

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Iterative Self-adaptive Filtering Algorithm for Reducing Impulsive Noise in Color Images

^{1,2,3}Wang Shitong, ^{1,3}Li Yueyang, ²F.L. Chung and ¹Chen Shu

¹School of Information, Southern Yangtse University, WuXi, JiangSu, China

²Department of Computing, Hong Kong Polytechnic University, Hong Kong, China

³National Key Laboratory of Computer Science, Institute of Software, CAS SINICA, China

Abstract: In this study, an iterative self-adaptive filtering algorithm based on local extreme value noise detection for impulsive noise reduction in color images was presented. First, the new filter detects the corrupted image pixels; then, it replaces the corrupted pixels and preserves the original undistorted image pixels. The results here demonstrate that the proposed algorithm outperforms the standard multichannel filters in reducing impulsive noise and retaining edges and corners in color images. The distinctive advantage of the proposed algorithm is its very simplicity yet excellent performance.

Key words: Self-adaptive algorithm, color image processing, impulsive noise, multichannel filters

INTRODUCTION

During the last decade, much attention has been devoted to the study of the arising field of vector (or multichannel) image processing. The most common image processing tasks are noise filtering and image enhancement.

Typically, a color image is represented in each pixel by a three components vector. The vector components are in general the amounts of pure red, green and blue that compose the local color. It has been widely recognized that the processing of color image data as vector fields is desirable due to the correlation that exists among the image channels.

A number of color image filtering techniques have been developed to utilize correlations among color vectors effectively^[1-3]. The popular nonlinear, multichannel filters are based on the ordering of vectors in a predefined moving window. Let $F(x)$ 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 is denoted as F_j , $j = 0, 1, \dots, n-1$. If the distance between two vectors F_i, F_j is denoted as $\rho(F_i, F_j)$, then the scalar quantity $R_i = \sum_{j=0}^{n-1} \rho(F_i, F_j)$

is the distance associated with the vectors F_i . The ordering of the R_i 's: $R_{(0)} \leq \dots \leq R_{(n-1)}$, implies the same ordering to the corresponding vectors F_i : $F_{(0)} \leq \dots \leq F_{(n-1)}$. Non-linear ranked type multichannel estimators define the vector $F_{(0)}$ as the filter output.

The most typical nonlinear vector filter is the Vector Median Filter (VMF). VMF uses the L_1 or L_2 norm to order

vectors according to their relative magnitude differences. The output of the VMF is that vector from the input set, which minimizes the sum of the distances with the other vectors^[9]. Trahanias *et al.*^[10] developed the Basic Vector Directional Filter (BVDF), in which the directional information among vectors is utilized instead of the distance information in VMF. BVDF outputs the vector that minimizes the sum of the angles with all the other vectors. Having combined BVDF and VMF, Karakos and Trahanias^[5] proposed the Directional-distance Filter (DDF), which retains the structure of the BVDF but utilizes a new distance criterion to order the vectors inside the processing window^[5].

Although the well-known vector filters hold good impulsive noise attenuation characteristics, their performance is often accompanied with undesired processing of noise-free image pixels, which results in edge and texture blurring. The reason is that these nonlinear filters do not satisfy the superposition property (its nonlinearity is caused by the ordering operation) and thus, the optimal filtering situation can be never fully achieved.

In the case of the impulsive noise corruption, the aim of the optimal filtering is to design noise reduction algorithms that would affect only corrupted image pixels, whereas the undistorted image pixels should be invariant under the filtering operation. In this study, we present an iterative self-adaptive filtering algorithm based on local extreme value noise detection of impulsive noise reduction. The new filter removes disturbed pixels, but has the ability of preserving noise-free pixels. The new filter can be easily implemented and our experimental

results demonstrate its excellent performance in reducing impulsive noise in color images.

The iterative self-adaptive filtering algorithm: Let us start from a gray scale images in order to better explain how the new algorithm is constructed. Let the gray scale image be represented by a matrix F of size $N_1 \times N_2$, $F = \{F(i,j) \in \{0,1,\dots,255\}, i = 1,2,\dots,N_1, j = 1,2,\dots,N_2\}$. When the image is corrupted with impulsive *salt and pepper* noise (pixel values are randomly replaced by 0 or 255 with equal probability $p/2$), the pixel $F(i, j)$ is denoted as:

$$F(i,j) = \begin{cases} 255, & p/2 \\ F(i,j) & 1-p \\ 0 & p/2 \end{cases} \quad (1)$$

The basic idea of the new algorithm can be described as follows:

- (1) Detecting the distorted pixels;
- (2) Designing noise reduction algorithm that would only affect assumed corrupted image pixels.

Noise detector: Let $W[F(i, j)]$ be the noise detection window, the central pixel of which is $F(i,j)$. In the window $W[F(i, j)]$, if the gray value of the central pixel $F(i, j)$ is far from the value of its neighbors (namely, the similarity between them is very small), it may perhaps be distorted by the impulsive noise. So we sequentially select the moving detection window $W[F(i, j)]$ and find the pixel, whose gray value is maximum or minimum in $W[F(i, j)]$. If the gray value of the central pixel $F(i, j)$ is equal to the maximum or minimum, $F(i, j)$ is assumed to be corrupted; otherwise, $F(i, j)$ is original pixel. It can be denoted as:

$$f_{ij} = \begin{cases} 1, & \text{if } F(i, j) = \max(W[F(i, j)]) \text{ or } \min(W[F(i, j)]) \\ 0, & \text{else} \end{cases} \quad (2)$$

where, $\max(W[F(i, j)])$ and $\min(W[i, j])$ is the maximal and minimal gray value in the window $W[F(i, j)]$, respectively. $[f_{ij}]$ is a binary matrix, which represents the noise distribution corresponding to the corrupted image. When $f_{ij} = 1$, it implies that the pixel $F(i, j)$ is corrupted by noise; otherwise ($f_{ij} = 0$), $F(i, j)$ is noise-free.

In this way, we can obtain all pixels corrupted by impulsive noise. It must be noted here that there may be a small number of original pixels assumed to be noise pixels. After noise reduction as follows, the output of the operating window is the pixel, which is one of the neighbors considered noise-free. So in this case, the effect to the results is small. On the other hand, we also can expand the detection window to reduce the possibility

that we may consider some of original pixels as noise pixels.

Noise reduction algorithm: The similarity function $u(x)$ is introduced here with the following assumptions for $u(x)^{[7]}$:

1. $u(x)$ is non-ascending in $[0, \infty)$;
2. $u(x)$ is convex in $[0, \infty)$;
3. $u(0) = 1, u(\infty) = 0$.

The argument of the function $u(x)$ is a distance between pixels in gray scale space. The fact that $u(x)$ must be non-ascending means that the similarity between two pixels is small if the distance between them in a given space is large. Assumption 3 is just a normalization of the similarity function. In this way, the similarity between two pixels with the same gray scale value is 1, the similarity between pixels with far distance intensities is 0.

In the construction of our new filter, the central pixels in the window W assumed to be corrupted by impulsive noise is replaced by that one, which maximizes the sum of similarities between all its noise-free neighbors. The output pixel must be taken from the filtering window W and introducing pixels, which do not occur in the image, are prohibited.

Smolka *et al.*^[7] proposed the following similarity function $u(x)$

$$u(x) = e^{-\beta x}, \quad \beta \in (0, \infty) \quad (3)$$

Adaptive selection of the parameter β is also given by Smolka *et al.*^[7].

For the gray scale images, we may adopt the following fuzzy measure of similarity between two pixels F_i and F_j ^[8]:

$$\rho(F_i, F_j) = u(|F_i - F_j|) \quad (4)$$

We assume that F_0 is the central pixel in the window W and that the pixels F_1, F_2, \dots, F_{n-1} are surrounding F_0 . The central pixel F_0 is replaced by the F_{i^*} from the neighborhood of F_0 , for which the total similarity function R_{i^*} reaches its maximum. Generally, according to the method by Smolka *et al.*^[7], the pixel F_0 is given by the value F_{i^*} , where, $i^* = \text{argmax } R_i$ and

$$R_i = \delta_{i,0} \sum_{j=1}^{n-1} \rho(F_i, F_j) + (1 - \delta_{i,0}) \sum_{j=1}^{n-1} (1 - \delta_{i,j}) \rho(F_i, F_j) \quad (5)$$

where, $\delta_{i,j}$ is defined as: $\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$

This approach can also be applied in a straightforward way to color images. We use the similarity function defined by $\rho(F_i, F_j) = u(\|F_i - F_j\|)$ where, $\|\cdot\|$ is the specific vector norm. Here we use L_2 norm, since it has been demonstrated to be the best choice^[7].

Let $F = \{F(i, j, k) \in \{0, 1, \dots, 255\}, i = 1, 2, \dots, N_1, k = 1, 2, 3\}$ represent a color image in RGB color space. The iterative self-adaptive filtering algorithm based on local extreme value noise detection is described as follows:

Noise detecting: All test noisy images are generated by MATLAB 6.x, each channel of which is corrupted independently with impulsive salt and pepper noise. So we detect corrupted pixels in three channels of the images independently and get the binary matrix $[f_{ijk}]$.

Noise filtering: We defined $\{F^{(0)}, F^{(1)}, \dots, F^{(N)}\}$ as the output image set and $\{[f_{ijk}^{(0)}], [f_{ijk}^{(1)}], L, [f_{ijk}^{(N)}]\}$ as the binary matrix set. $F^{(0)}$ represents the original corrupted image; $F^{(m)}$ ($m=1, 2, \dots, N$) represents the filtering image after m iterations. $[f_{ijk}^{(0)}]$ denotes the binary matrix $[f_{ijk}]$ after noise detection; $[f_{ijk}^{(m)}]$ denotes the binary matrix after m iterations.

The filter works in three channels of the image independently. In iteration m , the novel algorithm is described as follows:

1. If $F(i, j, k)^{(m-1)}$ is noisy pixel ($f_{ijk}^{(m-1)}$ equals 1), let $W[F(i, j, k)^{(m-1)}]$ be a filtering window, and its central pixel is $F(i, j, k)^{(m-1)}$.
2. Selecting n noise-free pixels in $W[F(i, j, k)^{(m-1)}]$, where, n denotes the number of the noise-free pixels.
3. If $n > 0$, computing the output $F(i, j, k)^{(m)}$ of the window $W[F(i, j, k)^{(m-1)}]$ using (5); otherwise, preserving the original pixel. Totally, we may summarized as:

$$F(i, j, k)^{(m)} = \begin{cases} F(i, j, k)^{(m-1)}, & \text{if } f_{ijk}^{(m-1)} = 1 \text{ and } n > 0 \\ F(i, j, k)^{(m-1)}, & \text{else} \end{cases} \quad (7)$$

4. If the pixel $F(i, j, k)^{(m-1)}$ is replaced by one of its neighbors, then $f_{ijk}^{(m)} = 0$. Otherwise, $f_{ijk}^{(m)}$ keeps unchangeable. That is to say:

$$f_{ijk}^{(m)} = \begin{cases} 0, & \text{if } F(i, j, k)^{(m)} = F(i, j, k)^{(m-1)} \\ f_{ijk}^{(m-1)}, & \text{if } F(i, j, k)^{(m)} \neq F(i, j, k)^{(m-1)} \end{cases} \quad (8)$$

After N iterations, if all the noisy pixels are replaced

($\sum_i \sum_j \sum_k f_{ijk}^{(N)} = 0$), then stop and $F^{(N)}$ is taken as the final output.

Obviously, the algorithm here can be easily realized. For an image corrupted by impulsive noise, because not

all pixels are corrupted, the new algorithm keeps the original value if a pixel is noise-free while the corrupted pixel is replaced by one of its noise-free neighbors. In this way, the new algorithm replaces the pixel only when it is really noisy, and meanwhile, it preserves the original undistorted image structures very well.

RESULTS

Several experimental results are presented in this section to demonstrate the power of our new filtering algorithm. Comparisons among the new filter and three standard filters of noise reduction in color image processing are also done here.

In order to make the comparisons fair, we utilize the following quantitative measures to evaluate the image quality. They are:

$$MSE = \frac{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \|F(i, j) - \hat{F}(i, j)\|^2}{3 \times N_1 \times N_2}$$

$$RMSE = \sqrt{MSE}$$

$$PSNR = 20 \times \log_{10} \left(\frac{255}{RMSE} \right)$$

$$NMSE = \frac{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \|F(i, j) - \hat{F}(i, j)\|}{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \|F(i, j)\|}$$

$$MAE = \frac{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |F(i, j) - \hat{F}(i, j)|}{3 \times N_1 \times N_2}$$

where, $F(i, j)$ and $\hat{F}(i, j)$ are the original pixel vector and the restored vector, respectively. $\|\cdot\|$ and $|\cdot|$ denote the L_2 and L_1 norm, respectively.

The effectiveness of the new filter was tested using the standard LENA and PEPPERS images, which have been contaminated by 5, 10, 20, 40 and 60% impulsive noise. The performance of the presented method was compared with VMF, BVDF, DDF and the adaptive filter^[7].

Table 1 and 2; Fig. 1-3 show the denoising effect of the new filter algorithm for impulsive noise ranging from 5 to 60% in color images. Table 1 and 2 depict the results of image filtering using the new algorithm in comparison with VMF, BVDF, DDF and the adaptive filter^[7] based on PSNR measure. Figure 1 and 2 depict the performances of the new filter for the LENA and PEPPERS images distorted with 5% impulsive noise in comparison with the other four filters.



Fig. 1: Noise reduction effect of the proposed filter as compared with the adaptive filter^[7], VMF, BVDF and DDF: (a) color test image LENA, (b) image distorted by 5% impulsive noise, (c) the proposed filter, (d) the adaptive filter^[7], (e) VMF, (f) BVDF, (g) DDF

Table 1: Results of noise reduction of the new filter compared with the other four filters. The test image LENA was contaminated by impulsive noise ranging from 5 to 60%

	RMSE	PSNR	NMSE (10 ⁻³)	MAE
No filtering				
5%	31.485	41.835	50.480	6.417
10%	44.614	34.864	101.352	12.935
20%	62.405	28.153	198.302	25.307
40%	89.268	20.992	405.779	51.565
60%	108.434	17.102	598.726	76.281
VMF				
5%	9.981	64.813	5.072	5.059
10%	11.027	62.819	6.191	5.714
20%	15.273	56.304	11.877	7.521
40%	38.475	37.824	75.381	17.036
60%	70.437	25.731	252.639	39.341
BVDF				
5%	11.632	61.750	6.890	5.782
10%	13.724	58.442	9.591	6.819
20%	21.832	49.157	24.272	9.827
40%	57.841	29.671	170.362	28.985
60%	93.359	20.096	443.816	61.842
DDF				
5%	10.036	64.701	5.129	5.008
10%	11.209	62.490	6.398	5.750
20%	15.966	55.417	12.980	7.635
40%	40.791	36.656	84.727	18.205
60%	74.448	24.623	282.232	42.704
The adaptive filter ^[7]				
5%	6.716	72.736	2.297	1.390
10%	9.617	65.554	4.710	2.753
20%	13.402	58.917	9.146	5.453
40%	21.998	49.006	24.642	10.537
60%	48.159	33.335	118.101	24.604
The proposed filter				
5%	4.549	80.527	1.054	0.828
10%	4.886	79.096	1.216	1.001
20%	6.394	73.718	2.082	1.655
40%	10.449	63.895	5.560	3.730
60%	19.605	51.310	19.572	7.322

Table 2: Results of noise reduction of the new filter compared with the other four filters. The test image PEPPERS was contaminated by impulsive noise ranging from 5 to 60%

	RMSE	PSNR	NMSE (10 ⁻³)	MAE
No filtering				
5%	31.811	41.629	65.474	6.336
10%	45.213	34.598	132.262	12.662
20%	64.534	27.481	269.454	25.840
40%	90.957	20.618	535.280	51.174
60%	110.941	16.645	796.331	76.322
VMF				
5%	10.205	64.367	6.739	5.219
10%	11.518	61.947	8.583	6.063
20%	16.201	55.123	16.982	8.141
40%	39.978	37.059	103.409	17.954
60%	71.435	25.450	330.165	39.394
BVDF				
5%	14.316	57.598	13.260	6.697
10%	17.574	53.497	19.982	8.136
20%	26.772	45.079	46.372	11.801
40%	65.874	27.070	280.762	33.152
60%	104.116	17.915	701.367	69.094
DDF				
5%	10.335	64.114	6.911	5.207
10%	11.892	61.307	9.150	6.095
20%	17.087	54.059	18.891	8.395
40%	44.639	34.853	128.928	19.947
60%	78.731	23.504	401.053	44.762
The adaptive filter ^[7]				
5%	6.813	72.450	3.003	1.392
10%	9.677	65.431	6.058	2.650
20%	14.833	56.888	14.235	6.053
40%	24.023	47.245	37.338	11.595
60%	49.848	32.646	160.767	25.620
The proposed filter				
5%	4.952	78.831	1.586	0.954
10%	5.399	77.100	1.886	1.169
20%	7.427	70.722	3.569	2.020
40%	11.639	61.739	8.764	4.185
60%	21.318	49.634	29.405	8.204

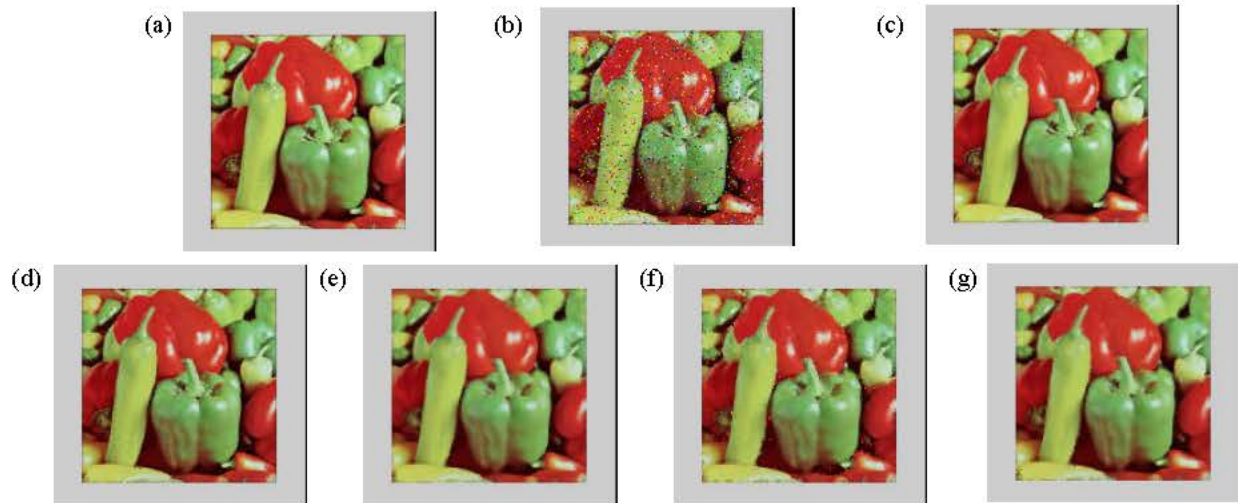


Fig. 2: Noise reduction effect of the proposed filter as compared with the adaptive filter^[7], VMF, BVDF and DDF: (a) color test image PEPPERS, (b) image distorted by 5% impulsive noise, (c) the proposed filter, (d) the adaptive filter^[7], (e) VMF, (f) BVDF, (g) DDF

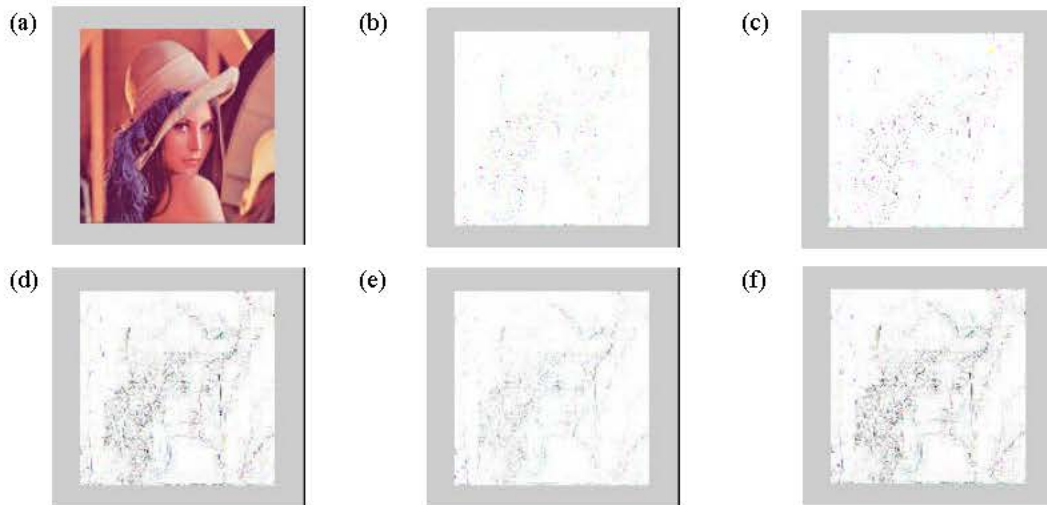


Fig. 3: The effect of preserving edges and details of the proposed filter as compared with the adaptive filter^[7], VMF, BVDF and DDF: (a) color test image LENA, (b)-(f) the absolute difference between the original and filtered image ((b) the proposed filter, (c) the adaptive filter^[7], (d) VMF, (e) BVDF, (f) DDF)

Figure 3 shows the absolute difference between the original and each filtered image for the corrupted image LENA. It can be seen that the difference between the original and the new-filter-based image has the minimal magnitude, which implies that the ability of preserving edges and details of the new algorithm is quite well.

CONCLUSIONS

The new filtering algorithm proposed in this study indeed improves the popular nonlinear, multichannel filters in reducing impulsive noise in color images. Based on local extreme value noise detection, the new filter only

affects corrupted image pixels, whereas the noise-free pixels are invariant under the filtering operation. In this way, the new filter well outperforms the standard multichannel filters commonly used in color image processing and achieves better capability of preserving the original structure of the image. Besides, very easy implementation is also another distinctive advantage.

ACKNOWLEDGEMENTS

This study here is supported by the Hong Kong PolyU CRG (grant No. G-T912) and HongKong RGC Competitive Earmarked Research Grant (grant No.

5065/98E), National Science Foundation of China (grant No. 60225015), Natural Science Foundation of JiangSu Province (grant No. BK2003017), National Key Lab. of Novel Software Technologies at NanJing University, The JiangSu Key Lab. of Information Processing, National Key Lab. of Computer Science at Institute of Software of CAS SINICA (Grant No. SYSKF0406) and Excellent Young Teacher Research Grant of Ministry of Education, China.

REFERENCES

1. Venetsanopoulos, A.N. and K.N. Plataniotis, 1995. Multichannel image processing. Proceedings of the IEEE Workshop on Nonlinear Signal/Image Processing, pp: 2-6.
2. Pitas, I. and A.N. Venetsanopoulos, 1990. *Nonlinear Digital Filters: Principles and Applications*, Kluwer Academic Publishers. Boston, MA, 1990.
3. Plataniotis, K.N. and A.N. Venetsanopoulos, 2000. *Color image processing and applications*. ISBN 3-540-66953-1, Springer, Berlin, 2000.
4. Pitas, I. and P. Tsakalides, 1991. Multivariate ordering in color image processing. *IEEE Trans. Circuits and Systems for Video Technology*, 1: 247–256.
5. Karakos, D.G. and P.E. Trahanias, 1995. Combining vector median and vector directional filters: The directional-distance filters. *Proc. IEEE Conf. Image Process. ICIP-95*, pp: 171-174.
6. Pitas, I. and A.N. Venetsanopoulos, 1992. Order statistics in digital image processing. *Proc. IEEE* 80, 12: 1893-1923.
7. Smolka, B., K.N. Plataniotis, A. Chydzinski, M. Szczepanski, A.N. Venetsanopoulos and K. Wojciechowski, 2002. Self-adaptive algorithm of impulsive noise reduction in color images. *Pattern Recognition*, 35: 1771-1784.
8. Trahanias, P.E. and A.N. Venetsanopoulos, 1993. Vector directional filters: A new class of multichannel image processing filters. *IEEE Trans. Image Proc.*, 2,4: 528-534.
9. Astola, J., P. Haavisto and Y. Neuovo, 1990. Vector median filters. *IEEE Proc.*, 78: 678-689.
10. Smolka, B. and K. Wojciechowski, 2001. Random walk approach to image enhancement. *Signal Process.*, 81: 465-482.