International Journal of Innovative Computing, Information and Control
Volume **13**, Number **6**, December **2017**

ICIC International ©2017 ISSN 1349-4198
pp. 1965–1980

EFFICIENT NOVEL VECTOR MEDIAN FILTER DESIGN FOR IMPULSE NOISE SUPPRESSION IN COLOR IMAGES

Benjamin Asubam Weyori[1,2], Kwame Osei Boateng[2] and Sampson Twumasi-Ankrah[3]

[1]Department of Computer Science and Informatics
School of Sciences
University of Energy and Natural Resources
Box 214, Sunyani, Brong Ahafo Region, Ghana
benjamin.weyori@uenr.edu.gh

[2]Department of Computer Engineering
College of Engineering
[3]Department of Mathematics
College of Science
Kwame Nkrumah University of Science and Technology
PMB, Kumasi, Ghana
{ boat.soe; stankrah.cos }@knust.edu.gh

Received September 2016; revised February 2017

Abstract. *Although different types of efficient vectors' median filters have been proposed in literature, researchers have not resolved the problem identified in vector filters: (i) detecting impulse noise in individual color component with the smallest dissimilarity measure in the window; (ii) detecting impulsive pixels in regions of highly dense impulse noise mixed with bright image pixels and edge pixels. Following the establishment of this shortcoming identified in existing vector filters, this paper aims at developing a novel vector filter that will solve this existing problem. This work proposes an intelligent switching based vector median filter that adapts and combines different techniques for noise detection, noise filtering. The algorithm adopts the K-nearest neighbor clustering method to perform some simple clustering of the image data in the window into noisy and noise-free pixels. This algorithm aids a rigorous separation of pixels into two parts. The primary task of the proposed filter is to recover the dead edge pixels, polluted pixels and recover the corrupted image detail. The NVMF is implemented in MATLAB and applied to several noisy color images to test the performance of the filter. The results of the experimental analysis shown in both the subjective and objective analysis prove that the proposed vector filter outperforms the state-of-the-art filters in noise (impulse) detection and removal.*

Keywords: Clustering, Vector filter, Impulse noise, Pixels, Switching filters, Distance measure

1. Introduction.

The rise in the use of color images in modern audiovisual technologies and the telecommunication industry has enhanced these sectors ominously in recent years. This massive improvement in these sectors is complemented by the proliferation of color image capturing devices. These color image capturing devices are responsible for the quality of the image. The capturing device introduces noise into a color image as a result of faulty sensors [1].

One main factor that degrades the quality of color images is the noise introduced by the capturing devices. Color images are also degraded by noise from other sources like errors in transmission, and electrical disturbances. Different types of noise pollute images, but the commonest and most difficult to suppress is impulse due to its variations and randomness. Thus, to maintain the quality of color images to aid processing in the multimedia and telecommunication industries, noise must be filtered out. This noise is the undesirable signal introduced in the image. The purposes of filtering are to restore the image and enhance them (in the logic of improved visual feature) to attain superior pre-conditions for image clustering, compression, object detection, etc. [2,3].

Different types of filters have been proposed to reduce noise in color images. For Gaussian noise the best filters that suppress the unwanted signals in order to obtain a relatively clean image are linear filters. However, for the random varying type of noise (impulse noise) the nonlinear filters perform much better than the linear filters. Such a variety of approaches is explained by several reasons. One of such reasons is usually as a result of the spread of the noise signal in the image. Some types of noise (e.g., Gaussian noise) are uniformly distributed in the image while others (e.g., impulse noise) are varying and distributed in a random manner in the image [2,3].

The vector median filter (VMF) [4] is the first filter to consider pixels of a color image as a vector field. This concept was development due to the problems encountered by the scalar median filters on color image. These scalar median filters are applied to the individual components of a color image. Though they perform well by suppressing the unwanted signal, they introduce color artifacts in the image [5]. This happens because the scalar filters do not take consideration of the inherent correlation of the color components. However, the introduction of the vector median filter solved the problem of color artifacts introduced in color images by their scalar counterpart. This vector median filter ranks the pixels in a window using the reduced ordering method. This method selects the vector with the lowest cumulative distance as the median. The distance measure, applied in performing the reduced ordering by computing the relative difference between sets of coordinates or points, is the L1 norm known as the Manhattan distance and the L2 norm also known as the Euclidean distance.

The family of vector filters developed based on the VMF includes the basic vector directional filter (BVDF) [6]. The BVDF applies the angular measure in place of the distance measure for the ranking of the pixel in the window. Other extensions of the VMF are distance-directional filter (DDF) [7], and generalized vector directional filter (GVDF) [8], hybrid directional filter (HDF) [9], weighted vector median filter (WVMF) [10]. These filters are all developed on the theory of robust statistics [1,11,12]; hence they treat each pixel of the color image as a vector. These filters belong to the class of uniformly applying vector nonlinear filters. This class of filters performs the computation of the median value and replaces very pivot pixel of the window with the median. This process is done as the window is applied to covering the entire image introducing distortions in the image.

Vector filters have been categorized into several class, some of which are the adaptive vector median filter class [13], hybrid vector median class [9], fuzzy vector filter class [14], vector marginal median filter class [15], the switching vector median filter class [16,17] and the decision based vector median filter class [18,19]. Among these classes mentioned, the switching vector median class [16,17] and the decision based median filter class [18,19] are the most recent group of the proposed filters. These classes of filters are able to improve upon the existing filters in suppressing impulse noise.

In [20], the decision based median filter (DBMF) is proposed to solve the problems encountered by the uniform applying vector filters and the blurring, introduced by subsequent filters, that distorts the quality of the color in an image when the fuzzy switching filter is applied [20]. The DBMF adapts a technique which makes its window adjustable and not fixed like in most cases. The computational time of the filter depends heavily on the size of the window. The window size increases depending on the ratio of noisy pixel to clean pixel in the window (if the ratio of noisy pixel is relatively high, the window size is increased). The technique works by defining a minimum or lower threshold (Tl) and a maximum or upper threshold (Tu). A test is carried out by comparing the central value in the window to the lower and upper thresholds. The value is considered noisy or noiseless depending on the Tl and Tu defined in [20]. An extension to improve the DBMF is presented in (ref).

The decision based unsymmetric trimmed median filter is proposed in [21], and this filter applies a simple local noise detection technique by checking if the center pixel is corrupted or not. The center pixel is tested and deleted if and only if it is noisy. An optimal pixel value (median value) is inserted in the center position of the window. In regions where the noise intensity is very high, the filter replaces the center pixel by either 0 or 255 which in most cases is also a noisy candidate. One method to overcome this issue is the modified decision based unsymmetric trimmed median filter (MDBUTMF) [11,22]. This filter performs better than the state-of-the-art switching median filter in high noise intensity regions.

The rank weighted switching filter (RWSF) [23] belongs to the class switching filters, which are developed based on decisions. Their intelligence is based on decision set for operation. Thus, if the decision for a class is not met, then the switching function moves it from that class to another class for testing. An extension of this filter is the adaptive rank weighted switching filter (RWASF) [12], which is proposed to improve upon the RWSF. The uniqueness of the RWASF is achieved in the introduction of adaptiveness in its design. The adaptiveness of the design aids the filter in tuning its parameters depending on the ratio of impulsive noise corrupting in the image.

These classes of filters mentioned above have their shortcomings when applied to image. The most challenging difficulty of the state-of-the-art filters is detecting noise in one color component if it appears with the smallest dissimilarity measure. Hence, they introduce distortions in the image details and their edges. Another problem is that the state-of-the-art filters are not able to completely differentiate between high dense impulse noise from fine image details and their edges. This happens mostly in regions of highly dense impulse noise mixed with bright image pixels and edge pixels [12,20,24,25]. The novel vector median filter (NVMF) developed is different from the existing vector filters because the NVMF applies some sort of learning of the pixels by employing some clustering methods to separate the noisy pixels from the noiseless ones. Based on this intelligence, the noisy pixels are replaced with the most optimum noiseless pixel to aid the quality of the image.

The main objective of this paper is to design an effective filter to restore or recover an image that has been degraded by impulse noise (salt and pepper noise) using prior knowledge of the type of noise and also considering that not all the image data are completely polluted. The filter is designed to detect impulse noise in individual color component with the smallest dissimilarity measure in the window and to detect impulsive pixels in regions of highly dense impulse noise mixed with bright image pixels and edge pixels. Based on the intelligence employed, the proposed filter is able to detect corrupted/distorted and uncorrupted/undistorted objects in the color images.

Our proposed method is an intelligent switching based vector median filter that adapts and combines different techniques for noise detection, noise filtering and correction of artificial color generated by certain distortions in the image. The product of this work is a simple intelligent novel vector filter that improves upon the switching median filter to produce bright image and clean color images with good quality.

The rest of the paper is organized into four sections as follows. Section 2 introduces the novel vector median filter (NVMF). Section 3 focuses on the mode of operation of the NVMF. The same section contains the algorithms of the various phases of the NVMF with explanation using the flowchart. Section 4 is devoted to the analysis of the proposed NVMF framework with ten (10) different filters. The analysis is based on comparing the simulation results of the NVMF with other proposed filters in literature. The image quality measure is depicted in both the subjective and objective simulations test and filtering results presented in tables and graphs. Finally, the conclusion is presented in Section 5.

2. Novel Vector Median Filter Methodology.

The novel vector median filtering algorithm is a newly developed filter that adopts some mechanism for effective operation from the vector median filter method and the K-nearest neighbor clustering method [26,27] to perform noise detection and filtering of impulsive noise. The KNN algorithm is used to perform some simple clustering of the image data in the window into noisy and noise-free pixels. If this similarity detection method fails to selecting the most effective value, the vector median filter [4] is applied to selecting the most suitable pixel value.

The cluster-based vector median filter basically consists of three main sections: (1) image training, (2) noise detection stage and (3) noise suppression.

The three sections of the algorithm are introduced below.

Section I: The uncorrupted image data is used to model the distribution of pixel applying a 3×3 block to computing and set the value of *Train_Dist*(i) for all the fixed non-overlapping blocks of the image.

Section II: The pixels in the window are grouped into clean pixel and noisy pixels using the *Train_Dist*(i) as a measure (noise detection stage).

Section III: Replace the noisy pixels with the optimal pixel selected (noise filtering).

Train_Dist(i) *is the cumulative distances computed by the pixels in Window*(i) *in the uncorrupted image. Window*(i) *is any chosen* 3×3 *window and* $i = 1, \ldots, n$.

The filtering algorithm proposed is novel because it uses a clustering algorithm to separate the noise from the clean pixels and then applies a method of replacing the noisy pixel with the selected clean pixels. The novelty lies the modification of the clustering algorithm and its application for the clustering.

3. Development of the Proposed Filter.

3.1. Design of the extraction stage.

This section deals with learning the pattern of distribution of the uncorrupted image data by some simple method of feature extraction. The brain behind this process is to study the distribution of the pixel patterns in a window as well as their inherent correlation. Thus, the pattern and correlation aid in the corrections of corrupted pixels in the color image. The moving non-overlapping window technique is employed in modeling the pattern of the original uncorrupted color image. The technique is quite different from the well-known moving window technique adapted by most of the non-linear order statistics filter algorithm. This moving non-overlapping window covers the entire image with fixed 3-by-3 non-overlapping windows and moves from pixel to pixel with a specific interval. This operation is attributed to the non-overlapping nature. Figure 1 shows the operation of the moving 3-by-3 non-overlapping window.

The training phase deals with computing the most deviating distance and the most deviating similarity angle for each window. The most deviating distance *Train_Dist*(i) and the most deviating similarity angle *Train_Omg*(i) are computed by applying the reduced sorting method. This method computes the cumulative distances from each pixel to its neighboring pixels and the maximum distance is set as *Train_Dist*(i) for the *Window*(i), and then the corresponding pixel to the maximum distance is used as the base pixel to compute the similarity angles. Thus, similarity angles in *Window*(i) are computed

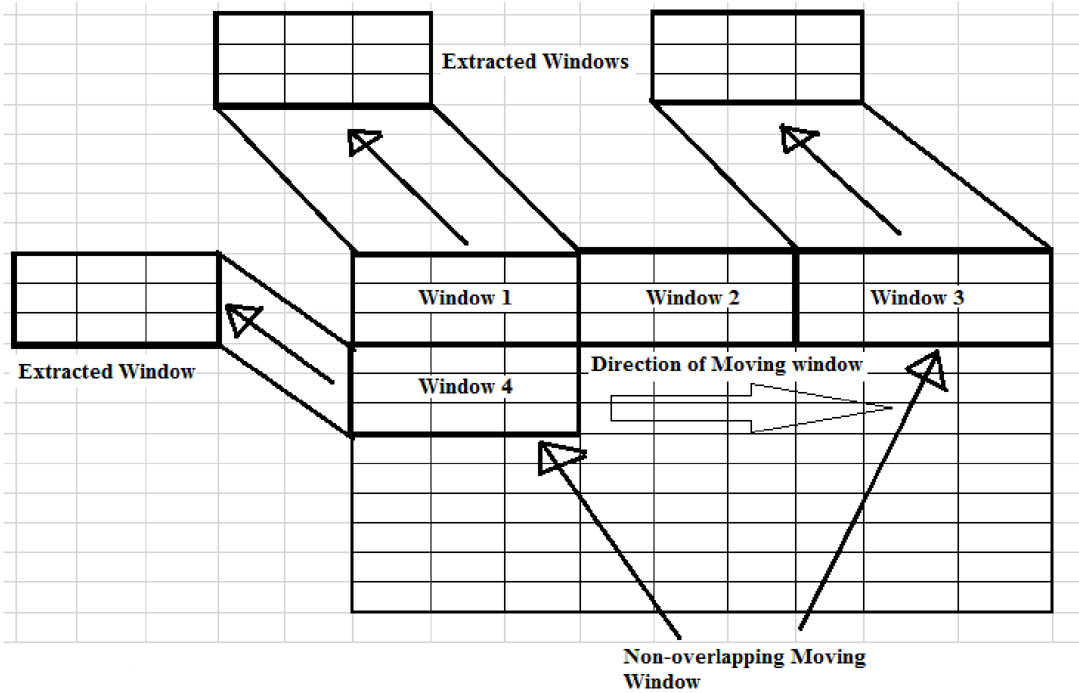

FIGURE 1. Operations performed in the non-overlapping window

between the base pixel and neighboring pixels. The maximum similarity angle is selected and set to $Train\_Omg(i)$. These values are computed locally for each non-overlapping window and saved.

Algorithm of the Feature Extraction Stage

The process of computing the value of $Train\_Dist(i)$ and $Train\_Omg(i)$ is as follows:

- $\boldsymbol{Train\_Dist(i)}$ $\leftarrow$ The cumulative distances computed by the pixels in $Window(i)$ in the uncorrupted image
- $Train\_Omg(i)$ $\leftarrow$ Maximum similarity angle between the most deviating pixels (pixels with the largest similarity angle) in $Window(i)$ in the uncorrupted image
- A and B $\leftarrow$ The vector pixels in $Window(i)$
- $\{a_1, a_2, a_3\}$ and $\{b_1, b_2, b_3\}$ $\leftarrow$ The RGB components of the pixels

1) The input image is a color image
2) The moving 3-by-3 non-overlapping window is moved through the image
3) For each window sort the pixel using reducing sorting method (the cumulative distance metric applied in the sorting process is computed using Equation (1)). For each pixel x_j with neighbors x_r in current window

$$Dist(j) = \sum_{r=1}^{n} \|x_j - x_r\| \tag{1}$$

where n is the number of neighbors of x_j in the current window.
Maximum of $Dist(j)$ is set as $Train\_Dist(i)$ for current window, $Window(i)$.

4) The pixel corresponding to the maximum cumulative distance is used as the base pixel A to compute the similarity angles. For each neighbor $B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$ of

base pixel $A = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$, in current window, compute the similarity angle as per Equation (2) below.

$$\theta = \cos^{-1}\left(\frac{a_1b_1 + a_2b_2 + a_3b_3}{\sqrt{a_1^2 + a_2^2 + a_3^2} \times \sqrt{b_1^2 + b_2^2 + b_3^2}}\right) \tag{2}$$

The maximum similarity angle is set as *Train_Omg*(i) for current window, *window*(i).

3.2. **Design of the noise detection and noise filtering phase.** The result from the feature extraction section is fed into this section for the noise detection process. This phase is a data dependent stage that is built on the results obtained from the first phase. The feature extraction phase is very novel in the computation of the maximum cumulative distance *Train_Dist*(i) and the maximum similarity angle *Train_Omg*(i).

The noise detection phase and the noise filtering phase employ the moving 3-by-3 overlapping window. The noise detection phase is applied to aiding the testing and categorization of pixels in the window while the noise filtering suppresses the noise leaving the clean pixels.

3.2.1. *Noise detection phase.* The distance metric is applied in computing the cumulative distances for each pixel to all its neighbors in the window. These cumulative distances computed are stored as *Dist*(j). This process is applied to the corrupted image to aiding the noise detection and suppression.

The noise detection employs a region-cluster-based method for each window, which creates two clusters; *P cluster* and *N cluster for each pixel.* The *P cluster* groups all the clean pixels while the noisy pixels or outliers are grouped in the *N cluster*. The testing stage is the main step of the noise detection phase; it works by comparing all the cumulative distance computed *Dist*(j) in the first window to their corresponding *Train_Dist*(i) from the feature extraction stage. The *Dist*(j) from the window greater than *Train_Dist*(i) is moved into the Cluster N while the *Dist*(j) that is less than or equal to *Train_Dist*(i) are moved into Cluster P. This process is performed for each window locally. After the pixels in the first window are grouped into clusters, this is fed into the noise filtering phase.

Noise Detection Algorithm

- $j \leftarrow$ the number of current pixels in the current sliding window
- $Dist(j) \leftarrow$ the cumulative distances computed by s pixel in current window (*Window*(j))
- Cluster $P \leftarrow$ cluster of good pixels
- Cluster $N \leftarrow$ cluster of noisy pixels or outliers
- $x_j \leftarrow$ current pixel value
- $x_k \leftarrow$ selected pixel

1) For each pixel j in the current window compute the distance metric as per Equation (3)

$$Dist(j) = \sum_{j=1}^{9} \|x_k - x_j\| \tag{3}$$

2) The centroid of cluster P is x_{med}, the pixel with the least *Dist* measure, and $N = \Phi$ initially
3) For all $x_j \in Window(j)$ repeat steps a) and b) until all pixels have been considered

a) If $Dist(j) \leq Train\_Dist(i)$ then
   i. Update cluster label P with feature vector x_j thus $\{P = P \cup x_j\}$
b) If $Dist(j) > Train\_Dist(i)$ then
   i. Update cluster label N with feature vector x_j thus $\{N = N \cup x_j\}$

4) Return

3.2.2. *Noise filtering phase.* The filtering of the noise and outlier detection is based on the classification of the pixels in the window. If the entire pixel values in the window are grouped in P, then it is assumed there are no noisy pixels or outliers present in that window implying that the center pixel also belongs to cluster P; hence the moving window is moved and centered on the next pixel to its right. Then the noise detection stage is

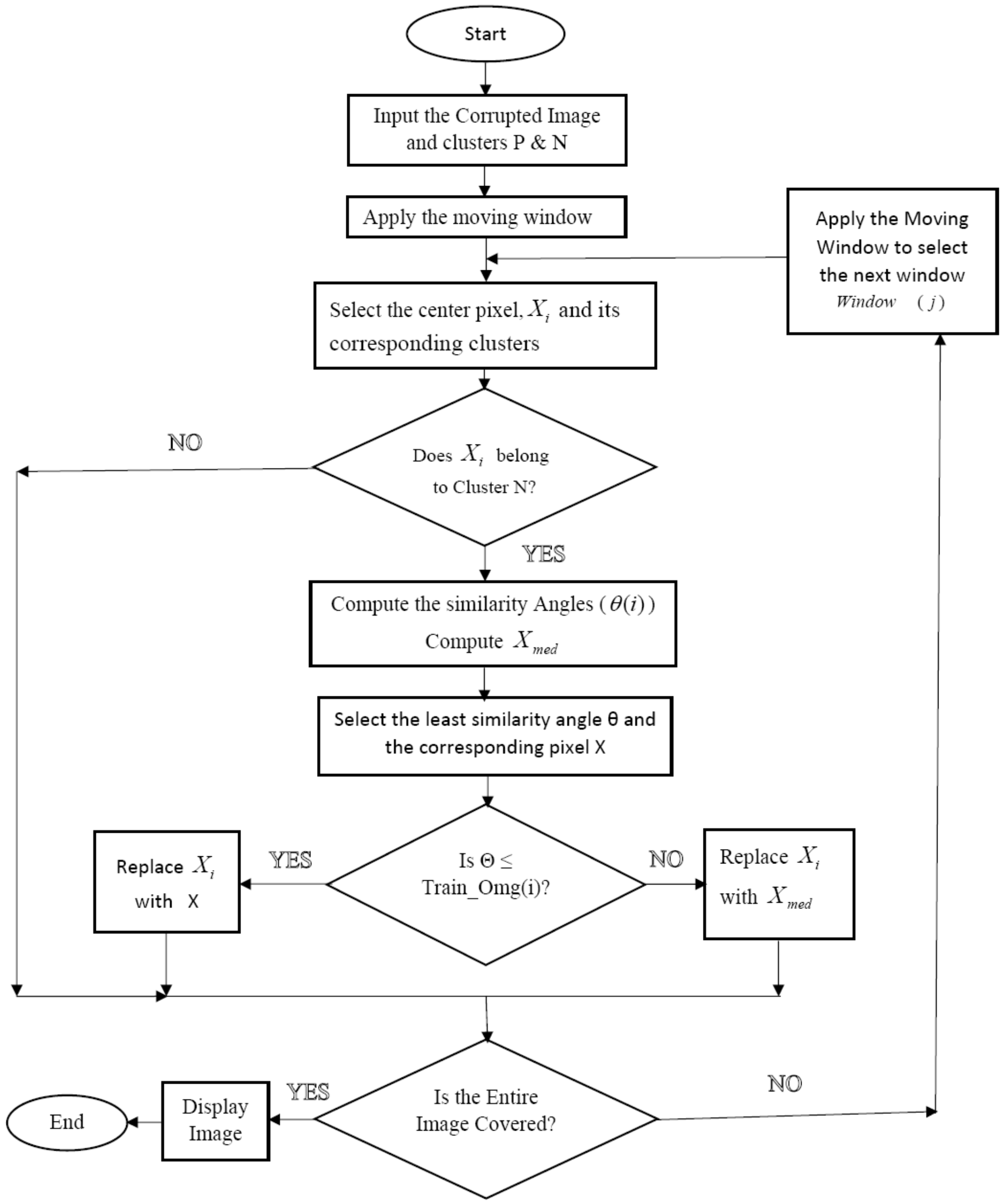

FIGURE 2. A chart showing the logical flow of the noise detection process

repeated. However, when the center pixel X_i of window i is included in cluster N, then it implies that the center pixel is noisy or an outlier. Once the center pixel is detected to be impulsive, a similarity measure between the center pixel and all its neighboring pixels in the window that are classified in cluster P is computed. The similarity measure angles are arranged in an order of priority, and the most similar pair of vector is taken. The final similarity measure angle is compared to its corresponding angle computed from the feature extraction data. If the angle computed far deviates from that computed from the result of the training, then an alternate is to replace the central pixel by the vector median. Otherwise, it implies that the computed similarity measure is closely related to the similarity measure computed from its corresponding window. After the process of the suppression of the noise in this window is completed, the filter moves to the next window and the noise detection mechanism is applied again. The mode of operation of the noise filtering phase is explained further in Figure 3.

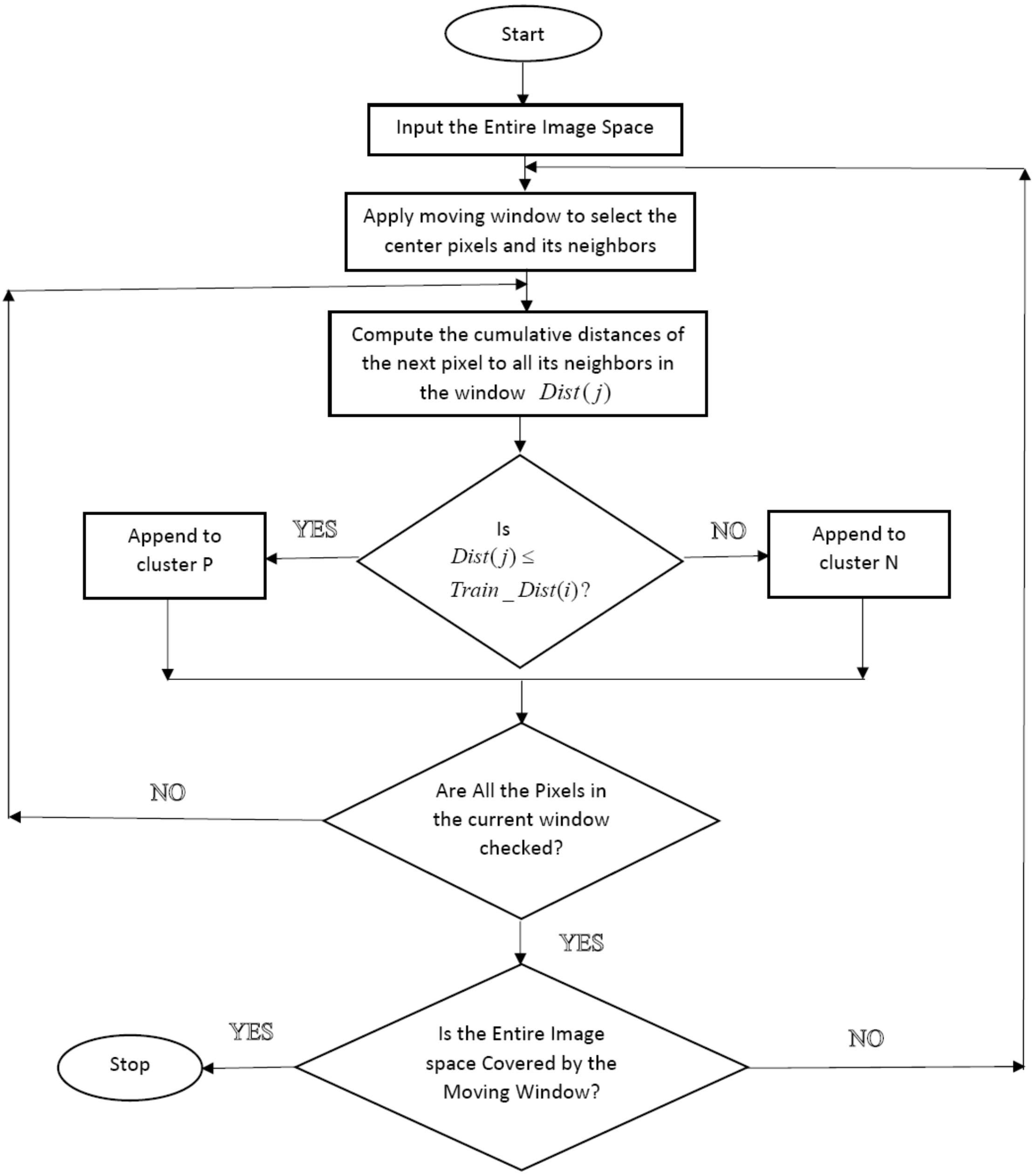

FIGURE 3. A chart showing the logical flow of the filtering process

Noise Filtering Algorithm

1) If X_i belongs to set P exit and move to the next window.
2) If X_i belongs to set N then proceed
3) Compute the similarity angle between the noisy center pixel and each of the other pixels x_j of current window that are in cluster P.
4) Sort the similarity angles into ascending order and select pixel X with the least similarity angle
5) Compute the vector median value X_{med}, in the active window.
6) Replace the noisy center pixel as per Equation (4).

$$X_i = \begin{cases} X & \text{for } \theta \leq \mathit{Train\_omg} \\ X_{med} & \text{for } \theta > \mathit{Train\_omg} \end{cases} \tag{4}$$

7) Return

The noise detection and noise filtering stage work together in stepwise manner. After the noise detection phase is completed for a window, the clusters of that window are fed in the noise filtering phase and the filtering operation is performed. When the filtering operation also completes the moving overlapping, window is moved to the next window location and the two processes are repeated again. This overlapping window moves from one location to the other to cover the entire image aiding the filtering process at each window local.

The moving window is an operator proposed by the low-pass filters. The operator usually affects one pixel of the image at a time, changing its value by some function of a "local" region of pixels ("covered" by the window). The operator "moves" over the image to affect all the pixels in the image. The operator moves throughout the image horizontally from the left-hand side to the right-hand side as shown in Figure 4. The process starts with the first pixel on the left corner of the image. That pixel is taken as the center pixel and then a square window is drawn around it forming a 3-by-3 or a 5-by-5 or a 7-by-7. The point moves from the first pixel to the last pixel horizontally on a straight line as shown in Figure 5. When the window gets to the last pixel of the image, the process increases by one step in the vertical direction and starts from the next pixel below the first pixel on the left-hand side.

The moving overlapping window produces more windows than the non-overlapping windows during the noise detection phase. However, the windows produced in between the two non-overlapping windows are handled by applying a simple vote count to finding out which window has more of its pixels members in the moving window. The *Train_Dist*(i)

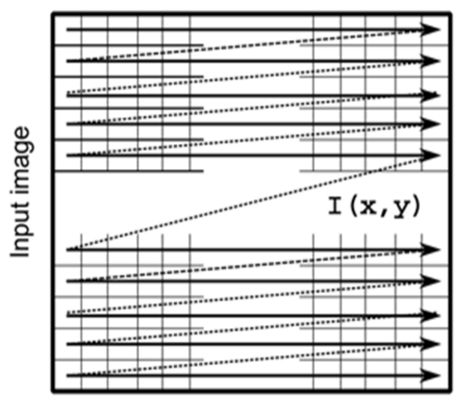

FIGURE 4. Graphical representation showing the direction of the sliding window technique

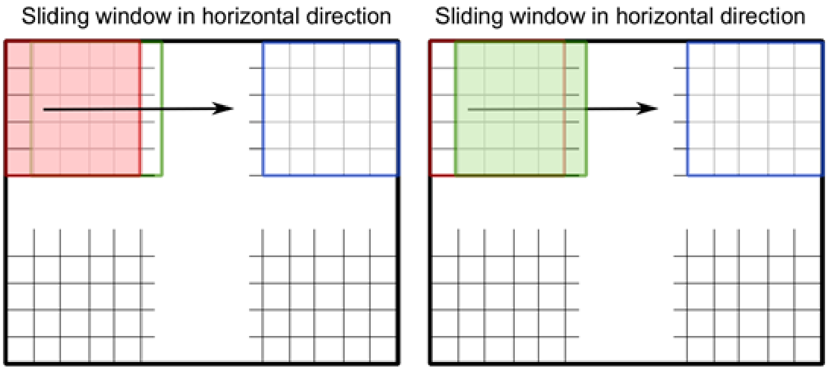

FIGURE 5. Graphical representation of the movement of the sliding window

of the non-overlapping window is applied to the present overlapping window to clustering the pixel into P and N. The method of the major votes considers which non-overlapping window has more pixels members in the current overlapping window.

This filter achieves very good results depending on the relationship of the feature extraction data and the corrupted image. When the original image is used as the feature extraction phase data, then the novel vector filter achieves a much-improved result compared with most of the state-of-the-art vector filters.

4. Simulation Results.

4.1. Subjective analysis.

The test result of the application of the novel vector median filter is displayed above for visual analysis. The proposed novel vector median filter achieves excellent results. If the features extracted from the image are the same as that of the corrupted image, then the features are used to restore the noise in the corrupted image. In this simulation, the bench mark images are selected and presented for analysis. Figure 6 to Figure 9 present the subjective evaluation of performance of the novel vector median filter as applied to the Lena image and Koala image. The filtered images are compared visually with the original image and the corrupted image as well.

4.2. Objective results.

Image enhancement or improving the visual quality of a digital image can be subjective. One method that provides a better-quality image could vary. For this reason, it is necessary to establish quantitative/empirical measures to compare the effects of image enhancement algorithms on image quality. Using the same set of

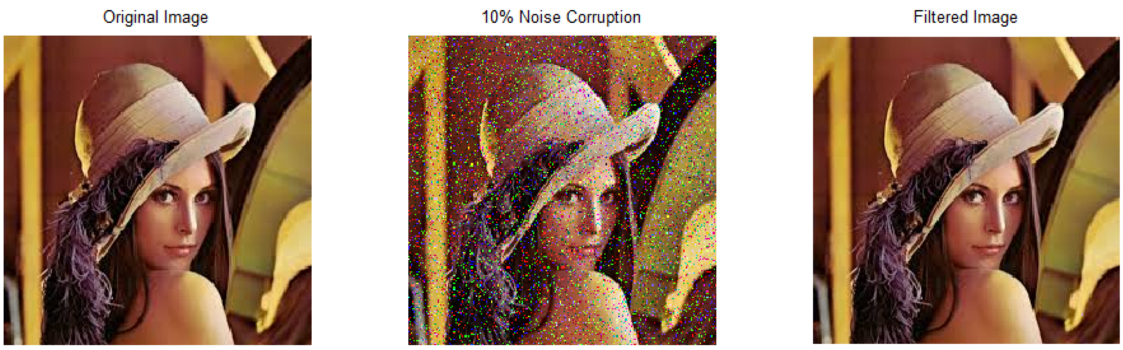

FIGURE 6. Application of the NVMF on the Lena image corrupted with 10% impulse noise

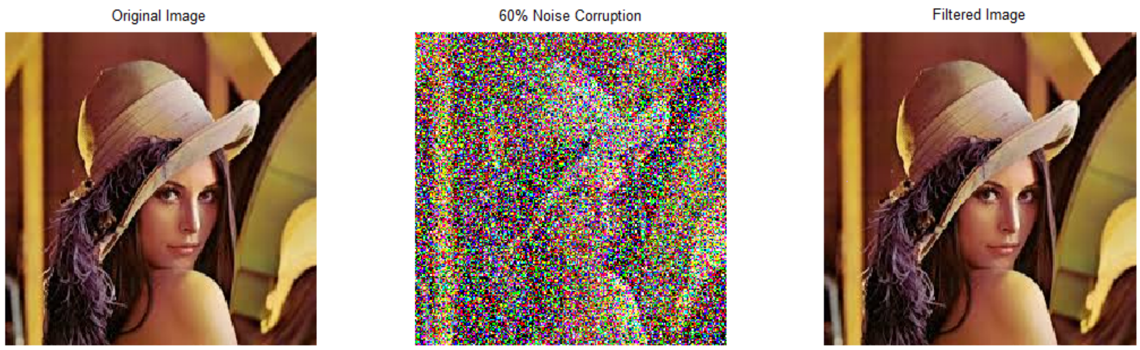

FIGURE 7. Application of the NVMF on the Lena image corrupted with 60% impulse noise

FIGURE 8. Application of the NVMF on the Koala image corrupted with 10% impulse noise

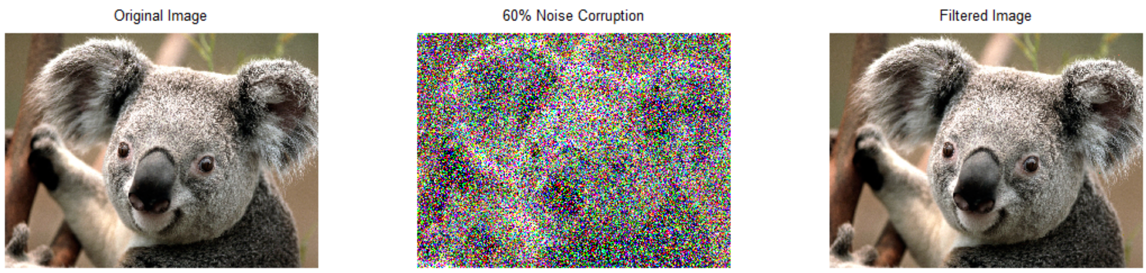

FIGURE 9. Application of the NVMF on the Koala image corrupted with 60% impulse noise

tests on images, different image enhancement algorithms can be compared systematically to identify whether a particular algorithm produces better results. The metric that is applied is the **peak-signal-to-noise ratio**. This ratio shows which algorithm or set of algorithms enhances a degraded known image. The more closely the image quality resembles the original image, the more accurately one can conclude that it is a better algorithm. The bigger the value of the PSNR of a particular algorithm when applied to a degraded image is, the better the performance of that algorithm as compared to the others is.

In this work, we narrow our evaluation to the use of the peak-signal-to-noise ratio. The PSNR of two separate images is computed for various types of filters. The mathematical formula for the PSNR is expressed in Equation (5) below,

$$PSNR = 20 * \log\left(\frac{255^2}{MSE}\right) \tag{5}$$

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [I(i,j) - K(i,j)]^2 \tag{6}$$

where M and N are the total number of pixels in the column and row of the image respectively. I denotes the noisy image and K denotes he filtered image.

Table 1. Filters taken for comparison with the proposed block matching hybrid median filter and the novel vector median filter

Filter Abbreviation	**Filter**
PGF	Peer Group Filter
ACWVMF	Adaptive Center-Weighted Vector Median Filter
RODSVMF	Rank-Ordered Differences Statistic Based Switching Vector Median Filter
ACWDDF	Adaptive Center-Weighted Directional-Distance Filter
SDDF	Sigma Directional Distance Filter
AVMF	Adaptive Vector Median Filter
ABVDF	Adaptive Basic Vector Directional Filter
SVMF	Sigma Vector Median Filter
FMVMF	Fast Modified Vector Median Filter
RWASF	Adaptive Rank Weighted Switching Filter

Table 2. Comparison of PSNR of filters on the Lena image

S/No.	**Filters**	**10%** [dB]	**20%** [dB]	**30%** [dB]
1	RWASF	**38.89**	**35.58**	**32.76**
2	PGF	37.89	33.22	29.04
3	ACWVMF	38.47	34.05	29.45
4	RODSVMF	37.96	34.72	32.01
5	ACWDDF	36.92	33.69	29.69
6	SDDFr	37.15	31.93	26.5
7	AVMF	36.94	33.97	31.26
8	ABVDF	35.36	31.93	28.67
9	SVMFr	36.91	32.28	26.89
10	FMVMF	35.93	34.09	32.14
11	NVMF	**40.33**	**37.43**	**32.97**

4.3. **Discussion of experimental results.** This novel vector median filters are designed with a robust mechanism in detecting impulse noise in a color image. This impulse noise detection mechanism and the computing of an optimal value to replace the impulsive center pixel are explained in the design phase of this work. The impulse noise detector uses the distance measure coupled with some intelligent techniques to detect pixels corrupted with impulse noise (salt and pepper) from very bright uncorrupted pixels and also edge pixels. The uncorrupted bright pixels and edge pixels are left untouched in the image preserving the smoothness and quality of the image. The evidence that this filter is able to detect impulse noise and replace it with a noise free pixel is shown in the subjective analysis and its quantitative results shown in the objective analysis. This process restores

TABLE 3. Comparison of PSNR of filters on the Parrot image

S/No.	**Filters**	**10%** [dB]	**20%** [dB]	**30%** [dB]
1	**RWASF**	**38.22**	**35.03**	**32.52**
2	PGF	37.46	32.41	28.8
3	ACWVMF	37.9	33.69	29.5
4	RODSVMF	36.66	33.94	31.69
5	ACWDDF	37.91	34.05	30.08
6	SDDFr	38.18	32.74	27.23
7	AVMF	35.66	33.07	30.74
8	ABVDF	31.97	28.5	26.15
9	SVMFr	37.07	32.19	27.17
10	FMVMF	35.77	34.15	32.35
11	**NVMF**	**40.73**	**37.22**	**33.99**

the fine image details and resolves the blurring problems encountered by other filters presented in the experimental analysis.

The results of the objective analysis demonstrate the effectiveness of the novel vector median filter vis-à-vis the RWASF, PGF, ACWVMF, RODSVMF, ACWDDF, SDDF, AVMF, ABVDF, SVMFr, FMVMF. For example, the NVMF, when applied to the Lena color image corrupted with 10% impulse noise resulted in a filtered image with a PSNR of 40.33dB. Applied to the same corrupted image, the ABVMF performed poorly with 35.36dB PSNR while the RWASF produced the best result among the reference filters with a PSNR of 38.89dB. For 20% impulse noise corruption NVMF performed better with PSNRs of 37.43dB whereas RWASF and ABVMF resulted in PSNRs of 35.58dB and 31.93dB respectively. For 30% impulse noise corruption, NVMF still performed better with PSNRs of 32.97dB and 32.51dB while using RWASF, which obtained the best PSNR

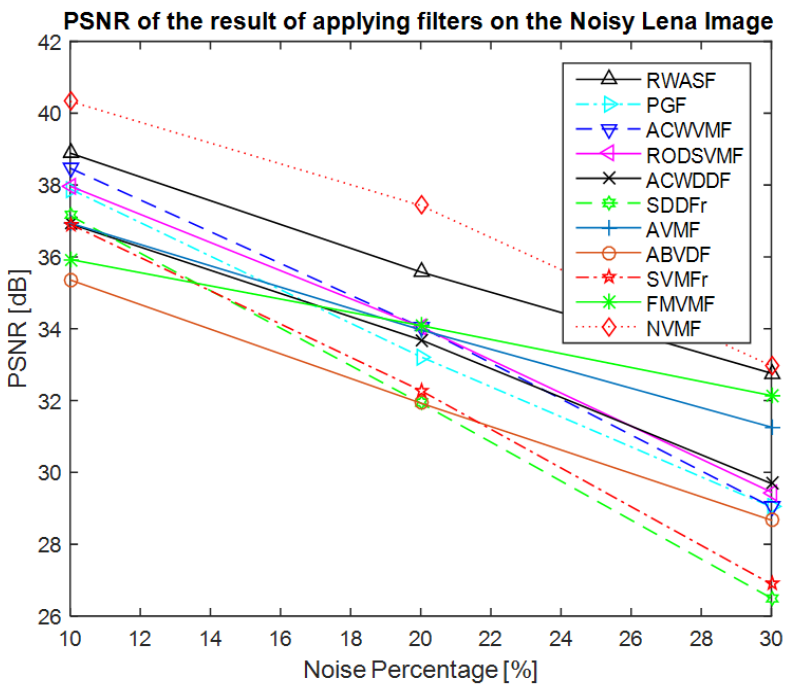

FIGURE 10. Graphical representation of the PSNR of the filters on the Lena image

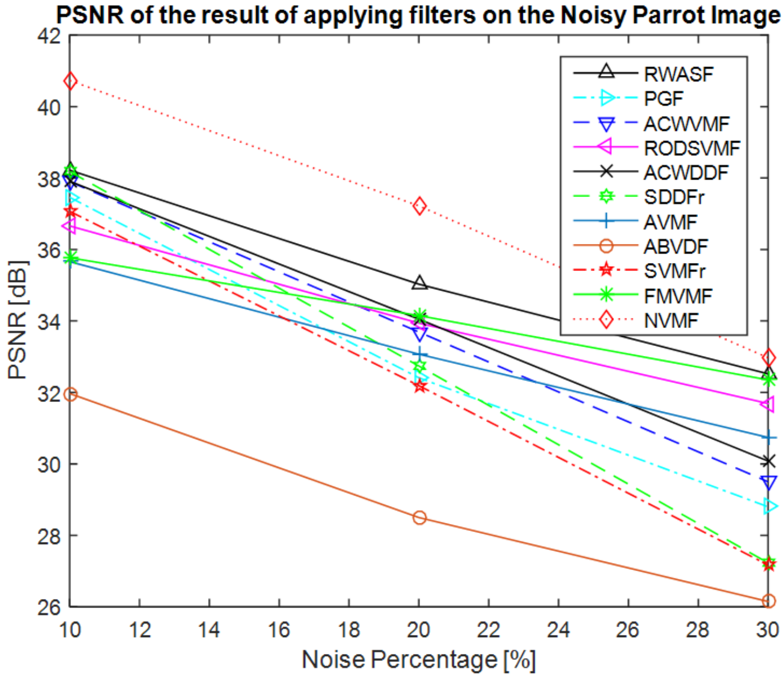

FIGURE 11. Graphical representation of the PSNR of the filters on the Parrot image

of 32.76dB among the filters considered, and the adaptive basic vector directional filter and SVMFr obtained PSNRs of 28.67dB and 26.89dB respectively.

The bigger the peak signal-to-noise ratio is, the better the performance of the filter is. Based on that our proposed method outperforms the best state-of-the-art filter by a 2% quality in most cases. In some cases where the state-of-the-art filter performs extremely well, the NVMF still outperforms it but with a small percent margin in quality.

5. **Conclusion.** The NVMF has shown, from the outcome of the filtering in the subjective analysis, its efficiency in detecting distortions/impulses in the individual RGB color components and how efficient it is in the correction of these distortions. The undistorted correlation of the RGB vector components in the color image is not tampered. The efficiency of these filters in identifying the distortion in the RGB components is engineered from the outcome of the simulations performed on the standard images. The results from the subjective analysis show from a visual point of view that a greater percentage of the impulse noise induced in the image is detected and filtered out. Also, bright and edge pixels are as well detected and untouched. This is evident when the filtered image is compared to the noisy image and the original image. The proposed filter has the ability to detect noise in the color image and replace it with the most optimal pixel value in regions with high intensities of impulse noise. This processing is to reduce the blurring problem encountered by the state-of-the-art family of vector filters. More so, it has the tendency to detect impulse noise corrupted pixels from bright image pixels and edge pixels in brighter image regions. Moreover, it is to replace the noisy pixels, and preserve the edge pixels and the uncorrupted bright pixels. Furthermore, it is to detect the distortions/impulses in the individual RGB color components and correct them as well as to preserve the undistorted correlation of the RGB vector components in the color image.

The objective evaluation is a quantitative analysis of the results obtained. The peak signal-to-noise ratios of some state-of-the-art filtering methods are compared to the PSNR of the novel vector median filters. The results of the PSNR show the filtering effectiveness of the intelligent switching novel vector median filter.

REFERENCES

[1] C. Fernandez-Maloigne, *Advanced Color Image Processing and Analysis*, 1st Edition, Springer-Verlag, New York, 2013.

[2] L. W. MacDonald and C. L. Mayne, *Color Image Processing*, Google Patents, 1999.

[3] K. N. Plataniotis and A. N. Venetsanopoulos, *Color Image Processing and Applications*, Springer-Verlag Berlin Heidelberg, New York, 2013.

[4] J. Astola, P. Haavisto and Y. Neuvo, Vector median filters, *Proc. of the IEEE*, vol.78, pp.678-689, 1990.

[5] V. D. Nguyen, D. T. Nguyen, T. D. Nguyen, V. A. Phan and Q. D. Truong, Filter-based feature selection and support vector machine for false positive reduction in computer-aided mass detection in mammograms, *Proc. of SPIE, the 7th International Conference on Machine Vision*, vol.9445, pp.94451H-1-94451H-5, 2015.

[6] S. Morillas, Fuzzy metrics and peer groups for impulsive noise reduction in color images, *The 14th European Signal Processing Conference*, pp.1-5, 2006.

[7] L. Jin and D. Li, Improved directional-distance filter, *Frontiers of Mechanical Engineering in China*, vol.3, pp.205-211, 2008.

[8] D. A. Konstantinos, N. Plataniotis and A. N. Venetsanopoulos, Color image filters: The vector directional approach, *Optical Engineering*, vol.36, pp.2375-2383, 1997.

[9] L. Khriji and M. Gabbouj, Vector median-rational hybrid filters for multichannel image processing, *IEEE Signal Processing Letters*, vol.6, pp.186-190, 1999.

[10] L. Alparone, M. Barni, F. Bartolini and V. Cappellini, Adaptively weighted vector-median filters for motion-fields smoothing, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp.2267-2270, 1996.

[11] N. K. Chaitanya and P. Sreenivasulu, Removal of salt and pepper noise using advanced modified decision based unsymmetric trimmed median filter, *International Conference on Electronics and Communication Systems*, pp.1-4, 2014.

[12] B. Smolka, K. Malik and D. Malik, Adaptive rank weighted switching filter for impulsive noise removal in color images, *Journal of Real-Time Image Processing*, vol.10, pp.289-311, 2015.

[13] R. Lukac, Adaptive color image filtering based on center-weighted vector directional filters, *Multidimensional Systems and Signal Processing*, vol.15, pp.169-196, 2004.

[14] Z. Ma, H. R. Wu and D. Feng, Fuzzy vector partition filtering technique for color image restoration, *Comput. Vis. Image Underst.*, vol.107, pp.26-37, 2007.

[15] S. Morillas, V. Gregori and A. Sapena, Adaptive marginal median filter for colour images, *Sensors*, vol.11, pp.3205-3213, 2011.

[16] K. M. Singh, Vector median filter based on non-causal linear prediction for detection of impulse noise from images, *Int. J. Comput. Sci. Eng.*, vol.7, pp.345-355, 2012.

[17] M. E. Celebi, Robust switching vector median filter for impulsive noise removal, *Journal of Electronic Imaging*, vol.17, pp.043006-1-043006-9, 2008.

[18] S. Patnaik, A. K. Samantaray and P. Mallick, Decision based adaptive neighborhood median filter, *International Conference on Computer, Communication and Convergence (ICCC 2015), Procedia Computer Science*, vol.48, pp.222-227, 2015.

[19] G. M. Daiyan, M. A. Mottalib and M. M. Rahman, High performance decision based median filter for salt and pepper noise removal in images, *The 15th International Conference on Computer and Information Technology (ICCIT)*, pp.107-112, 2012.

[20] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters Principles and Applications*, Kluwer Academic Publishers, 1990.

[21] S. Esakkirajan, T. Veerakumar, A. N. Subramanyam and C. H. PremChand, Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter, *IEEE Signal Processing Letters*, vol.18, pp.287-290, 2011.

[22] P. D. Shenoy, S. S. Iyengar, K. B. Raja et al., A decision based unsymmetrical trimmed modified winsorized mean filter for the removal of high density salt and pepper noise in images and videos, *The 11th International Conference on Communication Networks, ICCN 2015, the 11th International Conference on Data Mining and Warehousing, ICDMW 2015, the 11th International Conference on*

Image and Signal Processing, ICISP 2015, Procedia Computer Science, Bangalore, India, vol.54, pp.595-604, 2015.

[23] B. Smolka, Rank order weighted vector median filter, *The 17th International Conference on Systems, Signals and Image Processing*, pp.252-255, 2010.

[24] P. Kaur and M. Kaur, Decision based trimmed adaptive windows median filter, *International Journal of Science and Research (IJSR)*, vol.6, pp.40-43, 2013.

[25] B. Smolka, Adaptive rank based impulsive noise reduction in color images, *The 4th International Conference on Communications and Electronics (ICCE)*, pp.355-359, 2012.

[26] M. Sarkar and T. Y. Leong, Application of K-nearest neighbors algorithm on breast cancer diagnosis problem, *Proc. of the AMIA Symposium*, pp.759-763, 2000.

[27] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda et al., Top 10 algorithms in data mining, *Knowledge and Information Systems*, vol.14, pp.1-37, 2008.