colour image enhancement,
impulsive noise removal, image restoration

Bogdan SMOLKA*, Sebastian BUDZAN*,
Rastislav LUKAČ**

NONPARAMETRIC DESIGN OF IMPULSIVE NOISE REMOVAL IN COLOUR IMAGES

In this paper the problem of nonparametric impulsive noise removal in multichannel images is addressed. The proposed filter class is based on the nonparametric estimation of the density probability function in a sliding filter window. The obtained results show good noise removal capabilities and excellent structure preserving properties of the new impulsive noise removal technique.

1. INTRODUCTION

The majority of the nonlinear, multichannel filters are based on the ordering of vectors in a sliding filter window. The output of these filters is defined as the lowest ranked vector according to a specific vector ordering technique.

Let the colour images be represented in the commonly used RGB colour space and let $x_1, x_2, ..., x_N$ be N samples from the sliding filter window W. Each of the x_i is an m-dimensional multichannel vector, (in our case m = 3). The goal of the vector ordering is to arrange the set of N vectors $\{x_1, x_2,..., x_N\}$ belonging to W using some sorting criterion.

In [1,2] the ordering based on the cumulative distance function $R(x_i)$ has been proposed: $R(x_i) = \sum_{i=1}^{N} \rho(x_i, x_j)$, where $\rho(x_i, x_j)$ is a function of the distance between x_i and x_j. The ordering of the scalar quantities according to $R(x_i)$ generates the ordered set of vectors. The most commonly used measure to quantify distance between two multichannel signals is the Minkowski norm $\rho_\gamma = (x_i, x_j) = \left[\sum_{k=1}^{m} \left|x_{ik} - x_{jk}\right|^{\gamma}\right]^{1/\gamma}$. The Minkowski metric includes the city-block distance (γ = 1), Euclidean distance (γ = 2) and chess-board distance ($\gamma = \infty$) as the special cases.

One of the most important noise reduction filter is the vector median. In the case of grey scale images, given a set W containing N samples, the median of the set is defined as $x_{(1)} \in W$ such that

* Silesian University of Technology, Dept. of Automatic Control, Akademicka 16, 44-100 Gliwice, Poland
** The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto ON, M5S 3G4, Canada

$$\sum_j \left| x_{(1)} - x_j \right| < \sum_j \left| x_i - x_j \right|, \qquad \forall x_i, x_j \in W \tag{1}$$

Median filters exhibit good noise reduction capabilities, (especially when long tailed noise is involved) and outperform simple nonadaptive linear filters in preserving signal discontinuities. As in many applications the signal is multidimensional, in [4] the *Vector Median Filter* (VMF) was introduced, by generalizing the definition (1) using a suitable vector norm. Given a set *W of N* vectors, the vector median of the set is defined as $x_{(1)}$ *W* satisfying

$$\sum_j \left\| x_{(1)} - x_j \right\| < \sum_j \left\| x_i - x_j \right\|, \qquad \forall x_i, x_j \in W \tag{2}$$

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, [3]. The *Basic Vector Directional Filter* (BVDF) is a ranked-order, nonlinear filter which parallelizes the VMF operation. However, a distance criterion, different from the distance norms used in VMF is used 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. To improve the efficiency of the directional filters, another method called *Directional-Distance Filter* (DDF) was proposed. This filter retains the structure of the BVDF, but uses the combined distance criterions to order the vectors inside the processing window, [3,5].

2. NONPARAMETRIC ESTIMATION

Application of statistical pattern recognition techniques requires estimation of the probability density function of the data samples. Nonparametric techniques do not assume a particular form of density function since the underlying density of real data rarely fits common density models.

Nonparametric Density Estimation is based on placing a kernel function on every sample and on the summation of the values of all kernel function values at each point in the sample space, [6,7]. The nonparametric approach to estimating multichannel densities can be introduced by assuming that the colour space occupied by the multichannel image pixels is divided into m-dimensional hypercubes. If h_N is the length of an edge of a hypercube, then its volume is given by $V_N = h_N^m$. If we are interested in estimating the number of pixels falling in the hypercube of volume V_N, then we can define the window function $\phi(x_i) = 1, if \left| x_{ij} \right| \le 1/2, j = 1,\ldots,m$ and 0 otherwise, which defines a unit hypercube centered in the origin.

The function $\phi(\left\| x - x_i \right\| / h_N)$ is equal to unity if the pixel xi falls within the hypercube V_N centered at x and is zero otherwise. The number of pixels in the hypercube with the length of edges equal to h_N is then $k_N = \sum_{i=1}^{N} \phi(\|x - x_i\| / h_N)$ and the estimate of the probability that a sample x is within the hypercube is $p_N = k_N/NV_N$, which gives

$$p_N(x) = (NV_N)^{-1} \sum_{i=1}^{N} \phi(\|x - x_i\| / h_N) \tag{3}$$

This estimate can be generalized by using a smooth kernel function K in place of $\phi(\cdot)$ and the width parameter h_N satisfying: $K(x) = K(-x), K(x) \geq 0, \int K(x)dx = 1$ and $\lim_{N\to\infty} h_N = 0$ $\lim_{N\to\infty} h_N^m = \infty$

The multivariate estimator in the m-dimensional case is defined as

$$p_N^*(x) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{h_1 \ldots h_q} \mathrm{K}\left(\frac{|x_1 - x_{i1}|}{h_1}, \ldots, \frac{|x_q - x_{iq}|}{h_m}\right) \tag{4}$$

with K denoting a multidimensional kernel function $K : \Re^m \to \Re, h_1, \ldots, h_m$ denoting bandwidths for each dimension and N being the number of samples in W. A common approach to build multidimensional kernel functions is to use a product kernel $K(u_1, \ldots, u_m) = \prod_{i=1}^{m} K(u_i)$, where K is a one-dimensional kernel function

$$p_N^*(x) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{m}\left(\frac{|x_{ij} - x_j|}{h_i}\right) \tag{5}$$

The shape of the approximated density function depends heavily on the bandwidth chosen for the density estimation. Small values of h lead to spiky density estimates showing spurious features. On the other hand, too large values of h produce over-smoothed estimates that hide structural features.

If we chose the Gaussian kernel, then the density estimate of the unknown probability density function at x is obtained as a sum of kernel functions placed at each sample x_i

$$p_N(x,h) = \frac{1}{N(h\sqrt{2\pi})^m}\sum_{i=1}^{N} \exp\left(-\frac{\|x - x_i\|^2}{2h^2}\right) \tag{6}$$

The smoothing parameter h depends on the local density estimate of the sample data. The form of the data dependent smoothing parameter is of great importance for the non-parametric estimator. Choosing the Gaussian kernel function for K, the optimal bandwidth is

$$h^* = (4/(m+2))^{-\frac{1}{m+4}}\, \hat{\sigma} \quad N^{-\frac{1}{m+4}} \tag{7}$$

where σ denotes the approximation of the standard deviation of the samples. In one dimensional case (7) reduces to the well known, 'rule of thumb', $h^* = 1.06 N^{-\frac{1}{5}} \hat{\sigma}$, $[6,7]$. A version which is more robust against outliers in the sample set can be constructed if the interquartile range is used as a measure of spread instead of the variance, [6]. This modified estimator is $h^* = 0.79 \rho N^{-\frac{1}{5}} \hat{\sigma}$, where ρ is the inter-quartile range. Another robust estimate of the optimal bandwidth is $h^* = 0.9 A N^{-\frac{1}{5}} \hat{\sigma}$ with $A = \min(\hat{\sigma}, \rho / 1.34)$. Generally the simplified rule of choosing the optimal bandwidth *h* can be written as

$$h_1^* = C\ \hat{\sigma}\ N^{-\frac{1}{m+4}} \tag{8}$$

where *C* is an appropriate weighting coefficient. From the maximum likelihood principle and assuming independence of the samples, one can write the likelihood of drawing the complete dataset as the product of the densities of one sample

$$L(h) = \prod_{j=1}^{N} p_N(x_j, h) = \prod_{j=1}^{N} \frac{1}{N} \sum_{i=1}^{N} \frac{1}{(h\sqrt{2\pi})^m} \exp\left(- \frac{\left\| x_j - x_i \right\|^2}{2h^2} \right) \tag{9}$$

As this likelihood function has a global maximum for h=0, in [8] a modified approach has been proposed

$$L^*(h) = \left[\prod_{j=1}^{N} \frac{1}{N} \sum_{i=1, i \neq j}^{N} \frac{1}{(h\sqrt{2\pi})^m} \exp\left(- \frac{\left\| x_j - x_i \right\|^2}{2h^2} \right) \right]^{\frac{1}{m}} \tag{10}$$

This function has one maximum for *h,* which can be found by setting to 0 the derivative of the logarithm of $L^*(h)$ with respect to h, which gives

$$\frac{1}{N} \sum_{j=1}^{N} \frac{\sum_{i \neq j}^{N} \frac{\left\| x_j - x_i \right\|^2}{h^3} \exp\left(- \frac{\left\| x_j - x_i \right\|^2}{2h^2} \right)}{\sum_{i \neq j}^{N} \exp\left(- \frac{\left\| x_j - x_i \right\|^2}{2h^2} \right)} = \frac{m}{h} \tag{11}$$

A crude but rather fast way to obtain an approximate solution of (11) is by assuming that the density estimate of Eq. (5) on a certain location **x** in the feature space is determined by the nearest kernel only, [8]. In this case

$$\frac{\partial \log(L^*(h))}{\partial h} = \frac{1}{N}\sum_{j=1}^{n} \frac{\left\|\tilde{x}_j - x_i\right\|^2}{h^3} = \frac{m}{h} \tag{12}$$

In this paper we use the optimal *h* derived from (12) defined as

$$h_2^* = C\left((mN)^{-1}\sum_{j=1}^{N}\left\|\tilde{x}_j - x_j\right\|^2\right)^{\frac{1}{2}} \tag{13}$$

where x˜i represents the nearest neighbour of the sample x_i, and *C* is a tuning parameter.

3. PROPOSED ALGORITHM

Let us assume a filtering window *W* containing *N* image pixels, $\{x_1,..., x_N\}$ and let us define the similarity function $\mu : [0;\infty) \to R$ which is non-ascending and convex in [0; ∞) 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 grey scale values should be very close to 0. The function $\mu(x_i, x_j)$ defined as $\mu(x_i, x_j) = \exp\{-[(x_i - x_j)/h]^2\}$, where *h* is the bandwidth of the Gaussian kernel, defined by (8) or (13), satisfies the 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 x_1 we introduce M_1 and for the neighbours of x_1 we define M_k as

$$M_1 = \sum_{j=2}^{N} \mu(x_1, x_j), \qquad M_k = \sum_{j=2, j\neq k}^{N} \mu(x_k, x_j), k > 1, \tag{14}$$

which means that for x_k, which are neighbours of x_1, we do not take into account the similarity between x_k and x_1, which is the main idea of this algorithm. The omission of the similarity $\mu(x_k, x_1)$ when calculating M_k, privileges the central pixel, as in the calculation of M_1 we have $N - 1$ similarities $\mu(x_1,x_k)$, $k > 2$ and for M_k, $k > 1$ we have only $N - 2$ similarity values, as the central pixel x_1 is excluded from the calculation of M_k, [9,10], (see Figs. 1, 2).

In the construction of the new filter, the reference pixel x_1 in the window *W* is replaced by one of its neighbours if $M_1 < M_k$, $k = 2,..., N$. If this is the case, then x_1 is replaced by that x_{k^*} for which $k^* = \arg\max M_k$, $k = 2,...,N$. In other words x_1 is detected as being corrupted if $M_1 < M_k$, $k = 2,..., N$ and is replaced by its neighbours x_k which maximizes the sum of similarities *M* between all the pixels from *W* excluding the central pixel.

The basic assumption is that a new pixel must be taken from the window *W*, (introducing pixels, that 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 $x_1, x_2,..., x_N$.

The working scheme of the new filter is presented in Fig. 2 for the grey scale case and in Fig. 1 for the two-dimensional data. In the example provided by Fig. 2, the supporting window W contains 9 pixels of intensities {15, 24, 33,41,45, 55, 72, 90, 95}, (their special arrangement in W is not relevant). Each of the graphs from **a)** to **i)** shows the dependence of M_1 and $Mi\,/_1$ on the grey scale value, ($M_{/1}$ < M_1), where $Mi\,/_1$ denotes the cumulative similarity value with rejected central pixel x_1, on the sample's intensity. Graph **a)** shows the plot of M_1 and $M_{/1}$ for x_1 = 15, plot **b)** for x_1 = 24 and so on till plot plot **i)**, which shows the graphs of M_1 and $M_{/1}$ for x_1 = 95. The central pixel will be replaced in cases: **(a)**, **(b)**, **(f) - (i)**, as in those cases there exists a pixel x_k for which $M_1 < M_k$. The continuous plots show that the extremum of the similarity function $M_{/1}$ is always obtained at points $x_k \in W$, which is an important feature of this algorithm. Because the function $M_{/1}$ is convex, the maximum can be found by calculating the similarity values in N points only, which makes the algorithm computationally attractive.

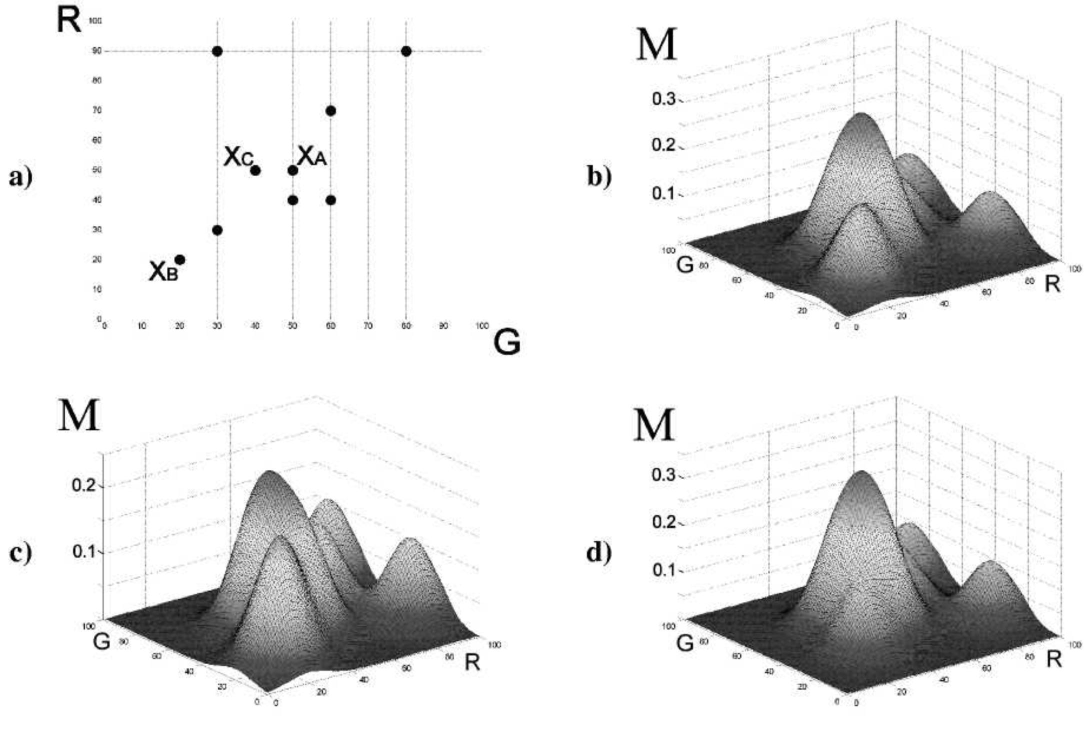

Fig. 1. Impulsive noise removal technique in the 2D case. Fig. **a)** depicts the arrangement of pixels in W and Fig. **b)** their nonparametric probability density estimation. Figs. **c)** and **d)** present the density plots for the cases when the central pixels X_A and X_B are removed from W. It can be seen that in the first case **c)** the pixel $x_1 = X_A$ will be retained and in the second case **d)** the pixel $x_1 = X_B$ will be replaced by X_A. The pixel XA will be preserved, as in Fig. **c)** the plot attains its maximum at x_C, but this maximum is less than the maximum for X_A in Fig. **b)**. Regarding sample X_B, its rejection causes that the maximum is attained at X_A and this pixel will replace the central pixel X_B.

The presented approach can be applied in a straightforward way to multichannel images using the similarity function defined as $\mu(x_i, x_j) = \exp\{-[\| x_i - x_j \| / h]^2\}$, where $\| \cdot \|$ denotes the specific vector norm and h denotes the bandwidth. Now in exactly the same way we can maximize the total similarity function M for the vector case.

4. RESULTS

The performance of the proposed impulsive noise reduction filters was evaluated using the widely used PSNR quality measure. Figure 3a) shows the dependence of the noise attenuation capability of the proposed filter class on the bandwidth type h_1^* and h_2^* defined by (8) and (13). Clearly the filter based on the h_2^* outperforms the technique based on the h_1 bandwidth for the whole range of used contamination probabilities *p, (p* = 0.01 - 0.1).

Figure 3b) presents the dependence of the PSNR restoration quality measure on the kind of the Minkowski norm. Surprisingly, the L_∞ norm yields significantly better results than the L_1 or L_2 norms. This is the result of the construction of the h_2 bandwidth, which depends on the nearest neighbour in the sliding filter window. This behaviour is advantageous, as the calculation of the L_∞ norm is much faster than the evaluation of distances determined by L_1, L_2 norms.

The efficiency of the filters based on adaptive h_1^* and h_2^* bandwidths are dependent, (especially for very small noise contamination) on the coefficient C in (8) and (13). Figure 3c) shows the dependence of PSNR for the filter based on h_2^* as a function of C in (13). For low noise intensity the parameter C should be significantly larger than for the case of images corrupted by heavy noise process. However, setting C to 4 is an acceptable trade-off, as can be seen in Fig. 3 d), which depicts the efficiency of the proposed filter in comparison with VMF, AMF and BVDF. It can be observed that although the $C = 4$ is not an optimal setting for the whole range of tested noise intensities, nevertheless the described filter yields much better results than the traditional techniques.

This is also testified by Fig. 4, which compares the filtering results obtained by the filter based on adaptive h_2 bandwidth with the performance of the *reference* VMF, BVDF, DDF filter. As can be observed the new filtering has much better detail preserving properties than VMF, BVDF and DDF.

5. CONCLUSIONS

In this paper a new nonparametric technique of impulsive noise removal in multichannel images has been proposed. The described filter class is based on the estimation of the kernel bandwidth using the technique proposed in [8]. The experiments revealed, that the proposed algorithm yields the best results when applying the L_∞ norm, which makes the filter computationally very attractive. The obtained results show that the proposed technique excels significantly over the standard techniques like VMF, BVDF and DDF.

BIBLIOGRAPHY

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

[2] K. TANG, J. ASTOLA, Y. NEUOVO, Nonlinear multivariate image filtering techniques, IEEE Trans. on Image Processing, 4, 6, 788-797, 1995.

[3] 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.

[4] J. ASTOLA, P. HAAVISTO, Y NEUVO, Vector median filters, Proceedings of the IEEE, 78, 678-689, 1990.

[5] K.N. PLATANIOTIS, A.N. VENETSANOPOULOS, "Color Image Processing and Applications", Springer Verlag, August 2000.

[6] B.W. SILVERMAN, "Density Estimation for Statistics and Data Analysis", London, Chapman and Hall, 1986 [7] D.W. SCOTT, "Multivariate Density Estimation", New York, John Wiley, 1992

[7] M.A. KRAAIJVELD, A Parzen classifier with an improved robustness against deviations between training and test data, Pattern Recognition Letters, 17, 679-689, 1996.

[8] B. SMOLKA, K.N. PLATANIOTIS, A. CHYDZINSKI, M. SZCZEPANSKI, A.N. VENETSANOPULOS, K. WOJCIECHOWSKI, Self-adaptive algorithm of impulsive noise reduction in color images, Pattern Recognition, 35, 1771-1784, 2002.

[9] B. SMOLKA, R. LUKAC, A. CHYDZINSKI, K.N. PLATANIOTIS, K. WOJCIECHOWSKI, Fast adaptive similarity based impulsive noise reduction filter, Real Time Imaging, 9, 261-276, 2003.

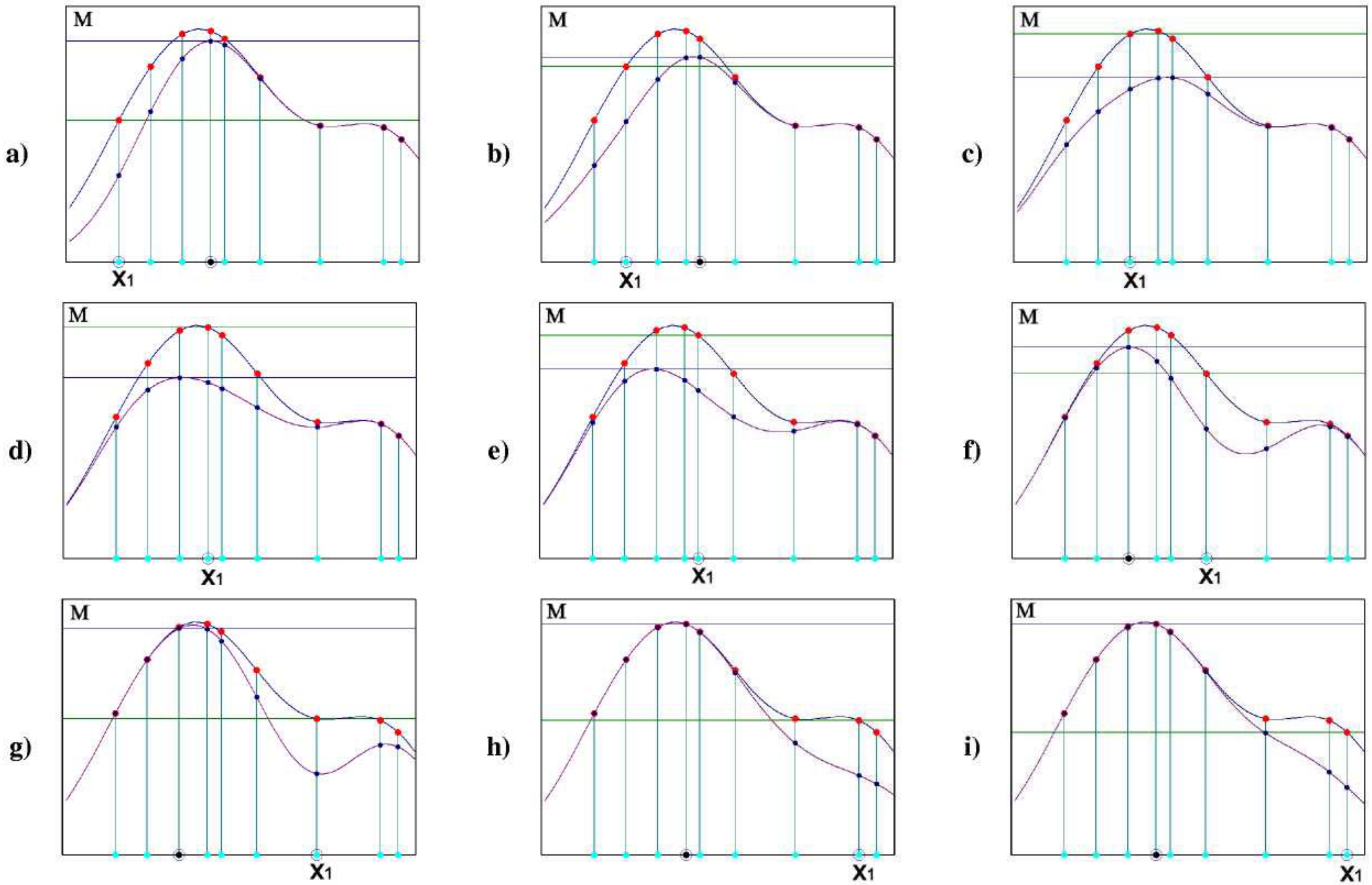

Figure 2: Illustration of the new filter construction using the Gaussian kernel. The supporting window W of size 3×3 contains 9 pixels of intensities $\{15, 24, 33, 41, 45, 55, 72, 90, 95\}$. Each of the graphs from **a**) to **i**) shows the dependence of M_1 and $M_{/1}$, ($M_{/1} < M_1$), where $M_{/1}$ denotes the cumulative similarity value with rejected central pixel x_1 on the gray scale value. Graph **a**) shows the plot of M_1 and $M_{/1}$ for $x_1 = 15$, plot **b**) for $x_1 = 24$ and so on till plot plot **i**) shows the graphs of M_1 and $M_{/1}$ for $x_1 = 95$. The arrangement of pixels surrounding the central pixel x_1 is not relevant. The central pixel will be replaced in cases: (**a**), (**b**), (**f** - **i**), as in those cases there exists a pixel x_k for which $M_1 < M_k$ or $R_1 > R_k$ is satisfied

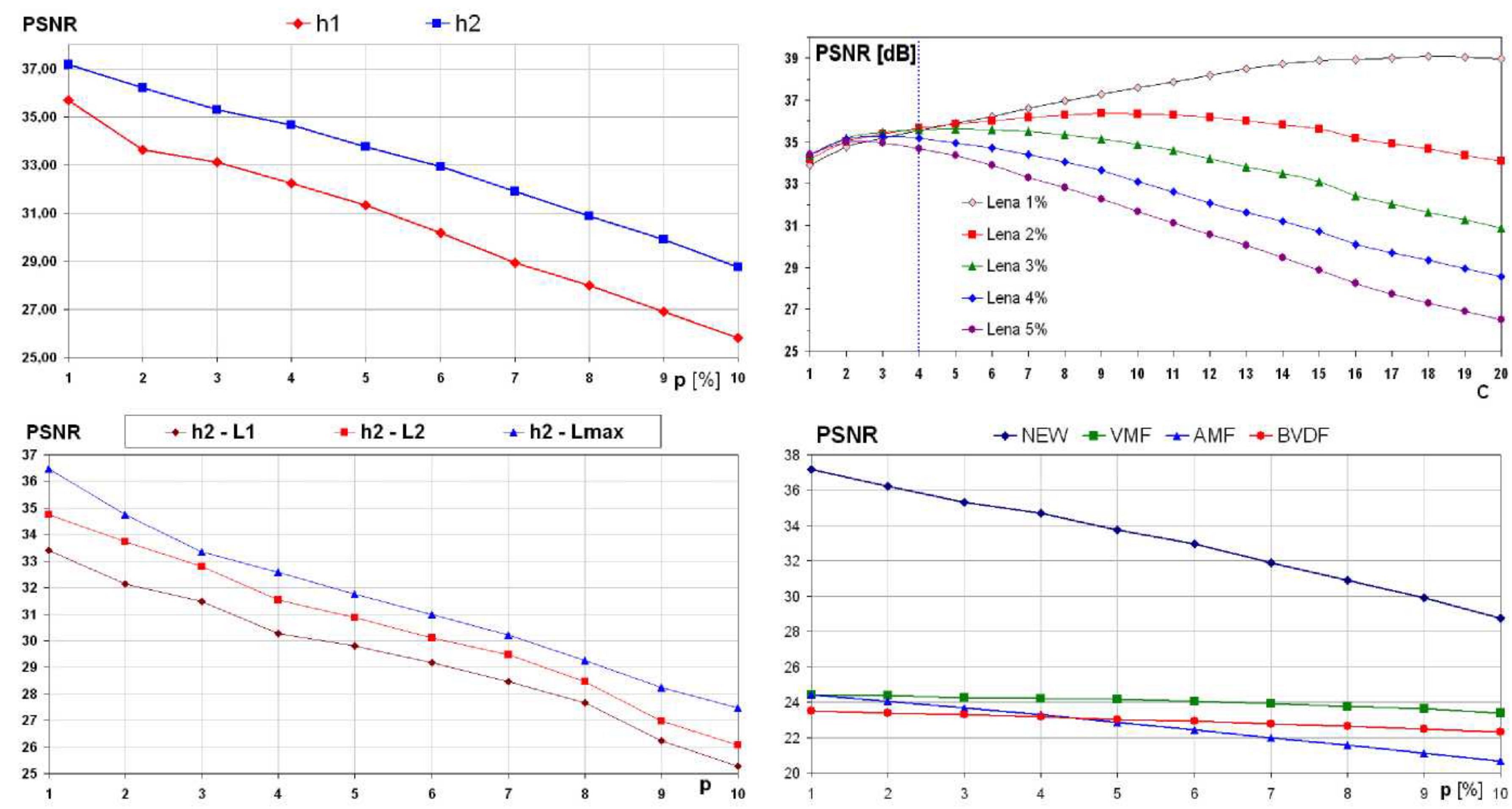

Figure 3: Dependence of the efficiency of the proposed filtering scheme on the bandwidth h_1^* (8) and h_2^* (13) - (**a**), below the dependence of the PSNR value on the value of the tuning parameter C in (13) - (**b**) and the dependence on the kind of Minkowski norm for the bandwidth h_2* - (**c**). At the bottom (**d**) the comparison of results obtained using the h_2^* bandwidth, L_∞ norm and $C = 4$ with the standard multichannel filters VMF and BVDF, (test were performed on the color image *LENA*); p denotes the probability of a pixel corruption - to RGB channels random, uniformly distributed values from the interval [0,255] were assigned.

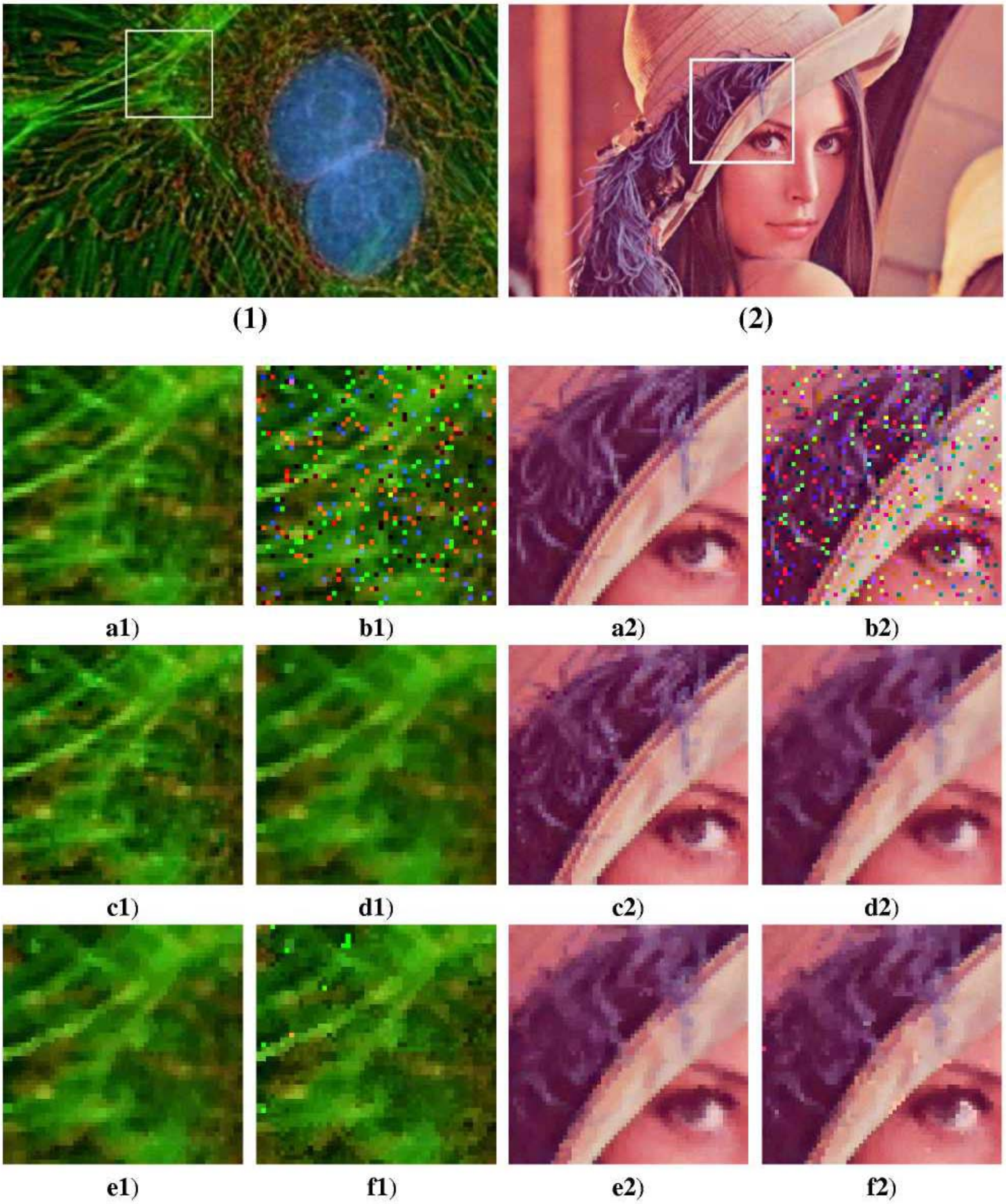

Figure 4: Illustrative example of the efficiency of the proposed algorithm: **a**) zoomed parts of a test images, **b**) image corrupted by 3% of impulsive noise, **c**) image after filtering with the proposed filter, **d**) VMF output, **e**) DDF output, **f**) BVDF output.