Journal of Engineering and Applied Sciences 15 (2): 350-364, 2020
ISSN: 1816-949X

A Two-Stage Quaternion Vector Median Filter for Removing Impulse Noise in Color Images

[1]P. Roji Chanu and [2]Kh. Manglem Singh
[1]Department of Electronics and Communication Engineering,
National Institute of Technology, Nagaland Nagaland, India
[2]Department of Computer Science and Engineering, National Institute of Technology
Manipur, Manipur, India

Abstract: This study presents a novel two-stage filtering algorithm for removing impulse noise in color images. Quaternion theory is used to represent the intensity and chromaticity differences of two color pixels. Use of quaternion treats color pixels as vectors and processes color images as single unit rather than as separated color components. This preserves the existing correlation and three dimensional vector natures of the color channels. In the first stage of noise detection, the color pixels are sorted and assigned a rank based on the aggregated sum of color pixel differences with other pixels inside the filtering window. The central pixel is considered as probably corrupted by an impulse if its rank is bigger than a predefined rank. In the second stage, the probably corrupted candidate is again checked for an edge or an impulse by using four Laplacian convolution kernels. If the minimum difference of these four convolution is larger than a predefined threshold, then the central pixel is regarded as an impulse. The noisy pixel is replaced by output of weighted vector median filter implemented using the quaternion distance. More weight is assigned to those pixels belonging to the direction of minimum difference. Experimental results indicate the improved performance of the proposed filter in suppressing the impulse noise while retaining the original image details comparing against other well-known filters.

Key words: Quaternion, chromaticity, convolution, impulse, laplacian, vector median

INTRODUCTION

Color information is of paramount importance used by computer vision systems in various fields like image understanding and pattern recognition. However, the quality of information is degraded by certain types of noise during image acquisition and transmission. Impulse noise is one such noise which occurs for a short duration. Presence of impulse noise complicates the subsequent stages of image processing such as edge detection, segmentation, etc. Hence, noise filtering is the most important task in signal processing. Vector Median Filters (VMF) (Astola *et al*., 1900), Basic Vector Directional Filters (BVDF) (Trahanias and Venetsanopoulos, 1993) and Directional Distance Filter (DDF) (Karakos and Trahanias, 1995) are the most common filters used for removing impulse noise. These are nonlinear filters based on order statistic. But they have the drawback of excessive blurring leading to loss of fine details. This is because they filter every pixel without checking the presence of an impulse. In order to improve the performance of VMF and its extensions, adaptive switching filters are developed. Switching filters detect the presence of an impulse and filtering is performed if found corrupted. Adaptive Center-Weighted VMF (ACWVMF), Adaptive Center-Weighted VDF (ACWVDF) and Adaptive Center Weighted DDF (ACWDDF) (Lukac and Smolka, 2003; Lukac, 2004) are efficient adaptive switching filters. Some other powerful switching filters are the Adaptive VMF (AVMF) (Lukac, 2003), Entropy VMF (EVMF), peer group filters (Smolka, 2010), Fuzzy VMF (Plataniotis *et al*., 1996) and vector sigma filters (Lukac *et al*., 2006).

For improving the efficiency of filters, certain color image filters based on quaternion theory have been developed (Evans *et al*., 2000; Sangwine and Ell, 2000; Cai and Mitra, 2000; Jin and Li, 2007; Jin *et al*., 2010; Geng *et al*., 2012; Wang *et al*., 2014). In this study, a switching quaternion vector median filter is proposed for removing impulse noise from color images. A quaternion considers a color pixel as a hyper-complex number treating it as a single unit rather than as separate color components. This handles the correlation existing among the color channels naturally (Subakan and Vemuri, 2011) thereby preserving its vector nature. Quaternion color representation characterizes the spatial and color aspects of the image pixel's textures (Shi and Funt, 2007). In this research, the difference between two color pixels is expressed in terms of the intensity and chromaticity differences using the quaternion theory. A color image is a vector signal having both the magnitude and direction. Quaternion representation considers the two aspects of vector by considering both the intensity (magnitude) and

Corresponding Author: P. Roji Chanu, Department of Electronics and Communication Engineering,
National Institute of Technology, Nagaland, Nagaland, India

chromaticity (direction). This improves the discrimination of vector pixels as compared with the commonly used Euclidean distance which considers only the intensity component. Based on the order statistics technique, the color pixels inside the sliding window are assigned a rank. This rank depends on the aggregated sum of color pixel differences from other neighboring pixels inside the filtering window. If the rank of central pixel is greater than a predefined rank, then it is treated as probably noisy and consider for the second stage of noise detection. In this stage, the minimum of the color window is calculated. The central pixel is considered as noisy if this distance is greater than a pre-defined threshold and is replaced by the output of the weighted VMF computed using the quaternion distance.

Quaternions: A quaternion q is a hyper complex number having a real part a and three imaginary parts b, c and d (Cai and Mitra, 2000) given by:

$$q = a+bi+cj+dk \tag{1}$$

where, i, j and k are complex operators which follow Hamilton's rules:

$$\begin{cases} i^2 = j^2 = k^2 = ijk = -1 \\ ij = k,\ jk = i,\ ki = j \\ ji = -k,\ kj = -i,\ ik = -j \end{cases} \tag{2}$$

A quaternion is split into its "Scalar" (S) and "Vector" (V) parts as follows:

$$q = Sq+Vq,\ Sq = a,\ Vq = bi+cj+dk \tag{3}$$

If $q_1 = a_1+b_1i+c_1j+d_1k$ and $q_2 = a_2+b_2i+c_2j+d_2k$ are two quaternions, then the addition is given by:

$$q_1+q_2 = (a_1+a_2)+(b_1+b_2)i+(c_1+c_2)j+(d_1+d_2)k \tag{4}$$

Quaternion multiplication follows the multiplicative properties of Hamilton's rule. It is expressed as follows:

$$\begin{aligned} q_1q_2 = {} & (a_1+b_1i+c_1j+d_1k)(a_2+b_2i+c_2j+d_2k) = \\ & (a_1a_2-b_1b_2-c_1c_2-d_1d_2)+(a_1b_2-b_1a_2-c_1d_2-d_1c_2)i+ \\ & (a_1c_2-b_1d_2-c_1a_2-d_1b_2)j+(a_1d_2-b_1c_2-c_1b_2-d_1a_2)k \end{aligned} \tag{5}$$

Conjugate of a quaternion is given by:

$$q^* = (a+bi+cj+dk)^* = (a-bi-cj-dk) \tag{6}$$

Also, the magnitude or norm is expressed as:

$$|q| = \sqrt{a^2+b^2+c^2+d^2} \tag{7}$$

A quaternion having a zero real part is called a pure quaternion and is denoted by:

$$q = bi+cj+dk \tag{8}$$

A quaternion with $|q| = 1$ is called a unit quaternion q_u. Any quaternion can be expressed in polar form as:

$$q = |q|e^{\mu\theta} = |q|(\cos\theta+\mu\sin\theta) \tag{9}$$

where, μ is a unit pure quaternion (a pure quaternion with unit magnitude). It is denoted by:

$$\mu = \frac{(bi+cj+dk)}{|bi+cj+dk|} = \frac{(i+j+k)}{\sqrt{1+1+1}} = \frac{(i+j+k)}{\sqrt{3}} \tag{10}$$

It is also referred to as eigenaxis which identifies the direction in the three-dimensional ($\mathbb{R}^3$) space of the vector part. θ is the angle between the real part and three dimensional imaginary parts. It is also known as the eigenangle or phase. It is computed as:

$$\begin{cases} \tan^{-1}\dfrac{\sqrt{b^2+c^2+d^2}}{a}, & a = 0 \\ \dfrac{\pi}{2}, & a = 0 \end{cases} \tag{11}$$

To use a quaternion which is defined in $\mathbb{R}^4$ space to operate on a color vector in $\mathbb{R}^3$ space, a pure quaternion is used. An RGB color vector is represented by a pure quaternion whose imaginary parts represent the red, green and blue component, respectively. This concept is supported by the coincidence between the three-space imaginary part of the quaternion and three-dimensional nature of the RGB color triplets. It is denoted by:

$$q = ri+gj+bk \tag{12}$$

where r, g and b are the red, green and blue amplitudes of the signal. The gray line of the unit RGB color space is denoted by the unit pure quaternion $\mu = (i+j+k)/\sqrt{3}$. A pixel in the gray line is achromatic where all the three components are equal c = g = b). By Cai and Mitra (2000) an operator called quaternion unit transform is derived:

$$\begin{aligned} UqU^* = {} & \left[\cos\theta+\frac{1}{\sqrt{3}}\sin\theta(i+j+k)\right](ri+gj+bk) \\ & \left[\cos\theta-\frac{1}{\sqrt{3}}\sin\theta(i+j+k)\right] \end{aligned} \tag{13}$$

UqU^* is rotation of vector q by 2θ around μ axis which represents the gray line in the RGB space. The color pixels q and its rotated form UqU* are positioned at equal distances from the gray line μ in the opposite direction. Thus, q+UqU* should lie on the gray line. The simplification of quaternion rotational operator is given by:

$$\begin{aligned}&Y = UpU^* = \left(\left(S(U)\right)^2 - \left\|V(U)\right\|^2\right)q + 2\left(V(U).q\right)V(U) + \\ &2S(U)\left(V(U)\times q\right) = \left(\cos^2\theta - \sin^2\theta\right)(ri+gj+bk) + \\ &2\left[\left(\frac{1}{\sqrt{3}}\sin\theta(i+j+k).(ri+gj+bk)\right]\left(\frac{1}{\sqrt{3}}\sin\theta(i+j+k)\right) + \\ &2\left(\cos\theta\times(ri+gj+bk)\right) = \cos 2\theta(ri+gj+bk) + \\ &\frac{2\sin^2\theta(r+g+b)}{3}(i+j+k) + \frac{\sin 2\theta}{\sqrt{3}}\left[(b-g)i+(r-b)j+(g-r)k\right] \\ &Y_{RGB}+Y_I+Y_\Delta\end{aligned} \tag{14}$$

where, $Y_{RGB} = \cos 2\theta(ri+gj+bk), Y_I \frac{2\sin^2\theta(r+g+b)}{3}\ (i+j+k) Y_\Delta = \frac{\sin 2\theta}{\sqrt{3}}\left[(b-g)i+(r-b)j+(g-r)k\right]$ denote the RGB space component, the intensity and the color difference. Y_Δ is the projection of the tristimuli values in the Maxwell triangle which is rotated by 90°. It represents the chromaticity component. When $\theta = \pi/4$, Y_{RGB} reduces to zero and the unit transform transfers the RGB space into Hue-Saturation-Intensity (HSI) like space (Cai and Mitra, 2000). Denoting then $T = U|_{\theta=\pi/4}$, then:

$$\begin{aligned}&G = TqT^* = UpU^*|_{\theta=\frac{\pi}{4}} = \frac{\mu}{3}(r+g+b) + \\ &\frac{1}{\sqrt{3}}\left[(b-g)i+(r-b)j+(g-r)k\right] = Y_I+Y_\Delta\end{aligned} \tag{15}$$

Similarly:

$$G' = T^*qT = \frac{\mu}{3}(r+g+b) - \frac{1}{\sqrt{3}}\left[(b-g)i+(r-b)j+(g-r)k\right] = Y_I+Y_\Delta \tag{16}$$

From Eq. 15 and 16, we have $Y_I = 1/2[TqT^*+T^*qT]$

$$Y_I = 1/2\left[TqT^*+T^*qT\right] \tag{17}$$

$$Y_\Delta = \frac{1}{2}\left[TqT^*-T^*qT\right] \tag{18}$$

For two pixels q_1 and q_2, the intensity and chromaticity parts are computed as follows:

$$Y_{I(q_1)} = \frac{1}{2}\left[Tq_1T^*T^*q_1T\right] \tag{19}$$

$$Y_{I(q_2)} = \frac{1}{2}\left[Tq_2T^*T^*q_2T\right] \tag{20}$$

$$Y_{\Delta(q_1)} = \frac{1}{2}\left[Tq_1T^*-T^*q_1T\right] \tag{21}$$

$$Y_{\Delta(q_2)} = \frac{1}{2}\left[Tq_2T^*-T^*q_2T\right] \tag{22}$$

Therefore, the intensity difference between q_1 and q_2 is derived as:

$$d_1(q_1, q_2) = Y_{I(q_1)}-Y_{I(q_2)} = \frac{1}{\sqrt{3}}(r_1+g_1+b_1-r_2-g_2-b_2) \tag{23}$$

Similarly, the chromaticity difference between q_1 and q_2 is derived as:

$$\begin{aligned}&d_2(q_1, q_2) = Y_{\Delta(q_1)}-Y_{\Delta(q_2)} = \\ &\frac{1}{\sqrt{3}}\left[\left((b_1-g_1)-(b_2-g_2)\right)i+\left((r_1-b_1)-(r_2-b_2)\right)j+\left((g_1-r_1)-(g_2-r_2)\right)k\right]\end{aligned} \tag{24}$$

When the two pixels are similar, both the intensity and chromaticity differences approach zero. Combining these two differences, the color pixel difference between two pixels is taken as their summation which is defined below:

$$d(q_1, q_2) = d_1(q_1, q_2) + d_2(q_1, q_2) \tag{25}$$

It quantifies both the intensity and chromaticity differences as compared with other distance similarity methods such as Euclidean distance, city-block distance, cosine distance and fuzzy magnitude similarity measure.

MATERIALS AND METHODS

The algorithm consists of two phases of impulse detection and filtering action. The algorithm is defined below. Two-phase noise detection: let us consider a 3×3 filtering window in Fig. 1. The color difference between two pixels is computed. Then the sum of aggregated color differences δ_m, (m = 1, 2, 3, …, N) assigned to a sample with the remaining pixels inside the filtering window is calculated as follows:

$$\delta_m = \sum_{n=1}^{N} d(q_m, q_n), (1 \le n \le N) \tag{26}$$

Then the aggregated distance $\delta_1, \delta_2, \delta_3, \ldots, \delta_N$ are sorted in an ascending order and assign a rank, i.e.:

$$\delta_{(1)} \le \delta_{(2)} \le, \ldots, \delta_{(N)} \tag{27}$$

This also implies similar ordering of the input set $q_1, q_2, q_3, \ldots, q_N$ resulting in an ordered input sequence:

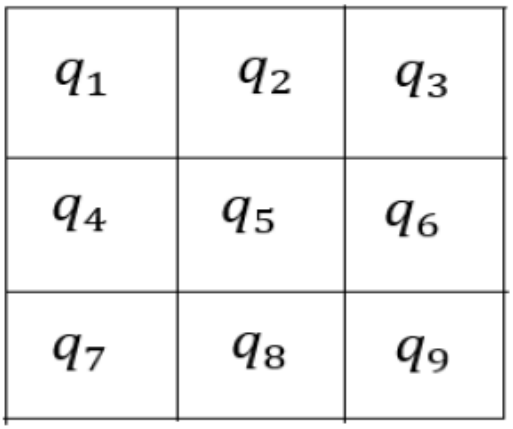

Fig. 1: A 3×3 filtering window

$$q_{(1)} \le q_{(2)} \le, \ldots, q_{(N)} \quad (28)$$

The lowest ranked sample $q_{(1)}$ is the output of classical VMF. If $q_{(1)}$, $q_{(2)}$, $q_{(3)}$, …, $q_{(k)}$ are set of k input ranked samples having highest similarity to input set, then it acts as a good measurement for detection of corruption probability of the central pixel. If the rank of the central pixel lies at the extreme end, it is regarded as an impulse. On the other hand, if it ranges in the middle then the central pixel is possibly not an impulse. The main advantage is that pixels with extreme rank which are likely to possess abnormal value are processed and pixel replacement is conditioned to their ranks. In the first stage of noise detection, if the central pixel has a rank r_c bigger than the predefined rank r_k, it is regarded as an impulse. However, it is quite common to mistaken a pixel as an impulse. For example in Fig. 2a, we consider a portion of a color image depicting the components of each channel. After calculating the aggregated sum of distances (Fig. 2b), we sorted the pixel values by assigning a rank as defined below:

$$q^1_{(1)} \le q^3_{(2)} \le q^2_{(3)} \le q^4_{(4)} \le q^8_{(5)} \le q^9_{(6)} \le q^3_{(7)} \le q^5_{(8)} \le q^7_{(9)} \quad (29)$$

where, $q_{(1)}{}^1$ corresponds to the lowest rank sample which is the first pixel q_1. The central pixel can occupy the rank 7, 8 or 9. If we set the predefined rank to be 4 or 5, the central pixel q_5 is consider as an impulse. But q_5 is actually an edge pixel. Thus, there is possibility of false detection resulting in removal or blurring of thin lines, fine details and edges. To overcome this problem, we go for second stage of noise detection using the four edge directional method. For a noisy pixel detected in the first stage, we use the four-dimensional Laplacian operators given in Fig. 3 to check if it is a noisy pixel or an edge. The input

(a)

R 223 G 120 B 100	R 223 G 120 B 100	R 226 G 129 B 116
R 223 G 120 B 100	R 226 G 129 B 116	R 223 G 120 B 100
R 226 G 129 B 116	R 223 G 120 B 100	R 223 G 120 B 100

(b)

84	84	168
84	168	84
168	84	84

Fig. 2: (a) A portion of a color image and (b) Aggregated sum of distances

$$\begin{bmatrix} -1 & 2 & -1 \end{bmatrix} \quad \begin{bmatrix} -1 & & \\ & 2 & \\ & & -1 \end{bmatrix}$$

$$\begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} \quad \begin{bmatrix} & & -1 \\ & 2 & \\ -1 & & \end{bmatrix}$$

Fig. 3: Four convolution kernels

image pixels ($I_{(x,\ y)}$) inside the sliding window is convolved with four convolution kernels W_t (t = 1, 2, 3, 4). The minimum difference ($L_{(x, y)}$) of these four convolutions is considered for detection of edge:

$$L_{x,y} = \min\left\{I_{(x,y)} \otimes W_t | : t = 1,2,3,4\right\} \quad (30)$$

If $L_{x,y}$ is greater than a pre-defined threshold Tol, the candidate is regarded as an impulse else it is an edge. Mathematically, it is depicted below:

(a)

R 223 G 120 B 100	R 223 G 120 B 100	R 226 G 129 B 116
R 223 G 120 B 100	R 0 G 74 B 81	R 223 G 120 B 100
R 226 G 129 B 116	R 223 G 120 B 100	R 223 G 120 B 100

(b)

342	342	484
342	2348	342
484	342	342

Fig. 4: (a) Portion of a noisy image and (b) Aggregated sum of distances

$$I^{output} = \begin{cases} I^{filter}, L_{x,y} \geq Tol \\ qs, otherwisex \end{cases} \quad (31)$$

Noisy pixel is considered for filtering action I^{filter} define in the next section. For example, given in Fig. 4, the direction with minimum difference corresponds to 45° which is 0. If we assume a Threshold T of 35, then $L_{x,y}$ <Tol and q_5 is not detected as an impulse. To show the noise detection capability of the filter, we again consider a portion of a color image depicting components of each channel in Fig. 4. The central pixel is corrupted by noise with R = 0, G = 74 and B = 81.

The central pixel is detected as corrupted by first-stage, since, it occupies the extreme rank after sorting the aggregated sum of distances. In the second-stage, the minimum difference ($L_{x, y}$) of the four convolutions is 291 in 45° direction. Thus, it is considered as noise, since, $L_{x, y}$>>Tol. This shows the capability of our noise detection algorithm.

Filtering: For filtering the noisy pixel, we consider a weighted vector median filter with 3×3 Window. The weight of each pixel is determined from the convolution values along the four edge directions. Let S represents the set of color pixels belonging to the direction with minimum difference, i.e., $L_{x, y}$. Therefore, the corrupted pixel is restored as:

$$I^{filter} = \text{Vector median}\left\{W_{x,y} \lozenge I_{x,y}\right\} \quad (32)$$

where, the weight $W_{x,y} = \begin{cases} 2, \text{ if } I_{x,y} \in S \\ 1, \text{ otherwise} \end{cases}$ and ◊ denotes the repetition operator. Here, the pixels belonging to the direction of minimum differences are given more importance by assigning larger weight.

Use of quaternion distance: In order to differentiate the color pixels accurately, a vector's magnitude (intensity) and direction (chromaticity) are both considered holistically. The proposed algorithm is implemented using the quaternion distance derived in Eq. 25. The commonly used Euclidean distance considers only the intensity as compared with quaternion.

RESULTS AND DISCUSSION

The impulse noise model (Celebi *et al*., 2007) is used for simulation. It has the following form:

$$n(x) = \begin{cases} o & \text{with probability } 1-\varphi \\ (d, o_2, o_3)^T & \text{with probability } \varphi_1 \cdot \varphi \\ (o_1, d, o_3)^T & \text{with probability } \varphi_2 \cdot \varphi \\ (o_1, o_2, d)^T & \text{with probability } \varphi_3 \cdot \varphi \\ (d, d, d)^T & \text{with probability } \varphi_1 \cdot \varphi_\varepsilon \end{cases} \quad (33)$$

where, n (x) is the noisy signal, $o = (o_1, o_2, o_3)^T$ represents the constant noise free vector pixel, φ is the sample corruption probability, φ_1, φ_2 and φ_3 are the corruption probabilities of the channel and $\varphi_\Sigma = 1-\varphi_1-\varphi_2-\varphi_3$. d represents the impulse value which is either 0 or 255 in case of fixed-valued impulse noise having equal probability whereas for random-valued impulse noise it can take any value in the range [0, 255]. The reliability of the proposed filter is evaluated by Peak-Signal-to-Noise-Ratio (PSNR) in dB, Mean Absolute Error (MAE), Normalized Color Distance (NCD) and Structural Similarity Index (SSIM). PSNR expresses the quality of the image. An image possesses good quality when the signal ratio is higher than the noise ratio. MAE depicts the erroneous value of the filtered image compared with the original image. To measure the objective criteria related to human perception, Normalized Color Distance (NCD) is used. NCD is computed in L*u*v* color space. It is the color

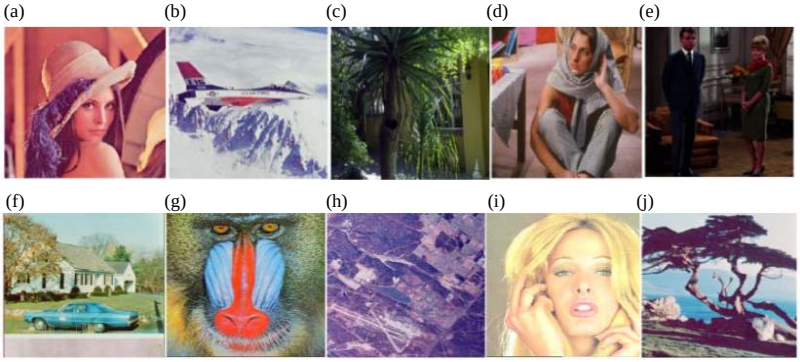

Fig. 5(a-j): Standard test images (a) Lena, (b) Airplane, (c) Aptus, (d) Barbara, (e) Couple, (f) House, (g) Mandrill, (h) Miramar, (i) Tiffany and (j) Tree

space standardized by the Commission International l' Eclairage (CIE) which is closely related to the characteristic of human perception (Lazhar *et al*., 1999). It is also suitable in defining the appropriate measures of the perceptual error between color spaces. Structural Similarity Index (SSIM) is another objective image quality measure based on human visual system metrics. They are defined below:

$$MAE = \frac{1}{3M_1N_1}\sum_{m=1}^{M_1}\sum_{n=1}^{N_1}\left|q(m,n)-f(m,n)\right| \quad (34)$$

where, $M_1 \times N_1$ is the size of the image; q(m, n) and f(m, n) are the original and filtered pixel values at (m, n) location:

$$PSNR = 10\log_{10}\left(\frac{I_{max}^2}{MSE}\right) \quad (35)$$

where, I_{max} is the maximum pixel value of the original image and MSE stands for Mean Squared Error given as:

$$MSE = \frac{1}{3M_1N_1}\sum_{m=1}^{M_1}\sum_{n=1}^{N_1}\left|q(m,n)-f(m,n)\right|^2 \quad (36)$$

$$NCD = \frac{\sum_{m=1}^{M_1}\sum_{n=1}^{N_1}\sqrt{\left(L^o(m,n)-L^x(m,n)\right)^2+\left(u^o(m,n)-u^x(m,n)\right)^2+\left(v^o(m,n)-v^x(m,n)\right)^2}}{\sum_{m=1}^{M_1}\sum_{n=1}^{N_1}\sqrt{\left(L^o(m,n)\right)^2+\left(u^o(m,n)\right)^2+\left(v^o(m,n)\right)^2}} \quad (37)$$

which is defined in the Lu*v* color space where $L^o(m, n)$, $u^o(m, n)$, $v^o(m, n)$ and $L^x(m, n)$, $u^x(m, n)$, $v^x(m, n)$ are values of the lightness and two chrominance components of the original image sample q(m, n) and filtered image sample f(m, n), respectively:

$$SSIM = \frac{(2\mu_x\mu_y+C_1)(2\mu_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \quad (38)$$

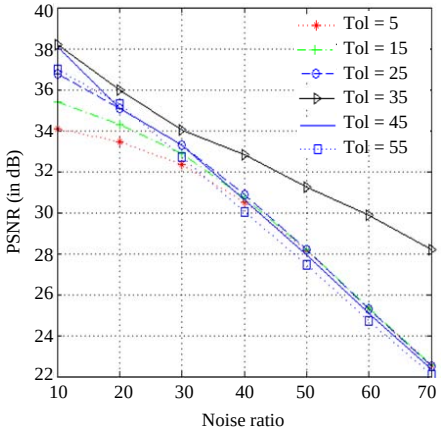

Fig. 6: PSNR values of Lena at rank = 5

which measures the similarity between two images where μ_x and μ_y are mean of the original and filtered image, σ_{xy}, σ_x^2 and σ_y^2, represent the corresponding covariance and variance of the original and filtered images. C_1 and C_2 are the constants (Wang *et al*., 2004).

The Proposed F ilter (PF) is compared with 8 different filters. They are VMF (Astola *et al*., 1900), AVMF (Lukac, 2003), Entropy VMF (EVMF) (Lukac *et al*., 2003), PGF (Smolka, 2010), ACWVMF (Lukac, 2004), Fuzzy VMF (FVMF) (Plataniotis *et al*., 1996), Rank Weighted Adaptive Switching Filter (RWASF) (Smolka *et al*., 2015) and Quaternion Switching Filter (QSF) (Geng *et al*., 2012). We have used different standard images shown in Fig. 5 and 6.

All the experiments are run in Matlab version 10 on a PC equipped with i5 processor at 2.00 GHz. Noise densities ranging between 10 and 60% are uniformly injected. To achieve the best performance of the filter, it is required to find the optimum values of both the rank and threshold. These optimum threshold and rank should give the best restoration results of the proposed filter. Various simulations on the proposed filter are carried out for different values of rank and threshold.

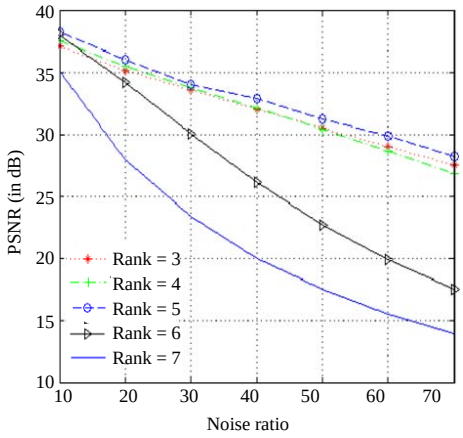

Fig. 7: PSNR values of Lena at Tol =35

Table 1: Parameters for various vector filters

Filters	Parameters
AVMF (Lukac, 2003)	Tol = 35
FVMF (Plataniotis *et al*., 1996)	δ = 0.5
PGF (Smolka, 2010)	Tol = 60
ACWVMF (Lukac, 2004)	w = 3, Tol = 80
RWASF (Smolka *et al*., 2015)	Tol = 65
QSF (Geng *et al*., 2012)	Tol = 35
Proposed Filter (PF)	r_c = 5, Tol = 35

Figure 6 and 7 give the PSNR values of the PF of Lena for fixed-valued impulse noise. The PF has the highest PSNR at rank = 5 and Tol = 35. Too lower values of rank consider healthy pixels as impulse and cause excessive filtering while much higher values will omit some impulses and avoid filtering leading to errors. The same case happens with threshold values. The parameters given in Table 1 gives the best compromised results for all vector filters in terms of MAE, PSNR, NCD and SSIM. Their parameters are set in accordance to the appropriate recommendation given in the references.

The PF is compared with 8 different filters in which rank is set at 5 and threshold at 35. All the filters are implemented in a 3×3 window since bigger window size cause excessive blurring. Table 2 gives the comparison of PF for Lena at impulse noise ranging from 10-60%. At 20%, the PF has PSNR of 36.03 dB while it is 31.34 dB for VMF and 33.55 dB for QSF. It can be seen that the PF is giving almost higher PSNR and SSIM values. Similarly, the NCD and MAE values are lower while comparing with the other filters as shown in Table 2 (bold values indicate the best results). Table 3-11 depict the performance of the PF compare with other well-known filters for other images. The superior performance of the PF can be seen from the tables. In Table 12, the average values of the results of 10 images is given. It is found that

Table 2: Comparison of different filters in removal of fixed-valued impulse noise from Lena

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0216	0.0055	0.0078	0.0188	0.0051	0.0052	0.0042	0.0047	**0.0039**
PSNR	31.76	34.95	33.41	32.13	35.50	36.19	38.61	35.73	**38.82**
MAE	3.72	1.00	2.57	3.69	0.87	0.76	**0.52**	0.83	0.54
SSIM	0.9938	0.9970	0.9957	0.9943	0.9974	0.9978	0.9987	0.9975	**0.9988**
20									
NCD	0.0184	0.0096	0.0078	0.0193	0.0062	0.0142	0.0156	0.0061	**0.0050**
PSNR	31.34	33.85	33.16	31.67	34.01	34.85	35.16	33.55	**36.03**
MAE	3.94	1.47	2.54	3.88	1.39	1.22	1.06	1.47	**1.06**
SSIM	0.9931	0.9962	0.9955	0.9936	0.9964	0.9969	0.9972	0.9959	**0.9975**
30									
NCD	0.0189	0.0088	0.0084	0.0198	0.0171	0.0166	0.0166	0.0081	**0.0065**
PSNR	30.82	32.67	32.41	31.05	32.56	33.57	32.36	31.56	**34.09**
MAE	4.18	1.98	2.69	4.13	1.95	1.72	1.70	2.23	**1.61**
SSIM	0.9922	0.9950	0.9946	0.9926	0.9948	0.9959	0.9946	0.9935	**0.9963**
40									
NCD	0.0181	**0.0100**	0.0178	0.0193	0.0179	0.0173	0.0179	0.0192	0.0172
PSNR	30.10	31.47	31.22	30.30	31.10	32.15	29.71	29.83	**32.44**
MAE	4.48	2.55	3.00	4.43	2.57	2.29	2.52	3.11	**2.23**
SSIM	0.9908	0.9934	0.9930	0.9913	0.9943	0.9943	0.9902	0.9903	**0.9948**
50									
NCD	0.0189	**0.0113**	0.0187	0.0202	0.0190	0.0182	0.0200	0.0210	0.0181
PSNR	29.32	30.33	29.87	29.48	30.01	30.92	27.10	28.26	**31.06**
MAE	4.84	3.16	3.45	4.81	3.23	**2.89**	3.65	4.15	2.90
SSIM	0.9890	0.9914	0.9904	0.9894	0.9901	0.9925	0.9825	0.9860	**0.9930**
60									
NCD	0.0205	**0.0130**	0.0200	0.0213	0.0202	0.0193	0.0233	0.0237	0.0193
PSNR	28.40	28.98	28.26	28.41	28.58	29.43	24.55	26.51	**29.61**
MAE	5.26	3.87	4.10	5.33	3.98	**3.61**	5.29	5.50	3.67
SSIM	0.9864	0.9883	0.9862	0.9865	0.9856	0.9894	0.9692	0.9790	**0.9896**
70									
NCD	0.0223	0.0215	0.0218	0.0220	0.0219	**0.0209**	0.0286	0.0270	0.0210
PSNR	27.34	27.69	26.64	27.29	27.10	28.00	22.11	25.03	**28.17**
MAE	5.81	4.69	4.97	6.03	4.86	4.44	7.72	7.08	**4.17**
SSIM	0.9827	0.9842	0.9802	0.9826	0.9790	0.9853	0.9480	0.9704	**0.9859**

Table 3 Comparison of different filters in removal of fixed-valued impulse noise from airplane

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0086	0.0191	0.0170	0.0089	0.0193	**0.0025**	0.0180	0.0349	0.0345
PSNR	28.15	28.77	28.72	28.20	28.83	29.13	31.30	30.48	**31.54**
MAE	3.28	1.38	2.36	3.36	1.31	1.10	0.78	1.12	**0.73**
SSIM	0.9811	0.9848	0.9843	0.9815	0.9852	0.9867	0.9876	0.9849	**0.9877**
20									
NCD	**0.0088**	0.0195	0.0351	0.0095	0.0197	0.0136	0.0344	0.0352	0.0345
PSNR	27.85	28.40	28.49	27.89	28.37	28.77	29.98	29.52	**30.66**
MAE	3.52	2.21	2.64	3.89	2.27	1.91	2.09	2.44	**1.23**
SSIM	0.9794	0.9830	0.9832	0.9795	0.9827	0.9851	0.9833	0.9811	**0.9851**
30									
NCD	**0.0096**	0.0200	0.0352	0.0105	0.0354	0.0152	0.0343	0.0357	0.0345
PSNR	27.49	27.99	28.08	27.52	27.86	28.35	28.59	28.42	**29.73**
MAE	3.57	2.21	2.64	3.89	2.27	1.91	2.09	2.44	**1.77**
SSIM	0.9771	0.9806	0.9811	0.9769	0.9798	**0.9828**	0.9772	0.9756	0.9814
40									
NCD	**0.0147**	0.0205	0.0353	0.0191	0.0358	0.0357	0.0343	0.0364	0.0347
PSNR	27.02	27.44	27.47	27.02	27.16	27.82	26.92	27.22	**28.75**
MAE	4.13	2.72	3.00	4.28	2.84	2.40	3.08	3.35	**2.40**
SSIM	0.9740	0.9775	0.9771	0.9736	0.9756	**0.9800**	0.9672	0.9678	0.9742
50									
NCD	**0.0163**	0.0363	0.0355	0.0213	0.0362	0.0361	0.0343	0.0370	0.0348
PSNR	26.42	26.78	26.66	26.36	26.47	27.17	25.02	25.92	**27.72**
MAE	4.56	3.30	3.53	4.77	3.50	**3.02**	4.48	4.48	3.15
SSIM	**0.9693**	0.9726	0.9711	0.9683	0.9692	0.9752	0.9513	0.9565	0.9608
60									
NCD	0.0373	0.0370	0.0359	**0.0277**	0.0370	0.0368	0.0346	0.0388	0.0352
PSNR	25.62	25.89	25.62	25.50	25.54	26.38	23.14	24.55	**26.62**
MAE	5.10	4.03	4.30	5.43	4.29	3.71	6.46	5.91	**3.60**
SSIM	0.9625	0.9657	0.9624	0.9608	0.9594	**0.9683**	0.9274	0.9401	0.9659
70									
NCD	0.0383	0.0380	0.0365	0.0361	0.0381	0.0376	0.0359	0.0409	**0.0358**
PSNR	24.68	24.86	24.45	24.56	24.38	25.44	21.01	23.19	**25.32**
MAE	5.79	4.91	5.28	6.31	5.24	4.54	9.39	7.77	**4.48**
SSIM	0.9519	0.9553	0.9490	0.9494	0.9430	0.9574	0.8888	0.9172	**0.9569**

Table 4: Comparison of different filters in removal of fixed-valued impulse noise from Aptus

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0338	0.0193	0.0243	0.0349	0.0172	0.0147	0.0108	0.0148	**0.087**
PSNR	25.32	26.09	26.82	25.34	26.30	26.88	28.21	26.63	**29.86**
MAE	6.34	4.04	4.54	6.50	3.68	3.17	2.05	3.21	**1.66**
SSIM	0.9414	0.9521	0.9593	0.9416	0.9547	0.9605	0.9719	0.9580	**0.9726**
20									
NCD	0.0357	0.0232	0.0250	0.0373	0.0221	0.0187	0.0173	0.0211	**0.0147**
PSNR	25.01	25.62	26.58	24.94	25.70	26.34	26.92	25.73	**28.43**
MAE	6.68	4.64	4.65	6.93	4.41	3.79	2.90	4.20	**1.66**
SSIM	0.9370	0.9465	0.9572	0.9357	0.9479	0.9551	0.9623	0.9481	**0.9660**
30									
NCD	0.0380	0.0274	0.0270	0.0401	0.0272	0.0232	0.0254	0.0285	**0.0213**
PSNR	24.63	25.12	26.11	24.48	25.13	25.75	25.55	24.84	**27.16**
MAE	7.09	5.28	4.94	7.41	5.17	4.47	4.12	5.33	**3.46**
SSIM	0.9310	0.9400	0.9525	0.9283	0.9404	0.9485	0.9486	0.9358	**0.9573**
40									
NCD	0.0408	0.0322	0.0304	0.0434	0.0328	**0.0282**	0.0355	0.0369	0.0290
PSNR	24.16	24.52	25.48	23.96	24.42	25.06	24.14	23.94	**25.95**
MAE	7.59	6.03	5.45	7.99	6.03	5.26	5.47	6.61	**4.52**
SSIM	0.9200	0.9310	**0.9451**	0.9187	0.9298	0.9394	0.9297	0.9207	0.9439
50									
NCD	0.0443	0.0375	0.0352	0.0473	0.0391	**0.0338**	0.0481	0.0467	0.0369
PSNR	23.61	23.89	24.72	23.41	23.67	24.35	22.75	23.01	**24.82**
MAE	8.19	6.86	6.14	8.66	6.98	6.13	7.11	8.10	**5.67**
SSIM	0.9122	0.9200	0.9350	0.9074	0.9165	0.9285	0.9048	0.9010	**0.9354**
60									
NCD	0.0486	0.0438	0.0416	0.0521	0.0462	**0.0403**	0.0649	0.0580	0.0425
PSNR	22.89	23.09	23.77	22.73	22.77	23.51	21.18	22.06	**23.77**

Table 4: Conitnue

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
MAE	8.97	7.87	7.07	9.50	8.09	7.17	9.30	9.85	**6.93**
SSIM	0.8963	0.9039	0.9198	0.8914	0.8974	0.9134	0.8673	0.8760	**0.9202**
70									
NCD	0.0536	0.0511	0.0498	0.0579	0.0538	**0.0483**	0.0877	0.0714	0.0519
PSNR	22.16	22.28	**22.76**	22.02	22.03	22.59	19.63	21.02	22.68
MAE	9.86	9.04	8.28	10.51	9.33	8.58	12.15	11.97	**8.25**
SSIM	0.8770	0.8838	0.8997	0.8720	0.8739	0.8918	0.8177	0.8408	**0.8992**

Table 5: Comparison of different filters in removal of fixed-valued impulse noise from Barbara

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0324	0.0245	0.0284	0.0300	0.0265	0.0188	0.0197	0.0206	**0.0169**
PSNR	23.01	23.70	23.86	23.21	24.04	24.18	25.76	24.22	**26.72**
MAE	10.23	7.61	8.45	10.31	6.63	6.64	3.75	6.07	**3.45**
SSIM	0.9400	0.9493	0.9508	0.9424	0.9534	0.9547	**0.9692**	0.9554	0.9658
20									
NCD	0.0329	0.0296	0.0285	0.0308	0.0280	0.0263	0.0389	0.0269	**0.0243**
PSNR	22.92	23.51	23.89	23.11	23.74	23.93	24.95	23.71	**25.56**
MAE	10.39	8.07	8.14	10.54	7.33	7.19	4.91	7.13	**4.79**
SSIM	0.9388	0.9471	0.9513	0.9411	0.9500	0.9520	**0.9630**	0.9497	0.9613
30									
NCD	0.0336	0.0331	0.0317	0.0322	0.0434	**0.0301**	0.0411	0.0315	0.0402
PSNR	22.79	23.31	23.82	22.99	23.44	23.68	24.13	23.26	**24.73**
MAE	10.62	8.54	8.07	10.81	8.03	7.74	6.13	8.19	**5.87**
SSIM	0.9370	0.9446	0.9508	0.9394	0.9464	0.9492	0.9556	0.9441	**0.9560**
40									
NCD	0.0360	0.0343	**0.0324**	0.0353	0.0445	0.0314	0.0439	0.0461	0.0417
PSNR	22.67	23.07	23.61	22.82	23.11	23.40	23.27	22.77	**24.29**
MAE	10.84	9.08	8.25	11.14	8.75	8.34	7.47	9.32	**6.59**
SSIM	0.9326	0.9415	0.9485	0.9371	0.9422	0.9458	0.9463	0.9372	**0.9485**
50									
NCD	0.0354	0.0336	0.0336	0.0383	0.0456	0.0329	0.0476	0.0485	**0.0303**
PSNR	22.50	22.82	23.30	22.64	22.78	23.10	22.26	22.23	**23.60**
MAE	11.14	9.61	8.64	11.54	9.46	8.97	9.05	10.59	**7.73**
SSIM	0.9326	0.9380	**0.9449**	0.9342	0.9377	0.9419	0.9333	0.9286	0.9448
60									
NCD	0.0471	0.0469	0.0457	0.0396	0.0469	**0.0330**	0.0528	0.0515	0.0466
PSNR	22.23	22.46	22.81	22.34	22.41	22.74	21.11	21.60	**22.92**
MAE	11.53	10.33	9.30	12.09	10.28	9.87	10.97	12.60	**9.19**
SSIM	0.9283	0.9327	0.9388	0.9295	0.9310	0.9366	0.9156	0.9170	**0.9389**
70									
NCD	0.0482	0.0483	0.0479	0.0478	0.0487	**0.0345**	0.0605	0.0553	0.0491
PSNR	21.85	22.01	22.19	21.94	21.90	22.37	19.69	20.88	**22.49**
MAE	12.06	11.15	**10.22**	12.85	11.19	10.66	13.58	13.85	10.24
SSIM	0.9221	0.9256	0.9301	0.9229	0.9212	0.9293	0.8882	0.9015	**0.9302**

Table 6: Comparison of different filters in removal of fixed-valued impulse noise from couple

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0226	0.0048	0.0156	0.0213	0.0130	**0.0047**	0.0139	0.0131	0.0128
PSNR	39.99	42.28	42.34	40.66	41.61	**43.74**	40.47	42.05	43.03
MAE	1.08	0.29	0.64	1.25	0.31	**0.25**	0.33	0.31	0.27
SSIM	0.9969	0.9982	0.9982	0.9974	0.9979	**0.9987**	0.9973	0.9981	0.9986
20									
NCD	0.0237	**0.0103**	0.0139	0.0246	0.0115	0.0103	0.0206	0.0125	0.0113
PSNR	38.97	39.42	40.35	39.40	38.55	**40.56**	36.53	38.34	39.78
MAE	1.25	0.57	0.77	1.44	0.62	**0.52**	0.72	0.68	0.55
SSIM	0.9961	0.9965	0.9972	0.9965	0.9958	**0.9973**	0.9933	0.9956	0.9968
30									
NCD	0.0253	**0.0150**	0.0168	0.0266	0.0290	0.0175	0.0288	0.0193	0.0169
PSNR	37.83	37.59	38.14	38.06	36.43	38.46	33.89	35.78	**38.56**
MAE	1.46	0.88	0.98	1.66	0.96	**0.81**	1.20	1.12	0.88
SSIM	0.9950	0.9947	0.9953	0.9952	0.9929	0.9957	**0.9987**	0.9920	0.9947
40									
NCD	0.0276	**0.0206**	0.0208	0.0281	0.0345	0.0227	0.0455	0.0277	0.0237
PSNR	36.16	35.69	35.80	36.37	34.60	36.36	31.36	33.49	**36.62**
MAE	1.74	1.24	1.28	1.95	1.35	**1.16**	1.84	1.69	1.26

Table 6: Continue

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
SSIM	0.9926	0.9919	0.9920	0.9929	0.9881	**0.9930**	0.9782	0.9866	0.9918
50									
NCD	0.0302	0.0279	**0.0229**	0.0309	0.0406	0.0279	0.0588	0.0471	0.0416
PSNR	34.71	34.54	33.62	34.67	32.81	**35.30**	28.99	31.49	35.41
MAE	2.25	1.65	1.68	2.30	1.81	**1.53**	2.69	2.39	1.70
SSIM	0.9893	0.9881	0.9869	**0.9895**	0.9810	0.9895	0.9627	0.9788	0.9873
60									
NCD	0.0361	0.0329	**0.0286**	0.0350	0.0479	0.0354	0.0765	0.0576	0.0493
PSNR	33.37	33.09	35.19	32.58	30.58	32.37	26.60	29.60	**37.69**
MAE	2.61	2.13	**1.97**	2.79	2.39	2.07	3.87	3.29	2.23
SSIM	0.9827	0.9811	0.9779	0.9831	0.9670	**0.9827**	0.9364	0.9673	0.9799
70									
NCD	0.0392	0.0408	0.0298	0.0457	0.0568	0.0513	**0.0100**	0.0710	0.0589
PSNR	**31.98**	31.61	33.62	30.77	28.64	30.36	24.31	27.79	30.52
MAE	3.03	2.72	**2.36**	3.43	3.10	2.71	5.52	4.46	2.91
SSIM	0.9736	0.9715	0.9658	**0.9743**	0.9480	0.9727	0.8946	0.9507	0.9691

Table 7: Comparison of different filters in removal of fixed-valued impulse noise from house

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0167	0.0116	0.0121	0.0164	0.0114	0.0099	**0.0093**	0.0253	0.0237
PSNR	26.20	27.03	26.98	26.33	27.18	27.59	28.25	27.09	**28.25**
MAE	5.02	2.52	3.57	5.16	2.30	1.85	**1.23**	2.22	1.25
SSIM	0.9735	0.9783	0.9779	0.9743	0.9791	0.9810	**0.9838**	0.9786	0.9825
20									
NCD	0.0166	0.0132	0.0258	0.0174	0.0129	**0.0120**	0.0243	0.0264	0.0244
PSNR	25.86	26.59	26.75	25.97	26.64	27.10	27.40	26.24	**27.57**
MAE	5.37	3.09	3.66	5.55	2.97	2.44	1.98	3.17	**1.95**
SSIM	0.9713	0.9760	0.9768	0.9720	0.9763	0.9787	0.9803	0.9740	0.9797
30									
NCD	0.0174	**0.0151**	0.0261	0.0184	0.0265	0.0259	0.0252	0.0279	0.0247
PSNR	25.45	26.08	26.36	25.55	26.06	26.59	26.45	25.28	**27.28**
MAE	5.79	3.75	3.94	6.03	3.70	3.09	2.80	4.30	**2.11**
SSIM	0.9686	0.9730	0.9747	0.9691	0.9730	0.9760	0.9757	0.9675	**0.9764**
40									
NCD	0.0198	**0.0164**	0.0267	0.0259	0.0275	0.0268	0.0265	0.0297	0.0263
PSNR	25.00	25.53	25.82	25.06	25.42	25.98	25.18	24.34	**26.06**
MAE	6.28	4.46	4.38	6.58	4.50	3.82	4.00	5.59	**3.66**
SSIM	0.9651	0.9693	0.9715	0.9654	0.9686	0.9724	0.9678	0.9595	**0.9741**
50									
NCD	0.0196	**0.0180**	0.0276	0.0271	0.0286	0.0278	0.0283	0.0320	0.0273
PSNR	24.40	24.85	25.10	24.43	24.67	25.28	23.69	23.26	**25.36**
MAE	6.91	5.32	5.06	7.31	5.42	4.66	5.53	7.20	**4.54**
SSIM	0.9599	0.9641	0.9664	0.9600	0.9627	0.9676	0.9555	0.9479	0.9680
60									
NCD	0.0207	**0.0198**	0.0287	0.0281	0.0301	0.0290	0.0310	0.0350	0.0286
PSNR	23.72	24.08	24.21	23.71	23.80	24.47	22.01	22.15	**24.52**
MAE	7.64	6.29	5.96	8.20	6.46	5.65	7.60	9.16	**5.51**
SSIM	0.9531	0.9572	0.9591	0.9527	0.9545	**0.9609**	0.9365	0.9322	0.9604
70									
NCD	0.0323	0.0315	0.0304	0.0318	0.0320	0.0308	0.0354	0.0388	**0.0303**
PSNR	22.92	23.18	23.15	22.84	22.93	**23.66**	20.09	20.94	23.50
MAE	8.60	7.52	7.18	9.38	7.75	**6.95**	10.74	11.63	6.73
SSIM	0.9437	0.9473	0.9484	0.9422	0.9415	**0.9511**	0.9058	0.9099	0.9486

Table 8: Comparison of different filters in removal of fixed-valued impulse noise from mandrill

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0574	0.0473	0.0369	0.0572	0.0413	0.0425	0.0154	0.0336	**0.0132**
PSNR	22.33	22.75	23.06	22.29	22.96	23.24	24.18	23.47	**25.93**
MAE	11.56	8.89	9.27	11.86	8.01	7.33	4.83	6.50	**3.53**
SSIM	0.9345	0.9412	0.9449	0.9336	0.9445	0.9479	0.9591	0.9511	**0.9600**
20									
NCD	0.0561	0.0481	0.0387	0.0584	0.0428	0.0422	**0.0197**	0.0348	0.0369
PSNR	22.14	22.51	22.98	22.04	22.63	22.95	23.53	22.80	**24.89**
MAE	11.97	9.54	9.24	12.37	8.94	8.11	6.04	8.01	**4.94**
SSIM	0.9314	0.9378	0.9441	0.9297	0.9400	0.9442	0.9525	0.9426	**0.9545**
30									

Table 8: Continue

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
NCD	0.0566	0.0437	**0.0299**	0.0595	0.0462	0.0371	0.0422	0.0384	0.0392
PSNR	21.88	22.22	22.78	21.76	22.28	22.61	22.76	22.18	**24.04**
MAE	12.48	10.28	9.47	12.95	9.90	8.97	7.43	9.54	**6.32**
SSIM	0.9273	0.9335	0.9417	0.9249	0.9347	0.9397	0.9436	0.9334	**0.9476**
40									
NCD	0.0540	0.0480	0.0451	0.0575	0.0476	0.0305	0.0451	0.0418	**0.0281**
PSNR	21.62	21.92	22.49	21.46	21.92	22.28	21.94	21.57	**23.28**
MAE	13.03	11.08	9.93	13.61	10.88	9.87	8.98	11.20	**7.75**
SSIM	0.9227	0.9287	0.9379	0.9193	0.9290	0.9348	0.9323	0.9228	**0.9380**
50									
NCD	0.0412	0.0358	0.0464	0.0469	0.0493	0.0328	0.0489	0.0519	**0.0289**
PSNR	21.29	21.56	22.08	21.10	21.51	21.86	20.97	20.93	**22.49**
MAE	13.72	12.00	10.65	14.38	11.96	10.94	10.85	13.05	**9.30**
SSIM	0.9167	0.9226	0.9320	0.9124	0.9219	0.9282	0.9168	0.9097	**0.9320**
60									
NCD	0.0418	0.0385	0.0481	0.0489	0.0512	**0.0359**	0.0538	0.0555	0.0478
PSNR	20.89	21.11	21.54	20.70	21.00	21.39	19.88	20.27	**21.68**
MAE	14.55	13.08	11.66	15.29	13.16	12.12	13.13	15.08	**10.95**
SSIM	0.9086	0.9142	0.9237	0.9040	0.9121	0.9199	0.8956	0.8940	**0.9215**
70									
NCD	0.0543	0.0532	0.0506	0.0555	0.0538	**0.0484**	0.0613	0.0597	0.0507
PSNR	20.37	20.57	**20.87**	20.21	20.36	20.79	18.52	19.49	20.82
MAE	15.65	14.43	13.02	16.45	14.61	13.86	16.24	17.52	**12.85**
SSIM	0.8973	0.9027	0.9116	0.8924	0.8984	0.9081	0.8623	0.8714	**0.9089**

Table 9: Comparison of different filters in removal of fixed-valued impulse noise from Miramar

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0276	0.0261	0.0280	0.0280	0.0268	**0.0108**	0.0228	0.0409	0.0231
PSNR	25.73	27.00	26.85	25.76	27.73	27.99	**30.36**	27.90	30.26
MAE	8.40	4.80	6.32	8.55	3.79	3.54	**1.93**	3.51	2.12
SSIM	0.9538	0.9665	0.9647	0.9541	0.9719	0.9734	**0.9848**	0.9729	0.9799
20									
NCD	0.0287	0.0273	0.0297	0.0290	0.0284	**0.0133**	0.0397	0.0424	0.0395
PSNR	25.41	26.52	26.76	25.42	26.98	27.36	28.83	26.72	**29.13**
MAE	8.78	5.52	6.22	8.95	4.77	4.33	**2.95**	4.79	3.03
SSIM	0.9503	0.9625	0.9641	0.9503	0.9665	0.9693	**0.9784**	0.9643	0.9746
30									
NCD	0.0299	0.0314	0.0302	0.0306	0.0434	**0.0211**	0.0412	0.0441	0.0406
PSNR	25.06	25.99	26.49	25.02	26.24	26.74	27.30	25.61	**27.93**
MAE	9.20	6.30	6.38	9.42	5.79	5.10	**4.10**	6.20	4.11
SSIM	0.9461	0.9574	0.9619	0.9454	0.9601	0.9644	**0.9694**	0.9535	0.9678
40									
NCD	0.0313	0.0312	0.0430	0.0315	0.0446	**0.0288**	0.0433	0.0463	0.0420
PSNR	24.65	25.41	26.03	24.55	25.49	26.04	25.71	24.57	**26.79**
MAE	9.70	7.15	6.75	9.98	6.85	6.12	5.44	7.76	**5.27**
SSIM	0.9407	0.9513	0.9578	0.9390	0.9524	**0.9581**	0.9563	0.9406	0.9566
50									
NCD	0.0355	0.0333	0.0442	0.0350	0.0460	**0.0304**	0.0461	0.0489	0.0435
PSNR	24.17	24.77	25.36	24.01	24.70	25.32	24.05	23.48	**25.68**
MAE	10.28	8.11	7.38	10.68	7.96	7.14	7.10	9.58	**6.54**
SSIM	0.9337	0.9433	0.9510	0.9307	0.9429	**0.9503**	0.9371	0.9227	0.9388
60									
NCD	0.0482	0.0473	0.0457	**0.0424**	0.0476	0.0462	0.0505	0.0523	0.0455
PSNR	23.62	24.10	24.50	23.40	23.88	24.53	22.19	22.39	**24.64**

Table 9: Conitnue

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
MAE	10.96	9.15	8.29	11.50	9.18	8.29	9.31	11.70	**7.93**
SSIM	0.9248	0.9337	**0.9408**	0.9203	0.9306	0.9403	0.9064	0.8992	0.9065
70									
NCD	0.0501	0.0492	0.0480	0.0510	0.0496	0.0483	0.0574	0.0566	**0.0474**
PSNR	22.95	23.32	23.49	22.71	23.13	23.77	20.22	21.26	**23.78**
MAE	11.85	10.40	9.49	12.57	10.52	9.76	12.34	14.25	**9.22**
SSIM	0.9123	0.9206	**0.9259**	0.9061	0.9136	0.9271	0.8610	0.8665	0.8548

Table 10: Comparison of different filters in removal of fixed-valued impulse noise from Tiffany

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0057	0.0026	0.0045	0.0057	0.0024	0.0023	0.0019	0.0023	**0.0019**
PSNR	27.64	28.47	28.14	27.74	28.55	28.71	29.10	28.65	**29.17**
MAE	3.67	1.35	8.79	3.65	1.25	1.16	0.90	1.20	**0.90**
SSIM	0.9673	0.9732	0.9709	0.9679	0.9736	0.9746	**0.9768**	0.9742	0.9763
20									
NCD	0.0059	0.0031	0.0044	0.0059	0.0029	0.0028	0.0026	0.0031	**0.0025**
PSNR	27.56	28.25	28.09	27.64	28.30	28.48	28.56	28.23	**28.74**
MAE	3.79	1.70	2.73	3.77	1.63	1.51	1.35	1.68	**1.33**
SSIM	0.9667	0.9718	0.9706	0.9672	0.9722	0.9733	0.9739	0.9716	**0.9744**
30									
NCD	0.0062	0.0036	0.0046	0.0062	0.0036	0.0034	0.0034	0.0039	**0.0032**
PSNR	27.37	27.99	27.89	27.51	27.91	28.20	27.78	27.70	**28.30**
MAE	3.96	2.07	2.81	3.92	2.04	1.89	1.90	2.24	**1.78**
SSIM	0.9652	0.9701	0.9693	0.9662	0.9696	0.9715	0.9689	0.9679	**0.9715**
40									
NCD	0.0065	0.0042	0.0049	0.0065	0.0043	0.0040	0.0044	0.0048	**0.0039**
PSNR	27.14	27.66	27.53	27.33	27.44	27.81	26.74	27.19	**27.92**
MAE	4.15	2.47	3.00	4.11	2.50	2.28	2.60	2.83	**2.24**
SSIM	0.9634	0.9678	0.9668	0.9648	0.9662	0.9694	0.9608	0.9638	**0.9697**
50									
NCD	0.0069	0.0049	0.0054	0.0069	0.0051	0.0047	0.0058	0.0059	**0.0046**
PSNR	26.87	27.28	27.05	27.08	27.02	27.47	25.44	26.60	**27.53**
MAE	4.37	2.91	3.31	4.35	2.98	2.73	3.52	3.53	**2.71**
SSIM	0.9610	0.9649	0.9630	0.9626	0.9613	0.9664	0.9480	0.9586	**0.9669**
60									
NCD	0.0074	0.0057	0.0061	0.0075	0.0060	0.0054	0.0078	0.0073	**0.0054**
PSNR	26.51	26.81	26.35	26.72	26.40	27.18	23.88	25.82	**26.98**
MAE	4.64	3.39	3.75	4.66	3.53	3.23	4.82	4.39	**3.22**
SSIM	0.9578	0.9611	0.9569	0.9595	0.9541	0.9622	0.9270	0.9504	**0.9626**
70									
NCD	0.0080	0.0065	0.0071	0.0083	0.0072	0.0063	0.0108	0.0091	**0.0063**
PSNR	26.24	26.52	25.39	26.10	25.37	**26.67**	22.05	24.88	26.52
MAE	5.07	3.93	4.36	5.14	4.21	**3.77**	6.74	5.49	3.78
SSIM	0.9513	0.9540	0.9466	0.9534	0.9392	0.9548	0.8923	0.9383	**0.9586**

Table 11: Comparison of different filters in removal of fixed-valued impulse noise from Tree

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	**0.0079**	0.0118	0.0235	0.0085	0.0118	0.0115	0.0228	0.0230	0.0227
PSNR	34.38	37.70	36.60	34.83	37.50	**40.09**	38.43	37.73	40.04
MAE	2.27	0.63	1.37	2.52	0.65	0.43	0.47	0.60	**0.42**
SSIM	0.9973	0.9987	0.9984	0.9976	0.9987	**0.9993**	0.9989	0.9987	0.9992
20									
NCD	0.0106	0.0131	0.0237	**0.0095**	0.0132	0.0126	0.0233	0.0239	0.0231
PSNR	32.13	35.51	35.51	33.61	35.01	**37.60**	34.80	34.33	37.17
MAE	3.17	1.08	1.56	2.88	1.15	0.81	0.98	1.23	**0.81**
SSIM	0.9965	0.9979	0.9979	0.9968	0.9977	**0.9987**	0.9976	0.9973	0.9986
30									
NCD	**0.0100**	0.0141	0.0242	0.0108	0.0242	0.0237	0.0244	0.0253	0.0236
PSNR	32.14	33.65	33.93	32.23	32.97	35.51	31.72	31.30	**34.90**
MAE	3.05	1.61	1.89	3.33	1.73	1.25	1.68	2.13	**1.30**
SSIM	0.9955	0.9968	0.9970	0.9956	0.9963	**0.9979**	0.9951	0.9945	0.9976
40									
NCD	0.0170	0.0250	0.0249	**0.0151**	0.0253	0.0244	0.0260	0.0273	0.0243
PSNR	30.75	31.72	32.02	30.63	30.96	**33.36**	28.99	28.77	32.84
MAE	3.56	2.26	2.36	3.90	2.43	**1.81**	2.62	3.29	1.87
SSIM	0.9938	0.9950	0.9954	0.9936	0.9941	**0.9966**	0.9909	0.9902	0.9961
50									
NCD	0.0186	0.0263	0.0261	**0.0172**	0.0266	0.0255	0.0286	0.0303	0.0253
PSNR	29.10	29.81	29.97	28.83	29.20	**31.28**	26.38	26.47	30.90
MAE	4.22	3.04	3.02	4.64	3.28	**2.52**	3.98	4.82	2.57
SSIM	0.9909	0.9923	0.9926	0.9903	0.9906	**0.9942**	0.9833	0.9926	0.9936
60									
NCD	0.0291	0.0279	0.0277	**0.0215**	0.0284	0.0270	0.0330	0.0340	0.0266
PSNR	27.52	27.95	28.01	27.11	27.37	**28.99**	23.77	24.52	28.91
MAE	5.02	3.99	3.89	5.60	4.29	**3.37**	6.00	6.71	3.44

Table 11: Continue

Noise (%)/parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
SSIM	0.9869	0.9882	0.9884	0.9856	0.9852	**0.9907**	0.9709	0.9738	0.9894
70									
NCD	0.0314	0.0302	0.0302	0.0317	0.0309	0.0291	0.0406	0.0399	**0.0287**
PSNR	25.85	26.17	26.02	25.40	25.61	27.44	21.26	22.50	**26.69**
MAE	6.04	5.49	5.07	6.85	5.55	**4.50**	9.03	9.41	4.59
SSIM	0.9807	0.9822	0.9818	0.9786	0.9773	**0.9848**	0.9501	0.9581	0.9812

Table 12: Comparison of filters based on average NCD. PSNR, MAE and SSIM for 10 images corrupted with fixed-valued impulse noise

Noise (%)/Parameter	VMF	AVMF	EVMF	FVMF	PGF	ACWVMF	RWASF	QSF	PF
10									
NCD	0.0234	0.0172	0.0198	0.0229	0.0174	**0.0122**	0.0138	0.0213	0.0239
PSNR	28.45	29.87	29.67	28.64	30.02	30.77	31.46	30.39	**32.36**
MAE	5.55	3.25	4.78	5.68	2.88	2.62	1.67	2.55	**1.48**
SSIM	0.9679	0.9739	0.9745	0.9684	0.9756	0.9774	**0.9828**	0.9769	0.9821
20									
NCD	0.0237	0.0197	0.0232	0.0241	0.0187	**0.0166**	0.0236	0.0232	0.0216
PSNR	27.91	29.01	29.25	28.16	28.99	29.79	29.66	28.91	**30.79**
MAE	5.88	3.78	4.21	6.02	3.54	3.18	2.49	3.48	**2.13**
SSIM	0.9660	0.9715	0.9737	0.9662	0.9725	0.9750	0.9781	0.9720	**0.9788**
30									
NCD	0.0245	**0.0212**	0.0234	0.0254	0.0296	0.0213	0.0282	0.0262	0.0250
PSNR	27.54	28.26	28.60	27.61	28.08	28.94	28.05	27.59	**29.67**
MAE	6.14	4.29	4.38	6.35	4.15	3.69	3.31	4.37	**2.92**
SSIM	0.9635	0.9685	0.9718	0.9633	0.9688	0.9721	0.9727	0.9657	**0.9746**
40									
NCD	0.0265	**0.0242**	0.0281	0.0281	0.0314	0.0249	0.0322	0.0316	0.0270
PSNR	26.92	27.44	27.74	26.95	27.16	28.02	26.39	26.36	**28.49**
MAE	6.55	4.90	4.74	6.79	4.87	4.33	4.40	5.47	**3.77**
SSIM	0.9595	0.9647	0.9685	0.9595	0.9640	0.9683	0.9619	0.9579	**0.9687**
50									
NCD	**0.0266**	0.0264	0.0295	0.0291	0.0336	0.0270	0.0366	0.0369	0.0291
PSNR	26.23	26.66	26.77	26.20	26.28	27.20	24.66	25.16	**27.45**
MAE	7.04	5.59	5.28	7.34	5.65	5.05	5.79	6.78	**4.68**
SSIM	0.9554	0.9597	0.9633	0.9544	0.9573	**0.9634**	0.9475	0.9482	0.9620
60									
NCD	0.0336	**0.0312**	0.0328	0.0324	0.0361	0.0308	0.0428	0.0413	0.0346
PSNR	25.47	25.75	26.02	25.32	25.23	26.09	22.83	23.94	**26.73**
MAE	7.62	6.41	6.02	8.03	6.56	5.90	7.67	8.41	**5.66**
SSIM	0.9487	0.9526	0.9554	0.9473	0.9476	**0.9564**	0.9252	0.9329	0.9534
70									
NCD	0.0377	0.0370	**0.0352**	0.0387	0.0392	0.0355	0.0427	0.0469	0.0380
PSNR	24.63	24.82	24.85	24.38	24.14	**25.10**	20.88	22.69	25.04
MAE	8.37	7.42	7.02	8.95	7.63	6.97	10.34	10.34	**6.72**
SSIM	0.9392	0.9427	0.9439	0.9373	0.9335	**0.9462**	0.8908	0.9124	0.9393

the PF has almost superior values as compared with other filters. ACWVMF occupies the second rank in terms of performance.

In order to show the improved performance due to quaternion distance, we have implemented the same proposed filter implemented using the Euclidean distance. Table 13 and 14 depict the comparison between the Proposed Filter implemented in Quaternion, (PF)(Q) and the Proposed Filter implemented in Euclidean distance, PF(E), of Lena image for fixed-valued impulse noise at rank = 5 and Tol = 35. It can be seen from Table 13 that at 10% impulse noise, the PF(Q) has NCD of 0.0039 and PSNR of 38.82 dB while it is 0.0106 and 36.16 dB for PF(E). Also, at 70% impulse noise, the PF(Q) has NCD of 0.0210 and PSNR of 28.17 dB whereas the PF(E) has NCD of 0.0216 and PSNR of 26.81dB. Similarly, from Table 14, it is found that the PF(Q) has MAE of 0.54 and SSIM of 0.9988 at 10% impulse noise while the PF(E) has MAE of 0.95 and SSIM of 0.9936. At 70% impulse noise, the PF(Q) has MAE of 4.17 and SSIM of 0.9859 whereas for PF(E), the values are 4.84 and 0.9594. This shows that the PF(Q) has better results than the PF(E).

Table 13: Comparison of NCD and PSNR between proposed filter (quaternion) and proposed filter (Euclidean)

Noise (%)	PF (E)		PF (Q)	
	NCD	PSNR	NCD	PSNR
10	0.0106	36.16	**0.0039**	**38.82**
20	**0.0043**	34.69	0.0050	**36.03**
30	0.0101	33.28	**0.0065**	**34.09**
40	0.0176	31.83	**0.0172**	**32.44**
50	0.0185	30.32	**0.0181**	**31.06**
60	0.0198	28.58	**0.0193**	**29.61**
70	0.0216	26.81	**0.0210**	**28.17**

Table 14: Comparison of MAE and SSIM between proposed filter (quaternion) and proposed filter (Euclidean)

Noise (%)	PF (E)		PF (Q)	
	MAE	SSIM	MAE	SSIM
10	0.95	0.9936	**0.54**	**0.9988**
20	1.43	0.9912	**1.06**	**0.9975**
30	1.95	**0.9882**	**1.61**	0.9963
40	2.52	0.9841	**2.23**	**0.9948**
50	3.16	0.9789	**2.90**	**0.9930**
60	3.93	0.9708	**3.67**	**0.9896**
70	4.84	0.9594	**4.17**	**0.9859**

CONCLUSION

In this study, we have proposed a two-stage filter for removing impulse noise from color images. Quaternion theory is used for calculating the intensity and chromaticity component of a color pixel. This helps in processing color image as single unit instead of separate components. In the first stage of noise detection using the quaternion distance, the color pixels are sorted and assigned a rank based on the aggregated sum of color pixel differences with other pixels inside the filtering window. The central pixel is declared as probably corrupted by an impulse if its rank is larger than a predefined rank. In the second stage, the probably noisy pixel is again checked for an impulse or an edge by using four Laplacian convolution kernels. The minimum of these four convolutions is calculated. If it is bigger than a predefined threshold, then the central pixel is considered as an impulse. The noisy pixel is filtered by output of weighted vector median filter computed using the quaternion distance. Pixels belonging to the direction of minimum difference has been assigned more weights. Experimental results indicate the improved performance of the proposed filter in terms of NCD, PSNR, MAE and SSIM in suppressing the impulse noise while retaining the original image details comparing against other well-known filters.

REFERENCES

Astola, J., P. Haavisto and Y. Neuovo, 1990. Vector median filters. IEEE Proc., 78: 678-689.

Cai, C. and S.K. Mitra, 2000. A normalized color difference edge detector based on quaternion representation. Proceedings of the 2000 International Conference on Image Processing (Cat. No. 00CH37101) Vol. 2, September 10-13, 2000, IEEE, Vancouver, Canada, pp: 816-819.

Celebi, M.E., H.A. Kingravi and Y.A. Aslandogan, 2007. Nonlinear vector filtering for impulsive noise removal from color images. J. Electron. Imaging, 16: 033008-1-033008-21.

Evans, C.J., S.J. Sangwine and T.A. Ell, 2000. Hypercomplex color-sensitive smoothing filters. Proceedings of the 2000 International Conference on Image Processing (Cat. No. 00CH37101) Vol. 1, September 10-13, 2000, IEEE, Vancouver, Canada, pp: 541-544.

Geng, X., X. Hu and J. Xiao, 2012. Quaternion switching filter for impulse noise reduction in color image. Signal Process., 92: 150-162.

Jin, L. and D. Li, 2007. An efficient color-impulse detector and its application to color images. IEEE. Signal Process. Lett., 14: 397-400.

Jin, L., H. Liu, X. Xu and E. Song, 2010. Quaternion-based color image filtering for impulsive noise suppression. J. Electron. Imaging, 19: 1-12.

Karakos, D.G. and P.E. Trahanias, 1995. Combining vector median and vector directional filters: The directional-distance filters. Proceedings of the IEEE Conferenec Image Processing, October 23-26, 1995, IEEE Computer Society Washington, DC, USA., pp: 171-174.

Lazhar, K., A.C. Faouzi and G. Moncef, 1999. High-resolution digital resampling using vector rational filters. Opt. Eng., 38: 893-901.

Lukac, R. and B. Smolka, 2003. Application of the adaptive center-weighted vector median framework for the enhancement of cDNA microarray images. Intl. J. Appl. Math. Comput. Sci., 13: 369-383.

Lukac, R., 2003. Adaptive vector median filtering. Pattern Recognit. Lett., 24: 1889-1899.

Lukac, R., 2004. Adaptive color image filtering based on center-weighted vector directional filters. Multidimension. Syst. Signal Process., 15: 169-196.

Lukac, R., B. Smolka, K.N. Plataniotis and A.N. Venetsanopoulos, 2003. Entropy vector median filter. Proceedings of the 1st Internatinal Iberian Conference on Pattern Recognition and Image Analysis, June 4-6, 2003, Springer, Berlin, Germany, ISBN:978-3-540-40217-6, pp: 1117-1125.

Lukac, R., B. Smolka, K.N. Plataniotis and A.N. Venetsanopoulos, 2006. Vector sigma filters for noise detection and removal in color images. J. Visual Commun. Image Represent., 17: 1-26.

Plataniotis, K.N., D. Androutsos and A.N. Venetsanopoulos, 1996. Fuzzy adaptive filters for multichannel image processing. Signal Process., 55: 93-106.

Sangwine, S.J. and T.A. Ell, 2000. Colour image filters based on hypercomplex convolution. IEE. Proc. Vision Image Signal Process., 147: 89-93.

Shi, L. and B. Funt, 2007. Quaternion color texture segmentation. Comput. Vision Image Understanding, 107: 88-96.

Smolka, B., 2010. Peer group switching filter for impulse noise reduction incolor images. Pattern Recognition Lett., 31: 484-495.

Smolka, B., K. Malik and D. Malik, 2015. Adaptive rank weighted switching filter for impulsive noise removal in color images. J. Real Time Image Process., 10: 289-311.

Subakan, O.N. and B.C. Vemuri, 2011. A quaternion framework for color image smoothing and segmentation. Intl. J. Comput. Vision, 91: 233-250.

Trahanias, P.E. and A.N. Venetsanopoulos, 1993. Vector directional filters: A new class of multichannel image processing filters. IEEE Trans. Image Proc., 2: 528-534.

Wang, G., Y. Liu and T. Zhao, 2014. A quaternion-based switching filter for colour image denoising. Signal Process., 102: 216-225.

Wang, Z., A.C. Bovik, H.R. Sheikh and E.P. Simoncelli, 2004. Image quality assessment: From error visibility to structural similarity. IEEE. Trans. Image Proc., 13: 600-612.