# An Algorithm for Impulsive Noise Removal in Color images

Jianjun Zhang[1] and Xuehua Tang.

**Abstract.** We propose an effective algorithm for impulsive noise removal in color images. The proposed method is a two-phase scheme. In the first phase, an impulse detector is designed to identify color pixels which are likely to be contaminated by noise; in the second phase, the image is restored by using a detail-preserving regularization method that applies only to those selected noise candidates. Extensive computer simulations indicate that our proposed algorithm provides significant improvement over many of the existing filtering algorithms in suppressing impulsive noise in color images.

**Keywords:** Color images, impulsive noise, vector median filter, denoising.

## 1 Introduction

Obtained images are often corrupted by impulsive noise during the acquisition or transmission, due to the image system error and transmission effects. So it is very important to eliminate noise in images before subsequent processing such as image segmentation.

When an image is corrupted by impulsive noise, only part of the pixels is changed. Based on trichromatic color theory, color pixels are encoded in three scalar values, namely, red, green and blue (RGB color space). Let $x_{ij}$ be the vector that characterizes a pixel of a noisy color image, $z_{ij}$ the vector describing impulsive noise model. Let $o_{ij}$ be the noise-free color pixel and $p$ be the impulsive noise probability. Then impulsive noise model is described as,

[1] J. Zhang (✉)
Shanghai University, Shanghai 200444, China
e-mail: jjzhang@staff.shu.edu.cn

$$x_{ij} = \begin{cases} z_{ij} & \text{with probablity } p, \\ o_{ij} & \text{with probablity 1-}p. \end{cases} \tag{1}$$

In this paper, we consider random-valued noise model. For this model, each component of $z_{ij}$ has random value in [0, 255].

Numerous vector filtering techniques have been proposed to remove impulsive noise for color image processing since the development of the well-known vector median filter (VMF) [2]. For example, vector directional filters (VDF) [3], directional-distance filters (DDF) [4], hybrid directional filter (HDF) [5].

Similar to the scalar median filter, these vector filters have undesirable side effects that tend to blur fine details or destroy edges while filtering out impulses because they introduce too much smoothing [6]. To trade off detail preservation against noise reduction, many improved vector filtering techniques have been developed. See for example [7] [8] and references therein. In recent years, the so-called "decision-based" or "switching" mechanism has been introduced into these filters. These filters first identify possible noisy pixels and then replace them by using the vector median filter or its variants, while leaving all other pixels unchanged. In [9], fast peer group filter was proposed. In [6], based on the quaternion representation of color difference, an efficient color-impulse detector for switching vector median filters was presented. In [10], based on vector marginal median filter, two switching-based filters were proposed. Many other switching-based filter were also proposed, see for example [8][11][12][13].

In this paper, we propose an effective algorithm for impulsive noise removal in color images. The proposed method is a two-phase scheme. In the first phase, an impulse detector is designed to identify pixels which are likely to be contaminated by noise; in the second phase, the image is restored by using a detail-preserving regularization method that applies only to those selected noise candidates. Extensive computer simulations indicate that our proposed algorithm provides significant improvement over many of the existing filtering algorithms in suppressing impulsive noise in color images.

The paper is organized as follows. In Section 2, we give our impulsive noise removal algorithm. Experimental results are presented in Section 3.

## 2 Our method

Let $S_{ij}^r$ be a window of size $(2r+1)\times(2r+1)$ centered at $(i, j)$, i.e.

$$S_{ij}^r = \{(k,l) : |k-i| \le r, |l-j| \le r\}$$

The observed image pixels in the window $S_{ij}^r$ are $x_{kl}$, for $(k,l)\in S_{ij}^r$. By reordering $x_{kl}$ for $(k,l)\in S_{ij}^r$ in lexicographical order for the convenience of exposition, we get $R$ samples: $X_1, X_2, \cdots, X_R$, with $x_{ij} = X_{(R+1)/2}$, where $R=(2r+1)^2$.

For $n=1,2,\cdots,R, n\neq (R+1)/2$, define $d_n$ as the distance between $X_n$ and $X_{(R+1)/2}$, i.e.,

$$d_n = d(X_n, X_{(R+1)/2}) \tag{1}$$

Then sort the $d_n$ values in increasing order:

$$d_{(1)} \le d_{(2)} \le \cdots \le d_{(R-1)}, \tag{2}$$

and define

$$\mathrm{ROPD}_m(i,j)=\sum_{n=1}^{m} d_{(n)}, \tag{3}$$

where $2\le m\le R-2$. The statistic defined in (3) is called Rank-Ordered Pixels Distance (ROPD) of the pixel $x_{ij}$. In this paper, we will consider $m=2, r=1$ and $d_n = \left\| X_n, X_{(R+1)/2} \right\|_1$, and set $\mathrm{ROPD}(i,j)=\mathrm{ROPD}_2(i,j)$.

Similar to ROAD statistic defined in [14], the ROPD statistic provides a measure of how close a color pixel is to its eight most similar neighbours, and the ROPD statistic can be used to detect impulsive noise as in the following algorithm.

**Algorithm 1** (Noise detection) For color image pixel $x_{ij}=(x_{ij}^R, x_{ij}^G, x_{ij}^B)$, if its ROPD statistic is greater than a predetermined threshold Tol, then the color pixel $x_{ij}$ is contaminated by noise. Otherwise, this color pixel is noise-free.

To implement the algorithm, a very important step is to select the threshold Tol. If Tol is too small, then many noise-free pixels may be classified as noise pixels. If Tol is too large, then many noise pixels may be classified as noise-free pixels. So the selected Tol should balance the above two situations. On the other hand, Tol should magnify the impulsive noise probability. Considering these conditions, we select Tol according to the following rule: Set iTol = 140, let q be the proportion of the pixels in the image whose ROPD values are less than iTol. Then we choose Tol as

$$\mathrm{Tol}=(\mathrm{iTol}-q\times 100)\times q. \tag{4}$$

Extensive simulations conducted on standard test images indicate that with Tol selected according to (4), our algorithm is very effective.

In the following, the set of noise candidates is denoted by

$$\mathrm{N}=\{(i,j): x_{ij} \text{ is contaminated by niose}\}.$$

The set of all uncorrupted pixels is denoted by $\mathrm{N}^c$ .

After the noise detection, the noise pixels can be restored by some nonlinear regularization method. Here we use the ideas in [15][16], to design our algorithm.

**Algorithm 2** (Two-phase impulsive noise removal algorithm for color images):

1. Initialize $x$ to be the observed image.

2. Apply algorithm 1 to the image $x$ to get the noise candidate set $\mathrm{N}$ .

3. For all $(i, j) \notin \mathrm{N}$ , take $\hat{y}_{ij} = x_{ij}$ . For all pixels in $\mathrm{N}$ , take $\hat{y}_{ij}$ be the minimizing point of the following functional over $\mathrm{N}$ :

$$f(y) = \sum_{(i,j)\in\mathrm{N}} \left\{ \sum_{(m,n)\in V_{ij}\cap\mathrm{N}} \phi\left(\left\|y_{ij} - y_{mn}\right\|_p\right) + 2 \sum_{(m,n)\in V_{ij}\setminus\mathrm{N}} \phi\left(\left\|y_{ij} - x_{mn}\right\|_p\right) \right\} \quad (5)$$

where $\phi(t) = |t|^{\alpha}, 1 < \alpha < 2$ $V_{ij}$ is the four nearest neighbors of $(i, j)$ , $V_{ij} \setminus \mathrm{N}$ is composed of those neighbors of $(i, j)$ which have been detected as signal candidates.

4. Output $\hat{y}$ .

We use the GBB method to find the minimizer of the objective function (5). More details can be found in M. Raydan [17].

In our experiments, we use 1-norm in (5), that is $p = 1$. And we set $\gamma = 10^{-3}$ , $\varepsilon = 10^{-5}, \sigma_1 = 0.1, \sigma_2 = 0.5$, $\alpha_0 = 1, \sigma = 0.3, M = 1, \delta = 1$, and all runs are stopped when $\left|\frac{f(x_k) - f(x_{k-1})}{f(x_{k-1})}\right| \le 10^{-4}$, or $\frac{\|x_k - x_{k-1}\|_2}{\|x_{k-1}\|_2} \le 10^{-4}$, or $k > 500$ .

## 3 Experimental results

In this section, we exam the performance of our proposed algorithms using two color images “Lena” and “Peppers”, which are shown in Figure 1. For comparison, we also test many other algorithms presented in the literature. These include VMF [2], VDF [3], DDF [4], HDF [5], AVDF [11], AVMF [8], MAVMF [8], FPGF [9], AVMMF [13], MVMF [10], RVMF [10], FHSF [12] and quaternion based algorithm, denoted by QVMF [6].

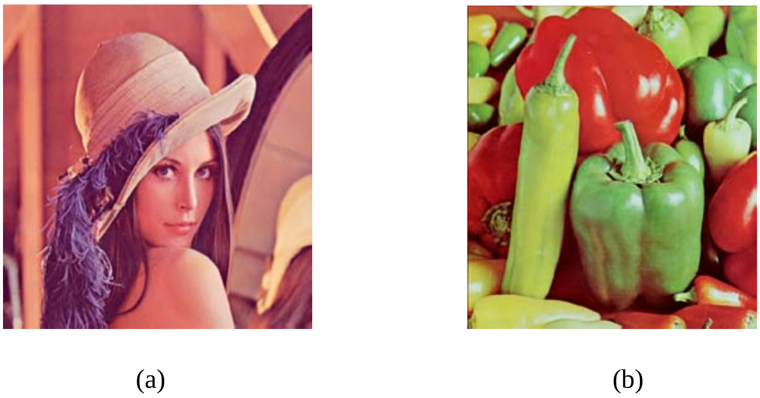
In the experiments, all above-mentioned techniques use $3 \times 3$ filter window. For DDF, we choose $p = 0.75$ as in [4], see [4] for details. For AVDF, we use AVDF2 in [11], since it has the best performance in removing impulsive noise among all AVDFs in [11]. For AVMF, we choose $\lambda_1 = 4$, and for MAVMF, we choose $\lambda_2 = 12$ as in [8]. For QVMF, the parameter Tol takes the value of 22 as in [6], see [6] for details. For FPGF, we choose Tol = 45, $m = 3$ as in [9]. For AVMMF, we set $m = 3$ as in [13]. For MVMF and RVMF, we set $t = 3$ as in [10]. And for FHSF, we set $Ht = 10, St = 11, Lt = 48$ and $m = 3$ as in [12].

For the measurement of the restoration quality the commonly used root mean squared error (RMSE) expressed through the peak signal-to-noise ratio (PSNR) was used as the RMSE is a good measure of the efficiency of impulsive noise suppression. The PSNR is defined as

$$\mathrm{PSNR} = 20\log_{10}\left(\frac{255}{\sqrt{\mathrm{MSE}}}\right)$$

$$\mathrm{MSE} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left\| x_{ij} - \tilde{x}_{ij} \right\|_2^2}{m \times n \times 3}$$

where $m, n$ are the image dimensions, and $x_{ij}$ and $\tilde{x}_{ij}$ denote the original image vector and its estimation at location $(i, j)$, respectively.



(a) (b)

**Fig. 1** Test images. (a) Lena, (b) Peppers.

Since RGB is not a perceptually uniform space in the sense that differences between colors in this space do not correspond to color differences perceived by humans, the restoration errors are often analyzed using the perceptually uniform color spaces [9]. So in this paper the restoration errors were also analyzed using the Normalized Color Difference (NCD) defined as [1],

$$\mathrm{NCD}=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\sqrt{(L^*_{x_{ij}}-L^*_{\tilde{x}_{ij}})^2+(u^*_{x_{ij}}-u^*_{\tilde{x}_{ij}})^2+(v^*_{x_{ij}}-v^*_{\tilde{x}_{ij}})^2}}{\sum_{i=1}^{m}\sum_{j=1}^{n}\sqrt{(L^*_{x_{ij}})^2+(u^*_{x_{ij}})^2+(v^*_{x_{ij}})^2}},$$

where $L^*$ represents lightness values and $(u^*,v^*)$ represents chrominance values corresponding to original $x_{ij}$ and noisy (or filtered) $\tilde{x}_{ij}$ samples expressed in CIELUV color space.

All our experiments are run when MATLAB 7.7 is used on a PC equipped with Intel Core(TM) 2.80 GHz, 3.48 GB RAM memory.

In the experiments, the two original test images are contaminated by random-valued impulsive noise of noise levels ranging from 10% to 30% with increments of 10% using the noise model (1). We summarize the performance of different methods in Table 1 and Table 2. From the tables, we see that our proposed algorithm achieves a significantly high PSNR and low NCD.

Based the above results, we can conclude that our proposed algorithm is very effective and superior to the other existing methods in random-valued impulsive noise removal.

**Table 1** Comparison of the presented algorithms for the image Lena.

| noise level | 0% | | 10% | | 20% | | 30% | |
|---|---|---|---|---|---|---|---|---|
| Method | NCD | PSNR | NCD | PSNR | NCD | PSNR | NCD | PSNR |
| NONE | 0 | ∞ | 0.0775 | 18.59 | 0.1527 | 15.66 | 0.2320 | 13.83 |
| VMF | 0.0300 | 33.10 | 0.0320 | 32.41 | 0.0346 | 31.48 | 0.0389 | 29.63 |
| VDF | 0.0329 | 32.50 | 0.0359 | 31.32 | 0.0413 | 28.97 | 0.0533 | 25.26 |
| DDF | 0.0316 | 32.81 | 0.0338 | 32.00 | 0.0377 | 30.35 | 0.0460 | 27.25 |
| HDF | 0.0309 | 33.21 | 0.0336 | 32.44 | 0.0377 | 31.11 | 0.0462 | 28.56 |
| AVDF | 0.0323 | 33.53 | 0.0350 | 32.64 | 0.0404 | 30.89 | 0.0533 | 27.76 |
| AVMF | 0.0009 | 46.20 | 0.0070 | 33.62 | 0.0220 | 26.43 | 0.0604 | 20.91 |
| MAVF | 0.0250 | 33.77 | 0.0299 | 32.66 | 0.0353 | 31.05 | 0.0444 | 28.16 |
| FPGF | 0.0011 | 41.61 | 0.0062 | 36.90 | 0.0120 | 34.00 | 0.0199 | 30.78 |
| AVMMF | 0.0322 | 32.72 | 0.0344 | 31.97 | 0.0372 | 31.03 | 0.0416 | 29.23 |
| MVMF | 0.0317 | 32.95 | 0.0342 | 32.05 | 0.0377 | 30.66 | 0.0457 | 27.76 |
| RVMF | 0.0317 | 32.95 | 0.0343 | 32.05 | 0.0377 | 30.65 | 0.0458 | 27.75 |
| FHSF | 0.0035 | 39.34 | 0.0110 | 31.59 | 0.0188 | 28.90 | 0.0284 | 26.88 |
| QVMF | 0.0004 | 46.61 | 0.0067 | 34.56 | 0.0153 | 31.10 | 0.0290 | 27.98 |
| PROPOSED | 0.0001 | 52.32 | 0.0048 | 39.36 | 0.0099 | 35.92 | 0.0178 | 32.66 |

**Table 2** Comparison of the presented algorithms for the image Peppers.

| noise level | 0% | | 10% | | 20% | | 30% | |
|---|---|---|---|---|---|---|---|---|
| method | NCD | PSNR | NCD | PSNR | NCD | PSNR | NCD | PSNR |
| NONE | 0 | ∞ | 0.1404 | 18.07 | 0.2094 | 15.10 | 0.2794 | 13.31 |
| VMF | 0.0393 | 32.02 | 0.0629 | 31.30 | 0.0832 | 29.97 | 0.0596 | 28.10 |
| VDF | 0.0457 | 30.60 | 0.0866 | 29.11 | 0.0921 | 26.54 | 0.1071 | 22.51 |
| DDF | 0.0421 | 31.39 | 0.0758 | 30.27 | 0.0888 | 28.19 | 0.0991 | 24.55 |
| HDF | 0.0725 | 32.29 | 0.0846 | 31.22 | 0.0894 | 29.34 | 0.0997 | 26.49 |
| AVDF | 0.0400 | 32.64 | 0.0432 | 31.33 | 0.0543 | 28.94 | 0.0671 | 25.27 |
| AVMF | 0.0011 | 45.16 | 0.0752 | 32.34 | 0.0888 | 25.14 | 0.1241 | 19.91 |
| MAVF | 0.0347 | 32.63 | 0.0663 | 31.33 | 0.0687 | 29.25 | 0.0808 | 26.44 |
| FPGF | 0.0014 | 40.38 | 0.0524 | 35.68 | 0.0775 | 32.21 | 0.0659 | 29.20 |
| AVMMF | 0.0410 | 31.63 | 0.0661 | 30.93 | 0.0865 | 29.61 | 0.0904 | 27.79 |
| MVMF | 0.0408 | 31.92 | 0.0427 | 30.87 | 0.0851 | 29.23 | 0.0921 | 26.72 |
| RVMF | 0.0408 | 31.92 | 0.0427 | 30.86 | 0.0852 | 29.22 | 0.0922 | 26.71 |
| FHSF | 0.0058 | 37.80 | 0.0777 | 31.53 | 0.0826 | 28.69 | 0.0889 | 26.58 |
| QVMF | 0.0011 | 42.30 | 0.0754 | 33.30 | 0.0816 | 29.70 | 0.0920 | 26.44 |
| PROPOSED | 0.0002 | 49.35 | 0.0733 | 37.37 | 0.0764 | 33.93 | 0.0637 | 31.68 |

## Acknowledgements

This work is supported by the grant “The First-class Discipline of Universities in Shanghai”.

## References

1. K. N. Plataniotis, A. N. Venetsanopoulos (2000), Color image processing and applications. Berlin: Springer Verlag
2. J. Astola, P. Haavisto, Y. Neuvo (1990) Vector median filters. Proc. IEEE 78:678-689
3. P. E. Trahanias, D. G. Karakos, A. N. Venetsanopoulos(1996) Directional processing of color images: Theory and experimental results. IEEE Trans. Image Process. 5:868-880
4. D. G. Karakos, P. E. Trahanias (1997) Generalized multichannel image-filtering structures. IEEE Trans. Image Process. 6:1038-1045
5. M. Gabbouj, F.A. Cheickh(1996) Vector median-vector directional hybrid filter for color image restoration. Proceedings of EUSIPCO-96, 879-881
6. L. Jin, D. Li (2007) An efficient color-impulse detector and its application to color images. IEEE Signal Process. Lett. 14:397-400
7. R. Lukac (2003) Adaptive vector median filtering. Pattern Recognit. Lett. 24:1889- 1899
8. R. Lukac, K. N. Plataniotis, A. N. Venetsanopoulos, B. Smolka (2005) A statistically-switched adaptive vector median filter. J. Intell. Robot. Syst. Theory Appl. 42:361-391

9. B. Smolka, A. Chydzinski (2005) Fast detection and impulsive noise removal in color images. Real-Time Imag. 11:389-402

10. S. Morillas, V. Gregori (2011) Robustifying vector median filter. Sensors 11:8115-8126

11. K. N. Plataniotis, D. Androutsos, A. N. Venetsanopoulos (1998) Color image processing using adaptive vector directional filters. IEEE Trans. Circuits and Syst. II 45:1414-1419

12. M. E. Celebi, H. A. Kingravi, B. Uddin, Y. A. Aslandogan (2007) A fast switching filter for impulsive noise removal from color images. J. Imaging Science and Technology 51:155-165

13. S. Morillas, V. Gregori, A. Sapna (2011) Adaptive marginal median filter for color images. Sensors 11:3205-3213

14. R. Garnett, T. Huegerich, C. Chui, W.-J. He (2006) A universal noise removal algorithm with an impulse detector. IEEE Trans. Image Process. 14:1747-1754

15. M. Nikolova (2004) A variational approach to remove outliers and impulse noise. Journal of Mathematical Imaging and Vision 20:99-120

16. R. H. Chan, C. W. Ho, M. Nikolova (2005) Salt-and-pepper noise removal by median- type noise detectors and detail-preserving regularization. IEEE Transactions on Image Processing 14:1479-1485

17. M. Raydan (1997) The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J. Optim. 7:26-33