Wavelet Domain Order Statistics Filters for Image Denoising

Jesús Martínez-Valdes, Francisco Javier Gallegos Funes, Volodymyr Ponomaryov

Section of Senior Studies and Research of Superior School of Mechanical and Electrical Engineering Higher School of the National Polytechnic Institute, Mexico D.F., Mexico.
jesusipnmex@yahoo.com.mx, fgallegosf@ipn.mx

Abstract. This paper shows the performance results to apply the analysis wavelet with the order statistics in the task of the suppression of impulsive noise in color images. The proposed filtering scheme in defined as two filters in the wavelet domain to conform the structure of a general filter that can be modified in some headings, the first filter is based on redundancy of approaches, the second one is the wavelet domain iterative center weighted median algorithm. With the structure of the proposed filter different implementations for the estimation of the noisy sample are already carried out using different order statistics algorithms that by their good performance can be beneficial in image processing applications.

1 Introduction

Many different classes of filters have been proposed for removing noise from images [1, 2]. They are classified into several categories depending on specific applications. The order statistics filters are designed to suppress noise of different nature, they can remove impulsive noise and guarantee detail preservation [1, 2]. By other hand, the filters based in the wavelet domain provide a better performance in terms of noise suppression in comparison with different filters in the spice domain [3, 4].

In this paper we proposed a filter that works in the wavelet domain and to use different order statistics algorithms in its filtering scheme.

The proposed filter is conformed by two filters that carry out the impulsive noise suppression in the wavelet domain. The first filter based on redundancy of approaches [5] smoothes the LF of the noisy image by means of double convolution operation (first of decomposition and after reconstruction) between the wavelet coefficients and the samples of the corrupted image, the second one is based wavelet domain iterative center weighted median filter [6] that provides an analysis of the histograms of the wavelet coefficients of several pairs of images (original and corrupted) through different scales and carries out an improved estimation of the variance field of the noisy wavelet coefficients of the image and with aid of the estimator of the minimum mean square error finally obtains the filtered wavelet coefficients of the approaches and details of the image.

The structure of the proposed filter is designed so that it can be modified in the sections of the first and second detection of the noisy sample in addition to the estimation algorithm, all contained in the wavelet domain iterative center weighted median filter [6] to increase its robustness and to improve its performance in the task of the impulsive noise suppression. We introduce in the proposed filter the algorithms of the SD-ROM, Tri-state Median, Adaptive Center Weighed Median, Multi-stage Median, FIR Median Hybrid, and MM-KNN Filters [7-11], and adapt them to work in the wavelet domain into the two detection blocks proposed in this paper.

2 Wavelet Domain Order Statistics Filtering Scheme

The structure of the proposed filter is constituted by two filters, the filter based on redundancy of approaches [5] and the wavelet domain iterative center weighted median (ICWMF) filter [6] as it is shown in the Fig. 1. For each color component of the noisy image is necessary to apply all the steps contained by this structure. This technique apply up to 5 scaling levels for the details and only 1 scaling level for the approaches, it obeys to the differences that can be found between them and their importance at the moment of the image reconstruction. Other operations are indicated to make clearer the wavelet analysis that it is carried out in this paper. We modify this structure in the block of the ICWMF. For that reason, the expressions used by the ICWMF to calculate the improved estimation of the variance field of the noisy wavelet coefficients will be required to indicate when and where different proposed filtering algorithms will take place into it to improve the performance of the proposed filter.

The first stage of the ICWMF [6] that detects if a sample contains noise or not is:

$$\hat{\sigma}_{gs}^2(k)=\begin{cases}\tilde{\sigma}_{gs}^2(k) & \text{if } \lambda_s \geq \lambda_{th} \\ med_{cw}(\tilde{\sigma}_{gs}^2(j)) & \text{if } \lambda_s < \lambda_{th}\end{cases} \tag{1}$$

where $\tilde{\sigma}_{gs}^2$ is the variance field estimated previously, k is central sample in the filter window, j is one of the N sample contained into the window, λ_s is the standard deviation of the preliminary estimate of the signal coefficients variances $\tilde{\sigma}_s^2(k)$ in each scale and λ_{th} is the discriminating threshold defined as [6]:

$$\lambda_{th}=\frac{\sum_s \lambda_s 2^{-s}}{\sum_s 2^{-s}} \tag{2}$$

where s is the scale used in the wavelet analysis and 2^{-s} is the weighting function.

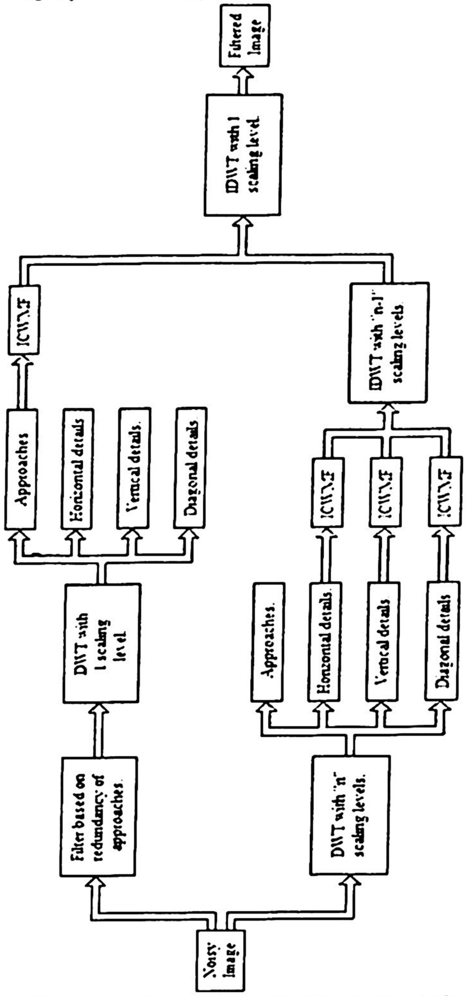

Fig. 1. Block diagram of the proposed filtering scheme of the Wavelet Domain Order Statistics Filter.

2.1 Wavelet Domain Signal Dependent Rank-Ordered Mean Filter

Consider a 3x3 filter window size with a central sample $\tilde{\sigma}^2_{g1}(m)=\tilde{\sigma}^2_{g1}(k)$ where $m=[m_1,m_2]$ is the location of the sample in the image, and a vector $\varphi(m)$ which contains the neighbor pixels of $\tilde{\sigma}^2_{g1}(m)$ into the window [1],

$$\varphi(m) = [\varphi_1(m), \varphi_2(m), \varphi_3(m), \varphi_4(m), \varphi_5(m), \varphi_6(m), \varphi_7(m), \varphi_8(m)]$$
$$\begin{aligned}\varphi(m) = [&\tilde{\sigma}^2_{g'}(m_1-1, m_2-1), \tilde{\sigma}^2_{g'}(m_1-1, m_2), \tilde{\sigma}^2_{g'}(m_1-1, m_2+1), \\ &\tilde{\sigma}^2_{g'}(m_1, m_2-1), \qquad \tilde{\sigma}^2_{g'}(m_1, m_2+1), \\ &\tilde{\sigma}^2_{g'}(m_1+1, m_2-1), \tilde{\sigma}^2_{g'}(m_1+1, m_2), \tilde{\sigma}^2_{g'}(m_1+1, m_2+1)]\end{aligned} \tag{3}$$

The samples of the previous vector can be ordered to obtain the next vector [1]:

$$v(m) = [v_1(m), v_2(m), \ldots, v_8(m)] \tag{4}$$

where $v_1(m), v_2(m), \ldots, v_8(m)$ are the rank ordered elements of $\varphi(m)$ that satisfy the condition $v_1(m) \le v_2(m) \le \ldots \le v_8(m)$.

Then, the Rank Ordered Mean is defined as follows [1],

$$rom(m) = \frac{v_4(m) + v_5(m)}{2} \tag{5}$$

and the vector that contains the rank ordered differences $rod(m)$ is [1]:

$$rod(m) = [rod_1(m), rod_2(m), rod_3(m), rod_4(m)] \tag{6}$$

where $rod_i(m) = \begin{cases} v_i(m) - \tilde{\sigma}^2_{g'}(m) & \text{for } \tilde{\sigma}^2_{g'}(n) \le rom(m) \\ \tilde{\sigma}^2_{g'}(m) - v_{9-i}(m) & \text{for } \tilde{\sigma}^2_{g'}(n) > rom(m) \end{cases}$, and $i=1,\ldots,4$.

Finally, the SD-ROM algorithm detects to $\tilde{\sigma}^2_{g'}(m)$ as a noisy sample if any of the following conditions is true [1]:

$$rod_i(m) > U_i \tag{7}$$

where $i=1,\ldots,4$, U_1, U_2, U_3 y U_4 are threshold, $U_1 < U_2 < U_3 < U_4$, and $U_1 = 8$, $U_2 = 20$, $U_3 = 40$, $U_4 = 50$ [1].

2.2 Wavelet Domain Tri-State Median Filter

The Tri-State Median (TSM) Filter joints the Median Filter to the Center Weighted Median (CWM) Filter into a noise detector that decides if a sample is corrupted or not. The TSM Filter algorithm is defined as [7]:

$$\theta_{TSM} = \begin{cases} \tilde{\sigma}^2_{g'} & U \ge d_1 \\ \theta_{CWM} & d_2 \le U < d_1 \\ \theta_{Med} & U < d_2 \end{cases} \tag{8}$$

where θ_{CWM} y θ_{Med} are the values of CWM and median filters respectively, $\tilde{\sigma}_{gs}^2$ is the value of the filtering center window, $d_1 = \left|\tilde{\sigma}_{gs}^2 - \theta_{Med}\right|$, $d_2 = \left|\tilde{\sigma}_{gs}^2 - \theta_{CWM}\right|$ with $d_2 \le d_1$, U is a selected threshold depending on the image and the noise level, a rank between 10 and 30 is defined [4] based on different simulations, in this case takes an intermediate value of $U = 20$.

2.3 Wavelet Domain Adaptive Center Weighed Median Filter.

The Adaptive Center Weighed Median (ACWM) Filter uses an adaptive operator which forms estimations based on the differences between the central sample of the filter window and the values from the CWM Filter with variations of its central weight. The ACWM Filter is defined as [8]:

$$\theta_{ACWM} = \begin{cases} \theta_{CWM}^{1} & d_k > U_k \\ \tilde{\sigma}_{gs}^2 & \text{otherwise} \end{cases} \tag{9}$$

where $d_k = \left|\theta_{CWM}^{m} - \tilde{\sigma}_{gs}^2\right|$ are the differences between the Variable CWM Filter and the central sample in the analysis window, $\theta_{CWM}^{m} = med\left\{\left[\tilde{\sigma}_{gs}^2\right]_j, m\Diamond\tilde{\sigma}_{gs}^2\right\}$ is the Variable CWM Filter being m its weight and $\left[\tilde{\sigma}_{gs}^2\right]_j$ are the samples contained in the filter window, $U_k = s \cdot MAD + \vartheta_k$ are the filter thresholds and $MAD = med\left\{\left|\left[\tilde{\sigma}_{gs}^2\right]_j - \theta_{CWM}^{1}\right|\right\}$ is the median absolute deviations from median, the parameter s gives necessary robustness to the filter and varies between $0 \le s \le 0.6$. The optimal values for these parameters are $s = 0.3$ and $\vartheta_k = [\vartheta_0, \vartheta_1, \vartheta_2, \vartheta_3] = [40, 25, 10, 5]$.

2.4 Wavelet Domain Median M-type K-Nearest Neighbor Filter

The algorithm of Median M-type KNN (MM-KNN) filter can be written as [9, 10]:

$$\hat{e}_{MMKNN}^{(q)}(i,j) = med\left\{g^{(q)}(i+m, j+n)\right\} \tag{10}$$

where $g^{(q)}(i+m, j+n)$ it is a set of K_c samples with weight according to the used function $\psi(\tilde{\sigma}_{gs}^2)$ required to carry out a comparison with the estimation of the previous step $\hat{e}_{MMKNN}^{(q-1)}(i,j)$. The initial estimator is $\hat{e}_{MMKNN}^{(0)}(i,j) = \tilde{\sigma}_{gs}^2(i,j)$, $\tilde{\sigma}_{gs}^2(i,j)$ is the central value of the filtering window, $\hat{e}_{MMKNN}^{(q)}(i,j)$ is the estimation in the iteration q.

The parameter K_c reflects the local data activity and the impulse presence [10],

$$K_c(i,j) = \left[K_{min} + aS(\tilde{\sigma}_{gs}^2(i,j))\right] \le K_{max} \tag{11}$$

where a controls the detail detection, K_{min} is the minimum number of nearest neighbors to remove the noise, and K_{max} is the maximum number of neighbors used for the detection of edges and fine details. The optimal values for these parameters are $a = 2$ and $K_{min}=5$ and $S(\tilde{\sigma}_{g'}^2(i,j))$ is the impulsive detector defined as [9, 10]:

$$S(\tilde{\sigma}_{g'}^2(i,j)) = \frac{med\{|\tilde{\sigma}_{g'}^2(i,j) - \tilde{\sigma}_{g'}^2(i+m,j+n)|\}}{MAD\{\tilde{\sigma}_{g'}^2(i,j)\}} + 0.5\frac{MAD\{\tilde{\sigma}_{g'}^2(i,j)\}}{med\{\tilde{\sigma}_{g'}^2(i+k,j+l)\}} \quad (12)$$

where $MAD_n = med\{|(\tilde{\sigma}_{g'}^2)_i - M_n|\}$ is the median of absolute deviations from median, $M_n = med\{(\tilde{\sigma}_{g'}^2)_i\}$, $(\tilde{\sigma}_{g'}^2)_i$ is the sample of the window that goes from left to right and of above to down, and M_n is the median of samples in the filtering window.

We also use the simple influence function in the MM-KNN filter [9],

$$\psi_{cut(r)}(\tilde{\sigma}_{g'}^2) = \begin{cases} \tilde{\sigma}_{g'}^2, & |\tilde{\sigma}_{g'}^2| \le r \\ 0, & |\tilde{\sigma}_{g'}^2| > r \end{cases} \quad (13)$$

where r is a parameter between 0 to 256.

2.5 Wavelet Domain Multi-Stage Median Filter

The proposed algorithms mentioned before were applied to the proposed filter as a first detection block, but the following two algorithms were applied as a second detection block due that these algorithms only constitute the part of estimation of the noisy sample value (only if the sample was detected of this way) and the proposed filter can continue operating in all its sections in the same way. For this reason it is necessary to present the expression for the second detection block contained in the proposed filter structure [6]:

$$med'_{\alpha}(\tilde{\sigma}_{g'}^2(j)) = \begin{cases} med(\tilde{\sigma}_{g'}^2(j)) & \text{if } \tilde{\sigma}_{g'}^2(k) \le \gamma\sigma_n^2 \\ \tilde{\sigma}_{g'}^2(k) + med(\tilde{\sigma}_{g'}^2(j) - \tilde{\sigma}_{g'}^2(k)) & \text{if } \tilde{\sigma}_{g'}^2(k) > \gamma\sigma_n^2 \end{cases} \quad (14)$$

The proposed filter uses the median algorithm represented as $med(\tilde{\sigma}_{g'}^2(j))$ to estimate the value of the central sample in a filter window if the sample is detected as noisy, and it is possible to use other estimation algorithms such as the Multi-Stage Median and FIR Median Hybrid (Filters that retain more information of the image.

The Multi-Stage Median (MSM) filter is based on the obtaining of the median of medians in such form [12, 11, 12]:

$$FMM = med\{h - med, v - med, d45 - med, d135 - med\} \quad (15)$$

where the parameters of filter are displayed in the Fig. 2.

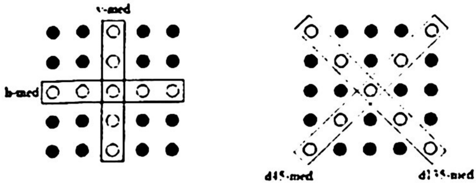

Fig. 2. Median Filters *h-med, v-med, d45-med, d135-med.*

2.6 Wavelet Domain FIR Median Hybrid Filter

There exist different types of these filters which offer the possibility of choosing the number of sub-filters, its type, as well as the weights and the type of window. In this case, a FIR Median Hybrid (FIRMH) filter is defined by means of use the filter windows shown in the Fig. 3 and the following equations [2, 13]:

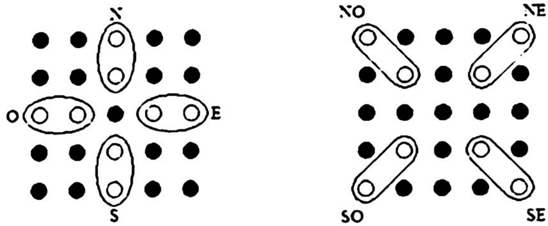

Fig. 3. Filter windows for the FIR Median Hybrid Filter.

$$F_1 = med\left\{Y_N, Y_E, Y_S, Y_O, \tilde{\sigma}^2_{gs_{m+1,m+1}}\right\} \quad (16)$$

$$F_1 = med\left\{\frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{m+1,i}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{m+1,i}}, \tilde{\sigma}^2_{gs_{m+1,m+1}}\right\} \quad (17)$$

$$F_1 = med\left\{\frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{m+1,i}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{m+1,i}}, \tilde{\sigma}^2_{gs_{m+1,m+1}}\right\} \quad (18)$$

$$F_1 = med\left\{\frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{m+1,i}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{gs_{i,m+1}}, \frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{gs_{m+1,i}}, \tilde{\sigma}^2_{gs_{m+1,m+1}}\right\} \quad (19)$$

$$F_2 = med\left\{Y_{NO}, Y_{SO}, Y_{SE}, Y_{NE}, \tilde{\sigma}^2_{gs_{m+1,m+1}}\right\} \quad (20)$$

$$F_2 = med\left\{\frac{1}{m}\sum_{i=1}^{m}\tilde{\sigma}^2_{g'_{i,j}}, \frac{1}{m}\sum_{i=m+2}^{m}\tilde{\sigma}^2_{g'_{z-i+1,j}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{g'_{i,j}}, \frac{1}{m}\sum_{i=m+2}^{z}\tilde{\sigma}^2_{g'_{z-i+1,j}}, \tilde{\sigma}^2_{g'_{m+1,m+1}}\right\} \quad (21)$$

$$F_3 = med\left\{F_1, F_2, \tilde{\sigma}^2_{g'_{m+1,m+1}}\right\} \quad (22)$$

where z is the measurement in pixels of a side of the used square window in the analysis and m it is obtained from $m = \frac{z-1}{2}$, Y are simply average filters, F_1 and F_2 are median filters and F_3 is the result of FIR Median Hybrid Filter.

3 Simulation Results

We obtained from the simulation experiments the properties of the proposed filter by means of use the following criteria: the crossed correlation index (CCI) to evaluate the correlation between the original and filtered images, the peak signal noise relation (PSNR) and the mean square error (MSE) to evaluate the noise suppression, the mean absolute error (MAE) to evaluate the detail preservation, the mean chromaticity error (MCRE) to evaluate the chromaticity retention, and the normalized color difference (NCD) to quantify the perceptual error [1, 2]. In the simulations results that we present here, the image "Lena" was degraded with 20% of impulsive noise, the wavelet used for the analysis was the dB5, and with 3x3 filtering window.

The performance results obtained for the Wavelet Domain Order Statistics Filter (section 2) is shown in the Table 1.

Table 1. Performance results for the proposed filter.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0.925930	29.759030 dB	31.189573 dB	31.602808 dB	[illegible]	[illegible]	[illegible]
2	0.971353	39.467616 dB	41.539996 dB	42.219814 dB	[illegible]	[illegible]	953.887441
3	0.984417	45.181795 dB	48.075334 dB	49.057513 dB	709.348935	531.123033	481.437204
4	0.987709	47.353714 dB	50.630914 dB	51.895372 dB	570.866701	411.340420	362.487470
5	0.988663	47.947656 dB	51.400423 dB	52.737680 dB	537.947796	380.880148	333.205385
Scale	**MAE Red channel**	**MAE Green channel**	**MAE Blue channel**	**MCRE Red channel**	**MCRE in Green channel**	**MCRE Blue channel**	**NCD**
1	34.941953	34.119675	34.309896	0.051001	0.044203	0.042414	0.480219
2	26.417234	24.171183	23.604334	0.019317	0.015701	0.014670	0.313019
3	20.736065	17.854867	17.026820	0.010909	0.008168	0.007404	0.227659
4	18.777056	15.732278	14.815044	0.008779	0.006326	0.005575	0.198971
5	18.215902	15.077929	14.170148	0.008273	0.005857	0.005124	0.191196

Tables 2 to 5 present the performance results to apply the proposed filter with only one detection block by means of use the SD-ROM, TSM, ACWM, MM-KNN algorithms, respectively.

The performance results for the use of a second detection block are shown in Tables 6 and 7 by means of use the MSM and FIRMH Filters, respectively. In these cases we use a 7x7 filtering window size.

One can see from the simulation results that the detection blocks (implementation of different filtering algorithms) provide good results in terms of noise suppression and detail preservation.

Finally, we can say that the proposed filters provide better properties in terms of CCI, PSNR, MSE, MAE, MCRE, and NCD in comparison with the performance of the thresholding Wavelet methods [3, 4] and Wavelet domain iterative center weighted median filter [6]. It can see in refs. [5, 14] where we demonstrate the better performance of the method of redundancy of approaches, and when we use it into the Wavelet domain iterative center weighted median filter in comparison with the filters described in [3, 4, 6]

Table 2. Performance results for the proposed filter using the SD-ROM Filter algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0 925999	29 768870 dB	31 200992 dB	31 614584 dB	3313 101568	2871 035178	2754 713506
2	0 971379	39 476371 dB	41 549986 dB	42 231109 dB	1254 998935	1019 969956	952 810607
3	0 984424	45 184984 dB	48 080248 dB	49 062561 dB	709 122811	530 862115	481 194231
4	0 987710	47 354953 dB	50 632708 dB	51 896504 dB	570 795947	411 272604	362 446420
5	0 988663	47 947747 dB	51 400072 dB	52 737976 dB	537 942899	380 893521	333 195533

Scale	MAE Red channel	MAE Green channel	MAE Blue channel	MCRE Red channel	MCRE Green channel	MCRE Blue channel	NCD
1	34 921154	34 096479	34 284837	0 050951	0 044153	0 042364	0 479900
2	26 404142	24 158269	23 589541	0 019300	0 015686	0 014653	0 312853
3	20 732692	17 850251	17 022308	0 010905	0 008164	0 007400	0 227608
4	18 775680	15 730385	14 813994	0 008778	0 006325	0 005574	0 198958
5	18 215976	15 078284	14 169941	0 008273	0 005858	0 005124	0 191197

Table 3. Performance results for the proposed filter using the TSM Filter algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0 925994	29 767953 dB	31 200629 dB	31 613487 dB	3313 405473	2871 139201	2755 015503
2	0 971377	39 475705 dB	41 549439 dB	42 230318 dB	1255 082470	1020 025769	952 886050
3	0 984423	45 184815 dB	48 079870 dB	49 062087 dB	709 134793	530 882189	481 217027
4	0 987710	47 354983 dB	50 632792 dB	51 896626 dB	570 794260	411 269157	362 442012
5	0 988663	47 947710 dB	51 400117 dB	52 737987 dB	537 944882	380 891820	333 195163

Scale	MAE Red channel	MAE Green channel	MAE Blue channel	MCRE Red channel	MCRE in Green channel	MCRE Blue channel	NCD
1	34 922722	34 096923	34 287249	0 050956	0 044154	0 042369	0 479918
2	26 405104	24 158935	23 590518	0 019302	0 015687	0 014654	0 312864
3	20 732959	17 850533	17 022530	0 010906	0 008164	0 007400	0 227611
4	18 775621	15 730340	14 813994	0 008778	0 006325	0 005574	0 198958
5	18 215947	15 078240	14 169926	0 008273	0 005858	0 005124	0 191197

Table 4. Performance results for the proposed filter using the ACWM Filter algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0 925997	29 768585 dB	31 200745 dB	31 614228 dB	3313 196139	2871 104021	2754 811450
2	0 971378	39 476028 dB	41 549664 dB	42 230851 dB	1255 041900	1020 002855	952 835192
3	0 984424	45 185074 dB	48 080198 dB	49 062361 dB	709 116376	530 864734	481 203861
4	0 987710	47 354917 dB	50 632708 dB	51 896521 dB	570 798033	411 272604	362 445814
5	0 988663	47 947747 dB	51 400072 dB	52 737976 dB	537 942899	380 893521	333 195533

Scale	MAE Red channel	MAE Green channel	MAE Blue channel	MCRE Red channel	MCRE in Green channel	MCRE Blue channel	NCD
1	34 921967	34 097027	34 285769	0 050953	0 044154	0 042365	0 479911
2	26 404512	24 158920	23 589985	0 019301	0 015686	0 014653	0 312857
3	20 732559	17 850266	17 022382	0 010905	0 008164	0 007400	0 227609
4	18 775695	15 730385	14 814068	0 008778	0 006325	0 005574	0 198958
5	18 215976	15 078284	14 169941	0 008273	0 005858	0 005124	0 191197

Table 5. Performance results for the proposed filter using the MM-KNN Filter algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0.925931	29.759522 dB	31.190093 dB	31.603201 dB	3316.200178	2874.165917	2757.850032
2	0.971354	39.467824 dB	41.540249 dB	42.220384 dB	1256.071982	1020.963586	953.833107
3	0.984417	45.181801 dB	48.075928 dB	49.057611 dB	709.348521	531.091464	481.432500
4	0.987709	47.353817 dB	50.631474 dB	51.895347 dB	570.860769	411.323373	362.488358
5	0.988663	47.947501 dB	51.400560 dB	52.738011 dB	537.956139	380.874941	333.194364
Scale	**MAE Red channel**	**MAE Green channel**	**MAE Blue channel**	**MCRE Red channel**	**MCRE in Green channel**	**MCRE Blue channel**	**NCD**
1	34.941124	34.118491	34.308979	0.050999	0.044201	0.042412	0.480207
2	26.417012	24.170621	23.603521	0.019317	0.015701	0.014669	0.313013
3	20.735917	17.854127	17.026790	0.010909	0.008167	0.007404	0.227653
4	18.776923	15.731746	14.815074	0.008779	0.006326	0.005575	0.198967
5	18.216050	15.077544	14.169926	0.008273	0.005857	0.005124	0.191195

Table 6. Performance results for the proposed filter using the MSM Filter estimation algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0.925943	29.758344 dB	31.194552 dB	31.608770 dB	3316.590062	2872.884571	2756.315533
2	0.971380	39.473387 dB	41.552420 dB	42.233572 dB	1255.373417	1019.721731	952.575976
3	0.984427	45.181500 dB	48.085745 dB	49.066113 dB	709.369882	530.570355	481.023343
4	0.987714	47.351208 dB	50.639627 dB	51.902085 dB	571.009734	410.988151	362.244201
5	0.988668	47.944820 dB	51.408736 dB	52.744168 dB	538.100340	380.563669	332.989290
Scale	**MAE Red channel**	**MAE Green channel**	**MAE Blue channel**	**MCRE Red channel**	**MCRE in Green channel**	**MCRE Blue channel**	**NCD**
1	34.945281	34.113713	34.302722	0.051005	0.044181	0.042389	0.480189
2	26.410725	24.158269	23.589763	0.019306	0.015682	0.014649	0.312862
3	20.737751	17.845976	17.019172	0.010909	0.008159	0.007398	0.227575
4	18.780680	15.725163	14.809645	0.008781	0.006320	0.005571	0.198914
5	18.219778	15.071509	14.165562	0.008275	0.005853	0.005121	0.191141

Table 7. Performance results for the proposed filter using the FIRMH Filter estimation algorithm.

Scale	CCI	PSNR(dB) Red channel	PSNR(dB) Green channel	PSNR(dB) Blue channel	MSE Red channel	MSE Green channel	MSE Blue channel
1	0.925943	29.758344 dB	31.194552 dB	31.608770 dB	3316.590062	2872.884571	2756.315533
2	0.971380	39.473387 dB	41.552420 dB	42.233572 dB	1255.373417	1019.721731	952.575976
3	0.984427	45.181500 dB	48.085745 dB	49.066113 dB	709.369882	530.570355	481.023343
4	0.987714	47.351208 dB	50.639627 dB	51.902085 dB	571.009734	410.988151	362.244201
5	0.988668	47.944820 dB	51.408736 dB	52.744168 dB	538.100340	380.563669	332.989290
Scale	**MAE Red channel**	**MAE Green channel**	**MAE Blue channel**	**MCRE Red channel**	**MCRE in Green channel**	**MCRE Blue channel**	**NCD**
1	34.945281	34.113713	34.302722	0.051005	0.044181	0.042389	0.480189
2	26.410725	24.158269	23.589763	0.019306	0.015682	0.014649	0.312862
3	20.737751	17.845976	17.019172	0.010909	0.008159	0.007398	0.227575
4	18.780680	15.725163	14.809645	0.008781	0.006320	0.005571	0.198914
5	18.219778	15.071509	14.165562	0.008275	0.005853	0.005121	0.191141

4 Conclusions

We present different implementations of order statistics filters in wavelet domain. The proposed filters constitute a good tool to the impulsive noise filtering in color image applications. It is obviously that the use of different order statistics filters into the proposed block detection in the wavelet domain can increase the processing time.

Acknowledgment. To National Polytechnic Institute of Mexico for all the given facilities to carry out this paper.

References

1. S. K. Mitra and G. L. Sicuranza, Nonlinear Image Processing, Academic Press, San Diego, CA, 2001, pp. 111-133.
2. J. Astola and P. Kuosmanen, Fundamentals of Nonlinear Digital Filtering, CRC Press, Boca Raton-New York, 1997.
3. Z. Cai, T.H. Cheng, C. Lu, K.R. Subramanium, Efficient wavelet-based image denoising algorithm, Electron. Lett. 37(11) 683-685, 2001.
4. S.G. Chang, B. Yu, M. Vettereli, Adaptive wavelet thresholding for image denoising and compression, IEEE Trans. Image Process. 9 (9) 1532-1546,.2000.
5. J. Martínez Valdes, F. J. Gallegos Funes, M. A. Acevedo Mosqueda, Reducción de Ruido Impulsivo en Imágenes Digitales a Color Utilizando Wavelets, Memorias de IEEE 15 Reunión de Otoño de Comunicación y Exposición Industrial ROC&C 2004, Acapulco, México, Noviembre 2004.
6. S. M. Mahbubur Rahman, Md. Kamrul Hasan, Wavelet-domain iterative center weighted median filter for image denoising, Signal Processing, 83, 1001-1012, 2003.
7. T. Chen, K. Ma, L. Chen, Tri-State Median Filter for image denoising, IEEE Trans. Image Process. 8 (12), 1834-1838, 1999.
8. T. Chen and H. R. Wu, Adaptive impulse detection using center-weighted median filters, IEEE Signal Processing Letters, 8(1), 1-3, 2001.
9. F. Gallegos, V. Ponomaryov, O. Pogrebnyak y L. Niño de Rivera, Filtros Robustos RM-KNN con Diferentes Funciones de Influencia para Supresión de Ruido Impulsivo en Imágenes Digitales, Computación y Sistemas, Vol. 6, No. 3, pp. 183-195, 2003.
10. F. Gallegos, V. Ponomaryov, Real-time image filtering scheme based on robust estimators in presence of impulsive noise, Real Time Imaging, 8(2), 78-90, 2004.
11. G.R. Arce, M. P. McLoughlin, Theoretical analysis of the max/median filter, IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-35, 60-69, Jan. 1987.
12. J. Astola, P. Heinonen, Y. Neuvo, Linear median hybrid filters, IEEE Trans. Circuits and Systems, vol. CAS-36, 1430-1438, Nov. 1989.
13. P. Heinonen, Y. Neuvo, Smoothed median filters with FIR substructures, Proc. 1985 IEEE Int. Conf. Acoust. Speech and Signal Processing, Tampa, March 1985, 49-52.
14. J. Martinez Valdes, F. Gallegos-Funes, M. A. Acevedo-Mosqueda, "Filtro Mediano con Peso Central Iterativo en el Dominio Wavelet utilizando el Método de Redundancia en las Aproximaciones", 6to. Congreso de Computación, CORE 2005, Research on Computing Science 13, Advances in Computer Science in Mexico, pp. 41-49, IPN, CIC, Mexico, D. F., Mayo 2005.