SIMILARITY BASED IMPULSIVE NOISE REMOVAL IN COLOR IMAGES

B. Smolka[1] *K.N. Plataniotis*[2] *R. Lukac*[2] *A.N. Venetsanopoulos*[2]

[1] Department of Automatic Control, Silesian University of Technology, Poland
[2] The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto ON, M5S 3G4, Canada

ABSTRACT

In this paper a novel approach to the problem of impulsive noise reduction in color images based on the nonparametric density estimation is presented. The basic idea behind the new image filtering technique is the maximization of the similarities between pixels in a predefined filtering window. The new method is faster than the standard vector median filter (VMF) and preserves better edges and fine image details. Simulation results show that the proposed method outperforms standard algorithms of the reduction of impulsive noise in color images.

1. NOISE REMOVAL IN COLOR IMAGES

A number of nonlinear, multichannel filters, which utilize correlation among multivariate vectors using various distance measures, have been proposed [1-7]. 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 $\mathbf{F}(x)$ represents a multichannel image and let W be a window of finite size $n+1$, (filter length). The noisy image vectors inside the filtering window W are denoted as $\mathbf{F}_j$, $j = 0, 1, ..., n$. 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 $R_i = \sum_{j=0}^{n} \rho(\mathbf{F}_i, \mathbf{F}_j)$ is the distance associated with the noisy vector $\mathbf{F}_i$. The ordering of the R_i 's: $R_{(0)} \leq R_{(1)} \leq ... \leq R_{(n)}$, implies the same ordering to the corresponding vectors $\mathbf{F}_i : \mathbf{F}_{(0)} \leq \mathbf{F}_{(1)} \leq ... \leq \mathbf{F}_{(n)}$. 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, [1,2].

The majority of 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. PROPOSED ALGORITHM

2.1. Gray-scale Images

Let us assume a filtering window W containing $n+1$ image pixels, $\{F_0, F_1, \ldots, F_n\}$, where n is the number of neighbors of the central pixel F_0, (see Fig. 2a) and let us define the similarity function $\mu : [0; \infty) \rightarrow \mathbf{R}$ which is non-ascending in $[0; \infty)$, convex in $[0; \infty)$ and satisfies $\mu(0) = 1$, $\mu(\infty) = 0$. The similarity between two pixels of the same intensity should be 1, and the similarity between pixels with far distant gray scale values should be very close to 0. The function $\mu(F_i, F_j)$ defined as $\mu(F_i, F_j) = \mu(|F_i - F_j|)$ satisfies the three above conditions.

Let us additionally define the cumulated sum M of similarities between the pixel F_k and all its neighbors. For the central pixel and its neighbors we define M_0 and M_k as

$$M_0 = \sum_{j=1}^{n} \mu(F_0, F_j), \quad M_k = \sum_{j=1, j\neq k}^{n} \mu(F_k, F_j), \quad (1)$$

which means that for F_k which are neighbors of F_0 we do not take into account the similarity between F_k and F_0, which is the main idea behind the new algorithm. The omission of the similarity $\mu(F_k, F_0)$ privileges the central pixel, as in the calculation of M_0 we have n similarities $\mu(F_0, F_k)$, $k = 1, 2, \ldots, n$ and for M_k, $k > 0$ we have only $n-1$ similarity values, as the central pixel F_0 is excluded from M_k.

In the construction of the new filter the reference pixel F_0 in the window W is replaced by one of its neighbors if $M_0 < M_k$, $k = 1, \ldots, n$. If this is the case, then F_0 is replaced by that F_k for which $k = \arg\max M_i$, $i = 1, \ldots, n$. In other words F_0 is detected as being corrupted if $M_0 < M_k$, $k = 1, \ldots, n$ and is replaced by its neighbors F_i which maximizes the sum of similarities M between all its neighbors excluding the central pixel. This is illustrated in Figs. 2 and 5.

The basic assumption is that a new pixel must be taken from the window W, (introducing pixels which do not occur in the image is prohibited like in VMF). For this purpose μ must be convex, which means that in order to find a maximum of the sum of similarity functions M it is sufficient to calculate the values of M only in points $F_0, \ldots, F_n$, [7].

0-7803-7750-8/03/$17.00

2.2. Color Images

The presented approach can be applied in a straightforward way to color images. We use the similarity function defined by $\mu\{\mathbf{F}_i, \mathbf{F}_j\} = \mu(||\mathbf{F}_i - \mathbf{F}_j)||$ where $||\cdot||$ denotes the specific vector norm. Now, in exactly the same way we maximize the total similarity function M for the vector case.

We have checked several convex functions in order to compare our approach with the standard filters used in color image processing presented in Tab. 1 and we have obtained very good results (Tab. 2), when applying the following similarity functions, which can be treated as kernels of non-parametric density estimation, [7-9], (see Fig. 4).

$$\mu_0 = \exp\left\{-\left(\frac{x}{h}\right)^2\right\},\ \mu_1 = \exp\left\{-\frac{x}{h}\right\},\ \mu_2 = \frac{1}{1+x/h},$$
$$\mu_3 = \frac{1}{(1+x)^h},\ \mu_4 = 1-\frac{2}{\pi}\arctan\left(\frac{x}{h}\right),\ \mu_5 = \frac{2}{1+\exp\left\{\frac{x}{h}\right\}},$$
$$\mu_6 = \frac{1}{1+x^h},\ \mu_7 = \begin{cases} 1-x/h & \text{if } x \le h, \\ 0 & \text{if } x > h, \end{cases}$$

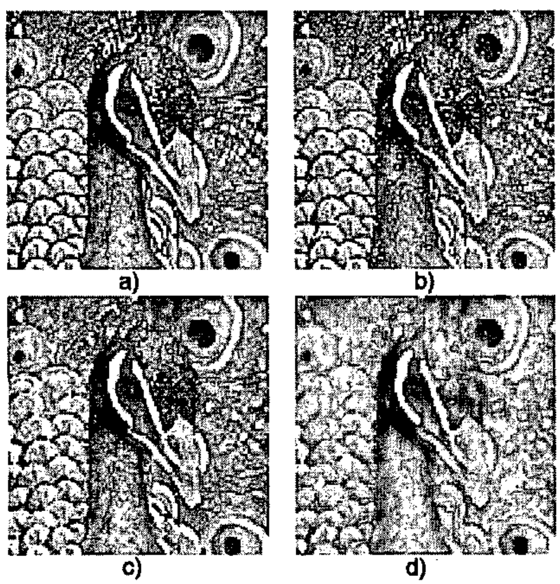

Fig. 1. Illustration of the efficiency of the new algorithm of impulsive noise reduction in color images: **a**) test image, **b**) image corrupted by 4% impulsive *salt & pepper* noise, **c**) new filter output, **d**) effect of median filtering (3×3 mask).

It is interesting to note, that the best results were achieved for the simplest similarity function $\mu_7(x)$, which allows to construct a fast noise reduction algorithm. In the multichannel case, we have

$$M_0 = \sum_{j=1}^{n} \mu(\mathbf{F}_0, \mathbf{F}_j), \quad M_k = \sum_{j=1, j\neq k}^{n} \mu(\mathbf{F}_k, \mathbf{F}_j), \quad (2)$$

where $\rho\{\mathbf{F}_i, \mathbf{F}_k\} = ||\mathbf{F}_k - \mathbf{F}_l)||$ and $||\cdot||$ is the L_2 norm.

Applying the linear similarity function μ_7 we obtain

$$\mu(\mathbf{F}_i, \mathbf{F}_k) = \begin{cases} 1 - \rho(\mathbf{F}_i, \mathbf{F}_k)/h & \text{for} \quad \rho(\mathbf{F}_i, \mathbf{F}_k) < h, \\ 0 & \text{otherwise.} \end{cases}$$

Then we have from (2), $M_0 = n - \frac{1}{h}\sum_{j=1}^{n} \rho(\mathbf{F}_0, \mathbf{F}_j)$, and

$$M_k = \sum_{j=1, j\neq k}^{n} \left(1 - \frac{\rho(\mathbf{F}_k, \mathbf{F}_j)}{h}\right) = n-1-\frac{1}{h}\sum_{j=1}^{n} \rho(\mathbf{F}_k, \mathbf{F}_j).$$

In this way the difference between M_0 and M_k is

$$M_0 - M_k = 1 - \frac{1}{h}\sum_{j=1}^{n} [\rho(\mathbf{F}_0, \mathbf{F}_j) - \rho(\mathbf{F}_k, \mathbf{F}_j)], \quad (3)$$

and

$$M_0 - M_k > 0 \ \text{ if } \quad h > \sum_{j=1}^{n} [\rho(\mathbf{F}_0, \mathbf{F}_j) - \rho(\mathbf{F}_k, \mathbf{F}_j)]. \quad (4)$$

If this condition is satisfied, then the central pixel is considered as not disturbed by the noise process, otherwise the pixel $\mathbf{F}_i$ for which the cumulative similarity value achieves maximum, replaces the central noisy pixel. In this way the filter replaces the central pixel only when it is really noisy and preserves the original undistorted image structures.

The parameter h can be set experimentally or can be determined adaptively using the technique described in [7].

F_1	F_2	F_3
F_8	F_0	F_4
F_7	F_6	F_5

a)

F_1	F_2	F_3
F_8		F_4
F_7	F_6	F_5

b)

Fig. 2. Illustration of the construction of the new filtering technique. First the cumulative similarity value M_0 between the central pixel F_0 and its neighbors is calculated (**a**), then the pixel F_0 is rejected from the filter window and the cumulative similarity values M_k, $k = 1, \ldots, n$ of the pixels $F_1, \ldots, F_n$ are determined, (**b**).

Notation	Filtering Technique
AMF	Arithmetic Mean Filter
VMF	Vector Median Filter
ANNF	Adaptive Nearest Neighbor Filter
BVDF	Basic Vector Directional Filter
HDF	Hybrid Directional Filter
AHDF	Adaptive Hybrid Directional Filter
DDF	Directional-Distance Filter
FVDF	Fuzzy Vector Directional Filter

Table 1. Filters taken for comparisons, [1-3].

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
FILTERING KERNELS			
$\mu_0(x)$	5.056	3.163	38.137
$\mu_1(x)$	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 2. Comparison of the new algorithm based on different kernel functions with the standard techniques, using the *LENA* color image contaminated by 5% of impulsive noise.

3. RESULTS AND CONCLUSION

The new algorithm presented in this paper can be seen as a modification and improvement of the Vector Median Filter. The comparison with standard color image processing filters, (Tab. 2, Fig. 1 and 3) shows that the new filter outperforms the standard procedures used in color image processing, when the impulse noise is to be eliminated. The new filter class based on the similarity functions and kernel density estimation is significantly faster than VMF and therefore it can be applied in various applications, in which the computational speed plays a crucial role.

4. 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, 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] J. Astola, P. Haavisto, Y. Neuovo, Vector median filters, IEEE Proceedings, 78, 678-689, 1990.

[5] K.N. Plataniotis, D. Androutsos, A.N. Venetsanopoulos, Colour Image Processing Using Fuzzy Vector Directional Filters, Proceedings of the IEEE Workshop on Nonlinear Signal/Image Processing, Greece, 535-538, 1995.

[6] K.N. Plataniotis, D. Androutsos, V. Sri, A.N. Venetsanopoulos, A nearest neighbour multichannel filter, Electronic Letters, 1910-1911, 1995.

[7] B. Smolka, A. Chydzinski, K. Wojciechowski, K. Plataniotis, A.N. Venetsanopoulos, On the reduction of impulsive noise in multichannel image processing, Optical Engineering, vol. 40, no. 6, pp. 902-908, 2001.

[8] B.W. Silverman, "Density Estimation for Statistics and Data Analysis", London, Chapman and Hall, 1986.

[9] D.W. Scott, "Multivariate Density Estimation", New York, John Wiley, 1992.

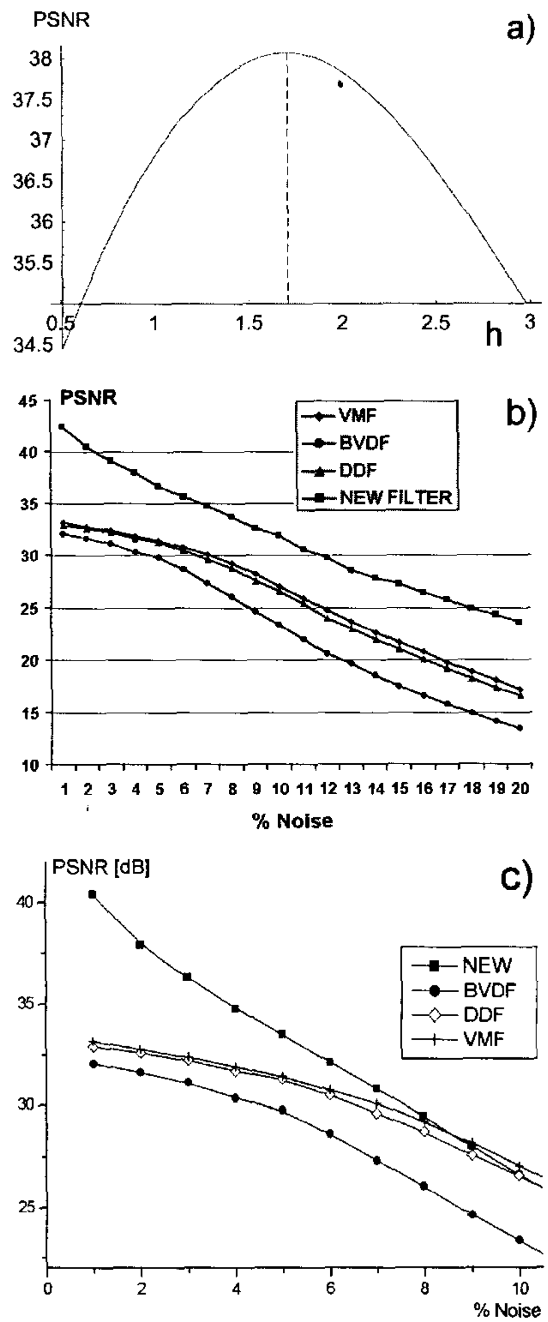

Fig. 3. **a**) Dependence of PSNR on the h parameter for *LENA* image with 12% of corrupted pixels, **b**) efficiency of the new algorithm in terms of PSNR in comparison with standard filters. *LENA* image was contaminated by impulsive noise with p from 1% to 20% and independently on each channel, p from 1% to 10%, **c**).

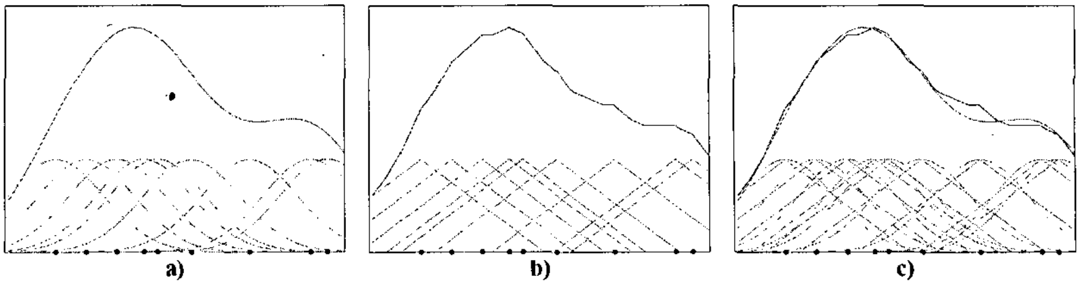

Fig. 4. Cumulative similarity values dependence on the pixel gray scale value for a window containing a set of pixels with intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$ using the μ_0 function (**a**) and μ_7 function (**b**). Plot (**c**) shows the comparison of the total similarity functions M_0 when using two different kernels.

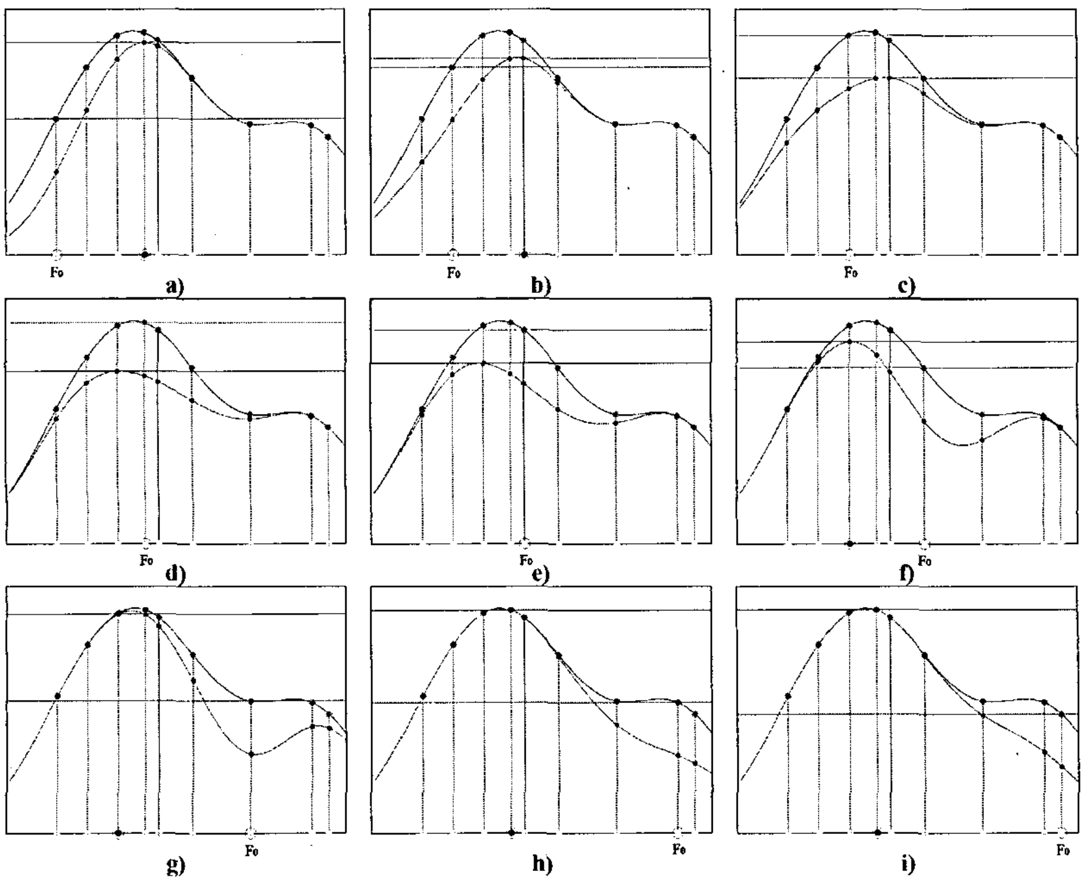

Fig. 5. Illustration of the new filter construction. The supporting window W of size 3×3 contains 9 pixels of intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$, $(Fig.4)$. Each of the graphs from **a**) to **i**) shows the dependence of M_0 and $M_{/0}$, ($M_{/0} < M_0$), where $M_{/0}$ denotes the cumulative similarity value with rejected central pixel F_0 on the pixel gray scale value. Graph **a**) shows the plot of M_0 and $M_{/0}$ for $F_0 = 15$, plot **b**) for $F_0 = 24$ and so on till plot plot **i**) which shows the graphs of M_0 and $M_{/0}$ for $F_0 = 95$. The arrangement of pixels surrounding the central pixel F_0 is not relevant. The central pixel will be replaced in cases: (**a**), (**b**), (**f**) - (**i**), as in those cases there exists a pixel F_i for which $M_0 < M_i$.