International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I8P229
Volume 70 Issue 8, 272-283, August 2022

Original Article

A Novel Methodology for Denoising Impulse Noise in Satellite Images through Isolated Vector Median Filter with k-means Clustering

Y.Vishnu Tej[1], M. James Stephen[2], P.V.G.D. Prasad Reddy[3], Praveen Choppala[4]

[1,3]*Department of Computer Science and Systems Engineering, Andhra University, Visakhapatnam, AP, INDIA.*
[2]*Department of Computer Science and Engineering, WISTM Engineering College, Visakhapatnam, AP, INDIA.*
[4]*Department of Electronics and Communications Engineering, WISTM Engineering College, Visakhapatnam, AP, INDIA.*

[1]vishnutej.br@gmail.com

Received: 31 May 2022 Revised: 01 August 2022 Accepted:13 August 2022 Published: 22 August 2022

Abstract *- Satellite image denoising is imperative for improving images' visual quality and making future image processing and analysis chores easier. Noise detection is critical after the noise has been discovered; appropriate filters are applied to eliminate the impulse noise from the image. By attenuating the high-frequency image components and removing noise from the image, essential features are also lost. An efficient denoising approach is necessary to retain important information, improve the visual appearance, and reliably categorize an image. This paper proposes a novel method for denoising satellite images using the isolated vector median filtering with the k-means clustering (IMF-KM) approach. The proposed method has given better performance when compared to the existing algorithms in terms of peak signal-to-noise ratio (PSNR), structure similarity index (SSIM), and root means square error (RMSE).*

Keywords *- Image De-noising, Vector median filter, Isolated vector median filter, Basic vector directional filter, Directional distance filter, Directional vector median filter, Isolated vector minimum distance filter.*

1. Introduction

Satellite image denoising has recently gained popularity among remote sensing experts. One of the most critical and difficult jobs in image processing is noise reduction. Various unwanted noises can influence the quality and resolution of a picture. Before completing any additional analysis and processing of the image, noise reduction plays a vital function as a preprocessing step in various applications, such as satellite and remote sensing image processing. Satellite images are valuable for multiple environmental applications, including earth resource tracking, geographical mapping, agricultural crop prediction, urban expansion, weather, flood and fire management, etc. The detection and analysis of objects in photographs collected from deep space probe missions are part of the space imaging application [1]. If salt and pepper noise is introduced into satellite images, it has random occurrences of both black and white intensity values and is frequently created by a noisy image threshold. The noise, which appears as black and white dots on satellite images, dramatically decreases the visual impression of the image. Standard filtering is insufficient to eliminate such noise and enhance image quality; it is critical to preserve features and edges while reducing noise in satellite images to be usable in the future [1].

One successful method for removing impulsive noise is to use the well-known median filter and its variations [2][3][4]. The effectiveness of median filters is due to two primary characteristics: edge retention and efficient noise reduction with impulsive noise resilience. Due to visual perception, edge preservation is critical in image processing [5][6].

An image is a two-dimensional numerical data set comprising the intensities of collected red, green, and blue hues. Images and electronic data are susceptible to noise, particularly during transmission from one location to another. Noise reduction (also known as image denoising) is essential to digital image processing. It allows for restoring the original image and reduces any information loss caused by noise. The impulse noise model is modeled using probabilistic measurements because of the unpredictability of pixel deterioration [7].

Several nonlinear filtering techniques may be used to reduce impulse noise. The median filter is a nonlinear digital noise reduction filter that may be used on satellite images [8,9]. A multilayer weighted graph model for image representation is created to define the grey or color difference between pixels and their adjacent pixels at

different scales. The noise detection is then transformed to locate the graph node with the lowest strength [10]. The median is derived by sorting all of the pixel values in the surrounding neighborhood into numerical order, then replacing the pixel in concern with the median pixel value. If there are an even number of pixels in the neighborhood, the average of the two middle pixel values is used [11]. The vector median filter (VMF) is a nonlinear approach for processing color and satellite images as a vector field to reduce multichannel dependency. Because they adequately address the color component correlation, the filters in this family, particularly the VMF, may perform pretty well in the impulse noise reduction without generating color artifacts [8][12]. This approach reduces the aggregate distance across all pixels by replacing the test pixel with another pixel in the window.

VDF (vector directional filter) separates image data processing into "directional processing" and "magnitude processing" based on the direction of image vectors. "directional processing" refers to image data processing that only considers directional information in the vector space. On the other hand, Magnitude processing relates to image data processing that solely considers the vector magnitudes. The separation of processing feature of VDF connects vector signal processing with single-channel image processing [13]. As proposed in the basic vector directional filter (BVDF), using direction is an alternative to using distance. This filter reduces the aggregate angular distance across all pixels in the window by using the angular distance between pixels rather than vector magnitudes [7][12]. Directional distance filtering (DDF), which uses a weighted product of VMF and BVDF, is another image filtering method [14]. Another directed approach is the directional VMF (DVMF), which applies vector median filtering over the pixels within degrees of the Centre test pixel and then calculates the VMF for the resulting [15]. By processing the test windows in isolation, the isolated vector minimum distance filter (IVMDF) is used to tackle this problem. The main concept is to keep the vector pixels isolated to minimize the aggregated distance between them and all the other pixels in the test window. Any smoothing done there impacts only the color component in consideration, not the others [16]. In the isolated vector median filtering (IVMF) approach, the primary idea is to isolate the joint vector pixels before applying median filtering to the most relevant pixels [17]. The adaptive rank-weighted switching filter (ARWSF) and adaptive switching trimmed (AST) filters have been proven to be effective in minimizing impulsive noise [15][18].

The proposed IMF-KM method is based on two main ideas: (a) categorizing the pixels in the sliding window into two groups, one that contributes to the signal and the other that contributes to the noise, and (b) performing median filtering over the pixel intensities that contribute to the signal, while ignoring the color components.

2. Signal Model

Noise is likely to degrade digital images during transmission. This research focuses on impulse noise, which is one of the various forms of noise that is significant [2,9]. In digital images, impulse noise is a random burst or sag of energy during transmission. With a certain degree of likelihood, a pixel will be distorted here.

2.1. The Impulse noise model

Let X be the color image of M X N pixels containing MN pixels, where M signifies the number of rows and N the number of columns. Then, image X will be a set of vector pixels.

$$X = \{x_{i,j}\ i = 1,2, \dots \dots, M, j = 1,2, \dots \dots, N\} \quad (1)$$

with each pixel consisting of a joint vector representing the color intensities in the red, green, and blue components as

$$x_{i,j} = \left(x_{i,j,r}, x_{i,j,g}, x_{i,j,b}\right), i = 1,2, \dots, M, j = 1,2, \dots, N \quad (2)$$

Let $Z = \{z_{i,j}, i = 1,2, \dots, M\ , j = 1,2, \dots, N\}$ be the image corrupted by impulse noise. Fixed valued impulse noise (also known as salt and pepper noise) and random valued impulse noise are the two basic types of impulse noise. A pixel in the fixed valued impulse noise is corrupted with probability p ∈ (0,1). A corrupted pixel indicates that one of its red, green, or blue components has been corrupted by changing to 0 (complete black) or 255 (complete white) with uniform probability throughout the color components. Instead of cycling to high or low values, the damaged pixels in random valued impulse noise take any random value between 0 and 255.

$$Zi,j = \begin{cases} \boldsymbol{x_{i,j}} & \text{If } q >= p \\ (\boldsymbol{x_{i,j,r}}, \boldsymbol{x_{i,j,g}}, \boldsymbol{a}) & \text{If } q < p, r < 1/3 \\ (\boldsymbol{x_{i,j,r}}, \boldsymbol{a}, \boldsymbol{x_{i,j,b}}) & \text{If } q < p, 1/3 <= r < 2/3 \\ (\boldsymbol{a}, \boldsymbol{x_{i,j,g}}, \boldsymbol{x_{i,j,b}}) & \text{If } q < p, 2/3 <= r \end{cases} \quad (3)$$

2.2. Conventional Noise Reduction Filters

This section summarises typical impulse noise filtering methods employed in the invention. Generally, the filtering procedure employs a sliding window W of n pixels and a size of $\sqrt{n}X\sqrt{n}$. It donates the set of pixels included in the window as,

$$W=\{x_i\ , i = 1, \dots \dots, n\} \quad (4)$$

where the joint vector is $x_i = (x_{i,r}, x_{i,g}, x_{i,b})$With this notation, the digital color image in (1) is now modified as

$$X= \{x_i, i = 1, \dots \dots, MN\ \} \quad (5)$$

The filtering algorithms work by identifying and processing the center pixel inside the test window W. The VMF technique is the most well-known of the filtering systems.

The aggregated distance between each pixel and every other pixel is calculated as

$$S_i = \sum_{j=1}^{n} d(x_i - x_j), i = 1, \dots, n \quad (6)$$

where $d(x_i - x_j)$ is the Minkowski's distance between two joint pixels x_i and x_j and the aggregate is reordered as

$$S_{i=1,\dots,n} \Rightarrow x_{i=1,\dots,n} : S_{i=1} \leq S_{i=2} \leq \cdots S_{i=n} \quad (7)$$

x_1	x_4	x_7
x_2	x_c=x_5	x_8
x_3	x_6	x_9

Fig. 1 A 3 x 3 test window with n = 32 = 9 and the test pixel in the center

The center pixel is then substituted with the pixel with the shortest aggregate distance from all other pixels as

$$x_c = x_{i-1} \quad (8)$$

The BVDF, on the other hand, uses the aggregated angular distance between pixels as

$$\theta_i = \sum_{j=1}^{n} \cos^{-1}\left(\frac{x_i\, x_j}{\|x_i\| \, \|x_j\|}\right), i = 1,2, \dots, n \quad (9)$$

And replaces the center pixel with the pixel that minimizes as specified in (7) and (8).

The DDF employs the weighted product of Minkowski's distance and the angular distance as

$$A_i = S_i^{\gamma} \theta_i^{1-\gamma}, i = 1,2, \dots, n \quad (10)$$

Where $\gamma \in (0,1)$.

The DVMF filter takes the VMF of the pixels in degrees to the center pixel, as illustrated in Fig.2, and then obtains the VMF of the resultant.

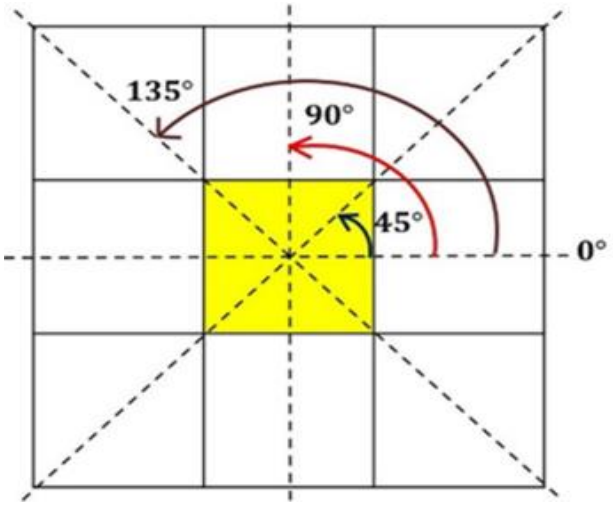

Fig. 2 A pictorial representation of DVMF

The VMFDD method adds the distances in four directions, then picks the pixels that lie at the angle that minimizes the sum from the center pixel, and then VMF is applied to those pixels. The alpha-trimming technique trims the distances by a predetermined trimming factor α, but the AST method uses a switching condition throughout the trimming operation.

All these strategies aim to reduce noise by using the VMF mechanism in various versions. There has been limited progress in detecting pixels that correlate with noise and minimizing their impact throughout the filtering process. The peer group filter explored this method; however, it was confined to grayscale images. The recently suggested ARWSF rankings combine the process of finding excellent pixels (those that contribute to the signal/information) by first measuring the distances of the pixel d(.) in the sliding window and then scaling the distances using a fading function. This scaling assigns high weight to the pixels with the shortest aggregate distance. When performing VMF over pixels using scaling, the most informative pixels are weighted heavily in the filtering process, resulting in more significant noise reduction. This paper proposes the IMF-KM approach, which improves the ARWSF method by fully marginalizing the noise pixels, resulting in better noise reduction.

3. Proposed Method

The proposed invention describes an isolated vector median filtering method that uses k-means clustering to reduce noise in digital color images. The proposed invention establishes notation and defines impulsive noise. The current technique is followed by traditional impulse noise reduction methods and the proposed isolated Vector Median Filtering using k-means clustering (IMF-KM) filter. The proposed innovation offers a unique way of filtering away impulsive noise in satellite images. Traditional filtering methods apply a noise reduction algorithm, often the vector median filtering

approach and its variations, to the center pixel of an appropriately chosen window that iteratively slides throughout the image. In the filtering process, these approaches consider the entire window into account. On the other hand, this approach considers the noise inside the noisy pixels in the filtering process and may damage the final result.

On the other hand, the current method works by grouping the pixels in the selected window into two groups, one for pixel intensities in the signal space and the other for those in the noise space. The motivation for this clustering approach is to marginalize those pixels in the noise space that appears to contribute nothing to the information in the image. The median filter is used for the pixels contributing to the signal in isolating the color components to filter out impulse noise.

The present state of the artworks is by splitting the pixels of a selected sliding window into two groups, one for the signal and the other for the noise. To achieve this grouping based on pixel intensities, the well-known k-means algorithm with K=2 was used. The main reason for dividing the pixels in a sliding window into signal and noise components is that the pixels encompassing the signal will be numerous and have relative intensities. In contrast, the pixels containing the noise will be few and have intensities distant from those corresponding to the signal space. According to this rationale, the cluster with the maximum number of pixels will include the signal space. To calculate the new value of the test (center) pixel, the pixels in this cluster are filtered using the median filter. The median filter is used on isolated color components to prevent the filtering process from being exploited by adjacent color components. In essence, this method employs those pixels that appear to contribute to the signal, thereby mitigating the impact of corrupted pixels in the filtering process. The current approach has a higher level of accuracy. The simulation results demonstrate that the proposed method outperforms state-of-the-art impulse noise reduction filters.

The proposed IMF-KM method is based on two key ideas: (a) categorizing the pixels in the sliding window into two groups, one that contributes to the signal and the other that contributes to the noise, and (b) performing median filtering over the pixel intensities that contribute to the signal in the isolation of the color components. The k-means clustering technique is used in step (a) to classify the pixels based on their spatial location of intensities. The median filter processes the cluster of intensities that contribute to the signal. This operation is executed in isolation over the colour components to avoid the impact of smoothing, softening, or smearing in one color component leveraging itself on the others.

Consider a sliding window $W = \{x_i, i = 1, \dots, n\}$ Containing n pixels. From W, the isolated color sliding windows is acquired as

$$W_r = \{x_{r,i}, i = 1, \dots, n\} \quad (11)$$

$$W_g = \{x_{g,i}, i = 1, \dots, n\} \quad (12)$$

$$W_b = \{x_{b,i}, i = 1, \dots, n\} \quad (13)$$

The approach described in the sequel relates to the red component and should be expanded to the green and blue components in the isolation of the color components. Consider the sliding window associated with the red component $W_r = \{x_{r,i}, i = 1, \dots, n\}$. The red component pixel intensities are displayed in this window. These intensity values are then grouped using the k-means approach with K = 2 and unsupervised learning. The k-means algorithm applied to the sliding window yields a responsibility vector,

$$R_r = \{j_i, j_i \epsilon (1,2), i = 1, \dots, n\} \quad (14)$$

$$R_r = \{j_{i=1,\dots,n} = 1\} \cup \{j_{i=1,\dots,n} = 2\} \quad (15)$$

$$R_r = R_r[1] \cup R_r[2] \quad (16)$$

The i^{th} responsibility variable j_i can take one of two values, 1 or 2, signifying the cluster number to which the i^{th} pixel intensity $x_{r,i}$ corresponds. The indices are contained in the set $R_r[1]$. As seen below, this allows us to divide the n pixels into two groups.

$$C_r[1] = \{x_{j,r} : j_i = 1\}, i = 1, \dots, n \quad (17)$$

$$C_r[2] = \{x_{j,r} : j_i = 2\}, i = 1, \dots, n \quad (18)$$

And then select the cluster with the highest pixel values according to

$$\eta_r = C_r[j] \; s.t. \; |C_r[j]| > |C_r[\, i \epsilon (1,2), i \neq j]| \quad (19)$$

where |.| signifies the cardinality of a set. Essentially, this approach divides the pixel intensities in the sliding window into two groups: the dominant set r, which is assumed to be in the signal space, and the second, which is assumed to be in the noise space. As the dominant group has higher pixel intensities spaced close together, they contribute to the image's signal while the other set adds to the noise. Then the median filtering can be applied to the pixel intensities in the dominating set and replace the red component of the center test pixel as

$$x_{c,r} = Median(\eta_r) \quad (20)$$

The process is repeated in isolation for the green and blue components, and the final test pixel is obtained as

$$x_c = (x_{c,r}, x_{c,g}, x_{c,b}) \quad (21)$$

The proposed Algorithm 1 describes the k-means algorithm used to cluster the pixel intensities in the red component and get the responsibility vector R_r in (14).

Algorithm 1: The k-means Algorithm

R_r = K – MEANS $[\{x_{i,r}, i = 1, \dots, n\}, K]$

Initialize centroids $c_k = 1, \dots, K$ at random.

While Flag Up **do**

Compute distance D(i,k)=d($x_{i,r} - c_k$),

i=1,…,n. k=1,…,K.

where d(.) is any distance measure.

Compute responsibility vector

$R_r = \{j_i = k : \arg min_k D(i,k), i = 1, \dots, n\}$

That is, the i[th] sample will belong to the j_i[th] cluster that minimizes its distance over the centroids.

Compute new centroids

$c_k = \sum x_{j_i=k,r} [\{j_i = k, i = 1, \dots, n\}], k = 1, \dots, K$

If $c_k, k = 1, \dots, k$ values unchanged, **then** Flag Down

end if

end while

IMF-KM method for the red component is outlined in Algorithm 2.

Algorithm 2: The proposed IMF-KM Algorithm

$x_{c,r} = IMF - KM[W_r = \{x_{i,r}, i = 1, \dots, n\}]$

Cluster the pixel intensities using the k-means algorithm and obtain the responsibility vector R_r.

$R_r = K - MEANS[\{x_{i,r}, i = 1, \dots, n\}, K = 2]$

Categorize the n pixels into two clusters

$C_r[j] = \{x_{j_{i,r}} : j_r = 1\}, i = 1, \dots, n. j = 1,2$

Determine the dominant cluster η_r according to (19)

The center test pixel is $x_{c,r} = Median(\eta_r)$

4. Experimental Results

This paper uses three test statistics to compare recent techniques' suggested IMF-KM against state-of-the-art vector median filters. The first is the root mean square error (RMSE), which is defined as

$$RMSE(X,Y) = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left\|X_{i,j} - Y_{i,j}\right\|^2} \quad (22)$$

where X and Y represent the original and filtered images, respectively. A small RMSE value indicates that the error between the filtered and original images is tiny. Hence, modest RMSE values are acquired. The second measure is the peak signal-to-noise ratio (PSNR) which is defined as

$$PSNR(X,Y) = 10\log_{10}\left(\frac{Max(X)^2}{MSE(X,Y)}\right) \quad (23)$$

where MSE (X, Y) is the filtered image's mean square error. A high PSNR is desired because it shows superior signal-to-noise recovery.

The third measure is the structural similarity index (SSIM) which is defined as

$$SSIM(X,Y) = \frac{(2\mu_x+\mu_y+c_1)(2C_{x,y}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (24)$$

where the image means are

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} X_i \quad (25)$$

$$\mu_y = \frac{1}{N}\sum_{i=1}^{N} Y_i \quad (26)$$

Where σ^2 denotes the variance and $C(.,.)$ denotes the covariance between the original and filtered images.

The SSIM value reflects the similarity between the original and the filtered pictures by combining perceptual variables such as luminance and contrast. A high SSIM value indicates that the original image was accurately reconstructed.

Fig. 3 is used to analyze the proposed denoising method experimentally.

Fig. 3 Original Test Image 1

Fig.4 depicts noisy and filtered images with various noise probabilities, and it can be seen that the last column, which corresponds to the suggested IMF-KM approach, outperforms the others.

Under an illustrative implementation of the disclosure, PSNR versus Noise Probability, RMSE versus Noise Probability, and SSIM versus Noise Probability for test image-1 are shown in Fig.5, Fig.6, and Fig.7. As can be observed, our IMF-KM method outperforms the joint vector approaches and the IVMDF approach in terms of accuracy (low RMSE), and the efficiency increases as the noise probability increases.

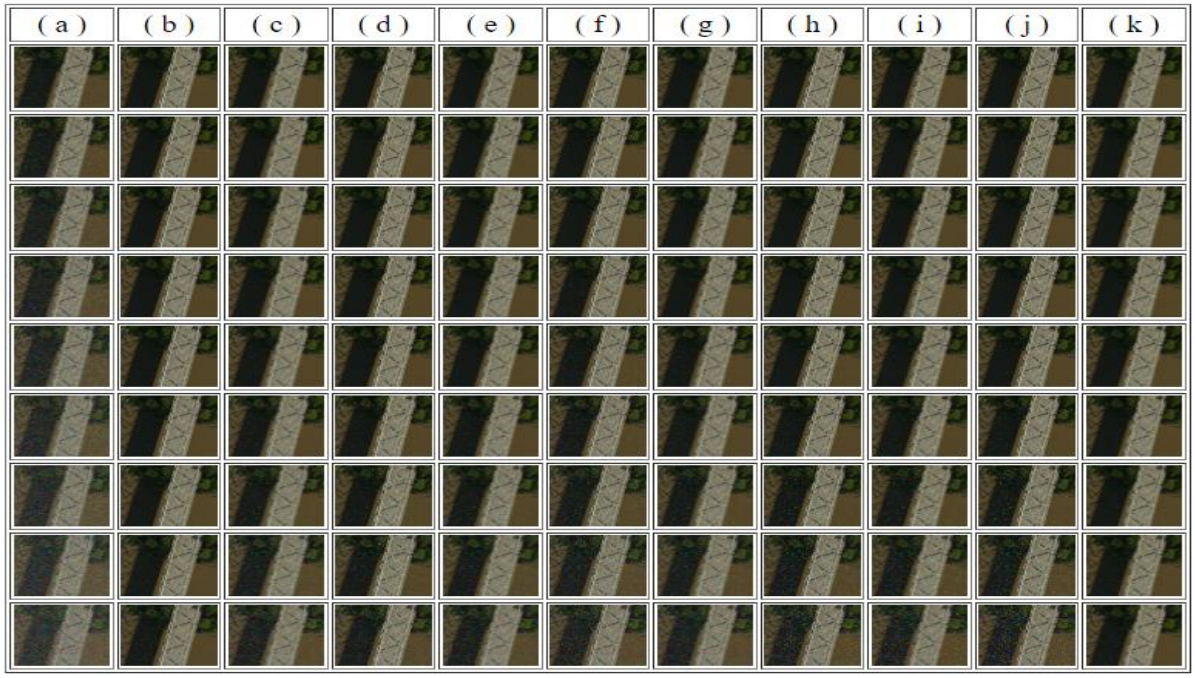

Fig. 4 Noise Reduction by various filters: Left to right: (a) Noisy image, (b) Median filter, (c) VMF, (d) BVDF, (e) DDF, (f) DVMF, (g) VMFDD, (h) ARWSF, (i) Alpha trim VMF, (j) ASTVMF, (k) IMF-KM. Top to bottom: Noise probability p = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, in accordance with an illustrative embodiment of the disclosure.

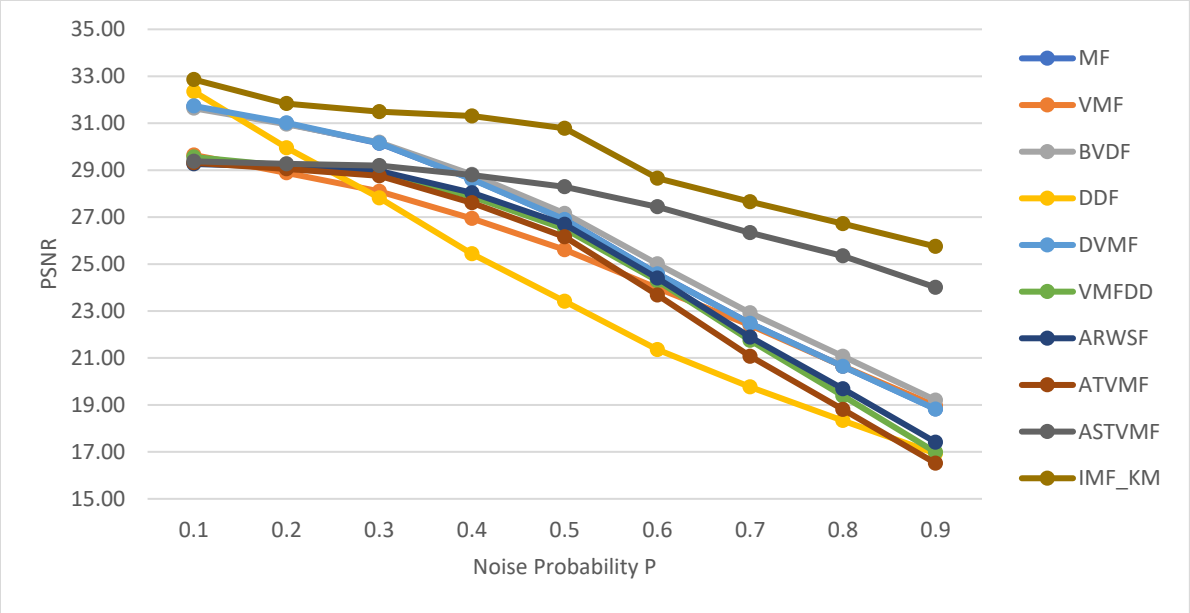

Fig. 5 The PSNR versus the Noise Probability for test image-1

Table 1. Shows the Psnr Values for the Images Shown in Fig.4

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	31.70	29.65	31.64	32.35	31.73	29.56	29.26	29.31	29.38	32.86
0.2	30.98	28.88	30.96	29.95	31.01	29.18	29.15	29.06	29.27	31.83
0.3	30.14	28.10	30.18	27.82	30.15	28.86	28.96	28.76	29.19	31.49
0.4	28.62	26.94	28.82	25.44	28.61	27.87	28.04	27.61	28.80	31.30
0.5	26.90	25.61	27.15	23.41	26.90	26.50	26.69	26.16	28.29	30.78
0.6	24.59	23.98	25.01	21.36	24.58	24.29	24.40	23.68	27.44	28.66
0.7	22.47	22.39	22.92	19.76	22.47	21.75	21.89	21.07	26.34	27.65
0.8	20.64	20.65	21.07	18.33	20.64	19.39	19.69	18.81	25.35	26.72
0.9	18.83	18.96	19.20	16.93	18.82	16.99	17.41	16.52	24.01	25.75

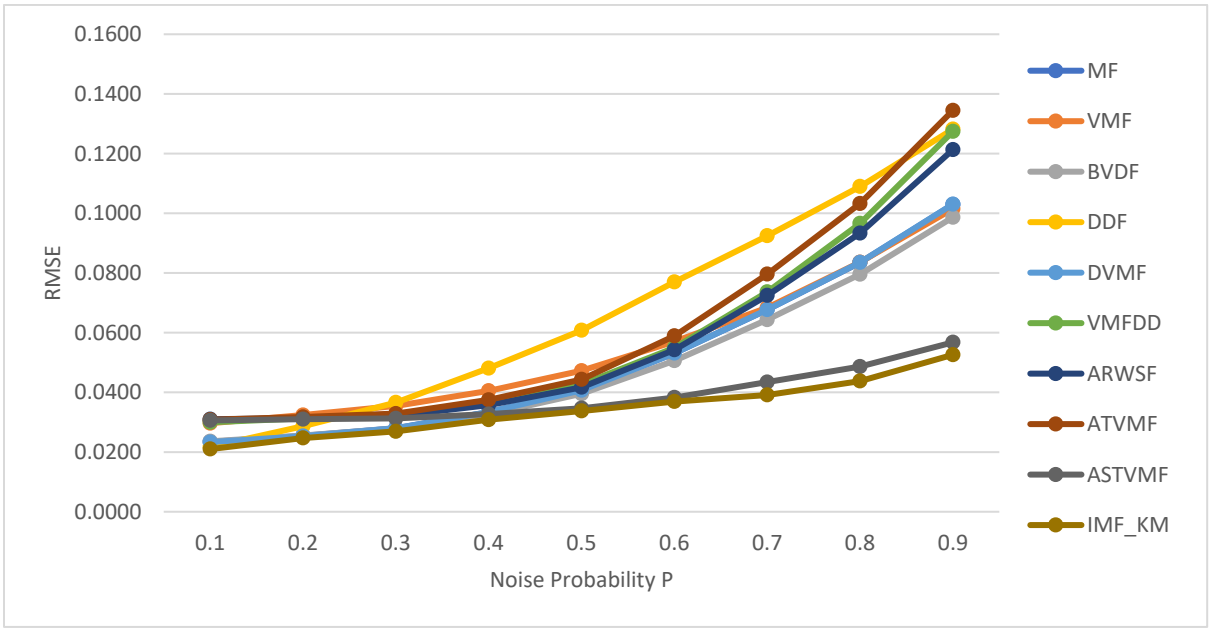

Fig. 6 The RMSE versus the Noise Probability for test image-1.

Table 2. Shows the RMSE values for the images shown in Fig.4

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	0.0234	0.0297	0.0236	0.0217	0.0233	0.0300	0.0310	0.0308	0.0306	0.0210
0.2	0.0254	0.0324	0.0255	0.0287	0.0254	0.0313	0.0314	0.0318	0.0310	0.0247
0.3	0.0280	0.0355	0.0279	0.0366	0.0280	0.0325	0.0321	0.0329	0.0313	0.0269
0.4	0.0334	0.0405	0.0326	0.0481	0.0334	0.0364	0.0357	0.0375	0.0327	0.0309
0.5	0.0407	0.0472	0.0395	0.0608	0.0407	0.0426	0.0417	0.0443	0.0347	0.0338
0.6	0.0531	0.0570	0.0506	0.0770	0.0531	0.0549	0.0543	0.0589	0.0383	0.0369
0.7	0.0677	0.0684	0.0644	0.0925	0.0677	0.0737	0.0725	0.0796	0.0434	0.0391
0.8	0.0835	0.0836	0.0796	0.1090	0.0835	0.0966	0.0934	0.1033	0.0486	0.0438
0.9	0.1029	0.1015	0.0987	0.1282	0.1031	0.1274	0.1213	0.1345	0.0568	0.0526

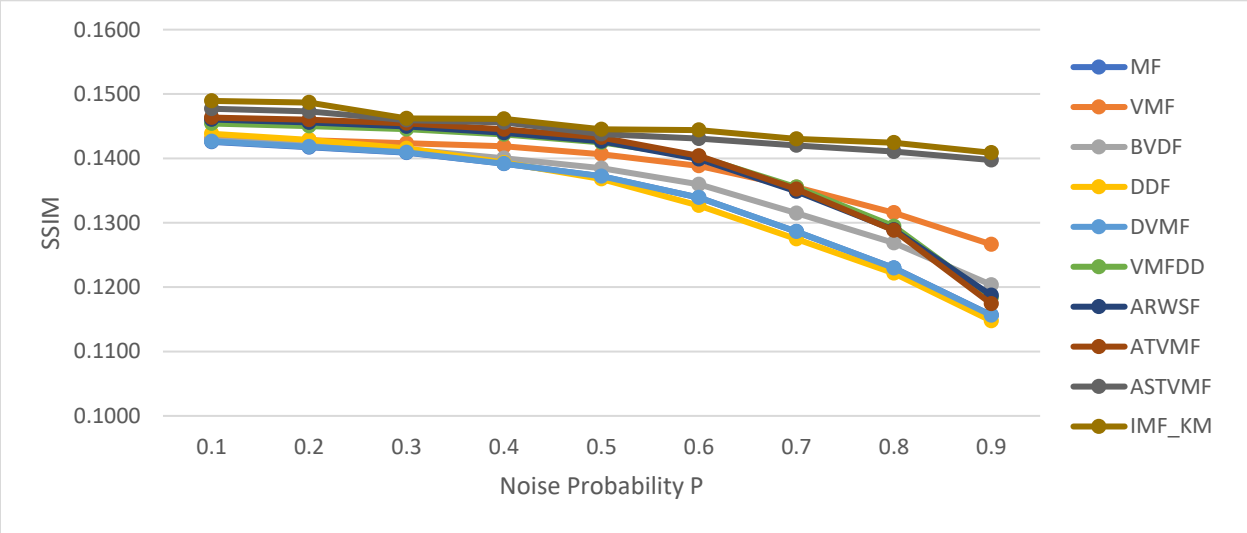

Fig. 7 The SSIM versus the Noise Probability for test image-1

Table 3. Shows the Ssim values for the Images Shown in Fig.4

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	0.1426	0.1431	0.1428	0.1438	0.1426	0.1454	0.1460	0.1463	0.1477	0.1489
0.2	0.1417	0.1428	0.1421	0.1429	0.1418	0.1450	0.1456	0.1460	0.1473	0.1487
0.3	0.1409	0.1423	0.1414	0.1415	0.1409	0.1445	0.1450	0.1454	0.1459	0.1462
0.4	0.1392	0.1419	0.1400	0.1393	0.1392	0.1437	0.1440	0.1445	0.1456	0.1461
0.5	0.1372	0.1406	0.1385	0.1368	0.1372	0.1425	0.1426	0.1432	0.1438	0.1445
0.6	0.1339	0.1388	0.1360	0.1327	0.1339	0.1403	0.1399	0.1404	0.1431	0.1444
0.7	0.1286	0.1355	0.1315	0.1275	0.1286	0.1355	0.1349	0.1352	0.1420	0.1430
0.8	0.1230	0.1316	0.1269	0.1222	0.1230	0.1294	0.1289	0.1288	0.1411	0.1424
0.9	0.1156	0.1266	0.1203	0.1148	0.1156	0.1184	0.1187	0.1174	0.1397	0.1409

Structural similarities can also be used to explain their superiority. At p= 0.2, the suggested IMF-KM approach outperforms the VMF method, but at p= 0.9, it becomes more efficient, reaching roughly 15% efficiency.

It demonstrates that isolated median filtering using k-means of just the valid pixels, i.e., those pixels in the signal space that allegedly carry meaningful information, improves impulse noise reduction.

.

Fig. 8 Test Image-2

Fig.9, Fig.10, Fig.11, and Fig.12 provide performance data for Test Images-2. Fig.10 depicts the PSNR versus the Noise Probability, Fig.11 depicts the RMSE versus the Noise Probability, and Fig.12 shows the SSIM versus the Noise Probability. Despite the suggested IMF-KM technique's vast advantage over the VMF and DVMF methods, the PSNR, RMSE, and SSIM performance data reveal that the VMF and DVMF methods compare well to the proposed IMF-KM approach.

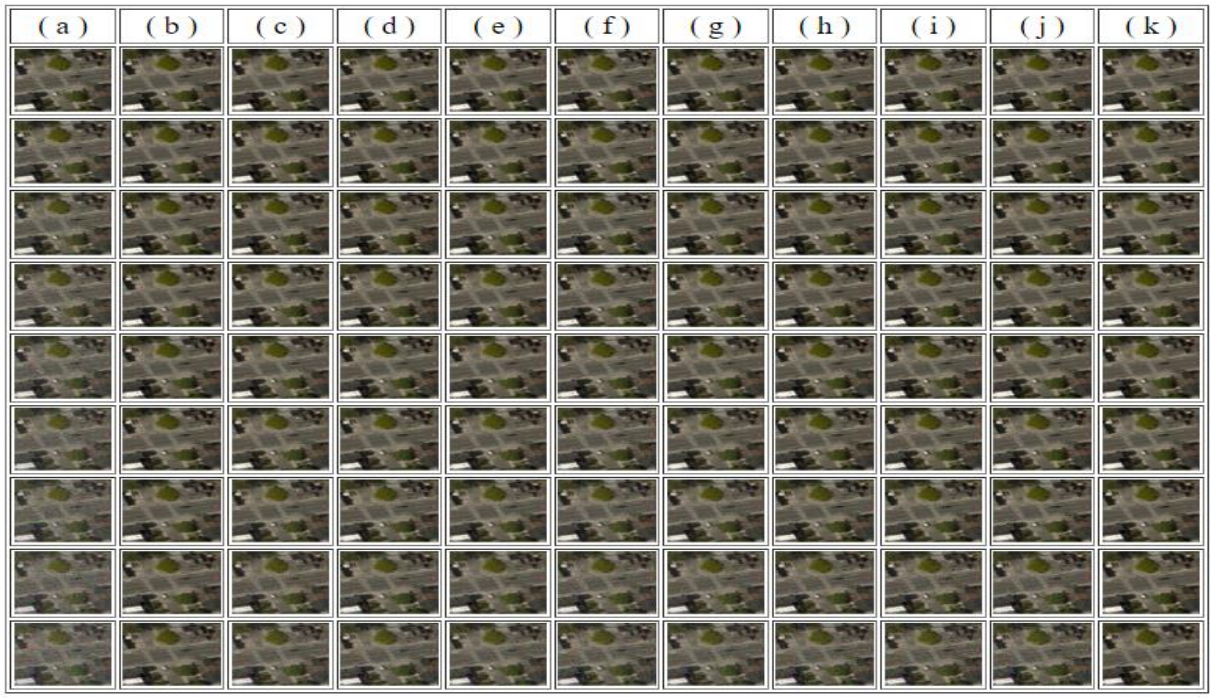

Fig. 9 Noise Reduction by various filters: Left to right: (a) Noisy image, (b) Median filter, (c) VMF, (d) BVDF, (e) DDF, (f) DVMF, (g) VMFDD, (h) ARWSF, (i) Alpha trim VMF, (j) ASTVMF, (k) IMF-KM. Top to bottom: Noise probability p = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, in accordance with an illustrative embodiment of the disclosure.

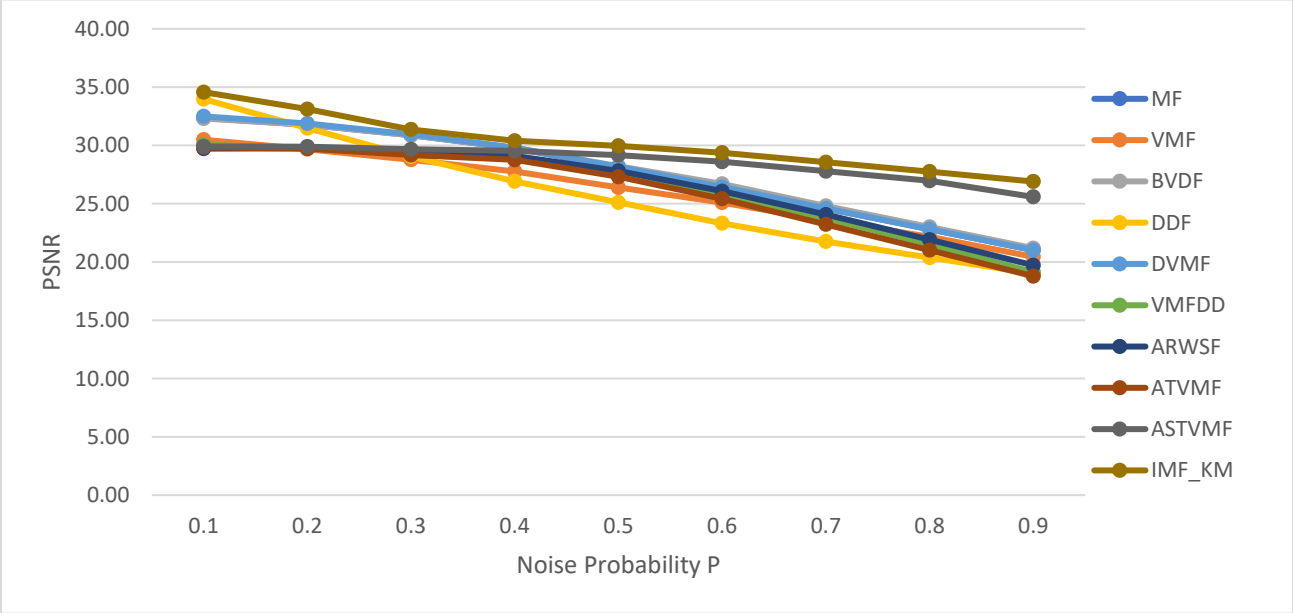

Fig. 10 The PSNR versus the Noise Probability for test image-2

Table 4. shows the PSNR values for the images shown in Fig.9

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	32.36	30.49	32.31	33.97	32.49	30.05	29.72	29.83	29.91	34.57
0.2	31.77	29.68	31.77	31.48	31.88	29.79	29.75	29.73	29.90	33.11
0.3	30.88	28.77	30.86	29.04	30.96	29.21	29.47	29.17	29.67	31.37
0.4	29.76	27.77	29.78	26.91	29.81	28.84	29.10	28.76	29.54	30.40
0.5	28.13	26.39	28.22	25.11	28.16	27.55	27.81	27.31	29.17	29.96
0.6	26.43	25.09	26.70	23.32	26.43	25.88	26.07	25.42	28.60	29.38
0.7	24.59	23.57	24.80	21.76	24.59	23.76	24.07	23.23	27.79	28.56
0.8	22.82	22.16	23.01	20.37	22.81	21.51	21.91	21.02	26.97	27.76
0.9	21.02	20.45	21.19	19.00	21.02	19.13	19.72	18.78	25.58	26.90

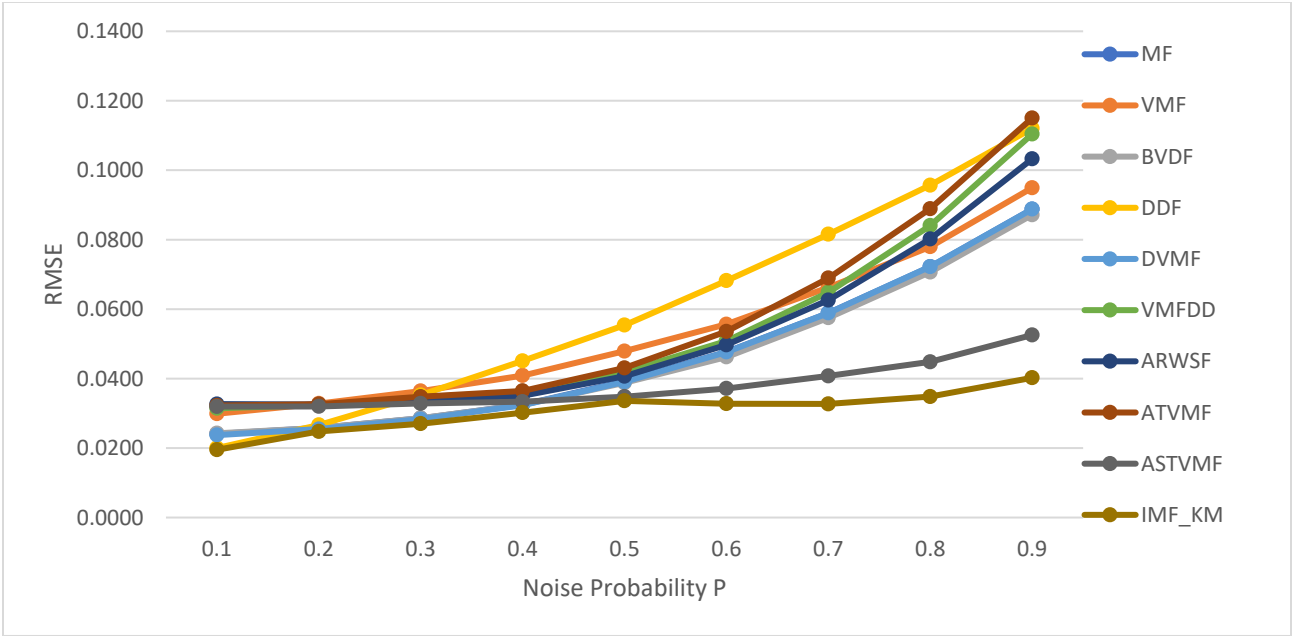

Fig. 11 The RMSE versus the Noise Probability for test image-2

Table 5. Shows the RMSE values for the images shown in Fig.9

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	0.0241	0.0299	0.0242	0.0200	0.0237	0.0314	0.0327	0.0322	0.0319	0.0195
0.2	0.0258	0.0328	0.0258	0.0267	0.0255	0.0324	0.0325	0.0326	0.0320	0.0248
0.3	0.0286	0.0364	0.0286	0.0353	0.0283	0.0346	0.0336	0.0348	0.0328	0.0270
0.4	0.0325	0.0409	0.0324	0.0451	0.0323	0.0361	0.0351	0.0365	0.0333	0.0302
0.5	0.0392	0.0479	0.0388	0.0554	0.0391	0.0419	0.0407	0.0431	0.0348	0.0336
0.6	0.0477	0.0556	0.0462	0.0682	0.0477	0.0508	0.0497	0.0536	0.0372	0.0328
0.7	0.0589	0.0663	0.0575	0.0815	0.0589	0.0648	0.0626	0.0689	0.0408	0.0327
0.8	0.0722	0.0780	0.0707	0.0957	0.0722	0.0840	0.0802	0.0889	0.0448	0.0348
0.9	0.0888	0.0949	0.0872	0.1120	0.0888	0.1104	0.1032	0.1150	0.0526	0.0402

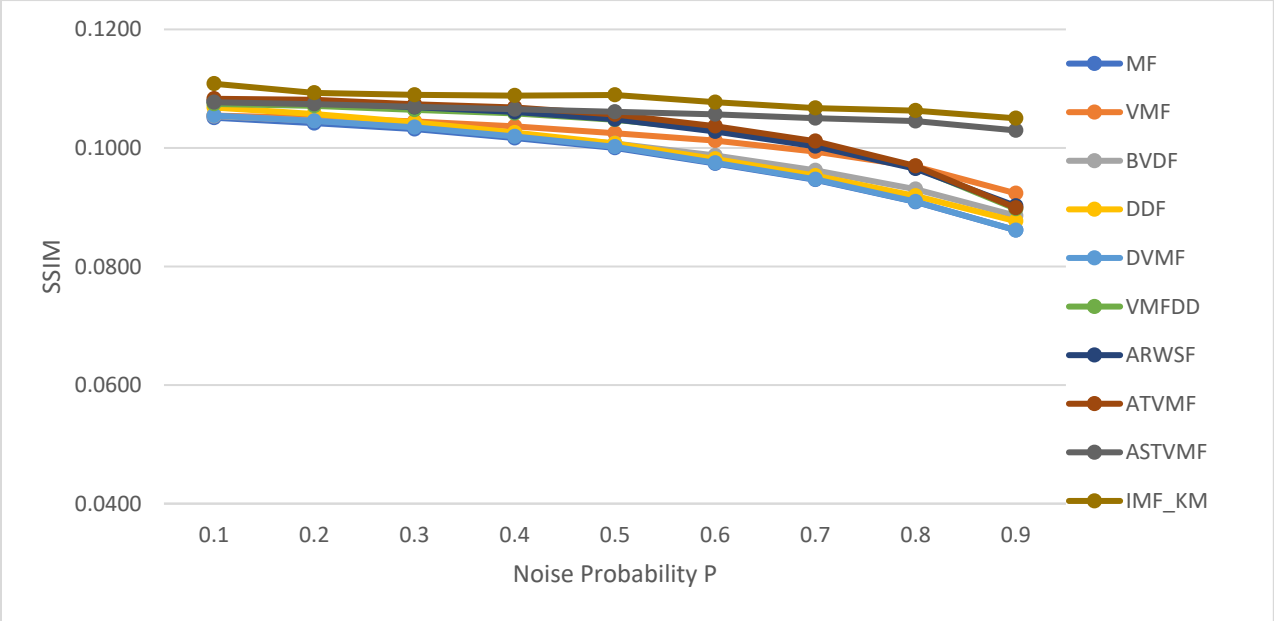

Fig. 12 The SSIM versus the Noise Probability for test image-2

Table 6. Shows the SSIM values for the images shown in Fig.9

	MF	VMF	BVDF	DDF	DVMF	VMFDD	ARWSF	ATVMF	ASTVMF	IMF_KM
0.1	0.1051	0.1055	0.1053	0.1068	0.1055	0.1074	0.1079	0.1083	0.1077	0.1108
0.2	0.1042	0.1051	0.1046	0.1057	0.1046	0.1071	0.1077	0.1081	0.1074	0.1093
0.3	0.1032	0.1045	0.1036	0.1043	0.1035	0.1064	0.1069	0.1074	0.1069	0.1090
0.4	0.1017	0.1036	0.1023	0.1026	0.1019	0.1058	0.1061	0.1068	0.1065	0.1088
0.5	0.1000	0.1025	0.1008	0.1007	0.1002	0.1048	0.1048	0.1057	0.1061	0.1089
0.6	0.0974	0.1012	0.0987	0.0981	0.0975	0.1030	0.1028	0.1037	0.1057	0.1077
0.7	0.0946	0.0994	0.0962	0.0953	0.0947	0.1007	0.1003	0.1011	0.1050	0.1067
0.8	0.0909	0.0969	0.0930	0.0919	0.0909	0.0967	0.0965	0.0970	0.1045	0.1063
0.9	0.0861	0.0923	0.0886	0.0876	0.0861	0.0897	0.0902	0.0899	0.1030	0.1050

In summary, our findings suggest that grouping pixels inside a test window to identify those containing information and subsequently filtering those pixels can reduce impulsive noise in satellite images. This innovation presented isolated median filtering with k means clustering for noise reduction in satellite images. The method consists of two steps: first, categorize the pixels in the sliding window into two groups, one that contributes to the signal and the other that contributes to the noise, and then perform median filtering over the pixel intensities that contribute to the signal, while isolating the color components. The primary benefit of this concept is that by median filtering just the usable pixels in the isolation of the color components, smoothing stays local and does not affect the other color components, resulting in better image reconstruction accuracy.

5. Conclusion

This paper proposed a novel methodology for denoising impulse noise in satellite Images through an Isolated Vector Median Filter with k-means clustering. Traditional filtering techniques apply a noise reduction algorithm to the center pixel of a well-chosen window that iteratively slides along the entire image, most often the vector median filtering approach and its variants. The technique described in this invention works by dividing the pixels in the selected window into two groups, one for pixel intensities in the signal space and the other for pixel intensities in the noise space. On the other hand, the current technique clusters the pixels in the selected window into two groups, one for pixel intensities in the signal space and the other for pixel intensities in the noise space. This paper demonstrated the superiority of the proposed method using satellite images and several statistical measures.

6. References

[1] Yogesh, V. and Yogendra, K, "Removal of Salt and Pepper Noise From Satellite Images," *International Journal of Engineering Research & Technology (IJERT)*, vol.2, pp.2051-2058.

[2] Pitas, I. and Venetsanopoulos, A.N, "Nonlinear Digital Filters: Principles and Applications ," Springer Science & Business Media.

[3] Sicuranza, G., "Nonlinear Image Processing. Elsevier," vol.84, 2000.

[4] Nodes, T. and Gallagher, N, " Median Filters: Some Modifications and Their Properties," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol.30, no.5 pp.739-746.

[5] Khryashchev, V.V., Priorov, A.L., Apalkov, I.V. and Zvonarev, P.S, "Image Denoising Using Adaptive Switching Median Filter," In *IEEE International Conference on Image Processing*,vol. 1, pp. I-117, 2005. IEEE.

[6] Yin, L., Yang, R., Gabbouj, M. and Neuvo, Y, "Weighted Median Filters: A Tutorial," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol.*43*, no.3, pp.157-192, 1996.

[7] Chanu, R. and Singh, K.M, " Vector Median Filters—A Survey," *International Journal of Computer Science and Network Security*, vol.16, no.12, pp.66-84, 2016.

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

[9] Tukey, J.W., 1974. Nonlinear (Nonsuperposable) Methods for Smoothing Data. *Proc. Cong. Rec. EASCOM'74*, pp.673-681, 1974.

[10] Xu, Q., Zhang, Q., Hu, D. and Liu, J, " Removal of Salt and Pepper Noise in Corrupted Image Based on Multilevel Weighted Graphs and IGOWA Operator," *Mathematical Problems in Engineering*, 2018.

[11] Kumar, N.R. and Kumar, J.U, "A Spatial Mean and Median Filter for Noise Removal in Digital Images," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol.4, no.1, pp.246-253, 2015.

[12] Khryashchev, V., Kuykin, D. and Studenova, A, " Vector Median Filter with Directional Detector for Color Image Denoising.," In *Proc. of the World Congress on Engineering*, vol. 2, pp. 1-6, 2011.

[13] Plataniotis, K.N., Androutsos, D. and Venetsanopoulos, A.N, "Vector Directional Filters: An Overview. In *CCECE'97*," *Canadian Conference on Electrical and Computer Engineering. Engineering Innovation: Voyage of Discovery. Conference Proceedings* , vol. 1, pp. 106-109, 1997. IEEE.

[14] Trahanias, P.E. and Venetsanopoulos, A.N, " Vector Directional Filters-A New Class of Multichannel Image Processing Filters," *IEEE Transactions on Image Processing*, vol.2, no.4, pp.528-534, 1993.

[15] Lukac, R, " Adaptive Color Image Filtering Based on Center-Weighted Vector Directional Filters," *Multidimensional Systems and Signal Processing*, vol.15, no.2, pp.169-196, 2004.

[16] Choppala, P., Meka, J.S. and PVGD, P.R, "Vector Isolated Minimum Distance Filtering for Image De-Noising In Digital Color Images," *International Journal of Recent Technology and Engineering*, vol.*8*, no.4, pp.2401-2405, 2019.

[17] Choppala, P., Meka, J.S. and PVGD, P.R, "Isolated Vector Median Filtering for Noise Reduction in Digital Color Images," 2020.

[18] Smolka, B., Malik, K. and Malik, D, " Adaptive Rank Weighted Switching Filter for Impulsive Noise Removal in Color Images," *Journal of Real-Time Image Processing*, vol.10, no.2 pp.289-31, 2015.

[19] Harini N, Shaik Majeeth S, Aswanth Kumar G and Abinaya J, "CT Image Denoising using DTCWT with Level Dependent Thresholding," *International Journal of Electronics and Communication Engineering*, vol. 5, no. 8, pp. 14-21, 2018. *Crossref*, https://doi.org/10.14445/23488549/IJECE-V5I8P103

[20] D.C. Shubhangi, Anita Totapnor, "Survey On Noise Detection Method", *International Journal of Engineering Trends and Technology* 67.8 (2019): 19-21.

[21] Johnsymol Joy, "Overview of Different Data Clustering Algorithms for Static and Dynamic Data Sets" *SSRG International Journal of Computer Science and Engineering*, vol. 5, no. 3 pp. 1-3, 2018. *Crossref*, https://doi.org/10.14445/23488387/IJCSE-V5I3P101