On the Similarity Based Impulsive Noise Removal Technique for Multichannel Images

Bogdan Smolka

Department of Automatic Control,
Silesian University of Technology,
Akademicka 16, 44-101, Gliwice, Poland

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 and preserves better edges and fine image details.

1 Introduction

A number of nonlinear, multichannel filters, which utilize correlation among multivariate vectors using various distance measures, have been proposed [1-11]. 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].

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 the construction of a simple, efficient and fast filter, which removes noisy pixels, but has the ability of preserving original image pixel values, is presented.

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, and let us define the

J. Bigun and T. Gustavsson (Eds.): SCIA 2003, LNCS 2749, pp. 1066–1073, 2003.

similarity function $\mu : [0;\infty) \to \mathbf{R}$ which is non-ascending and 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 minimal and maximal 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 required conditions.

Let us additionally define the cumulated sum M of similarities between a given pixel and all other pixels belonging to window W. For the central pixel let us introduce M_0 and for the neighbors of F_0 let us define M_k as

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

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 $\mu(F_k, F_0)$ when calculating M_k, 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 the calculation of the sum 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 $k^* = \arg\max M_k,\ k = 1, \ldots, n$.

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_k which maximizes the sum of similarities M between all pixels of W excluding the central pixel. This is illustrated in Fig. 1.

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 the 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 $F_0, F_1, \ldots, F_n$, [11].

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 can maximize the total similarity function M for the vector case.

Several convex functions were applied in order to compare the new filtering scheme with the standard filters used in color image processing presented in Tab. 1 and good results were obtained, (see Tab. 2), when applying the following similarity functions, which can be treated as kernels of nonparametric estimation, [12, 13].

$$\mu_0(x) = \exp\left\{-\left(\frac{x}{h}\right)^2\right\},\ \mu_1(x) = \exp\left\{-\frac{x}{h},\right\},\ \mu_2(x) = \frac{1}{1 + x/h},\ h \in (0;\infty),$$

$$\mu_3(x) = \frac{1}{(1+x)^h},\ \mu_4(x) = 1 - \frac{2}{\pi}\arctan\left(\frac{x}{h}\right),\ \mu_5(x) = \frac{2}{1+\exp\left\{\frac{x}{h}\right\}},\ h \in (0;\infty)\,,$$

$$\mu_6(x) = \frac{1}{1+x^h}, \quad \mu_7(x) = \begin{cases} 1 - x/h & \text{if } x \leq h, \\ 0 & \text{if } x > h, \end{cases} \quad h \in (0; \infty).$$

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

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

where $\rho\{\mathbf{F}_i, \mathbf{F}_k\} = ||\mathbf{F}_k - \mathbf{F}_l)||$ and $|| \cdot ||$ is the L_2 vector norm, as it yields the best results.

Applying the linear similarity function μ_7 we obtain

$$\mu(\mathbf{F}_i, \mathbf{F}_k) = \begin{cases} 1 - \frac{\rho(\mathbf{F}_i, \mathbf{F}_k)}{h} & \text{for} \quad \rho\left(\mathbf{F}_i, \mathbf{F}_k\right) < h \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Then we have from (2)

$$M_0 = n - \frac{1}{h} \sum_{j=1}^{n} \rho\left(\mathbf{F}_0, \mathbf{F}_j\right) \quad M_k = \sum_{\substack{j=1, \\ j \neq k}}^{n} \left(1 - \frac{\rho\left(\mathbf{F}_k, \mathbf{F}_j\right)}{h}\right) = n - 1 - \frac{1}{h} \sum_{j=1}^{n} \rho\left(\mathbf{F}_k, \mathbf{F}_j\right) \tag{4}$$

In this way the difference between M_0 and M_k is

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

$$= 1 - \frac{1}{h} \sum_{j=1}^{n} \left[\rho\left(\mathbf{F}_0, \mathbf{F}_j\right) - \rho\left(\mathbf{F}_k, \mathbf{F}_j\right) \right], \tag{5}$$

$$M_0 - M_k > 0 \;\; \text{if} \quad h > \sum_{j=1}^{n} \left[\rho\left(\mathbf{F}_0, \mathbf{F}_j\right) - \rho\left(\mathbf{F}_k, \mathbf{F}_j\right) \right]. \tag{6}$$

If this condition is satisfied, then the central pixel is considered as not disturbed by the noise process, otherwise the pixel $\mathbf{F}_k$ for which the cumulative similarity value achieves maximum, replaces the central noisy pixel. In this way the filter changes the central pixel only when it is really noisy and preserves the original undistorted image structures, (Figs. 1 and 2).

It is easy to observe that the construction of the new filter is quite similar to the standard VMF. Instead of the function R_k used in the VMF, a modified cumulative distance function R_0 is proposed

$$R_k = \begin{cases} -h + \sum\limits_{j=1}^{n} \rho(\mathbf{F}_k, \mathbf{F}_j)\,, & \text{for} \quad k = 0 \;\;, \\ & \\ \sum\limits_{j=1}^{n} \rho(\mathbf{F}_k, \mathbf{F}_j)\,, & \text{for} \quad k = 1, \ldots, n \end{cases}, \tag{7}$$

and in the same way as in VMF, the original vector $\mathbf{F}_0$ in the filter window W is being replaced by $\mathbf{F}_{k^*}$ such that $k^* = \arg\min_k R_k$.

It is easy to notice that the above construction is equivalent to the condition expressed in (6). Now, instead of maximizing the cumulative similarity M_k, we minimize the modified cumulative distance R_k. In this way, the condition for retaining the original image pixel is: $R_0 \leq R_k$, $k = 1, \ldots, n$, which leads to the condition

$$-h + \sum_{j=1}^{n} \rho(\mathbf{F}_0, \mathbf{F}_j) \leq \sum_{j=1}^{n} \rho(\mathbf{F}_k, \mathbf{F}_j)\,, \quad k = 1, \ldots, n\,, \tag{8}$$

$$R_0 \leq R_k \quad \text{if} \quad h \geq \sum_{j=1}^{n} \left[\rho(\mathbf{F}_0, \mathbf{F}_j) - \rho(\mathbf{F}_k, \mathbf{F}_j)\right]\,, \quad k = 1, \ldots, n\,. \tag{9}$$

The construction of the new filter is very similar to that of VMF, however the major difference is the omitting of the central pixel $\mathbf{F}_0$, when calculating R_k, $k > 0$. This scheme, based on the simple *leave-one-out* scheme, is the **most important feature** of the new algorithm. As the central pixel is suspected to be noisy, it is not taken into consideration, when calculating the distances associated with the neighbors of $\mathbf{F}_0$. In this way the filter replaces the central pixel only when it is really noisy, while preserving the original undisturbed image structures.

As widely known, VMF has the disadvantage of replacing too many uncorrupted image pixels. It is improved in the new filter design by setting positive h values, which forces the filter to preserve uncorrupted, original pixels, but still enables to remove corrupted ones. Moreover, h can be controlled for the best effectiveness of the filter depending on image structure and noise statistics.

It is easy to observe that the presented method is faster than VMF. It can be shown using a simple matrix representation, (for the sake of simplicity in the 4-neighborhood system case). In order to find R_{k^*} using the VMF method we have to add the elements in rows or columns of the symmetric matrix $\mathbf{T}_{VMF}$

$$\mathbf{T}_{VMF} = \begin{bmatrix} 0 & \rho_{01} & \rho_{02} & \rho_{03} & \rho_{04} \\ \rho_{10} & 0 & \rho_{12} & \rho_{13} & \rho_{14} \\ \rho_{20} & \rho_{21} & 0 & \rho_{23} & \rho_{24} \\ \rho_{30} & \rho_{31} & \rho_{32} & 0 & \rho_{34} \\ \rho_{40} & \rho_{41} & \rho_{42} & \rho_{43} & 0 \end{bmatrix}, \quad \mathbf{T}_{NEW} = \begin{bmatrix} -h & \rho_{01} & \rho_{02} & \rho_{03} & \rho_{04} \\ 0 & 0 & \rho_{12} & \rho_{13} & \rho_{14} \\ 0 & \rho_{21} & 0 & \rho_{23} & \rho_{24} \\ 0 & \rho_{31} & \rho_{32} & 0 & \rho_{34} \\ 0 & \rho_{41} & \rho_{42} & \rho_{43} & 0 \end{bmatrix}. \tag{10}$$

where $\rho_{ij} = \rho_{ji} = \rho(\mathbf{F}_i, \mathbf{F}_j)$. Obviously, the symmetry of the matrix $\mathbf{T}_{VMF}$ causes that effectively we have to compute 10 distances (36 in the 8-neighborhood case) and then to make 15 additions (63 in the 8-neighborhood case). For the implementation of the new filter (matrix $\mathbf{T}_{NEW}$), the number of distances we need, is still 10 but there is only 11 additions (55 in the 8-neighborhood case), so the new filter is faster than VMF.

3 Results

For evaluation purposes, the color test image *LENA* was corrupted with impulsive noise defined by $\mathbf{x}_{ij} = \mathbf{v}_{ij}$ with probability p, where i, j define a pixel position, p describes the intensity of the noise process, $\mathbf{x}_{ij}$ denotes the original image pixel and $\mathbf{v}_{ij}$ denotes a pixel corrupted by the noise process $\mathbf{v}_{ij} = \{\nu_R, \nu_G, \nu_B\}$, where ν_R, ν_G, ν_B are random integer variables from the interval $[0, 255]$ updated for each corrupted pixel.

The root of the mean squared error (RMSE), peak signal to noise ratio (PSNR), normalized mean square error (NMSE) [1,2] were used for the analysis. The comparison shows that the new filter outperforms by far the standard vector median filter, which can be treated as a reference filter, and other filters listed in Tab. 1. The efficiency of the new filtering technique is shown in Tab. 2 and in Fig. 2.

The new algorithm presented in this paper can be seen as a fast modification and improvement of the Vector Median Filter. The comparison with standard color image processing filters shows that the new filter outperforms the standard procedures used in color image processing. Another advantage of the proposed filtering class is its lower computational complexity compared to the VMF, which makes the new filter class interesting for real-time applications

4 Conclusions

The new algorithm presented in this paper can be seen as a fast modification and improvement of the Vector Median Filter. The comparison with standard color image processing filters, (Tab. 2, Fig. 2) shows that the new filter outperforms the standard procedures used in color image processing, when the impulse noise is to be eliminated.

Notation	**Filter**
AMF	Arithmetic Mean Filter
VMF	Vector Median Filter
BVDF	Basic Vector Directional Filter
HDF	Hybrid Directional Filter
AHDF	Adapt. Hybrid Directional Filter
DDF	Directional-Distance Filter

Table 1. Filters taken for comparison with the proposed adaptive filter, [1,2].

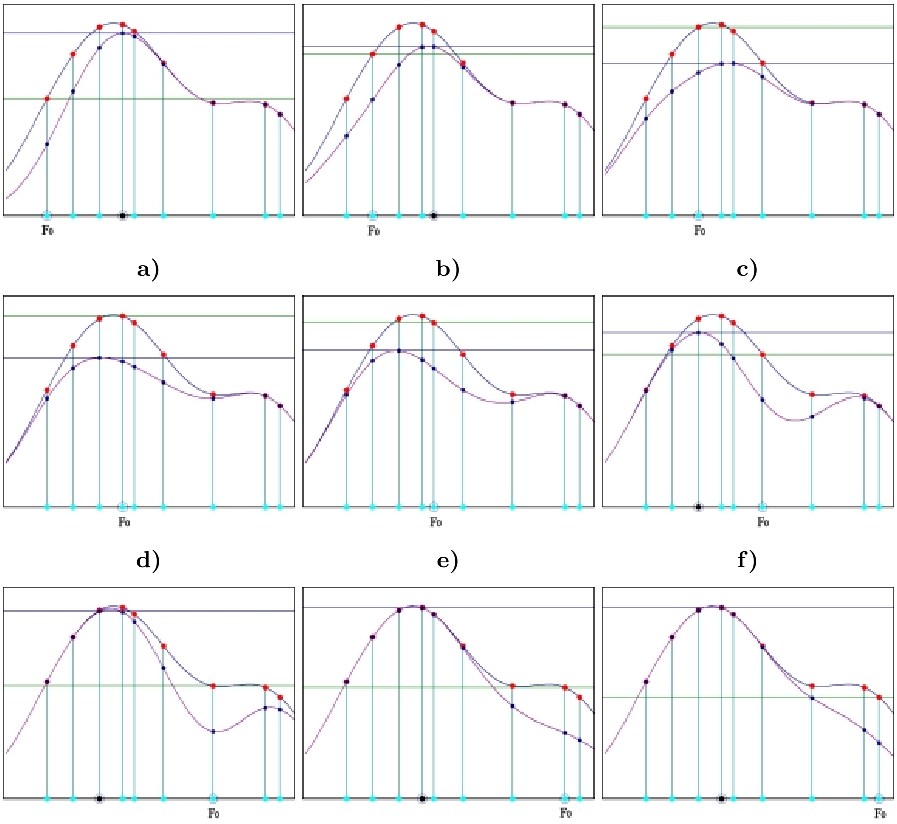
References

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, Combining vector median and vector directional filters: The directional - distance filters, Proceedings of the IEEE Conf. on Image Processing, ICIP-95, 171-174, (October 1995)
5. I. Pitas, A.N. Venetsanopoulos, Order statistics in digital image processing, Proceedings of IEEE, 80, 12, 1893-1923, (1992)
6. J. Astola, P. Haavisto, Y. Neuovo, Vector median filters, IEEE Proceedings, 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, Proceedings of EUSIPCO-96, 879-881, (1996)
9. 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)
10. K.N. Plataniotis, D. Androutsos, V. Sri, A.N. Venetsanopoulos, A nearest neighbour multichannel filter, Electronic Letters, 1910-1911, (1995)
11. B. Smolka, A. Chydzinski, K. Wojciechowski, K. Plataniotis, A.N. Venetsanopoulos, On the reduction of impulsive noise in multichannel image processing, Optical Engineering, 40, 6, 902-908, (2001)
12. B.W. Silverman, "Density Estimation for Statistics and Data Analysis", London, Chapman and Hall, (1986)
13. D.W. Scott, "Multivariate Density Estimation", New York, John Wiley, (1992)

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	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
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
$\boldsymbol{\mu_7(\mathbf{x})}$	**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 10% of impulsive noise.



a) **b)** **c)**

d) **e)** **f)**

g) **h)** **i)**

Fig. 1. Illustration of the new filter construction. The supporting window W of size 3×3 contains 9 pixels of intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$. 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 $M_0 < M_i$.

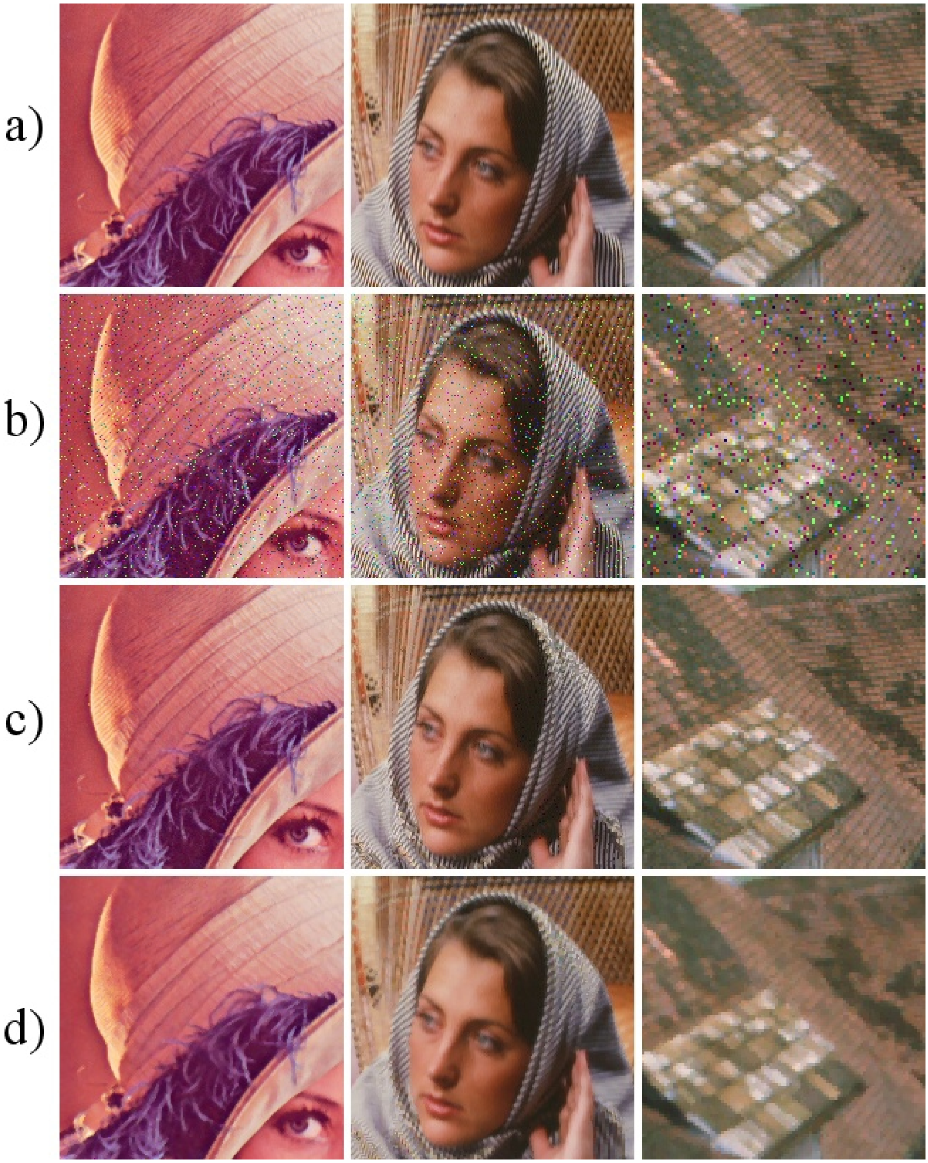


Fig. 2. Comparison of the efficiency of proposed filter with the VMF: **a)** parts of the *LENA*, *BARBARA* and *GOLDHIL* images, **b)** images contaminated by 6% impulsive noise, **c)** images restored using the proposed technique, **d)** the result of the filtering with the vector median.