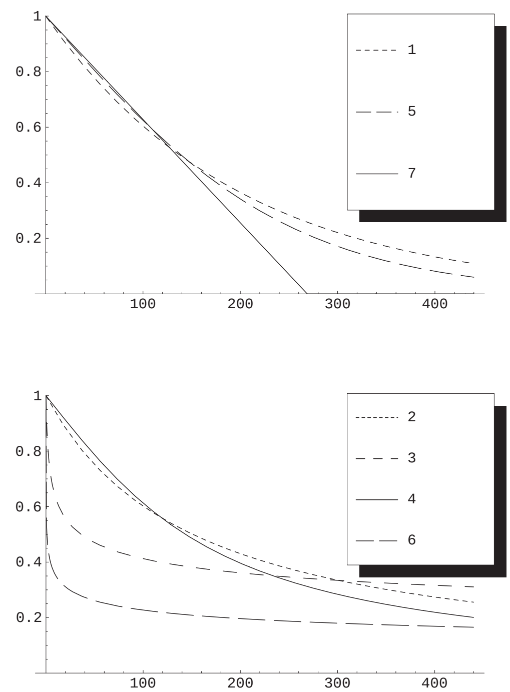


Fig. 1. Illustration of the construction of the new filtering technique for the 4-neighbourhood case. If the center pixel F_0 is replaced by its neighbour F_2, then the similarity measure $R_2 = \rho\{F_2, F_1\} + \rho\{F_2, F_3\} + \rho\{F_2, F_4\}$between F_2 (new center pixel) is calculated. If the total similarity R_2 is greater than $R_0 = \rho\{F_0, F_1\} + \rho\{F_0, F_2\} + \rho\{F_0, F_3\} + \rho\{F_0, F_4\}$ then the center pixel is replaced, otherwise it is retained.

Fig. 2. Similarity functions $\mu_1, \ldots, \mu_7$.



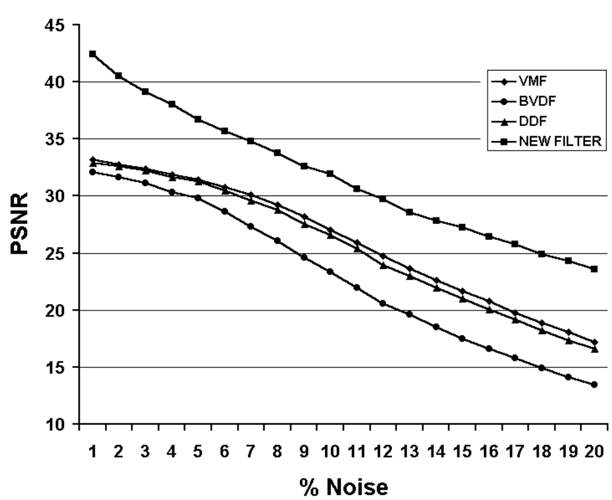


Fig. 3. Dependence of the noise reduction efficiency on the percentage of impulsive noise for the new method, VMF, BVDF and DDF, (*LENA* colour image, $\beta_1 = 5.04 \cdot 10^{-3}$).

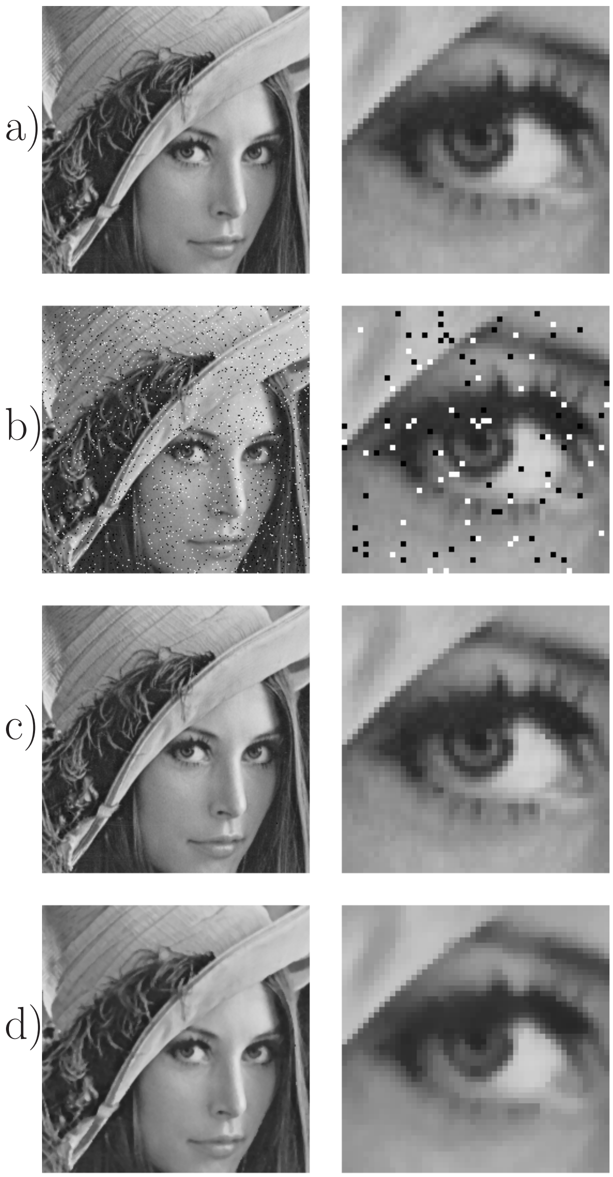


Fig. 4. Noise reduction effect of the proposed filter as compared with the median filter: a) gray scale test image *LENA*, b) image distorted by 4% 'salt & pepper' noise, c) filtered with the new method $\beta_1 = 5.04 \cdot 10^{-3}$ (PSNR=42.02), d) median filter (PSNR=34.08). To the right zoomed image portions.

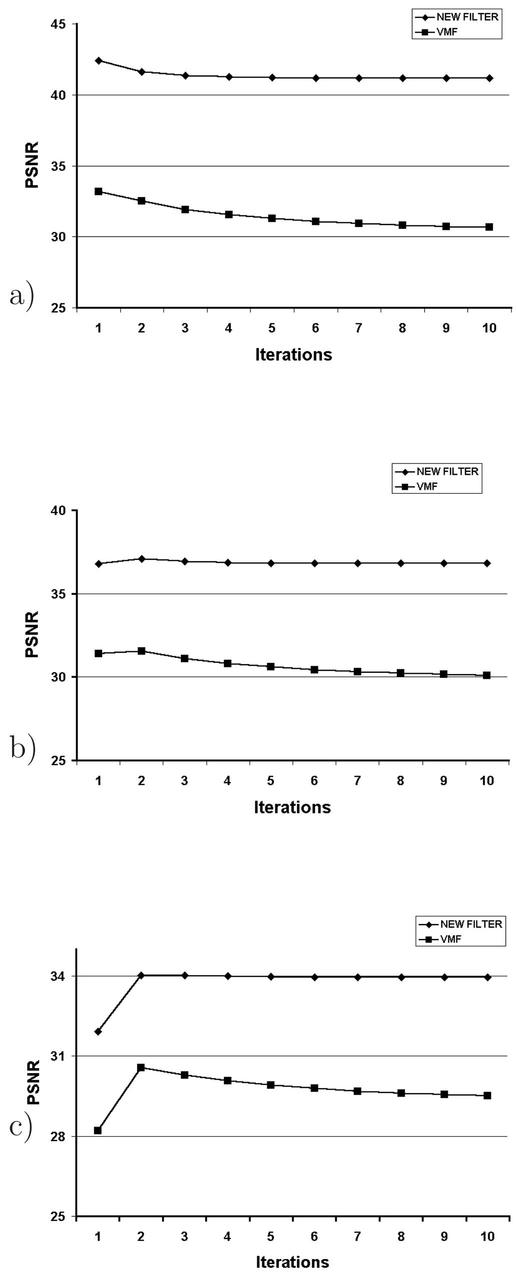


Fig. 5. Dependance of the noise reduction efficiency of the proposed filter and VMF on the number of iterations for colour test image distorted by: a) 1% impulsive noise b) 5% impulsive noise c) 10% impulsive noise. (*LENA* colour image, $\beta_1 = 5.04 \cdot 10^{-3}$).