International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

A Summary on Various Impulse Noise Removal Techniques

Keisham Pritamdas[1], Kh. Manglem Singh[2], L. Lolitkumar Singh[3]

[1]Department of Electronics & Communication Engineering, National Institute of TechnologyManipur

[2]Department of Computer Science & Engineering Engineering, National Institute of Technology Manipur

[3]Department of Electronics & Communication Engineering, Mizoram University, AiZawl, Mizoram

Abstract: ***This paper represents a comprehensive study of the various nonlinear filters used for impulse noise removal from color images. Depending on the working model of their algorithms, a classification of the filters into three broad catagories is shown, where each of the category is further devided into various sub groups. Each of the filter is properly formulated and detailed.***

Keywords: adaptive, cumulative,entropy, impulse noise, variance, Vector

1. Introduction

The application of image processing is very diverse in nature and is growing day by day which includes medical field, remote sensing, transmission and coding, video processing, microscopic imaging etc. For any efficient image processing application an image must contain the required data to show the correct information. But the images are generally deprived of the correct information due to the influence of external unwanted ingredients called noise. Filtering is one of the most essential steps in the applications of image processing. These unwanted information which are termed as noise must be removed properly from the image as a preprocessing step. Some of the most common type of noise is additive random noise (Gaussian noise) and salt and pepper noise. Impulse noise also called as salt and pepper noise which may be fixed valued noise (FVN) or random valued noise (RVN) is one of the most naturally occurring noises in digital images and it is induced in the image during image acquisition by faulty sensors or during transmission through communication channel. Depending on the types of noises and the percentage of noise present in the image, both the noise detection and the noise removal algorithms can be varied. A number of robust filters have been proposed in literature for filtering the color images corrupted with impulse noise. The most suitable filters which work in spatial domain are the non-linear filters [1].In this survey, a large number of nonlinear filters are broadly categorized into 3 families such that the first and second category is further divided into three and eight sub-groups respectively.

The current developments in vector median filters for impulse noise removal from color images are detailed and reviewed in this survey paper. Sections II demonstrates the categories and their various sub-divisions of vector filters. In Section III, a generally used impulse noise model is described. Section IV describes the various performance measuring criteria of the vector filters and finally conclusion is given in section V.

2. Classification of Filters

The filters are broadly classified into three categories namely adaptive-switching vector filters, non-adaptive switching filters and miscellaneous filters. Then each category is divided into various groups, such that each group is further divided into many sub-groups.

A. Non Adaptive-switching Vector Filters
- Basic Vector Filters
- Weighted Vector Filters
- Fuzzy Vector Filters

B. Adaptive-Switching Vector filters
- VMF Based on Non Causal Linear prediction
- Adaptive Weighted Vector Filters
- Peer Group Vector Filters
- Hybrid Vector Filters
- Vector Sigma Filters
- Entropy Vector Filters
- Rank-conditioned Vector Filters

C. Miscellaneous Filters

A. Non Adaptive-switching Vector Filters

This group of filters replaces the center pixel with the output of a vector filter without using a noise detection algorithm for checking whether the center pixel is noisy or not.

Basic Vector Filters

This group of filters used the concept of reduced ordering of the vectors in a sliding window, depending on their respective cumulative distance from the surrounding vectors. Let $\boldsymbol{x}_i$ representsthe vector pixel in a window W of size$m \times n$, where i goes from 1 to $m \times n$, whose corresponding cumulative distance is calculated as

$$CD(i) = \sum_{j=1}^{m\times n} d(\boldsymbol{x}_i, \boldsymbol{x}_j), i = 1,2, \dots \dots, m \times n \quad (1)$$

Where $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ represents the differenceor divergence function between the vector pixels $\boldsymbol{x}_i = (x_{iR}, x_{iG}, x_{iB})$ and $\boldsymbol{x}_j = (x_{jR}, x_{jG}, x_{jB})$, for R-red, G-green and B-blue components. The reduced ordering of the vector pixels in the window is derived from the ascending order of their respective accumulative distance as

$$CD(1) \le CD(2) \le \cdots \le CD(m \times n) \Rightarrow \boldsymbol{x}_{(1)} \le \boldsymbol{x}_{(2)} \le \cdots \le \boldsymbol{x}_{(m \times n)} \quad (2)$$

Vector Median Filter (VMF)

In [2] Vector Median filter (VMF), Generalized Vector Median Filter(GVMF) and ExtendedVector Median Filter(EVMF)work with the concept of nonlinear order statics where the output is the lowest ranked vector in the sliding window. Impulse noise usually has very high or very small intensity value as compared to the surrounding vector pixels, that makes sense that vector pixel which vary highly or having high cumulative difference from the surrounding vector pixels tends to be more impulse noise. Thus the lowest ranked vector or the vector which gives the least corresponding cumulative distance calculated based on Minkowski metric is the output of the VMF. They are based on the concept of nonlinear order statistics and derived as maximum likelihood estimates from exponential distributions.If $\boldsymbol{x}_1, \dots, \boldsymbol{x}_{m \times n}$ represent the vector inside the filtering window W, the vector median is computed as follows:

i) The corresponding magnitudinal cumulative distance $m(i) = CD(i)$for each vector element $\boldsymbol{x}_i$is calculated with the help of equation 1,using the Minkowski metric (either the L_1or L_2 norm) where

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_\mu \quad (3)$$

where $\mu = 1$ defines the city block distance and $\mu = 2$ gives the Euclidean distance.

ii) Then the cumulative distance associated with $x_i, m(i) =$

$$\sum_{j=1}^{m \times n} d(\boldsymbol{x}_i, \boldsymbol{x}_j) \; i = 1,2,3, \dots m \times n \quad (4)$$

are sorted in ascending order so that the corresponding element $\boldsymbol{x}_i$ which gives the least cumulative distance $m(i)_{minimun} = m(1)$, replaces the center vector pixel as the VMF output.The VMF properly smoothes noise present in the image while steel preserving the fine details and the edges in the image.

α-trimmed Vector Median Filter(α-VMF)

The trimmed vector median filters α-VMF considers the lowest ranked 1+α vectors from the window W as described in equation 2, as an input to an averaging filter. The output is defined as follows [2, 3]:

$$\boldsymbol{x}_{\alpha\mathrm{VMF}} = \sum_{i=1}^{1+\alpha} \frac{1}{(1+\alpha)} \boldsymbol{x}_i, \quad \alpha \in [0, (m \times n) - 1] \quad (5)$$

Impulse noise is removed efficiently with the trimming operation and the averaging function also helps in the removal of Gaussian noise.

Generalized Vector Median Filter (GVMF)

Let $\boldsymbol{x}_i, \dots \boldsymbol{x}_{m \times n}$ be the vector pixels present in the window$W$of size$m \times n$. If $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_\mu$ which is described in equation 3, then the output of GVMF [4], $\boldsymbol{x}_{GVMF} \in \{\boldsymbol{x}_i \mid i = 1, \dots, m \times n\}$ for $j = 1, \dots, m \times n$ is the vector pixel which satisfies the condition

$$\sum_{i=1}^{m \times n} d(\boldsymbol{x}_j, \boldsymbol{x}_i) \ge \sum_{i=1}^{m \times n} d(\boldsymbol{x}_{GVMF}, \boldsymbol{x}_i) \quad (6)$$

Directional Vector Median Filter (DVMF)

This particular filter [5] works in two stages. In the first stage VMF algorithm is applied in the four directions of the sliding window$W$, namely $\theta, \theta + \frac{\pi}{4}, \theta + \frac{\pi}{2}, \theta + \frac{3\pi}{4}$ to give the corresponding output as $\boldsymbol{x}_{VMF1}, \boldsymbol{x}_{VMF2}, \boldsymbol{x}_{VMF3}and\boldsymbol{x}_{VMF4}$. Then in the next stage the four VMF outputs are further given as inputs to the VMF to give the final output as $\boldsymbol{x}_{DVMF} = \boldsymbol{x}_{VMF}$, which is the vector median of $\boldsymbol{x}_{VMF1}, \boldsymbol{x}_{VMF2}, \boldsymbol{x}_{VMF3}$ and $\boldsymbol{x}_{VMF4}$. It is still efficient in preserving the fine details while removing the impulse noise.

Crossing Level Median Mean Filter (CLMMF)

This filter [6] works on the concept of VMF and arithmetic mean filter (AMF). As in the case of VMF the vector pixels are firstly ordered according to their cumulative distances of magnitudes as described in equation 1 and 2. Then depending on the order, each pixel $\boldsymbol{x}_i$is given a weight as

$$w(i) = \begin{cases} 1 - \dfrac{m \times n}{\sqrt{((m \times n)+1)((m \times n)+1+\partial)}}, \text{for } i = 1 \\ \dfrac{1}{\sqrt{((m \times n)+1)((m \times n)+1+\partial)}}, \text{for } i = 2,3, \dots, m \times n \end{cases} \quad (7)$$

Where $\partial$is the parameter to tune the amount of weight given to the pixels, when ∂tends to ∞ and 0, the filter takes the form of VMF and AMF respectively.

Finally using the concept of AMF, the output of the CLMMF is given as

$$\boldsymbol{x}_{CLMMF} = \sum_{i=1}^{m \times n} w(i) \boldsymbol{x}_i \quad (8)$$

Vector Directional Filter (VDF)

As the pixels in a color image are considered vectors, the difference or dissimilarity between two vector pixels can be measured in terms of magnitudinal distance or in terms of directional distance. The VMF considers the magnitudinal distance to form the ordered set of pixels. Whereas the VDF [7] considers the directional distance between two vector pixels x_i and x_j as

$$D(\boldsymbol{x}_i, \boldsymbol{x}_j) = \cos^{-1}\left(\frac{\boldsymbol{x}_i \boldsymbol{x}_j}{\|\boldsymbol{x}_i\| \|\boldsymbol{x}_j\|}\right) \quad (9)$$

Then the cumulative directional distance for each vector pixel is given as

$$\emptyset(i) = \sum_{j=1}^{m \times n} D(\boldsymbol{x}_i, \boldsymbol{x}_j), \; i = 1,2,3 \dots\dots, m \times n \quad (10)$$

To form the rank ordered set of the pixels in the window as

$$\emptyset(1) \le \emptyset(2) \le \cdots \emptyset(k) \dots \le \emptyset(m \times n) \rightarrow \boldsymbol{x}_1 \le \boldsymbol{x}_2 \le \cdots \boldsymbol{x}_k \dots \le \boldsymbol{x}_{m \times n}$$

Then the output of the VDF is $\boldsymbol{x}_1$which gives the minimum cumulative angular distance as compared to other vector pixels in the window.As the color chromaticity of a color image is defined by the vector's direction, the VDF is able to preserve the chromaticity better than the VMF.

Generalized Vector Directional Filter (GVDF)

This filter [8] considers both the aspects of a vector i.e. direction and magnitude, correspondingly in two stages. In the first stage the vectors are ordered and ranked depending on their cumulative angular distance as in the case of VDF, from which a set of low-ranked vectors are picked up as input to an additional filter to produce the final GVDF output, which forms the second stage. The additional filter can be any grayscale filter where only the magnitude concept is used, like AMF, the multistage median filter and some other morphological filters.

Directional Distance Filter (DDF)

The DDF [8,9] considers both the magnitude and the direction simultaneously, in calculating the difference or dissimilarity between the vector pixels.The cumulative distance for a vector pixel$x_i$is given as

$$\varphi(i) = m(i)^{\delta}\emptyset(i)^{1-\delta} \quad (11)$$

Where $m(i)$ and $\emptyset(i)$ are described in equations 1 and 10 respectively.And$\delta \in (0,1)$, is the parameter for adjusting the weight assigned to the intensityand the chromaticity components, in calculating the overall cumulative distance of a vector pixel with respect to its surrounding pixels in the sliding window. Therefore a vector pixel which is having the least magnitudinal and angular difference from its surrounding will be treated as the output of DDF.

Weighted Vector Filters

Weighted Vector Filters are extension of Weighted Standard Median Filters in which a non-negative weight is assigned to every pixel inside the filtering window offering more flexibility.

Weighted Vector Median Filter (WVMF)

The VMF [3, 10] is further generalized and made more flexible by assigning a non negative integer-valued weight to each pixel, during the calculation of the accumulative distance as given

$$CD_{WVMF}(i) = \sum_{j=1}^{m\times n} w(j)d\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right), i = 1,2,\ldots\ldots.., m\times n \quad (12)$$

Where $w(j)$ is the weight assigned to x_j and $d\left(x_i, x_j\right)$ is described in equation 3. Then the corresponding vector pixel which gives the least value of the cumulative distance is the output of the WVMF denoted as $\boldsymbol{x}_{WVMF}$.

α-Trimmed Weighted Vector Median Filter (α-TWVMF)

The output of α-TWVMF [10] of vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$with the corresponding weights as $w(1), w(2), w(3) \ldots. w(m\times n)$ is defined as

$$\boldsymbol{x}_{\alpha-TWVMF} = \begin{cases} \boldsymbol{x}_{\alpha}, \text{if} \sum_{i=1}^{m\times n} w(i)d(\boldsymbol{x}_{\alpha}, \boldsymbol{x}_i) < \sum_{i=1}^{m\times n} w(i)d(x_{VMF}, x_i) \\ \boldsymbol{x}_{WVM}, \quad \text{otherwise} \end{cases} \quad (13)$$

where$\boldsymbol{x}_{\alpha} = \frac{1}{|S_{\alpha}|}\sum_{x_i\in S_{\alpha}} \boldsymbol{x}_i$ and $S_{\alpha} = \{\boldsymbol{x}_i; \text{having } S_i < S m\times n) - \alpha$is the set of vector pixels whose corresponding weighted cumulative distance to other pixels in the window is less than a predefined threshold $S((m\times n))$ is the weighted cumulative distance of the vector $\boldsymbol{x}_i$ to all other vectors $\boldsymbol{x}_j, j = 1, \ldots, N$. $|S_{\alpha}|$ represents the number of elements in S_{α} and $S_{(i)}$ is the i^{th} smallest of $S_1, \ldots S_N$. α can have any value 0, 1, ………., *N*-1.

Rank Weighted Vector Median Filter (RWVMF)

In this filter [11] the distance of a pixel x_i to the other pixel $\boldsymbol{x}_j$, $a(i) = d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ for $i\&j = 1,2,3 \ldots. m\times n$ is calculated to be assigned a rank r to give $a_r(i)$ as $r = 1,2, \ldots. m\times n$. Then a cumulative distance for each pixel is calculated by assigning a weight $w(r)$that depends on the rank r, as

$$A(i) = \sum_{r=1}^{m\times n} w(r).a_r(i) \quad (14)$$

Then the output of the ROWVMF [23] is the corresponding vector pixel that gives the least value of cumulative distance, which can be expressed as$\boldsymbol{x}_{RWVMF} = argmin_{x_i\in W}\sum_{r=1}^{m\times n} w(r).a_r(i)$.

Another filter called Rank-based Vector Median Filter having similar concept is also designed in [12].

Sharpening Vector Median Filter (SVMF)

As seen in the case of RWVMF, the vector pixel which corresponds to the minimum value of the rank-weighted cumulative is the RWVMF output,

$$\boldsymbol{x}_{RWVMF} = argmin_{x_i\in W}\sum_{r=1}^{m\times n} w(r).a_r(i) \quad (15)$$

If a constant function is used as the weighting function as $w(r) = 1, r = 1,2, \ldots. m\times n$, we obtain $A(i) = m(i)$and$\boldsymbol{x}_{RWVMF} = \boldsymbol{x}_{VMF}$, where$m(i) = \sum_{r=1}^{m\times n} d(\boldsymbol{x}_i, \boldsymbol{x}_j)$, is the cumulative distance associated with$\boldsymbol{x}_i$which is explained in equation 3 and 4.Then for a step-like function

$$w(r) = \begin{cases} 1, for\ r \le \alpha, \alpha \le m\times n \\ 0, \quad otherwise \end{cases} \quad (16)$$

The resultant filter is the SVMF [13]. It can also be seen that using monotonously decreasing function, like $w(r) = {}^{1}/_{r}$ and $w(r) = {}^{1}/_{r^2}$ [14-16], results to very effective denoising.

Weighted Vector Directional Filters (WVDF)

As in the case of WVMF, the WVDF [17, 18] also assigns a non negative real valued weight for each vector pixel and thus the weighted cumulative angular distance is given as

$$\emptyset_{WVDF}(i) = \sum_{j=1}^{m\times n} w(j)D\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right), for\ i = 1,2,3 \ldots.., m\times n \quad (17)$$

Where $D\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$is already described in equation 9 and the center pixel is replaced with $x_i \in W$,denoted as x_{WVDF} which corresponds to the minimum$\emptyset_{WVDF}(i)$.

Similarly Weighted Directional Distance Filter (WDDF) is also obtained using both the magnitude and angular distance criteria.

Center-Weighted Vector Median Filter (CWVMF)

It [19, 20] is a special case of WVMF where the weight is assigned only to the center pixel as given below

$$w_j(l) = \begin{cases} (m\times n) - 2l + 2, \text{for } j = ((m\times n) + 1)/2 \\ 1, \quad \text{otherwise} \end{cases} \quad (18)$$

To give output of the CWVMF as

$$\boldsymbol{x}_{CWVMF}\, l = \text{argmin}_{x_i\in W}\left(\sum_{j=1}^{m\times n} w_j(l).d(\boldsymbol{x}_i, \boldsymbol{x}_j)\right) \quad (19)$$

Where $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the magnitudinal distance between $\boldsymbol{x}_i$and $\boldsymbol{x}_j$ for $i$and $j = 1,2, \ldots. m\times n$ and l is the parameter used for tuning the weight given to the center pixel $\boldsymbol{x}_c$.

Fuzzy Vector Filters

In various areas of engineering where there are cases of uncertainty and imprecision to dealt with, fuzzy systems [21] have been very efficient in finding the accurate solutions. Therefore if appropriate network topologies are provided and proper processing strategies are adopted Fuzzy systems are smart enough to overcome the uncertainty faced by the filters in the impulse noise removal from color images. A number of fuzzy techniques adopt a window-based, rule-driven approach leading to data-dependent fuzzy filters, which are constructed by fuzzy rules in order to remove impulse noise while preserving important image characteristics, such as edges and fine details. Local correlation in the data is utilized by applying the fuzzy rules directly on the pixel elements which lie within the operational window. Through the utilization of linguistic terms, a fuzzy rule-based approach to image processing allows for the incorporation of human knowledge and intuition into the design, which cannot be achieved via traditional mathematical modeling techniques.

Adaptive Fuzzy Filters (ADF)

Considering a window W consisting of vector pixels $\boldsymbol{x}_i$ for $i = 1,2 \ldots m \times n$, centered at $\boldsymbol{x}_c$ and performing an averaging operation on the vectors in the window for the center pixel to be replaced with a suitable output, resulting in smoothing is considered to be an efficient solution for random noise removal from an image. Therefore fuzzy weighted average filter (FWAF) [22-27], is one of the general forms of adaptive fuzzy filters, where each of the vector pixels is given a weight by an adaptive weighting function $w(\varphi_i) = \varphi_i^{\alpha}$, with φ_i the adaptive membership function for $x_i$determined based on the local context of the neighboring pixels and α a parameter to tune the weight such that $\alpha \epsilon [0, \propto)$. Then the output of the weighted average filter is given by the centroid as

$$\boldsymbol{x}_{FWAF} = \frac{\sum_{i=1}^{m \times n} w(\varphi_i)\boldsymbol{x}_i}{\sum_{i=1}^{m \times n} w(\varphi_i)} \quad (20)$$

Abiding by the normalization procedure, two constraints are necessary to make it sure that the output of FWAF is an unbiased one, namely: a) each weight $y_i = \frac{w(\varphi_i)}{\sum_{i=1}^{m \times n} w(\varphi_i)}$ for a respective vector pixel $\boldsymbol{x}_i$is a positive number,$y_i \geq 0$& b) $\sum_{i=1}^{m \times n} y_i = 1$.

Image Rotation and Fuzzy processing coupled with Recursive Noise Filters.

This approach [28] is based on the fact that the output of a recursive image filter varies depending on the orientation of the input image. The whole filtering approach is completed in the following steps:

a) Generation of four input images by rotating the noisy input image at integer multiples of90°.
b) Obtaining four corresponding restored images after processing the four input images by the specified noise filter.
c) Computation of the final enhanced output image from the filtered images by using a fuzzy system.

The fuzzy system used in this approach is a first order Sugeno type fuzzy system with four inputs and one output [29] where each input has two membership functions.

Fuzzy Vector Median Filter (FVMF)

In this filter [21, 22] each of the vector pixels in the window w is given a fuzzy weight by using a membership function [30] which is of exponential equation as

$$\varphi_i = \exp(-\frac{l(i)^{\alpha}}{\beta}) \quad (21)$$

Where $l(i) = CD(i) = \sum_{j=1}^{m \times n} \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_{\mu}$ represents the cumulative variation distance corresponding to $\boldsymbol{x}_i$, which may in L_1or L_2norm of the Minkowski metric for $\mu = 1$or $\mu = 2$ respectively. α and $\beta$are the parameters used to tune the weight given to each pixel such that $\beta$is the maximum value of α at which the membership function takes the maximum derivative.

Fuzzy Vector Directional Filter (FVDF)

In this filter the cumulative variation distance for a vector pixel is given by the vector angle metric $d(i) = \sum_{j=1}^{m \times n} cos^{-1}\left(\frac{\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle}{\|\boldsymbol{x}_i\| . \|\boldsymbol{x}_j\|}\right)$ and the fuzzy membership function has a sigmoidal form [21, 22] as

$$\varphi_i = \frac{\beta}{(1+\exp(d(i)))^{\alpha}} \quad (22)$$

Fuzzy Ordered Vector Filter (FOVF)

Fuzzy ordered vector filters [22, 31] are the generalization of α- trimmed filters such that only the $k$elements from the rank-ordered set of the vector pixels, with the highest fuzzy membership strength are given as input to the FWAF as described in equation 20. The fuzzy membership function $\varphi_i$giving a weight of $w(\varphi_i)$to each vector pixel $x_i$can be of exponential or sigmoidal form that results in fuzzy ordered VMF (FOVMF) or fuzzy ordered VDF (FOVDF) respectively. The k elements are ordered as $\boldsymbol{x}_k \leq \boldsymbol{x}_{k-1} \leq \ldots \ldots \boldsymbol{x}_1$ for their corresponding weights as $w(\varphi_k) \leq w(\varphi_{k-1}) \leq \cdots w(\varphi_1)$.

Fuzzy Hybrid Filter (FHF)

The FHF [6,32] firstly perform a nonlinear transformation on the sliding window$W$, resulting to a new set of vector pixels $W^{new} = \{\boldsymbol{x}_i^{new}; i = 1,2, \ldots . m \times n\}$ where the pixel with the least and highest intensity levels is replaced by the VMF output of the window$W$. Then each of the pixel in the new window is given a fuzzy weight using a membership function $\varphi_i = f(d(\boldsymbol{x}_i^{new}, \boldsymbol{x}_{VMF}))$ which is a function of the difference of the respective pixel to that the x_{VMF} such that the center pixel is finally replaced with the weighted average of the pixels of the new window as $x_{FHF} = \frac{\sum_{i=1}^{m \times n} \varphi_i \boldsymbol{x}_i^{new}}{\sum_{i=1}^{m \times n} \varphi_i}$ (23)

Adaptive Nearest-Neighbor Filter (ANNF)

In this filter each vector pixel $x_i$in the window $W$is given a weight $w(\varphi_i)$using the fuzzy membership function

$$\varphi_i = \frac{b(N) - b(i)}{b(N) - b(1)} \text{ for } i = 1,2 \ldots m \times n \quad (24)$$

Where $b(i)$ can be the cumulative distance of Minkowski, angular or directional-distance metric, such that $b(N)$and $b(i)$ are the maximum and minimum cumulative values correspondingly [26,33].

Adaptive Nearest-Neighbor Multichannel Filter (ANNMF)

It [34] is a modification over ANNF that uses a composite distance function rather than a magnitude or an angular metric, to measure the variation distance from a vector pixel x_i to another onex_j, which is expressed as

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \left(\frac{\langle x_i . x_j \rangle}{\|x_i\| . \|x_j\|}\right)\left(1 - \frac{\|x_i\| - \|x_j\|}{max\,(\|x_i\| . \|x_j\|)}\right) (25)$$

for $i \& j = 1,2 \ldots m \times n$

The first component represents the angular part whereas the second one represents the normalized magnitude measures such that when the vectors have equal intensity, only the angular component is considered.

B. Adaptive-Switching Vector Filters

This group of filters uses a noise detection algorithm to check whether the center pixel is noisy or not, prior to replacement of the center pixel with the output of a vector filter. It actually works by switching between the center pixel and the output of a vector filter.

Vector Filters Based on Non-Causal (NC) linear Prediction Technique

Depending on the concept that a strong correlation occurs among the neighborhood pixels of a certain region of an image centered at a vector pixel, the non-causal linear predictor based filters execute based on theconcept of linear prediction [35, 36] for thepredictionof the center pixel as a weighted combinationof pastand future vector pixels with respect to the center pixel.

c (5th) nc x(m-2,n-2)	c (4th) nc x(m-2,n-1)	c (3rd) nc x(m-2,n)	c (4th) nc x(m-2,n+1)	c (5th) nc x(m-2,n+2)
c (4th) nc x(m-1,n-2)	c (2nd) nc x(m-1,n-1)	c (1st) nc x(m-1,n)	c (2nd) nc x(m-1,n+1)	c (4th) nc x(m-1,n+2)
c (3rd) nc x(m,n-2)	c (1st) nc x(m,n-1)	nc x(m,n)	(1st) nc x(m,n+1)	(3rd) nc x(m,n+2)
(4th) nc x(m+1,n-2)	(2nd) nc x(m+1,n-1)	(1st) nc x(m+1,n)	(2nd) nc x(m+1,n+1)	(4th) nc x(m+1,n+2)
(5th) nc x(m+2,n-2)	(4th) nc x(m+2,n-1)	(3rd) nc x(m+2,n)	(4th) nc x(m+2,n+1)	(5th) nc x(m+2,n+2)

Figure 1: Block of causal and non causal regions showing different orders

Considering an analysis block of$p \times q$ centered at the vector pixel$\boldsymbol{x}(p,q)$, the group of non-causal vector median filters $NCVMF\_p \times q$ [37] first filters the image using VMF, then estimates the center pixel using constrained intra-channel linear predictor [38] given by the equation $\boldsymbol{x}_{est}(p,q) = X_{\eta\alpha_\eta} = \sum_{(i,j)\epsilon w_{nc}} \alpha(i,j)\boldsymbol{x}(p-i,q-j)$where X_η is a matrix of$C$rows and D columns , $D$is the number of vector pixels which will be used in the estimation,$\alpha_\eta$is the vector matrix formed by the prediction coefficients andw_{nc} is the non-causal region of support for the linear predictor and$\eta$is the order of prediction. And the prediction coefficients α_η is predicted using equation $R_{X_\eta}\alpha_\eta = r_{X_\eta}$, where R_{X_η} is the autocorrelation matrix of the input vector X_η and r_{X_η} is the cross-correlation of the input vector and the center vector pixel$\boldsymbol{x}(p,q)$. This group of filters uses 13 elements ofR_{X_η}, whose coordinates are described as $(p,q),(p,q-2),(p-$ *1.q+2,p−1,q−1,p−1,q,p−1,q−2,p−2,q−1,p−1,q+3,p−2, q,p,q−1,p−2,q+1,p−2,q−2and (p−2,q+2)* for calculating the values ofα_η. Then the $NCVMF_4 \times 4$estimates the center pixel $x_{est}(p,q)$ by considering a block size of 4×4 and the eight second order pixels (see Figure1) [39- 41]. Similarly the$NCVMF_8 \times 8$,$NCVMF\_16 \times 16$, $NCVMF_32 \times 32$and $NCVMF\_64 \times 64$ use block size of 8, 16, 32 and 64 respectively. Then if the difference between the estimated pixel and the center pixel is greater than or equal to a user-specified threshold, the center pixel is replaced by the output of VMF.

Adaptive Weighted Vector Filter

In this group of filters, prior to the detection algorithm the vector pixels are given weight by an adaptive function according to the characteristics of the pixel with respect to the local surrounding pixels in the sliding window.

Adaptive Center-Weighted Vector Filters

It is a very robust filter which is very efficient and flexible in removing impulse noise, which actually allows one to design an optimal filter for a particular domain by adjusting the weights assigned to the center pixel.To provide more flexibility for modification in the size and shape of the window, adaptive center weighted vector filters are designed in [19,42and 43].The weights are estimated using an optimization procedure by using a number of training images [1], using the equation

$$w_j(l) = \begin{cases} (m \times n) - 2l + 2, & \text{for } j = ((m \times n) + 1)/2 \\ 1, & \text{otherwise} \end{cases}$$
(26)

To give the output of the adaptive center weighted vector median filter x_{ACWVMF} as

$$\boldsymbol{x}_{ACWVMF} = \begin{cases} \boldsymbol{x}_{VMF}, if \ \sum_{l=u}^{u+2} \left\| \boldsymbol{x}_{CWVMF}^{l} - \boldsymbol{x}_c \right\| \\ \boldsymbol{x}_c, \quad otherwise \end{cases} \quad (27)$$

x_{CWVMF}^{l} is already defined in equation 19.l is used as the smoothing parameter such that if $l = 1$ the ACWVMF becomes an identity filter and so no smoothing will be performed and if the value of$l$ increases from 1 to c, the smoothing potential of the filter increases. When$l$ reaches the value of c , lastly the ACWVMF becomes the VMF where the maximum amount of smoothing is done. The value of $l$is chosen from$u$ tu $+ 2$, for which three respective CWVMF outputs are determined to be subtracted from their respective center pixel as seen in equation 27. And if the sum of the three differences is greater than the threshold T then the center pixel is replaced with the output of VMF. In the ACWVMF the center pixel is highlighted or given more importance without a prior knowledge of whether it is a noisy pixel or not, by assigning a variable weight by equation (28), that makes the ACWVMF more suitable to be considered for lower noised region of the image.If $\boldsymbol{x}_{CWVMF}^{l}$ in equation 27 is replaced by $\boldsymbol{x}_{CWVDF}^{l}$ and $\boldsymbol{x}_{CWDDF}^{l}$, for the noisy center pixel to be replaced by $\boldsymbol{x}_{VDF}$ and $\boldsymbol{x}_{DDF}$ correspondingly, it results to two other related filters namely ACWVDF and ACWDDF respectively.

Adaptive Rank Weighted Switching Filter (ARWSF)

The ARWSF [44] is a modification of the (RWVMF) such that the center pixel is checked to be noisy or not by comparing the difference $\partial = A(c) - A(x_{VMF})$ with a threshold$T$. Then the VMF output replaces the center pixel if $\partial > T$otherwise the center pixel is kept unchanged, where $A(i)$ for a particular pixel x_i is the weighted cumulative distance as explained in equation 14.

Extended Weighted Vector Median Filter (EXWVMF)

This filter [10, 45] called the EXVWVMF works by replacing the center pixel with the output of the WAVF,x_{WAVF} if the cumulative intensity distance of x_{WAVF} is less than that of the cumulative distance of thex_{WVMF}, with that of other vector pixels in the sliding window. The weighted average filter is defined as $\boldsymbol{x}_{WAVF} = \frac{\sum_{i=1}^{m \times n} w(i)\boldsymbol{x}_i}{\sum_{i=1}^{m \times n} w(i)}$, where each weight $y_i = \frac{w(i)}{\sum_{i=1}^{m \times n} w(i)}$ for a respective vector pixel $\boldsymbol{x}_i$ is a positive number, $y_i \geq 0$ and $\sum_{i}^{m \times n} y_i = 1$. EWVMF replaces the center pixel with the$\boldsymbol{x}_{WAVF}$ in the smooth region of the image, whereas with thex_{WVMF} in the edgy region correspondingly.

Peer Group Vector Filters

This group of adaptive switching filters is based on the concept of Peer group where for each pixel in the window a peer group consisting of the neighborhood pixels which are very similar to it, according to a distance measurement criterion, is formed [46-48].

Peer Group Vector Median Filter (PGVMF)

In this filter according to the respective distance of the pixels to the center pixel, a ranked ordered set of the pixels is formed, from which a peer group of $p = \sqrt{(m \times n)} + 1/2$ pixels of the lowest ranked is formed as $g(i) = d(\boldsymbol{x}_c, \boldsymbol{x}_i) for\ i = 1,2,\ldots,p$. Then to further highlight the presence of impulse noise a first order differential operator is applied over $d(i)$ to form $\partial(i) = d(i+1) - d(i) for\ i = 1,2,\ldots,p$ such that if one of $\partial(i)$ is greater than a user-specified threshold T, then the center pixel is replaced with the VMF output otherwise it is left unaltered.

Fast Peer Group Filter (FPGF)

It is the faster version of the PGF where the center pixel $\boldsymbol{x}_c$ is replaced with the output of VMF, $\boldsymbol{x}_{VMF}$ as soon as peer group of m pixels in the window W are sufficiently similar to the center pixel as given by the equation

$$\boldsymbol{x}_{FPGF} = \begin{cases} \boldsymbol{x}_{VMF} & if\ |\{\boldsymbol{x}_{i \neq c} \in W\ s.t.\ \|\boldsymbol{x}_c - \boldsymbol{x}_i\| \leq T\}| < m \\ \boldsymbol{x}_c & otherwise \end{cases} \quad (28)$$

Hybrid Vector Filters

The hybrid filters give an output which is not one of the input vector pixels in the sliding window, since they process the noisy pixel by combining different types of sub filters which may be linear or non-linear filters.

Hybrid directional filter (HDF)

The HDF [49] accompanies the concept of vectorial aspect processing in the DDF. It can be defined as a non linear combination of the VMF and VDF which can be expressed as

$$\boldsymbol{x}_{HDF} = \begin{cases} \boldsymbol{x}_{VMF}\ if\ \boldsymbol{x}_{VMF} = \boldsymbol{x}_{BVDF} \\ \frac{\|\boldsymbol{x}_{VMF}\|}{\|\boldsymbol{x}_{BVDF}\|}.\boldsymbol{x}_{BVDF}\ otherwise \end{cases} \quad (29)$$

$\|\boldsymbol{x}_{VMF}\| = (x^R.x^R + x^G.x^G + x^B.x^B)^{1/2}$ representingthe R=red, G=green and B=blue components and similarly for $\|\boldsymbol{x}_{BVDF}\|$.

Vector Median Rational Hybrid Filter (VMRHF)

The VMRHF [50-52] combines three sub-filters namely two VMF outputs and one CWVMF output in a rational function that also accompanies an edge sensing aspect characterized by a weighted function of the Euclidean distance between the two VMF outputs.The rational function giving the output of VMRHF, $\boldsymbol{x}_{VMHRF}$ is expressed as

$$\boldsymbol{x}_{VMRHF} = \boldsymbol{x}_{CWVMF} + \frac{a_1.\boldsymbol{x}_{VMF_1} + a_2.\boldsymbol{x}_{CWVMF} + a_3.\boldsymbol{x}_{VMF_2}}{b_1 + b_2.\left\|\boldsymbol{x}_{VMF_1} - \boldsymbol{x}_{VMF_2}\right\|} \quad (30)$$

The weighting coefficient vector $a = [a_1 a_2 a_3]$ in the numerator is predefined and is used to tune the different filter outputs by maintaining the numerical stability, whereas the coefficients b_1 and b_2 in the denominator are used as positive constants for regulating the nonlinearity.The following 3×3 sized masks

$$VMF_1 = \begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{matrix}, VMF_2 = \begin{matrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{matrix} CWVMF = \begin{matrix} 1 & 1 & 1 \\ 1 & 3 & 1 \\ 1 & 1 & 1 \end{matrix}$$

are used for the three sub-filters respectively to the image, before the center pixel x_c is processed with the equation 34.

Fuzzy Rational Hybrid Filters

This group of filters [31, 53,54] is the modified versions of VMRHF. In the fuzzy VMRHF (FVMRHF)the sub-filters are replaced by their fuzzy forms namely fuzzy $VMF_1(FVMF_1)$, $VMF_2(FVMF_2)$ and CWVMF (FCWVMF),which are processed by the same rational function used in VMRHF as described below

$$\boldsymbol{x}_{FVMRHF} = \boldsymbol{x}_{FCWVMF} + \frac{a_1.\boldsymbol{x}_{FVMF_1} + a_2.\boldsymbol{x}_{FCWVMF} + a_3.\boldsymbol{x}_{FVMF_2}}{b_1 + b_2.\left\|\boldsymbol{x}_{FVMF_1} - \boldsymbol{x}_{FVMF_2}\right\|} \quad (31)$$

And the membership function used in these sub-filters for weighting the vector pixels in the sliding window is given as $\varphi_i = \exp(-\frac{l(i)^{\propto}}{\beta})$ which is already described in equation 21. The fuzzy vector directional-rational hybrid filter (FVDRHF) and the fuzzy directional distance-rational hybrid filter (FDDRHF) utilizes membership functions which are functions of angular and directional-distance metrics respectively.

Kernel Vector Median Filter

The kernel vector median filter (KVMF) [55-59]uses theequation

$$\boldsymbol{x}_{KVMF} = \partial(\|\boldsymbol{x}_c - \boldsymbol{x}_{VMF}\|).\boldsymbol{x}_c + (1 - \partial(\|\boldsymbol{x}_c - \boldsymbol{x}VMF.\boldsymbol{x}VMF \quad (32)$$

that combines output of VMF and center pixel linearly, where the weighting coefficients are determined by a kernel ∂. A Laplacian kernel is defined as $\partial(b) = exp\left(b/l\right)$ where the kernel width is defined as $l = \frac{\gamma}{\left(\sum_{i=1}^{m \times n}\|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\| / 8(m \times n)^2\right)^{1/2}}$.

The value of the normalization factor γ depends on the type of the kernel used which can be Gaussian, Cauchy or Epanechnikov other than the Laplacian and $\bar{x}$ is the mean of all the vector pixels in the window.

Vector Sigma Filters

These filters are the multichannel extensions of the scalar sigma filters [50, 60] that make use of an approximated multivariate variance based on either the mean vector or the lowest-ranked vector, for checking if the center pixel is noisy or not.The vector sigma filters are broadly classified into adaptive and non adaptive depending on the use of a tuning parameter γ or determining the threshold adaptively.

Non Adaptive Vector Sigma Filters

The aggregated distance associated with the center pixel is compared with a threshold which is a combination of theapproximated variance and a tuning parameter

a. Non adaptive sigma filters based on lowest ranked vector

The sigma vector median filter [61-63]based on the lowest ranked vector (SVMF_rank) is given by the equation

$$\boldsymbol{x}_{SVMF\_rank} = \begin{cases} \boldsymbol{x}_{VMF}\ if\ m(c) \geq Th \\ \boldsymbol{x}_c\ otherwise \end{cases} \quad (33)$$

Where the threshold is given as

$$Th = m(x_{VMF}) + \delta\varphi_\gamma \quad (34)$$

$$\text{such that}\varphi_\gamma = \frac{m(x_{VMF})}{(m\times n)-1} \quad (35)$$

is the multivariate approximated variance which represents the mean distance from the vector median to all the other pixels in the window ensuring that measure of dispersal is not dependent of the window size, and $m(c) = \sum_{j=1}^{m\times n}\left\|\boldsymbol{x}_c - \boldsymbol{x}_j\right\|_\mu$ is the accumulative distance associated with the center pixel $\boldsymbol{x}_c$and similarly$m(x_{VMF})$ is associated with the VMF output $\boldsymbol{x}_{VMF}$.

If the distance criteria are of angular measure as described in equation 9 and 10 then the resultant filter is called as the sigma vector directional filter based on lowest rank (SVDF_rank) which is described as

$$\boldsymbol{x}_{SVDF\_rank} = \begin{cases} \boldsymbol{x}_{VDF} \ if \ \emptyset(c) \geq Th \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (36)$$

Where $Th = \emptyset(x_{VDF}) + \delta\varphi_\gamma$ such that $\varphi_\gamma = \frac{\emptyset(x_{VDF})}{(m\times n)-1}$ is the mean angular distance associated with the VDF output which represents the approximated multivariate variance for angular measure and $\emptyset(c)$ is the accumulative angular distance associated with the center pixel $\boldsymbol{x}_c$.

Similarly for sigma directional distance filter based on lowest rank (SDDF_rank) the required changes are made in equation 33, 34 and 35 accordingly with respect to the directional-distance measure described in equation 11.

b.Non Adaptive Sigma Filters Based on Sample Mean

This group of filter is based on approximated variance calculated based on the mean accumulative distance of the vector mean to other vector pixels in the window. The sigma vector median filters based on the sample mean (SVMF_mean) is given as

$$\boldsymbol{x}_{SVMF\_mean} = \begin{cases} \boldsymbol{x}_{VMF} \ if \ m(c) \geq Th \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (37)$$

Where the threshold is given as

$$Th = m(\bar{\boldsymbol{x}}) + \delta\varphi_\gamma \quad (38)$$

The approximated variance is given as

$$\varphi_\gamma = \frac{m(\bar{\boldsymbol{x}})}{m\times n} \quad (39)$$

where$m(c)$ and $m(\bar{\boldsymbol{x}})$ is the cumulative distance as given in equation 4, associated with the center pixel $\boldsymbol{x}_c$ and the mean$\bar{\boldsymbol{x}} = \frac{\sum_{i=1}^{m\times n} x_i}{m\times n}$ of the pixels in the window respectively.

Then the replacement of $\{m(c), m(\bar{\boldsymbol{x}})\}$by $\{\emptyset(c), \emptyset(\bar{x})\}$ and $\{\varphi(c), \varphi(\bar{x})\}$ in equation 37,38 and 39 results to the formation of sigma vector directional filter (SVDF_mean) and sigma directional distance filter (SDDF_mean) based on the sample mean,respectively such that $\emptyset(c) = \sum_{j=1}^{m\times n} D(\boldsymbol{x}_c, \boldsymbol{x}_j)$ is the cumulative angular distance of the center pixel to other vector pixels in the window and $\varphi(c) = m(i)^\delta \emptyset(i)^{1-\delta}$is the cumulative directional-distance measure, which is explained in equation 10 and 11. The calculation of the approximated variance based on the sample mean is computationally simpler than that of the lowest ranked since it avoids the determination of the aggregated distance of the VMF output to the other pixels in the window however the sample mean is less effective in impulsive environments as compared to that of the lowest ranked.

Adaptive Vector Sigma Filters.

The smoothing of the non adaptive approaches is controlled by the tuning parameter $\delta$and although its adaptation is done easily and is sufficiently robust, to adopt a fully adaptive sigma filtering [64-66] the adaptive sigma VMF (ASVMF), adaptive sigma VDF (ASVDF) and adaptive sigma DDF (ASDDF) are introduced where the threshold is computed adaptively.

a. Designed based on lowest ranked vector

The ASVMF based on lowest ranked vector (ASVMF_rank) replaces the center pixel with the VMF output if the distance of the center pixel from the VMF output is greater than the average distance of all the vector pixels from the VMF output, in the filtering window. The mathematical expression for the same filter is given below as

$$\boldsymbol{x}_{ASVMF\_rank} = \begin{cases} \boldsymbol{x}_{VMF} \ if \ \|x_c - x_{VMF}\|_\gamma \geq \sigma \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (40)$$

$$\text{Where}\sigma^2 = \frac{1}{m\times n}\sum_{i=1}^{m\times n}\left\|\boldsymbol{x}_i - \boldsymbol{x}_{VMF}\right\|_\mu^2 \quad (41)$$

is the variance of the vector pixels in the window.

The ASVDF based on lowest ranked vector (ASVDF_rank) uses the angular criterion to measure the dissimilarity, which can be seen as

$$\boldsymbol{x}_{ASVDF\_rank} = \begin{cases} x_{VMF} \ if \ D(x_c, x_{VDF}) \geq \sigma \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (42)$$

$$\text{And } \sigma^2 = \frac{1}{m\times n-1}\sum_{i=1}^{m\times n} D^2(x_i, x_{VDF}) \quad (43)$$

where $D(\boldsymbol{x}_c, \boldsymbol{x}_{VDF}) = \cos^{-1}\left(\frac{x_c x_{VDF}}{\|x_c\|\|x_{VDF}\|}\right)$ represents the angular distance between the center pixel and the VDF output and similarly for $D(x_i, x_{VDF})$as explained in equation 9.

Considering both the magnitude and the directional aspect in determining the adaptive approximated variance results in the formation of the adaptive SDDF based on the lowest ranked vector (ASDDF_rank) which is expressed as

$$\boldsymbol{x}_{ASDDF\_rank} = \begin{cases} \boldsymbol{x}_{VMF} \ if \ \delta_\gamma \geq \sigma_\gamma \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (44)$$

Where the approximated variance and the hybrid distance measure [63] are defined as

$\sigma_\gamma{}^2 = \left(\frac{1}{m\times n-1}\sum_{i=1}^{m\times n} D^2(\boldsymbol{x}_i, \boldsymbol{x}_{DDF})\right)^q \left(\frac{1}{m\times n-1}\sum_{i=1}^{m\times n}\|\boldsymbol{x}_i - \boldsymbol{x}DDF\mu 21-q\right.$and

$$\delta_\gamma = D^q(\boldsymbol{x}_c, \boldsymbol{x}_{DDF})\left\|\boldsymbol{x}_c - \boldsymbol{x}_{DDF}\right\|_\mu{}^{1-q} \quad (45)$$

respectively.

b. Designed Based on Sample Mean

This design uses an approximated variance determined based on the sample mean $\bar{x} = \frac{1}{\sum_{i=1}^{m\times n} x_i}$of the vector pixels in the window. Thus the approximated variance considering only the magnitude criteria is given as

$$\sigma_\gamma{}^2 = \frac{1}{m\times n}\sum_{i=1}^{m\times n}\left\|\boldsymbol{x}_i - \bar{x}\right\|_\mu{}^2 \quad (46)$$

Which when compared with the distance measure between the center pixel and the VMF output results to the replacement of the center pixel with the VMF output with the condition $\|\boldsymbol{x}_c - \boldsymbol{x}_{VMF}\| \geq \sigma_\gamma$ otherwise no change is performed. The above algorithm is collectively called as adaptive sigma vector median filter based on the sample mean (ASVMF_mean).

Then the adaptive sigma vector directional filters based on sample mean (ASVDF_mean) is defined as

$$\boldsymbol{x}_{ASVDF\_mean} = \begin{cases} x_{VDF} \text{ if } D(x_c,\bar{x}) \geq \sigma_\gamma \\ x_c \text{ otherwise} \end{cases} \quad (47)$$

Where the approximated variance is given as

$$\sigma_\gamma{}^2 = \frac{1}{m \times n - 1} \sum_{i=1}^{m \times n} D^2(\boldsymbol{x}_i, \bar{\boldsymbol{x}}) \quad (48)$$

Similarly the adaptive SDDF based on sample mean (ASDDF_mean) uses the same concept as explained by equation 45 and 46 with the replacement of DDF output x_{DDF} with the sample mean $\bar{x}$of the filtering window.

Entropy VectorFilters

This group of filters [67, 68] utilizes the fact that noisy pixels are more unstable and also posses more energy and entropy. They are the multichannel extensions of grayscale local contrast entropy filter [69]. In the entropy VMF (EVMF), the center pixel is replaced with the output of VMF if the local contrast entropy $P_c$related with the center pixel is greater than its corresponding adaptive local contrast entropic threshold Th_c . The EVMF is mathematically expressed as

$$\boldsymbol{x}_{EVMF} = \begin{cases} \boldsymbol{x}_{VMF} \text{ if } P_c > T_c \\ \boldsymbol{x}_c \text{ otherwise} \end{cases} \quad (49)$$

Such that local contrast entropy associated with a vector pixel $\boldsymbol{x}_i$ for $i = 1,2 \ldots m \times n$, in the filtering window W, is expressed as $P_i = \frac{\|x_i - \bar{x}\|_\mu}{\sum_{j=1}^{m \times n} \|x_j - \bar{x}\|_\mu}$ and the corresponding adaptive local entropic threshold as $Th_i = \frac{-P_i \log P_i}{-\sum_{j=1}^{m \times n} P_j \log P_j}$. If angular distance and directional distance criteria is used instead of the magnitudinal distance criteria, the resulting entropy filters are entropy VDF and entropy DDF respectively which are expressed as

$$\boldsymbol{x}_{EVDF} = \begin{cases} \boldsymbol{x}_{VDF} \text{ if } P_c > T_c \\ \boldsymbol{x}_c \text{ otherwise} \end{cases} \quad (50)$$

Where the local contrast entropy and the corresponding adaptive local contrast entropic threshold of $x_i$is given as

$$P_i = \frac{D(\boldsymbol{x}_i, \bar{\boldsymbol{x}})}{\sum_{j=1}^{m \times n} \|x_j - \bar{x}\|_\gamma} \quad (51)$$

$$\text{and } Th_i = \frac{-P_i \log P_i}{-\sum_{j=1}^{m \times n} P_j \log P_j} \quad (52)$$

respectively. Then similarly the entropy DDF (EDDF) considers the directional distance metric for determining the local contrast entropy and the local contrast entropic threshold, which is expressed as

$$P_i = \frac{D(\boldsymbol{x}_i, \bar{\boldsymbol{x}})^\delta \|x_i - \bar{x}\|_\gamma{}^{1-\delta}}{\sum_{j=1}^{m \times n} D(\boldsymbol{x}_j, \bar{\boldsymbol{x}})^\delta \|x_j - \bar{x}\|_\gamma{}^{1-\delta}} \quad (53)$$

$$\text{And } Th_i = \frac{-P_i \log P_i}{-\sum_{j=1}^{m \times n} P_j \log P_j} \quad (54)$$

such that the center pixel is replaced with the output of DDF if $P_c > T_c$.

Rank-Conditioned Vector Filters

This group of filters use impulse noise detection algorithm based on the rank of the vector pixels in the window.

Rank Conditioned Vector Median Filter (RCVMF)

For this filter [70] the aggregated or cumulative distances of the pixels, given by equation 1 and 3, in the sliding window, is calculated. Then a trimming ranked set S, which is an ordered subset , is obtained by excluding the extreme values from the ordered set pixels arranged in ascending order, depending on the magnitudes of the aggregated distances of the vector pixels from the surrounding pixels in the window. Let the rank or index of the pixels in the set S range from 1 *to* m and if the rank of the center pixel$l$ is greater than$m$, then it is replaced by the VMF output, such that m is the predefined rank of a healthy pixel.

Rank Conditioning and Threshold Vector Median Filter (RCTVMF)

This filter is an extension of the RCVMF. Considering $m$is the rank of the healthy vector pixel in the ordered set $S$and l be the corresponding rank or index of the center pixel, the RCTVMF [70,45] works with the condition

$$\boldsymbol{x}_{RCTVMF} = \begin{cases} \boldsymbol{x}_{VMF}, \text{if } l > m \text{ and} \|\boldsymbol{x}_c - \boldsymbol{x}_{VMF}\| > T \\ \boldsymbol{x}_c \quad \text{otherwise} \end{cases} \quad (55)$$

where$x_c$is the center pixel and$T$is the predefined threshold.

C. MiscellaneousFilters

These are the filters which do not belong to any of the above described categories of filters though some of them might have some similarities with certain filters in the above groups. It consists of filters from both switching and non-switching categories.

Vector Signal Dependent Rank Order Mean Filter (VSDROM)

In this filter [71] the vector pixels in the window are sorted according to their cumulative distance $m(i) = \sum_{j=1}^{m \times n} D(x_i, x_j)$ for $i = 1,2, \ldots . m \times n$. Then a set $S = \left\{\boldsymbol{x}_1, \boldsymbol{x}_2 \ldots . . \boldsymbol{x}_{(m \times n) - 1/2}\right\}$of lowest ranked $(m \times n) - 1/2$ vector pixels is considered such that each vector in the set is subtracted from the center pixel $x_c$to be compared with a correspondingly increasing threshold T_i for $i = 1,2, \ldots . (m \times n) - 1/2$.

Then if any of the difference $\|\boldsymbol{x}_i - \boldsymbol{x}_c\|_\mu$is greater than the threshold T_i, the center pixel is replaced with the VMF output otherwise it is left unchanged.

Fast Modified Vector Median Filter (FMVMF)

The FMVMF [72, 73] replaces the center pixel $\boldsymbol{x}_c$with the vector pixel$\boldsymbol{x}_l$ in the window which has the least cumulative distance to the other pixels excluding the center pixel, such that $m(c) - m(l) > T$and

$$\boldsymbol{x}_l = argmin_{x_l \in W} \sum_{i=1}^{m \times n} \|\boldsymbol{x}_l - \boldsymbol{x}_i\|_\gamma \text{ , for } i \neq c. \quad (56)$$

The cumulative distance is given by $m(i) = \sum_{j=1}^{m \times n} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_\gamma$which is already explained in equation 4.

Half Space Deepest Location Filter (HSDLF)

HSLDF [74] is a spatial domain filter which employs an algorithm that is substantially different to other nonlinear impulse noise removal filters and transforms domain filters. It is based on an adjusted version of the DEEPLOC algorithm [75] which calculates the approximate value of multivariate median (i.e. deepest location or the most central point within the window) in the RGB space. The spectral correlation between all the three channels is preserved by intrinsically considering them simultaneously in calculating the multivariate median. The efficiency in removal of the salt and pepper noise (fixed valued or random-valued noise) or the mixture of salt-and-pepper noise by the HSDLF doesn't depend on the source and/or distribution of the multichannel noise.

Fast Fuzzy Noise Reduction Filter (FFNRF)

The FFNRF [76, 77] replaces the center pixel $\boldsymbol{x}_c$ with the vector pixel $\boldsymbol{x}_p$ that has the maximum cumulative similarity to the other pixels in the window if $\sum_{i=1}^{m\times n} M(\boldsymbol{x}_c,\boldsymbol{x}_i) <$ *i=1i≠cm×nMxp,xi* otherwise the center pixel is kept unchanged such that

$$\boldsymbol{x}_p = argmax_{\boldsymbol{x}_p \in W} \sum_{\substack{j=1 \\ j\neq c}}^{m\times n} M(\boldsymbol{x}_p,\boldsymbol{x}_j) \quad (57)$$

The similarity between two pixels x_i and x_j is given as

$$M^{\beta}(\boldsymbol{x}_i,\boldsymbol{x}_j) = \prod_{k=1}^{3} \left(\frac{min\left(x_i^k,x_j^k\right)}{max\left(x_i^k,x_j^k\right)}\right)^{\beta} \quad (58)$$

which is special fuzzy metric [78] such that the value of each term in the product can be pre-calculated as

$$P^{\beta}(a,b) = \left(\frac{\min(a,b)+Q}{\max(a,b)+Q}\right)^{\beta} \quad (59)$$

Then the pre-calculated values can be used to compute the fuzzy similarity between the two vector pixels x_i and x_j as

$$M^{\beta}(\boldsymbol{x}_i,\boldsymbol{x}_j) = \prod_{k=1}^{3} P^{\beta}(\boldsymbol{x}_i,\boldsymbol{x}_j) \quad (60)$$

which actually results to a faster computation as compared to $L_1 - norm$. The Q and β are user defined parameters used in the computation of the fuzzy similarity.

Adaptive Vector Marginal Median Filter (AVMMF)

The AVMMF [79] compares the cumulative distance of the center pixel $m(c)$ with that of the sorted array of the cumulative distance of all the pixels $m(i)$ for $i = 1,2,3 \ldots . m \times n$ in the window, so that center pixel lies in the l^{th} index in the sorted array. $m(i)$ Is already explained in equation 4 and if l is greater than the index $c = (m\times n) + 1/2$ defining the center of the window, then the center pixel is replaced by the median of the respective vector medians for $n = 1, 2\ldots k$, where n is the index of the sorted array of cumulative distances of the pixels in the window and $k \epsilon\, n$.

Filters Based on Long Range Correlation

Normally most of the vector filters which works as spatial filtering consider the nearest neighbors in the local window, for checking whether the center pixel is noisy or not and subsequently replacing the center pixel with a filter that actually works by considering the localized image characteristics. But for this type of filter both the detection and the replacement process depends on the local window and also on the remote regions [80] of the image, depending on the fact that there exist a strong long range correlation within the natural images [81]. Other than that it firstly detects noisy pixel and then replaced or cancelled it. Any of the noise detection algorithms described in [82-84] and can be used for detecting the noisy pixel from the local window. Then once the center pixel is detected as noisy pixel, a remote window larger than the local window, around the impulse pixel is created in the search range such that the uncorrupted pixels in the local window and the remote window competes for the perfect match based on their mean square errors. Finally the noisy pixel is replaced with the center pixel in the remote window, with the least mean squared error.

Neuvo Vector Median Filter (NVMF)

The NVMF [82]replaces the center pixel with the VMF output if the distance of the vector median from the center pixel is greater than a predefined threshold Th which is expressed as

$$\boldsymbol{x}_{NVMF} = \begin{cases} \boldsymbol{x}_{VMF}\ if \|\boldsymbol{x}_c - \boldsymbol{x}_{VMF}\|_{\mu} > Th \\ \boldsymbol{x}_c \qquad otherwise \end{cases} \quad (61)$$

Vector Marginal Median Filters (VMMF)

The VMMF [85,86] calculates the scalar median of each of the sliding window in the R, G and B channels separately and replace the respective center pixels, without actually considering the correlation between the channels.

Noise Percentage Based Switching Filter (NPSF)

In this particular filter [87] prior to the application of noise detection algorithm, the localized probability of impulse noise in the sliding window is predicted after which two adaptive switching filters are switched between the low noised probability and the higher one, which is expressed as

$$\boldsymbol{x}_{NPSF} = \begin{cases} \boldsymbol{x}_{ACWVMF}\ if\ T_L \le P_r \le T_H \\ \boldsymbol{x}_{MASVMF\_MEAN}\ otherwise \end{cases} \quad (62)$$

Where $P_r = {N_{imp}}/{m\times n}$ is the localized noise probability such that N_{imp} is the number of 0s and 255s valued pixels in the window. Then x_{ACWVMF} is the output of ACWVMF which is already explained and x_{MASVMF_MEAN} is the output of modified adaptive sigma vector median filter which actually uses the same noise detection algorithm used by ASVMF_mean, and then the center pixel is replaced by the output of an exponentially weighted mean filter [88-90]as

$$\boldsymbol{x}_{Ewmf} = \frac{\sum_{i=1}^{m\times n} w_i \boldsymbol{x}_i}{\sum_{i=1}^{m\times n} w_i} \quad (63)$$

Where $w_i = e^{-\left(\frac{m(i)^{\alpha}+l(i)^{\eta}}{\beta}\right)}$ is the weight assigned to each pixel x_i of the window, which has the noisy pixel. Then $m(i)$ is the cumulative intensity distance which is explained in equation 4 and $l(i) = \sum_{j=1}^{m\times n} \|i-j\|_{\mu}^2$ is the cumulative spatial distance corresponding to x_i with respect to other pixels in the window.

Adaptive Vector Median filter (AVMF)

This filter [91] is described by the equation as

$$x_{AVMF} \begin{cases} \boldsymbol{x}_{VMF}\ if\ d\left(\boldsymbol{x}_c,\boldsymbol{x}_{k(mean)}\right) > Th \\ \boldsymbol{x}_c\ otherwise \end{cases} \quad (64)$$

Where $d\left(\boldsymbol{x}_c,\boldsymbol{x}_{k(mean)}\right) = \left\|\boldsymbol{x}_c - \boldsymbol{x}_{k(mean)}\right\|_{\mu}$ and $\boldsymbol{x}_{k(mean)} = \frac{1}{k}\sum_{i=1}^{k} \boldsymbol{x}_i$ is the mean of the first k elements from the set of rank ordered vector pixels as described in equation 2. Then x_c is the center pixel and Th is the predefined threshold. Another adaptive filter is the Adaptive Vector Directional Filter (AVDF) [92] which is the angular counterpart of AVMF.

Robust Switching Vector Filter (RSVF)

The robust switching vector median filter (RSVMF) [93]considers the center pixel to be a noisy pixel if the cumulative distance associated with the center pixel $m(c)$is greater than a predefined percentage β of the cumulative distance with respect to the vector median $m(x_{VMF})$. The RSVMF is expressed as

$$\boldsymbol{x}_{RSVMF} = \begin{cases} \boldsymbol{x}_{VMF}, & \text{if } m(c) > \beta m(x_{VMF}) \\ \boldsymbol{x}_c, & \text{otherwise} \end{cases} \quad (65)$$

If the cumulative distance is determined using the angular metric $\emptyset(i)$ and directional-distance metric $\varphi(i)$, then the noisy pixel is replaced with the output of VDF and DDF respectively that results in robust switching VDF (RSVDF) and robust switching DDF (RSDDF) correspondingly.

Adaptive Marginal Median Filter (AMMF)

The AMMF [94, 86] combines the noise removal efficiency of the Marginal median filter (MMF) and the capability of maintenance of vector correlation among the pixels in VMF. In this filter, a set $M = \{x_1, x_2, \ldots . x_l\}$ consisting of first l elements from the rank ordered set of pixels (equation 2) is formed, and then the set M is given as input to the MMF. The MMF treats its input as scalar or channel wise by finding the median of the set M for each channel R, G and B respectively [79] as

$$\boldsymbol{x}_{AMMF} = \left(\left(med(\{x_{(1)}^R, \ldots x_{(l)}^R\}) \right), \left(med(\{x_{(1)}^G, \ldots x_{(l)}^G\}) \right), \left(med(\{x_{(1)}^B, \ldots x_{(l)}^B\}) \right) \right) \quad (66)$$

Modified Switching Median Filter (MSMF)

The MSMF [95, 85] firstly detects the noisy pixels using the detection algorithm of AVMF, and then the detected pixels are further treated or convolved with n number of Laplacian operators to check their edginess. Laplacian operators are 2 dimensional differential operators that check the rate of change of the pixels locally with respect to the surrounding pixels. Then the minimum Laplacian difference $LD = \min\{\boldsymbol{x}_c * La_k \mid k = 1, \ldots, n\}$ is compared with a threshold TH, such that the center pixel is replaced with the output of VMF, $\boldsymbol{x}_{VMF}$ if $LD \geq TH$ otherwise the center pixel x_c is kept unchanged. La is the laplacian operator which is convolved with the center pixel.

Extended Vector Median Filter (EXVMF)

The EXVMF [2,3] replaces the center pixel with the mean filter output, $\boldsymbol{x}_{mean} = \frac{\sum_{i=1}^{m\times n} x_i}{m\times n}$ if the cumulative distance associated with the $\boldsymbol{x}_{mean}$, $m(\boldsymbol{x}_{mean})$ is less than that of the cumulative distance associated with the $\boldsymbol{x}_{VMF}$, $m(\boldsymbol{x}_{VMF})$, otherwise the $\boldsymbol{x}_{VMF}$ replaces the center pixel $\boldsymbol{x}_c$, such that $m(\boldsymbol{x}_i) = \sum_{j=1}^{m\times n} \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_\mu$ which is explained in equation 3 and 4.

Adaptive Hybrid Multichannel Filter (AHMF)

The AHMF [96] is made up of three components namely a) a Hybrid multichannel filter (HMF), b) a fuzzy rule based system and c) an adaptive learning algorithm such that the HMF works by applying a summation transformation on the VMF, VDF and the identity filter (IMF). Because of its special character, the AMMF is able to remove the impulse noise effectively by steel maintaining the chromaticity and preserving the fine details and edges of the image.

Filters Based on Hopfield Neural Network and Improved Vector Median Filter

This particular filter firstly check for noisy pixel using a Hopfield neural network (HNN) and then the noisy pixel is processed with an improved VMF in RGB space and in HSI space [97] in the following steps:

a) Vector pixels which are qualified to be vector median are collected by applying the VMF in the sliding window for the RGB space.

b) When more than one pixel is fit to be the vector median, the one which is closer to the mean of Hue in the HSI space is selected.

c) Then finally when more than one pixel is selected in step b, the pixel which is nearest to the mean of the Saturation in the HIS space is selected.

Modified Entropy Vector Median filter (MEVMF)

This filter [98] is a modification of the EVMF, by using the same noise detection algorithm used in EVMF, but the noisy pixel is replaced by the output of a fuzzy weighted filter $\boldsymbol{x}_{Fwf}$ instead of using $\boldsymbol{x}_{VMF}$ as described below

$$\boldsymbol{x}_{EVMF} = \begin{cases} \boldsymbol{x}_{Fwf} \ if \ P_c > T_c \\ \boldsymbol{x}_c \ otherwise \end{cases} \quad (67)$$

Where P_c is the local contrast entropy associated with the center pixel x_c and T_c is the corresponding local contrast entropic threshold such that $P_i = \frac{\|x_i - \bar{x}\|_\gamma}{\sum_{j=1}^{m\times n} \|x_j - \bar{x}\|_\gamma}$

And $Th_i = \frac{-P_i \log P_i}{-\sum_{j=1}^{m\times n} P_j \log P_j}$ are the corresponding local contrast entropy and local entropic threshold for vector pixel x_i. The fuzzy weighted filter is described as $\boldsymbol{x}_{Fwf} = \frac{\sum_{i=1}^{m\times n} w_i \boldsymbol{x}_i}{\sum_{i=1}^{m\times n} w_i}$, where membership function used to assign the weight is given as $w_i = exp\left(-\frac{d(i)^\alpha}{\beta}\right)$ for $i = 1,2,3 \ldots . . m \times n$, such that $d(i) = \sum_{j=1}^{m\times n} \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_\mu$ is the cumulative magnitude distance of $\boldsymbol{x}_i$ with respect to the vector pixels in the window.

3. Impulse Noise Model

The impulse noise can be varied according to the source, such that fixed valued impulse noise results from malfunction of sensor and the random valued noise are generally caused by electronic interference [99]. Impulse noise can be further divided as correlated impulse and uncorrelated impulse noise. The correlated type [3] can be described as

$$\boldsymbol{y} = \begin{cases} \boldsymbol{x} \text{with probability } 1 - p \\ \{n_R, \ x_G \ \ x_B\} \text{with probability } p_R p \\ \{x_R, \ n_G, \ x_B\} \text{with probability } p_G p \\ \{x_R, \ x_G, \ n_B\} \text{with probability } p_B p \\ \{n_R, \ n_G, \ n_B\} \text{with probability } p_a p \end{cases} \quad (68)$$

where $\boldsymbol{x} = \{x_R, x_R, x_B\}$ is the original uncontaminated vector pixel, $\boldsymbol{y} = \{y_R, y_G, y_B\}$ may be either contaminated or uncontaminated, $n_k = \{n_R, n_G, n_B\}$ equals 0 or 255 with equal probability for fixed-valued impulse noise, or takes any value in the range [0,255] for random-valued impulse noise; p is the probability of corruption of the color image with the impulse noise; p_R, p_G and p_B are the channel corruption probabilities for the R, G and B channel respectively and $p_a = 1 - p_R - p_G - p_B$.

And the uncorrelated impulse noise can be expressed as [100]

$$y_k' = \begin{cases} n_k \text{ with probability } p \\ q_k \text{ with probability } 1-p \end{cases} \qquad (69)$$

where $k = \{R, G, B\}$denotes the three channels in RGB color space; p is the channel corruption probability; $\boldsymbol{y}_k$ and n_k denote the original component and contaminated component respectively. n_k can take either 0 or 255 for fixed-valued impulse noise and can take any discrete value in [0,255] for random-valued impulse noise.

4. Filter Performance Measurements

Execution time, mean absolute error (MAE) [37], normalized color difference (NCD) [37] and peak signal to noise ratio (PSNR) [85] are the performance measuring parameters which will be used to evaluate the filters in comparison. Color chromaticity preservation capability of a filter is measured with NCD. A filtered image is said to preserve its chromaticity if it is free from the shadowy effects whereas MAE represents the noise suppression and the signal–detail preservation capability. MAE is mathematically expressed as

$$MAE = \frac{1}{3MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[|\boldsymbol{x}(i,j)^R - \boldsymbol{x}_F(i,j)^R| + |\boldsymbol{x}(i,j)^G - \boldsymbol{x}_F(i,j)^G| + |\boldsymbol{x}(i,j)^B - \boldsymbol{x}_F(i,j)^B|] \qquad (70)$$

Considering that $M \times N$ is the size of the image, where $\boldsymbol{x}(i,j)$ and $\boldsymbol{x}_F(i,j)$ are the original and the filtered image respectively. $\{\boldsymbol{x}(i,j)^R, \boldsymbol{x}(i,j)^G, \boldsymbol{x}(i,j)^B\}$ and $\{\boldsymbol{x}_F(i,j)^R, \boldsymbol{x}_F(i,j)^G, \boldsymbol{x}_F(i,j)^B\}$are the red, green and blue components of the original image$\boldsymbol{x}(i,j)$ and filtered image $\boldsymbol{x}_F(i,j)$ respectively. It can be seen that with the help of the above equation aslight difference between the original and filtered image can be highlighted properly for better comparison between the filters. The NCD is defined in theLu^*v^* ccolor space by

$$NCD = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left\{ \begin{matrix} |L(i,j) - L_F(i,j)|^2 + |u(i,j) - u_F(i,j)|^2 + \\ |v(i,j) - v_F(i,j)|^2 \end{matrix} \right\}^{1/2}}{\sum_{i=1}^{M}\sum_{j=1}^{N}\{|L(i,j)|^2 + u(i,j)^2 + v(i,j)^2\}^{1/2}} \qquad (71)$$

Where $\{L(i,j), u(i,j), v(i,j)\}$ and $\{L(i,j), u(i,j), v(i,j)\}$ are the respective values of the lightness and two chrominance components of the original image $\boldsymbol{x}(i,j)$ and the filtered image $\boldsymbol{x}_F(i,j)$. And the signal content of the image is described as

$$PSNR = 10\, log_{10} \frac{{x_{max}}^2}{\frac{1}{3MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\|\boldsymbol{x}(i,j) - \boldsymbol{x}_F(i,j)\|^2} \qquad (72)$$

Where $\boldsymbol{x}_{max} = 2^b - 1$isthe maximum intensity for an image in a particular channel and b is the number of bits per pixel for the particular channel.Then the similarity between two images is measured by Structural similarity index (SSIM), which is expressed as

$$SSIM = \frac{(2\mu_x\mu_y + D_1)(2\mu_{xy} + D_2)}{(\mu_x^2 + \mu_y^2 + D_1)(\sigma_x^2 + \sigma_y^2 + D_2)} \qquad (73)$$

where μ_x and μ_y are mean of the original and filtered image, whereas σ_x^2 and σ_y^2 characterize the variance of the original and filtered images respectively. $D_1$and D_2 are the constants. It is desired to a high value of PSNR and SSIM for a particular filter to be efficient in removing the impulse noise from an image. On the other hand MAE, MSE and NCD should bear the minimum values for the filter to be able to preserve the fine details, edges and chromaticity of the image.

5. Conclusions

A various form of vector filters for impulse noise removal have been surveyed and analyzed by grouping them into categories. Some recently proposed miscellaneous filters have also been included and briefed individually.

References

[1] M.E. Celebi, H.A. Kingravi and Y.A. Aslandogan, "Nonlinear vector filtering for impulsive noise removal from color images" *J. Electron Imaging* 16(3):033008,2007.

[2] J.Astola, P. Haavisto, and Y. Neuvo, "Vector median filters", *Proceedings of the IEEE,* vol. 78, no.4, pp. 678-689, 1990.

[3] T.Viero, K. Oistamo, and Y. Neuvo, "Three-dimensional median-related filters for color image sequence filtering", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.4, no. 2, pp.129-142, 1994.

[4] B.Smolka, and M. Perczak, "Generalized vector median filter", *Image and Signal Processing and Analysis, 2007. ISPA 2007.*

[5] S.Vinayagamoorthy, et al., "A multichannel filter for TV signal processing", *IEEE transactions on consumer electronics,* vol.42, no.2, pp. 199-202, 1996.

[6] B.Smolka, K.N. Plataniotis, and A.N. Venetsanopoulos, "Nonlinear techniques for color image processing", *Nonlinear Signal and Image Processing: Theory, Methods, and Applications,* pp.445-505, 2004.

[7] P.E Trahanias, and A.N. Venetsanopoulos, "Vector directional filters-a new class of multichannel image processing filters", *IEEE Transactions on Image Processing,* vol.2, no.4, pp.528-534, 1993.

[8] P.E Trahanias, D. Karakos, and A.N. Venetsanopoulos, "Directional processing of color images: theory and experimental results", *IEEE Transactions on Image Processing,* vol.5, no.6, pp. 868-880, 1996.

[9] D.G Karakos, and P.E. Trahanias, "Combining vector median and vector directional filters: The directional-distance filters", *Image Processing, Proceedings. International Conference on*. Vol. 1. IEEE, 1995.

[10] R.Wichman, et al.,"Weighted vector median operation for filtering multispectral data" , *Applications in Optical Science and Engineering*. International Society for Optics and Photonics, 1992.

[11] B. Smolka, "Rank Order Weighted Vector Median Filter", *17th International Conference on Systems, Signals and Image Processing, IWSSIP 2010.*

[12] B. Smolka, "Rank-Based Vector Median Filter", *Proceedings: ASIPA ASC 2009: Asia Pacific Signal and Information Processing Association, 2009 Annual Summit and Conference: 254-257.*

[13] R.Lukac, K.N.Plataniotis and B.Smolka, "Sharpening vector median filters". *Signal Process.* 87, 2085-2099 (2007)

[14] B. Smolka , "Adaptive edge enhancing technique of impulsive noise removal in color digital images ", *In: Proceedings of the Third International Conference on Computational Color Imaging, CCIW*, pp. 60-74. Springer, Berlin (2011)

[15] B. Smolka, "Adaptive rank based impulsive noise reduction in color images" *In: IEEE International Conference on Communications and Electronics (ICCE 2012)*, pp. 335-359 (2012)

[16] B.Smolka, "Adaptive truncated vector median filter " *In: IEEE International Conference on Computer Science and Automation Engineering, (CASE)*, pp.261-266 (2011)

[17] R. Lukac, et al., "Selection weighted vector directional filters", *Computer Vision and Image Understanding*, vol.94, no.1, pp.140-167, 2004.

[18] R. Lukac, et al., "Generalized selection weighted vector filters", *EURASIP Journal on Advances in Signal Processing* , vol.12, pp.1-16, 2004.

[19] R. Lukac, "Adaptive color image filtering based on center-weighted vector directional filters", *Multidimensional Systems and Signal Processing,* vol.15, no.2, pp.169-196, 2004.

[20] Z. Ma, H.R.Wu, and D.Feng, "Partition-based vector filtering technique for suppression of noise in digital color images", *IEEE transactions on image processing*, vol.15, no.8, pp.2324-2342, 2006.

[21] K.N.Plataniotis, D.Androutsos and A.N. Venetsanopulos, "Adaptive fuzzy systems for multichannel signal processing"*proceedings of IEEE,* Vol.87, No.9, pp. 1601-1622, 1999

[22] K.N.Plataniotis, D.Androutsos and A.N. Venetsanopulos, "Fuzzy adaptive filters for multichannel image processing,"*Signal Processing J*., vol. 55, no. 1, pp. 93-106, 1996.

[23] K.N.Plataniotis, D. Androutsos, S. Vinayagamoorthy and A.N. Venetsanopoulos, "An adaptive nearest neighbor multichannel filter," *IEEE Trans. Circuits Syt. Video Technol.*, vol. 6, pp.699-703, Dec 1996.

[24] K.N.Plataniotis, D. Androutsos and A.N. Venetsanopoulos, "Color image filters: The vector directional approach," *Opt. Eng*., vol. 36, no. 9, pp. 2375-2383, 1997.

[25] K.N.Plataniotis, D. Androutsos, S. Vinayagamoorthy and A.N. Venetsanopoulos, "Color image processing using adaptive multichannel filters," *IEEE Trans. Image Processing,* vol. 6, pp. 933-950, July 1997.

[26] K.N.Plataniotis, D. Androutsos, V. Sri, and A.N. Venetsanopoulos, "A nearest neighbor multichannel filter" *Electron Lett*., vol. 33, no. 22, pp. 1910-1911, Oct. 1995.

[27] K.N.Plataniotis, D. Androutsos and A.N. Venetsanopoulos, "Content-based image filters," Electron. Lett., vol. 33, no. 3, pp. 202-203, 1997.

[28] Y. Yuksel, M. Alci and M.Emin Yuksel, "Performance enhancement of impulse noise filters by image rotation and fuzzy processing," *Int. J. Electron. Commun. (AEU)* 64 pp. 329-338, 2010.

[29] J.SR. Jang, C.T. Sun and E. Mizutani, "Neuro fuzzy and soft computing," *NJ: Prentice-Hall International Inc.*; 1997

[30] V. Chatzis and I. Pitas, "Fuzzy scalar and vector median filters based on fuzzy distances," *IEEE Trans. Image Process.* 8(5). 731-734, 1999.

[31] L. Khriji and M. Gabbouj, "Adaptive fuzzy order statistics-rational hybrid filters for color image processing," *Fuzzy Sets syst.* 128(1). Pp. 35-46, 2002.

[32] S.Peng, and L.Lucke, "Fuzzy filtering for mixed noise removal during image procesisng", *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence. Proceedings of* the Third IEEE Conference on. IEEE, 1994.

[33] M.E. Celebi, G.Schaefer, and H.Zhou, "A new family of order-statistics based switching vector filters", *17th IEEE International Conference on Image Processing on Image Processing*, 2010.

[34] K.N. Plataniotis, D. Androutsos, S. Vinayagamoorthy, and A.N. Venetsanopoulos, "A nearest neighbor multichannel filter", *Electron. Lett*. vol.31, no.22, pp.1910-1911, 1995.

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

[36] Kh. M. Singh and P. K. Bora, "Noncausal vector linear prediction filter,"*WSEAS Trans. Circuits Syst.,* 2, 1211-1219, 2003

[37] Kh.Manglem Singh and P. K. Bora , "Switching vector median filters based on non-causal linear prediction for detection of impulse," *The Imaging Science Journal*. 62, pp. 313-325, 2014

[38] J.H. Hu, Y. Wang, and P. T. Cahill,"Multichannel application in magnetic resonance images,"*IEEE Trans. Image Process.* 6, 1155-155566, 1997.

[39] S. David and B. Ramamurthi,"Two-sided filters for frame-based prediction,"*IEEE Trans. Signal Process.* 39, pp. 789-7994,1991.

[40] A. Asif and J. M. F. Moura,"Image codec by noncausal prediction residual mean removal and cascaded VQ", *IEEE Trans. Circuits Syst. Video Technol.,* 6, pp. 42-55, 1996.

[41] A. Asif, "Fast Rauch- Tung- Striebelsmoother based image restoration for noncausal images,"*IEEE Signal Process. Lett*., 11, pp. 371-374, 2004.

[42] R.Lukac, "Optimised directional distance filter", *Machine Graphics & Vision International Journal*, vol.11, no.2/3, pp. 311-326, 2002.

[43] R.Lukac, and B.Smolka, "Application of the adaptive center-weighted vector median framework for the enhancement of cDNA microarray images", *International Journal of Applied Mathematics and Computer Science*, vol.13, no.3, pp. 369-384, 2003.

[44] 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, no.2, pp.289-311, 2015.

[45] R.Wichman, et al.,"Weighted vector median operation for filtering multispectral data" , *Applications in Optical Science and Engineering*. International Society for Optics and Photonics, 1992.

[46] Y.Deng, et al., "Peer group filtering and perceptual color image quantization", *Circuits and Systems, ISCAS'99. Proceedings of the 1999*

[47] C. Kenney, et al., "Peer group image enhancement", *IEEE Transactions on Image Processing,* vol. 10, no.2, pp. 326-334 ,2001.
[48] J.Y. Ho, "Peer region determination based impulsive noise detection", *Acoustics, Speech, and Signal Processing, 2003.*
[49] M. Gabbouj and F.A. Cheikh, "Vector median-vctor directional hybrid filter for color image restoration," *Proc. EUSIPCO'* 96, 2, pp. 879-882 ,1996.
[50] L. Khriji and M. Gabbouj, "A class of multichannel image processingfilters," *Electron Lett*. 35(4), pp. 285-287, 1999.
[51] L. Khriji and M. Gabbouj, "Vector median-rational hybrid filters for multichannel image processing," *IEEE Signal Process. Lett* 6(7), pp. 186-190, 1999.
[52] L. Khriji and M. Gabbouj, "A new class of multichannel image processing filters: vector median-rational hybrid filters," *IEICE Trans. Inf. Syst. E82* (12), pp. 1589-1596, 1999.
[53] L. Khriji and M. Gabbouj, "Multichannel image processing using fuzzy vector median-rational hybrid filters," *proc. EUSIPCO'00*, pp. 1345-1348, 2000.
[54] L. Khriji and M. Gabbouj, "Rational-based adaptive fuzzy filters," *Int. J. Comput. Cognition*, 2(1), pp. 113-132 , 2004.
[55] B. Smolka and K.N. Plataniotis, "Soft –switching adaptive technique of impulsive noise removal in color images," *Proc. 2nd Int. Conf. Image Analysis and Recog. (ICIAR), Lect. Notes Comput,* Sci. 3656, pp. 686-693, 2005.
[56] B. Smolka, R. Bieda, K.N. Plataniotis and R. Lukac, "Adaptive soft-switching filter for impulsive noise suppression in color images,"*Proc. EUSIPCO'05*, 2005.
[57] B. Smolka, K.N. Plataniotis, R. Lukac and A. N. Venetsanopoulus, "New class of impulsive noise reduction filters based on kernel density estimation," *Proc. Of the 28th IEEE Int. Conf. on Acoustics, Speech & Signal Process. (ICASSP'03)* 3, pp. 721- 724, 2003
[58] B. Smolka, R. Lukac, K.N. Plataniotis and A. N. Venetsanopoulus, "Application of kernel density estimation for color image filtering," *Proc, Vis, Commun. and Image Process. (VCIP'03), SPIE,* Vol. 5150, pp. 1650-1656, 2003.
[59] B. Smolka, K.N. Plataniotis, R. Lukac and A. N. Venetsanopoulus, "Kernel density estimation based impulsive noise reduction filter," *Proc. IEEE. Int. Conf. Image Process. (VCIP'03)*, 2, pp. 137-140, 2003.
[60] R. Lukac, et al. "Angular multichannel sigma filter", *Acoustics, Speech, and Signal Processing*, vol.3, 2003.
[61] R. Lukac, B. Smolka, et al. "Vector sigma filters for noise detection and removal in color images", *Journal of Visual Communication & Image Representation*, vol. 17, pp. 1-26, 2006.
[62] R. Lukac, et al., "A variety of multichannel sigma filters", *Optical Metrology, International Society for Optics and Photonics,* 2003.
[63] J.S Lee, "Digital image smoothing and the sigma filter", *Computer Vision, Graphics, and Image Processing*, vol. 24, no.2, pp.255-269, 1983.
[64] R. Lukac, B. Smolka and K. N. Plataniotis, "Color sigma filter," *Proceedings of the 9th Int. Workshop on Systems, Signals and Image Processing, IWSSIP'02*, pp. 559-565, 2002.
[65] R. Lukac, B. Smolka, K. N. Plataniotis, A. N. Venetsanopoulos and P. Zavarsky, "Angular multichannel sigma filter," *Procceding of the 28th Int. Conf. on Acoustic, Speech and Signal Processing ICASSP*, vol. III, pp. 745-748, 2003.
[66] R. Lukac, B. Smolka, K. N. Plataniotis and A. N. Venetsanopoulos, "Generalized adaptive vector sigma filters," *Proceedings of the IEEE Int. Conf. on Multimedia and Expo*, vol. 1, pp. 537-540, 2003.
[67] B. Smolka, R. Lukac, K.N. Plataniotis and A. N. Venetsanopoulus, "Entropy vector median filter," *Proc. 1st Iberian Conf. Patt. Recogn. Image Analy. (IbPRIA), Lect. Notes Comput. Sci*. 2652. Pp. 1117-1125, 2002.
[68] B. Smolka, R. Lukac, K.N. Plataniotis and A. N. Venetsanopoulus, "Gneralized entropy vector filters," *Proc. 4th EURASIP EC-VIP-MC Video Image Processing and Multimedia Commun. Conf*., pp. 239-244, 2003.
[69] A. Beghdadi and A.Khellaf, "A noise-filtering method using a local information measure," *IEEE Trans. Image Process.* 6(6), 879-882, 1997.
[70] K.M. Singh, and P.K. Bora, "Adaptive vector median filter for removal of impulse noise from color images", *Journal of Scientific Computing,* vol. 4, no.1, pp.1063-1072, 2004.
[71] M.A. Moore, M. Gabbouj, and S. K. Mitra Vector SD-ROM filter for removal of impulse noise from color images. Proc. *2nd EURASIP conf., Krakow, Poland, ECMCS*, No. 89,1999.
[72] B. smolka, M. Szezepanski, K.N. Plataniotis and A. N. Venetsanopoulus, "Fast modified vector median filter," *Proc. 9th Int. Conf. Compu. Analy. Images and patterns, Lect. Notes Comput. Sci*. 2124, pp. 570-580, 2001.
[73] B. smolka, M. Szezepanski, K.N. Plataniotis and A. N. Venetsanopoulus, "On the fast modification of the vector median filter," *Proc. 16th Int. Conf. Patt. Recog. ICPR'02*, 3, pp. 931-934, 2002.
[74] M. Emre Celebi and B. Smolka, "Advances in low level color image processing,"pp. 141-147.
[75] J. Lichtenauer, MJT. Reinders and E.A. Hendriks, "A self-calibrating chrominance model applied to skin color detection," *Proceedings of the VISAPP,* vol. 1, pp. 115-120.
[76] S. Morillas, V. Gregori, G. Peris- Fajarnes and P. Latorre, "A new vector median filter based on fuzzy metrics," *Proc. 2nd Int. Conf. on Image Anal. Recog. (ICIAR'05), Lect. Notes Comput. Sci.* 3656, pp. 82-91, 2005.
[77] S. Morillas, V. Gregori, G. Peris- Fajarnes and P. Latorre, "A fast impulsive noise color image filter using fuzzy metrics," Real-Time Imag. 11(5/6), pp. 417-428, 2005.
[78] A. George and P. Veeramani, "On some results in fuzzy metric spaces," Fuzzy Sets Syst. 64(3). Pp. 395-399, 1994.
[79] S. Morillas, V. Gregori and A. sapna,"Adaptive marginal median filter for color images,"*Sensors* 11:3205-3213, 2011.

[80] Z.Wang, and D.Zhang "Restoration of impulse noise corrupted images using long-range correlation", *IEEE Signal Processing Letters*, vol. 5, no.1, pp. 4-7, 1998.
[81] D. Zhang, and Z. Wang, "Image information restoration based on long-range correlation", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.12, no.5, pp.331-341, 2002.
[82] T. Sun, and Y. Neuvo, "Detail-preserving median based filters in image processing", *Pattern Recognition Letters*, vol.15, no.4, pp. 341-347, 1994.
[83] D.Zhang, and Z.Wang, "Impulse noise removal using polynomial approximation", *Optical Engineering*, vol.37, no.4, pp. 1275-1282, 1998.
[84] Roji Chanu and Kh. M. Singh, "Vector median filters: A survey," Int. Jour. Of Comput. Science and Network Security, vol. 16, no.12, pp. 66-84, 2016.
[85] K. Plataniotis and A. N. Venetsanopoulos, "color image processing and applications," Springer, 2000.
[86] I. Pitas, "Marginal order statistics in color image filtering," *Opt. Eng 29*. Pp. 495-503, 1990.
[87] K.Pritamdas, KH. Manglem singh and L. Lolit Kumar Singh, " An adaptive switching filter based on approximated variance for detection of impulse noise from color images", *SpringerPlus*, 10.1186/s40064-016-3644-9,vol.5, pp1-22,2016
[88] C. Tomshi and R.Manduchi " Bilateral filtering for gray and color images" *proc.IEEE int.conf . Computer vision,* pp 839-846, 1998.
[89] S.Daniel John, "Bilateral filter extension for removal of universal noise", *International Journal of latest trends in Engineering and Technology (IJLTET).* Vol.2 Issue 3, 235-241, 2013.
[90] M.Kaur, S. Gupta and B.Bhusan, "An improved adaptive bilateral filter to remove Gaussian noise from color images", *International Journal of Signal Processing, Image Processing and Pattern Recognition.* Vol.8, No.3,pp.49-64,2015.
[91] R.Lukac, "Adaptive vector median filtering," *Pattern Recognition Letters*, vol.24, no. 12, pp. 1889-1899, 2003.
[92] R.Lukac, "Colour image filtering by vector directional order statistics"*Pat. Rec. Imag. Anal*. 12, pp. 279-285, 2002.
[93] M.E. Celebi and Y. A. Aslandogan, "Robust switching median filter for impulsive noise removal," *J. Electron. Imaging,* 17, 043006-1-043006-9, 2008.
[94] I.Pitas, and A.N. Venetsanopoulos, "Nonlinear Digital Filters Principles and Applications,"*Kluwer Academic Publishers, Boston, M.A*, 1990.
[95] S.Wang, and C.H. Wu, "A new impulse detection and filtering method for removal of wide range impulse noise", *Pattern Recognition*, vol.42, no.9, pp. 2194-2202, 2009.
[96] H.T. Tsai and P.T. Yu, "Adaptive fuzzy hybrid multichannel filters for removal of impulse noise from color images", *Signal Processing*, vol.74, pp.127-151, 1999.
[97] G.P. Deepti, M.V. Borker, and J. Sivaswamy, "Impulse noise removal from color images with Hopfield neural network and improved vector median filter", *Sixth Indian Conference on Computer Vision, Graphics & Image Processing, 2008.*
[98] K. Pritamdas and Kh. M. Singh, "A new adaptive switching approach for impulse noise removal from color images," *Int. Conf. on Man and machine Interfacing (MAMI),IEE*, 978-1-5090-0225-2/15, pp. 1-6, 2015.
[99] X. Geng, X. Hu, and J. Xiao, "Quaternion switching filter for impulse noise reduction in color image" , *Signal Processing, vol.* 92, no.1, pp. 150-162, 2012.
[100]B.Smolka, "Peer group switching filter for impulse noise reduction incolor images", *Pattern Recognition Letters*, vol.31, no.6, pp. 484-495, 2010.

Author Profile

K.Pritamdas did B.E. and M.E. in Electronics and Communication Engineering and Applied Electronics from P.G.P College of Engineering and Technology and Sri Krishna Engineering College, under Anna University Chennai, India. He is currently working as an assistant Professor at NIT Manipur, India. His area of interest are Digital Image Processing, Digital signal Processing and Speech processing.

Kh. Manglem Singh did BE, MTech, MS and PhD in Electrical Engineering, Control & Instrumentation, Software Systems and Digital Image Processing from DEI, Agra, Delhi University, BITS Pilani and IIT Guwahati respectively. Currently, he is an Associate Professor at NIT Manipur. His areas of interest are Image processing, Digital Watermarking, Steganography, Information Security etc.

L.Lolit Kumar Singh did B.Tech, M.Tech and Phd in Electronics and Telecommunication Engineering from Amravati University, Jadavpur University and Jadavpur University, India. Currently he is an Associate Professor at Mizoram University.His area of interest are Microwave Engineering, Microstrip antenna and Semiconductor devices.