Wireless Personal Communications
https://doi.org/10.1007/s11277-021-08314-5

An Adaptive Weighted Min-Mid-Max Value Based Filter for Eliminating High Density Impulsive Noise

Nikhil Sharma[1] · **Prateek Jeet Singh Sohi**[1] · **Bharat Garg**[1]

Accepted: 18 February 2021

Abstract
This paper presents a novel algorithm to filter impulsive noise at very high noise density ($\geq$ 85%). The proposed algorithm initially makes an accurate decision and selects a window which has sufficient information for denoising. Within the selected window, the proposed algorithm computes maximum, minimum, middle values along with their weights to restore noisy pixel. The performance of proposed filter is evaluated on natural and medical images with varying noise density. The proposed filter showed tremendous performance at high noise densities in terms of quantitative metrics and visual representation. Even at noise densities as high as 97% and 99%, the proposed filter is able to retrieve the details of the image. The proposed filter on an average improves the peak signal to noise ratio value by 10% in medical images over the existing.

Keywords Impulse noise · Salt and pepper noise · Median filters · Mean filter · Image processing

1 Introduction

Images are often corrupted with random noise when it is transmitted through a noisy medium. The major constituent of this noise is salt and pepper (SAP) or impulse noise. This is a phenomenon, in which the pixels in the image randomly obtain extreme values $e.g.$ 0 and 255 for gray-scale images. This in turn, results in transfer of false information. To overcome this issue, various linear and non-linear filters are introduced. It is observed that non-linear filters specially median filters produce better outputs [1,2]. This is due to the fact that most pixels have high correlation with their neighbouring pixels and median filter restores the corrupted pixel by most appropriate original pixel.

Though, in some cases such as, when the noise corruption is very high (say $\geq$ 85%) mean filter performs better because information is less and mean of some particular pixels provides close results to actual pixels and hence better performance [3,4]. Further, the filtering algorithms having effective decisions also provide good results [5–9]. These filters make

✉ Bharat Garg
bharat.garg@thapar.edu

[1] Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Published online: 17 March 2021

a specific choice, whether to use mean or median of the pixels for noise removal. Other categories include probability-based filters, adaptive mean/median filters and filters with a combinational logic of mean and median. The next section presents brief comparative analysis of state-of-the-art filters for the removal of SAP noise.

The rest of the paper is organized as follows. Section 2 provides detailed literature review whereas, Sect. 3 presents proposed filtering approach. The simulation results and comparative analysis are given in Sect. 4 while Sect. 5 concludes the paper.

2 Related Work

Several filtering techniques are presented to remove impulsive noise such as the mean, median, weighted mean [2] and interpolation. In median filters, the median of a window centred around the noisy pixel is used to replace noisy-pixel. At very small noise density (ND) ($i.e. \leq 20\%$), median of this window is non-noisy but it may be noisy at high ND ($i.e. \geq 90\%$). An adaptive window median filter increases the window size based on the the localized ND to achieve non-extreme pixels for denoising [10]. This filter has the advantage of restoring original pixel using median from low to medium noise density ranges. At high noise density, the use of large window size reduces the edge information and the resulting filtered image is blurred. A modified decision based unsymmetrical trimmed median filter (MDBUTMF) presented in [5] uses the median of all uncorrupted values in a 3×3 window to restore the corrupted pixel. Further, the mean of all pixels restores the noisy pixel when all pixels of selected window are noisy which leads to the poor image quality.

A decision based coupled window median filter (DBCWMF) [11] either restores corrupted pixels by median value of all uncorrupted pixels of the 3×3 window or increases window size if no uncorrupted pixels are found in the selected window. The maximum window size is limited to 9×9 for median calculation. But if a pixel still remains unprocessed then mean of all pixels in the 9×9 window is calculated for replacing noisy pixels. However the consideration of variable window improves the performance of the filter, it provides blurred output images at high ND where the window size increases beyond 7×7. A fast switching based mean-median filter (FSBMMF) [12] is another state-of-the-art filter which has promoted research into to combinational mean and median filters. By introducing the idea of using previously processed pixels for mean calculation to deal with corruption that is left unprocessed. Though it is a novel approach, the only problem with FSBMM filter is that when the boundary pixels are not rectified using the medians it replaces them with the previous pixels in their respective row/column. Thus under high noise density conditions we observe repletion of pixels which in turn leads to streaking effect.

Filters such as unsymmetric trimmed mid point filter (UTMP) and unsymmetric trimmed median filter (UTMF) are presented as an extension of decision based median filter to achieve noise density range specific filtering [13]. The UTMP which evaluates value of denoised pixel using median performs well under high noise density condition. Whereas, the UTMF which evaluates value of denoised pixel using mean provides better denoising at lower noise densities. The most prominent problem with these filters is that their inability to provide good quality images over the entire range of noise density. Attempts to use other approximation techniques lead to the introduction of recursive spline interpolation filter (RSIF) [14]. This filter is similar to MDBUTMF, where in place of taking the median of uncorrupted pixels it uses cubic spline interpolation technique. It is observed that being a mathematically superior technique it produces better results from low to medium noise density ranges. But fails at high

ND due to lack of information availability for estimation. Moreover it requires at least two uncorrupted pixels for interpolation which are rarely available at high ND ($\geq$ 90%). Other major drawback of most the interpolation techniques is that they require a large execution time.

Keeping the above problems in mind various weighted filters were introduced. A popular category amongst these is the directional weighted filters. They fundamentally differ from other filters in the way by which corruption is identified. At first difference between the selected pixels and the pixels in the window are calculated. The directional sum of product of their differences with their weighting factor is calculated. If the minimum directional sum is less than threshold then it is considered to be a noise free pixel. Otherwise, the chosen pixel is considered to be noisy.

Adaptive switching weighted median filter [15] uses mainly two conditions for noise detection. The first one is that the pixel should either be 255 or 0 from a window whose mean is not equal to 255 and 0 respectively and the other one is if the number of uncorrupted pixels in the selected 3×3 window is odd then the corrupted pixel is replaced by median of all uncorrupted values. Otherwise, the most repeated value in the particular window is repeated one more time and the process is repeated. Further, it repeats nearest noise-free pixels (NFPs) to restore the noisy pixel which leads to streaking effect at high noise density.

A loss function for denoising convolutional neural network (DnCNN) for SAP is presented in [16]. Further, a mask-involved loss function is designed and utilized with DnCNN to achieve more effective denoising. An adaptive total variation (TV) regularization model based on general regularized image restoration model with L1 fidelity is presented for handling SAP noise in [17]. Further, the paper estimated regularization parameter based on SAP noise characteristics and implemented the adaptive TV-L1 regularization model using primal dual gradient method. This filter provides an artefact free denoised image while preserving edges of images.

A noise density range sensitive (NDRS) filtering algorithm is presented in [18] where the noisy pixel is replaced by mean, median or pre-processed values based on value of ND. The NDRS filter provides a unique approach restoring corrupted pixels even at very high ND. A four stage median filtering algorithm (FSMA) is presented in [19]. The FSMA algorithm performs median filtering with smaller window size, large window size, running average and replacement by previously processed pixels at first, second, third and fourth stages, respectively. The algorithm moves to next stages only when the current stage fails to denoise the candidate noisy pixel. The FSMA algorithm can effectively denoise the image by considering the first two stages at low noise density while it may requires all stages at higher noise density. This filter eliminates SAP noise along with better edge preservation as it considers NFPs while estimating the value of noisy pixels. An adaptive trimmed median (ATM) filter is presented in [20] can effectively denoise image corrupted with high noise density. This filter restores noisy pixel by computing median of NFPs of adaptive size window and by computing via interpolation based procedure at low and high noise densities, respectively. Further, this algorithm denoise candidate pixel using nearest processed pixel procedure for the rare scenarios of boundary pixels. A multi procedure min–max average pooling based filter (MMAPF) is presented in [21]. In this filter, after preprocessing, the noisy image segmented into two parts and passed through multiple layers of max and min pooling. Finally, to remove all residual noise, the algorithm combines the processed images from the previous procedures and performs average pooling. An adaptive min–max value based weighted median (AMMWM) filter is presented in [22]. This filter first determines two highly correlated groups of uncorrupted pixels using minimum and maximum value of the current window and then determines weighted medians of these groups to restore the noisy pixel. The filter increases

window size if the current window fails to provide any uncorrupted pixels. However the filter can restores noisy pixel at high ND, it exhibits high computational complexity.

Another weighted filter presented in [23] is the three-valued weighted filter (TVWA). It starts by considering a 3×3 window across the noisy pixel. Then all the noisy-free pixels in the window are divided into three groups that are made on the basis of their correlation with the middle, maximum and minimum values. The number of pixels of these groups determine their weighting factors. Finally, the noisy pixel is replaced by the weighted mean. The window size is increased to 5×5 and same process is repeated, if the selected 3×3 window is completely corrupted. The maximum size of the window considered is 7×7 which is the major drawback of this filter and leads to blurred output images. Apart from the weighted filters cascaded decision based filter is another type of filtering technique that have produced noteworthy results. It can be divided into two sub-sections. In the first part, the corrupted pixel is replaced by the median of all uncorrupted values in the 3×3 window. In the second part if possible the directional median is used to substitute corruption otherwise mean of all uncorrupted value in the window are used. When no uncorrupted pixels are present in the selected window, the window size is increased till uncorrupted pixels are not available.

To improve the image quality at higher noise density, the next section presents an adaptive weighted min-mid-max value based filter that effectively eliminates salt-and-paper noise at high noise density while preserving the image edge information.

3 Proposed Adaptive Weighted Min-Mid-Max Filtering (AWM3F) Approach

This section presents proposed AWM3F approach for the removal of impulse noise at high noise density. The following subsections first present the flowchart followed by algorithm and illustration of proposed filtering technique with the help of example.

3.1 Flowchart for Proposed Filter

The flowchart of proposed filter is shown in Fig. 1. The proposed filter first initializes the flag, threshold variables and does symmetric padding. Each pixel of the noisy image is first checked and filtering process is done for noisy pixel only. If current pixel is noisy, a small 3×3 window is considered. If the number of NFPs is less than predefined threshold α_1, the size of window is increased to 5×5. Similarly, if number of NFPs are less than threshold α_2, window size is increased to 7×7. The denoising process is done using original pixels of the appropriate widow where number of NFPs is greater than predefined threshold. In the denoising process, maximum, minimum and middle values from the selected window are computed. Further, the NFPs of the selected window are segmented into three groups based on their closeness to the maximum, minimum and middle values. The size of each segment determines the weight of these segments which are used to compute the denoised pixel value using the following expression.

$$Pix \leftarrow P_{\delta_1} * \beta_{max} + P_{\delta_2} * \beta_{min} + P_{\delta_3} * \beta_{mid} \tag{1}$$

where P_{δ_1}, P_{δ_2}, and P_{δ_3} represent the weight of maximum (β_{max}), minimum (β_{min}), and middle (β_{mid}) value respectively. The denoising process discussed above provides filtered image which may contains few noisy pixels at high noise density. Therefore, threshold variables are updated with new values and then similar steps are repeated. Further in the second

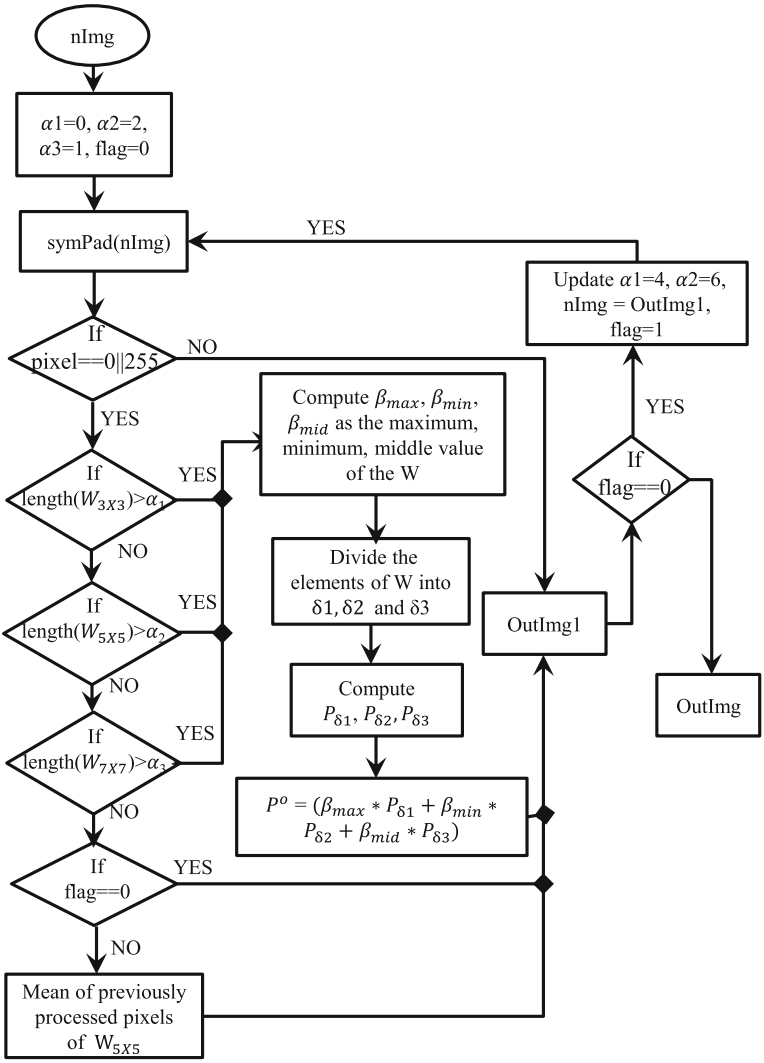


Fig. 1 Flowchart for proposed filter

iteration, if the number of NFPs are less the given threshold value in 7×7 window then mean of 5×5 is considered as denoised value.

3.2 Algorithm of the Proposed Filter

The pseudo code of the proposed algorithm is given in Algorithm 1. The algorithm first initializes the threshold variables (α_1, α_2 and α_3) and window size parameter (m). The proposed algorithm detects current pixel to be noisy or noise free and performs denoising process for each noisy pixel.

Algorithm 1 AWM3F($nImg$, $OutImg$)

1: **Input** $nImg$ ▷ *Input noisy Image of size* $M \times N$
2: **Output** $OutImg$ ▷ *Output filtered Image*
3: ***Initialize:*** $Flg = 0, m = 1, \eta \in \{0, 255\}$;
4: ***Initialize:*** $\alpha_1 = 0, \alpha_2 = 2, \alpha_3 = 1$;
5: $pnImg \leftarrow SymPad(nImg)$ ▷ *Symmetric padding*
6: ***for*** *each* $P_{i,j}$ ***do***
7: ***if*** $P_{i,j} \notin \eta$ ***then***
8: $P^o_{i,j} \leftarrow P_{i,j}$ ▷ $P^o_{i,j}$ *represents denoised pixel*
9: ***else***
10: $W_c \leftarrow W^{nf}_{(2m+1)\times(2m+1)}$
11: ***if*** $length(W_c) > \alpha_m$ ***then***
12: $\beta_{max} \leftarrow Max(W_c)$;
13: $\beta_{min} \leftarrow Min(W_c)$;
14: $\beta_{mid} \leftarrow MID\text{-}CAL(\beta_{max}, \beta_{min}, W_c)$;
15: $P^o_{i,j} \leftarrow PIX\text{-}CAL(\beta_{max}, \beta_{mid}, \beta_{min}, W_c)$;
16: ***else***
17: $m \leftarrow m + 1$;
18: ***if*** $m > 3$ ***then***
19: $m = 1$;
20: **go to** *Step 24;*
21: ***end if***
22: **go to** *Step 9;*
23: ***end if***
24: ***if*** $(Flg \neq 0)$ & $(P_{i,j} \in \eta)$ ***then***
25: $P^o_{i,j} \leftarrow \mu(W^{PP}_{5\times5})$
26: ***end if***
27: ***end if***
28: ***end for***
29: ***if*** $Flg == 0$ ***then***
30: $nImg = OutImg$; $Flg = Flg + 1$; $\alpha_1 = 4$; $\alpha_2 = 6$;
31: **go to** *Step 5;*
32: ***end if***
33: ***Return*** $OutImg$;

Initially, a small widow (3×3) centred around noisy pixel is considered and number of noisy-free pixels in the window are evaluated (line number 9 of Algorithm 1). If the number of NFPs are greater than given threshold (α_1), maximum (β_{max}) and minimum (β_{min}) value are computed (line number 11). Using maximum and minimum values, middle value (β_{mid}) is calculated using function MID-CAL. Finally, with the help of maximum, minimum and middle values, denoised pixel value is computed using function PIX-CAL. If the number of NFPs in the window are less then threshold, the size of window is increase and number of NFPs new window is compared against respective threshold. The size of window is increased up to 7×7. If the number of NFPs in 7×7 is less than α_3, the noisy pixel left unchanged in the first iteration. In the second iteration, all step of first iterations are repeated with updated value of thresholds (line number 28 and 29 of Algorithm 1). If the number of NFPs in maximum sized window are smaller than threshold (α_3), the mean of the original pixels of 5×5 window restores the noisy pixel.

Algorithm 2 Function MID-CAL($\beta_{max}, \beta_{min}, W_c$)

Initialize: $\gamma_1 \leftarrow 0$; $\gamma_2 \leftarrow 0$;
Initialize: $\delta_1 \leftarrow [\,]$; $\delta_2 \leftarrow [\,]$;
for each P_i **do**
  $\gamma_1 \leftarrow |\beta_{max} - P_i|$;
  $\gamma_2 \leftarrow |\beta_{min} - P_i|$;
  if $\gamma_1 \leq \gamma_2$ **then**
    $\delta_1 \leftarrow [\delta_1, P_i]$;
  else
    $\delta_2 \leftarrow [\delta_2, P_i]$;
  end if
end for
$P_{\delta_1} \leftarrow$length(δ_1)/length(W_c);
$P_{\delta_2} \leftarrow$length(δ_2)/length(W_c);
$\beta_{mid} = P_{\delta_1} * \beta_{max} + P_{\delta_2} * \beta_{min}$;
Return: β_{mid};

Algorithm 3 Function PIX-CAL($\beta_{max}, \beta_{mid}, \beta_{min}, W_c$)

Initialize: $\gamma_1 \leftarrow 0$; $\gamma_2 \leftarrow 0$; $\gamma_3 \leftarrow 0$;
Initialize: $\delta_1 \leftarrow [\,]$; $\delta_2 \leftarrow [\,]$; $\delta_3 \leftarrow [\,]$;
for each P_i **do**
  $\gamma_1 \leftarrow |\beta_{max} - P_i|$;
  $\gamma_2 \leftarrow |\beta_{min} - P_i|$;
  $\gamma_3 \leftarrow |\beta_{mid} - P_i|$;
  if $\gamma_1 \leq \gamma_2$ and $\gamma_1 \leq \gamma_3$ **then**
    $\delta_1 \leftarrow [\delta_1, P_i]$;
  else if $\gamma_2 \leq \gamma_1$ and $\gamma_2 \leq \gamma_3$ **then**
    $\delta_2 \leftarrow [\delta_2, P_i]$;
  else
    $\delta_3 \leftarrow [\delta_3, P_i]$;
  end if
end for
$P_{\delta_1} \leftarrow$length(δ_1)/length(W_c);
$P_{\delta_2} \leftarrow$length(δ_2)/length(W_c);
$P_{\delta_3} \leftarrow$length(δ_3)/length(W_c);
$Pix \leftarrow P_{\delta_1} * \beta_{max} + P_{\delta_2} * \beta_{min} + P_{\delta_3} * \beta_{mid}$;
Return: Pix;

3.3 Illustration of the Proposed Filter

The denoising efficacy of the proposed algorithm is demonstrated using an example shown in Fig. 2. An image segment of size 7×7 is considered and noisy image with 90% noise density is generated. This noisy image segment is filtered using proposed algorithm as shown in figure. The first noisy pixel (0) exhibits two NFPs (158, 158) while considering 3×3 window when image is symmetrically padded. These number of NFPs are greater than threshold (α_1), therefore, denoising process is started. The maximum, minimum value are calculated which become equal to $\beta_{max} = 158$ and $\beta_{min} = 158$. Using function MID-CAL, middle value (β_{mid}) is calculated which also becomes equal to 158. Finally, the denoising value of the pixel is calculated by passing β_{max}, β_{mid} and β_{min} to PIX-CAL function. The computed denoised value (158) replaces the noisy pixel as shown in Fig. 2. Same process is repeated to denoise all noisy pixels.

The next section presents an analysis of simulation results to evaluate the efficacy of the proposed filter over the existing.

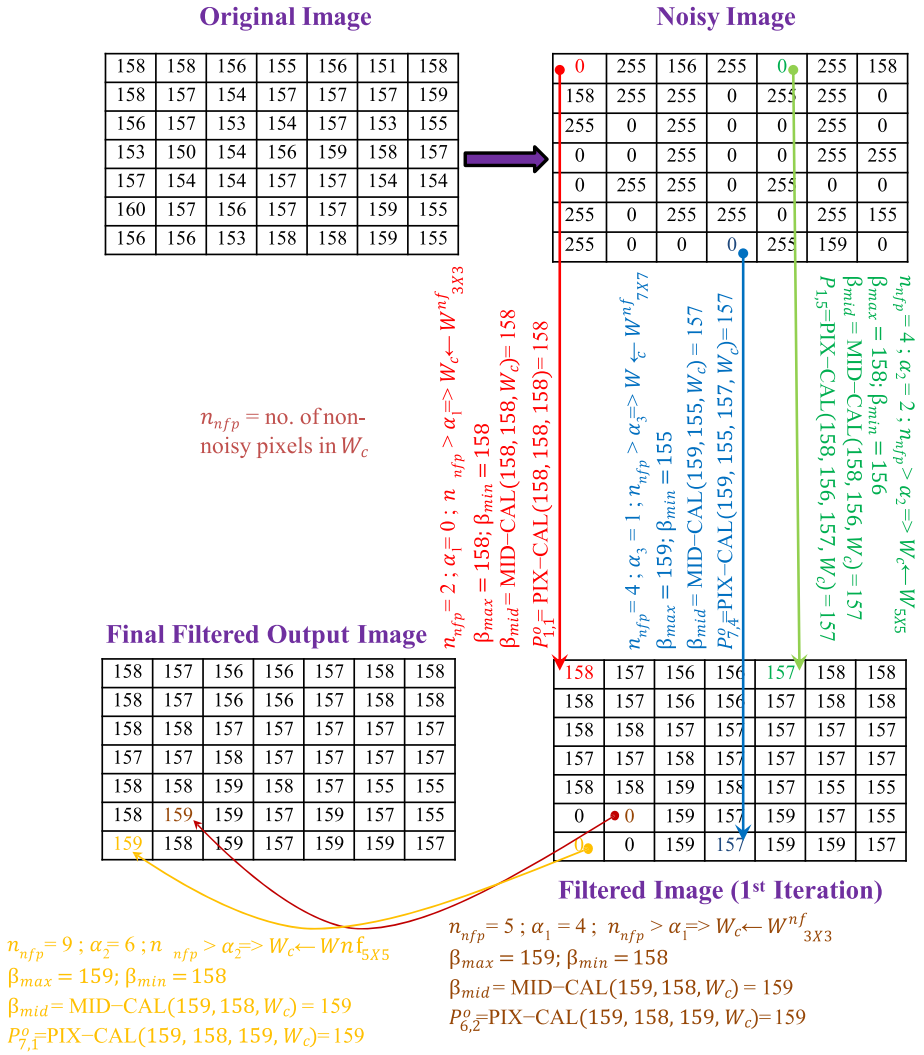


Fig. 2 Illustration of proposed filtering approach

4 Simulation

The performance of the proposed and existing filters like decision-based median filter (DBAMF), MDBUTMF [5], FSBMMF [12], RSIF [14], TVWA filter [23], adaptive switching weighted median filter (ASWMF) and different applied median filter (DAMF) are tested on various benchmark images. These images are corrupted with impulse noise with varying noise density from 85 to 99%. For the quantitative analysis, the peak signal to noise ratio (PSNR) and structural similarity index (SSIM) values are calculated. The mathematical expression of PSNR is given by Eq. 2.

$$PSNR = 10log_{10}(MAX^2/MSE) \tag{2}$$

where MSE is mean square error and Max is maximum pixel value of the image. In case of grayscale 8-bit images, the value of Max is 255.

$$MSE = \frac{1}{M * N} \sum_{i=1}^{M} \sum_{j=1}^{N} (x_{i,j} - y_{i,j}) \tag{3}$$

The parameter M and N represent dimensions of the image while x and y represent input and output images respectively.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{4}$$

where μ_x (σ_x) and μ_y (σ_y) represent mean (standard deviation) value in x and y directions, respectively. Further, C_1 and C_2 constants are considered to limit the SSIM value to 1.

The following subsections provide the qualitative and quantitative analysis of the proposed filter over the existing with different benchmark images.

1. Gray scale Lena image (512 × 512).
2. Four coloured images including Peppers, Fruits, Lena and Aeroplane each of size 512 × 512.
3. Seven grayscale images including Boat, Zelda, Lena, Baboon, Cameraman, Barbara and Goldhill each of size 512 × 512.
4. Medical images include X-ray (Lungs) image.

4.1 Simulation Results Analysis with Lena Image

Table 1 summarizes quality metrics of the proposed and existing filters using Lena image (512 × 512) with ND varied from 85 to 99%. It can be observed from the simulation results that the proposed filter has a superior performance over other filters at higher ND (even over 95% ND). Even at noise densities as high as 97% and 99% the proposed filter provides high PSNR values. Figure 3 shows the plot for the comparative analysis. The trend clearly demonstrates that the PSNR of the proposed filter is better than other existing filters. Further, for visual quality analysis, Fig. 4 shows the restored Lena images with the proposed and existing filters corrupted with 91% SAP noise density. It reveals that the proposed filter can recover the details of the image better than the existing filters even at very high noise density.

4.2 Comparative Analysis with Coloured Benchmark Images

Table 2 summarizes the value of average PSNR and SSIM of four different coloured images (including Peppers, Lena, fruits and aeroplane each of size 512 × 512) with varying noise density from 85 to 99%. The results show that even on coloured images the proposed filter provides magnificent performance over the existing filters. The proposed filter improves the PSNR values by 9.5% over the best known algorithm. Figure 5 shows the plots for Table 2. The graph also shows that the proposed filter outshines other filters in both metrics at each sample of noise density.

Figure 6 shows the restored Peppers image using different filters which are corrupted with 97% noise density. As we can see from the qualitative analysis, the proposed filter can efficiently preserve the edges and details of the image at very high noise density.

Table 1 Quality metrics of filtered Lena images using proposed and exiting filters

Metric	Noise (%)	[13]	[15]	[5]	[12]	[14]	[24]	[23]	Prop
PSNR	85	21.37	26.59	18.06	25.87	23.94	27.38	27.58	27.79
	87	20.71	25.92	17.20	25.18	23.35	26.88	27.08	27.38
	89	19.75	24.99	16.39	24.36	22.51	26.20	26.32	26.73
	91	18.90	23.87	15.51	23.60	21.71	25.49	25.80	26.29
	93	17.94	22.35	14.72	22.93	20.99	24.00	24.87	25.51
	95	16.60	20.24	13.92	21.67	19.70	20.63	23.65	24.55
	97	14.99	17.40	13.10	19.89	18.14	15.02	22.21	23.20
	99	12.85	14.81	12.38	16.95	15.76	8.85	18.68	20.13
	Avg	17.89	22.02	15.16	22.56	20.76	21.81	24.52	25.20
SSIM	85	0.7353	0.8980	0.4115	0.8696	0.8210	0.9067	0.9082	0.9115
	87	0.6973	0.8810	0.3528	0.8514	0.8008	0.8952	0.8971	0.9021
	89	0.6429	0.8516	0.3010	0.8254	0.7600	0.8800	0.8818	0.8879
	91	0.5806	0.8083	0.2560	0.7988	0.7236	0.8602	0.8652	0.8736
	93	0.5055	0.7122	0.2090	0.7660	0.6741	0.8167	0.8376	0.8477
	95	0.4003	0.5672	0.1546	0.7042	0.5954	0.7015	0.7998	0.8128
	97	0.2719	0.3615	0.1110	0.6165	0.4993	0.4373	0.7417	0.7511
	99	0.1275	0.1769	0.0674	0.4657	0.3471	0.1181	0.5932	0.6118
	Avg	0.4952	0.6571	0.2329	0.7372	0.6527	0.7020	0.8156	0.8248

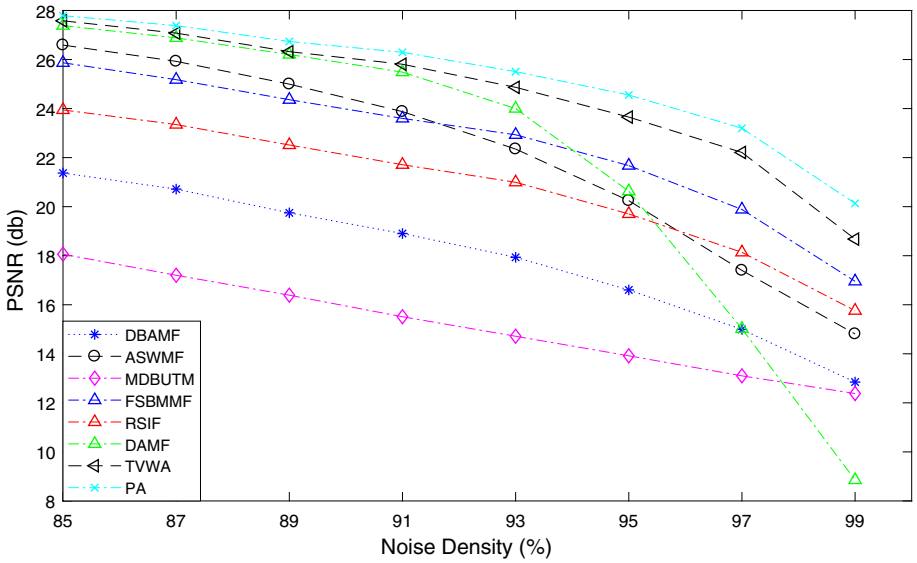


Fig. 3 PSNR for different filters with varying ND using Lena image (512 × 512)

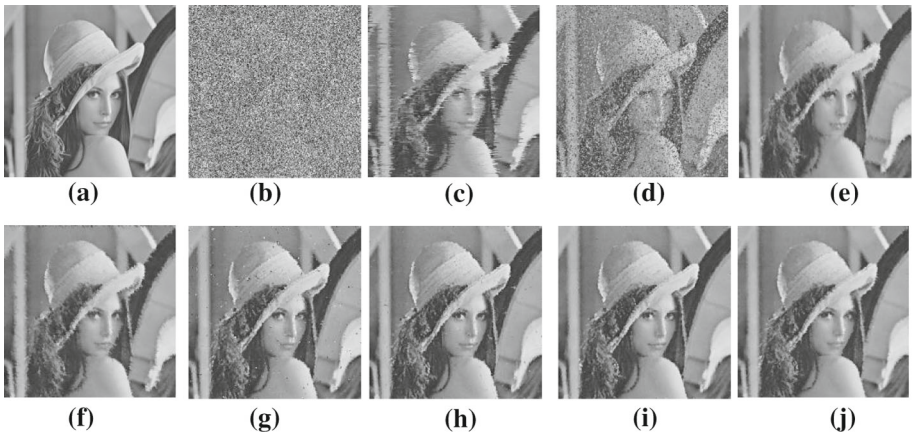


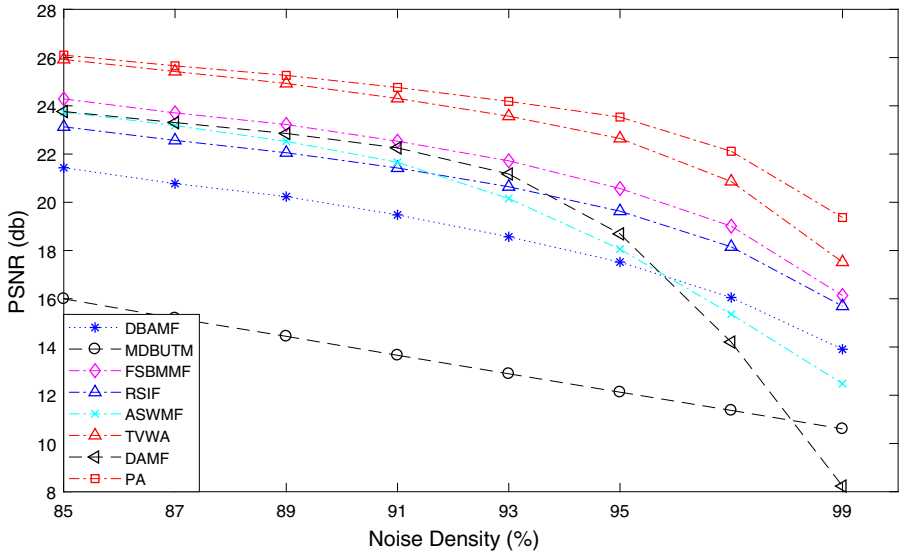
Fig. 4 Lena images **a** original **b** with 91% ND, and filtered images using: **c** DBAMF, **d** MDBUTM, **e** FSBMM, **f** RSIF, **g** TVWA, **h** ASWMF, **i** DAMF, and **j** Proposed median filters

Table 2 Quality metrics of filtered coloured images using proposed and exiting filters

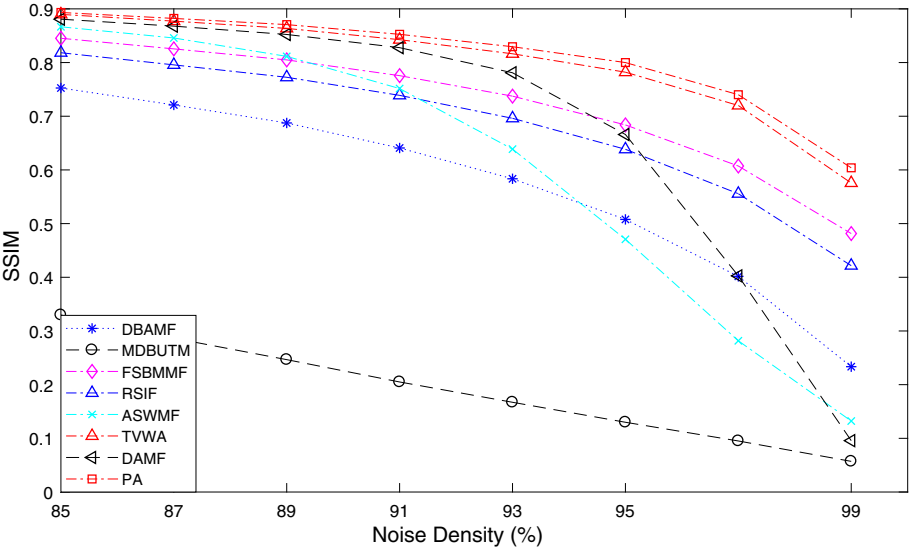
Metric	Noise (%)	[13]	[15]	[5]	[12]	[14]	[24]	[23]	Prop
PSNR	85	21.43	16.0	24.29	23.13	23.74	25.92	23.76	26.10
	87	20.77	15.19	23.70	22.57	23.18	25.42	23.30	25.69
	89	20.23	14.44	23.22	22.05	22.52	24.93	22.85	25.26
	91	19.48	13.66	22.53	21.42	21.66	24.31	22.25	24.76
	93	18.57	12.89	21.72	20.65	20.15	23.57	21.17	24.18
	95	17.52	12.13	20.57	19.63	18.06	22.65	18.69	23.53
	97	16.05	11.37	19.01	18.15	15.36	20.87	14.20	22.11
	99	13.91	10.60	16.13	15.70	12.48	17.52	8.23	19.37
	Avg	18.50	13.29	21.40	20.41	19.64	23.15	19.31	23.87
SSIM	85	0.7526	0.3297	0.8452	0.8183	0.8666	0.890	0.8808	0.8933
	87	0.7210	0.2865	0.8255	0.7957	0.8458	0.8771	0.8678	0.8822
	89	0.6875	0.2466	0.8049	0.7726	0.8115	0.8636	0.8523	0.8707
	91	0.6408	0.2050	0.7760	0.7389	0.7516	0.8426	0.8277	0.8526
	93	0.5835	0.1672	0.7374	0.6961	0.6387	0.8163	0.7810	0.8295
	95	0.5078	0.1299	0.6838	0.6384	0.4708	0.7821	0.6660	0.80
	97	0.4017	0.0951	0.6073	0.5556	0.2817	0.7200	0.4026	0.7401
	99	0.2335	0.0571	0.4815	0.4217	0.1322	0.5756	0.0957	0.6039
	Avg	0.5660	0.1896	0.7202	0.6797	0.5999	0.7959	0.6717	0.8090

4.3 Results Analysis with Seven Gray Scaled Images

Table 3 summarizes the average PSNR and SSIM values on seven grayscale images (including Boat, Zelda, Lena, Mandril, Cameramen, Barbara and Goldhill images) each of size 512×512. The ND is varied from 85 to 99%. The values demonstrate that the proposed filter is superior than other existing filters on a wide set of grayscale images. The average PSNR and SSIM trends are shown in Fig. 7 for comparative analysis. These figures show that the existing



(a) PSNR



(b) SSIM

Fig. 5 Average PSNR and SSIM value plots of four coloured images filtered using various filters with different noise density

TVWA provides better quality over the other filters. However, the proposed filter provides higher value of quality metrics at each sample of noise density.

The next subsection presents a quality analysis of the proposed filter over the existing using medical images.

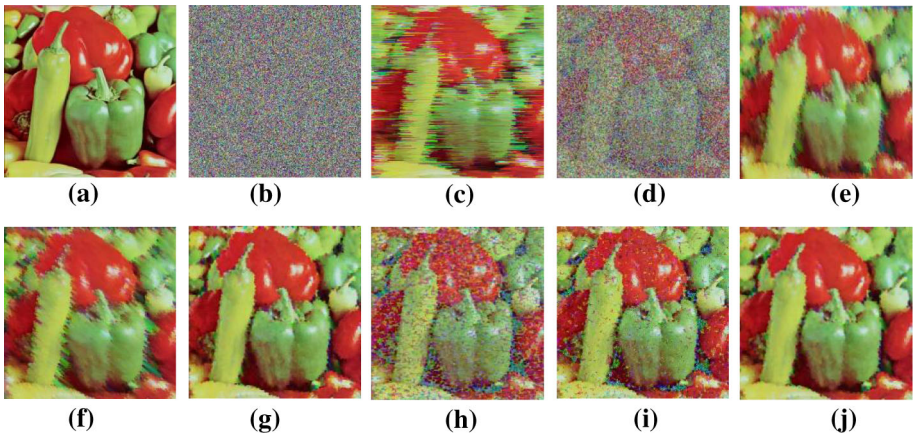


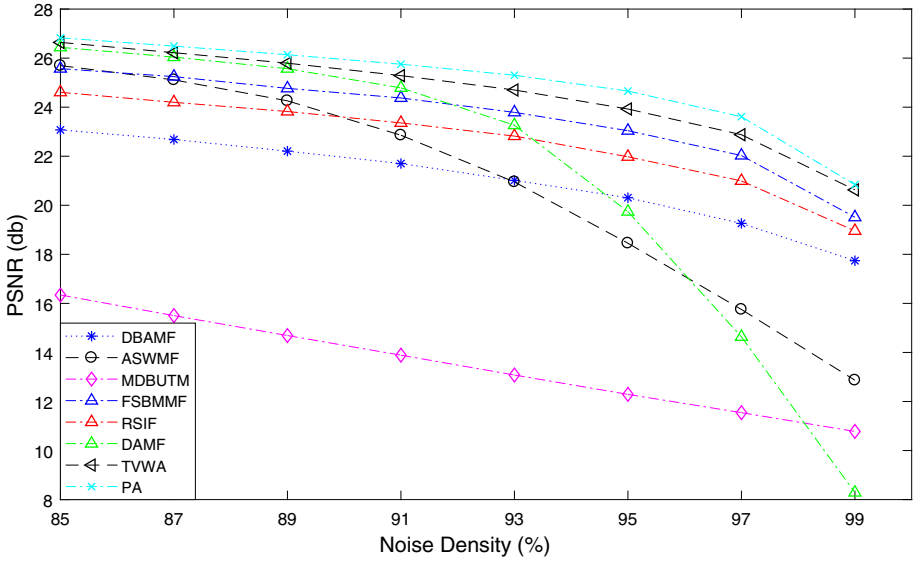
Fig. 6 Coloured Pepper images **a** original **b** with 97% ND, and filtered images using: **c** DBAMF, **d** MDBUTM, **e** FSBMM, **f** RSIF, **g** TVWA, **h** ASWMF, **i** DAMF, and **j** Proposed median filters

Table 3 Quality metrics of seven benchmark images using proposed and exiting filters

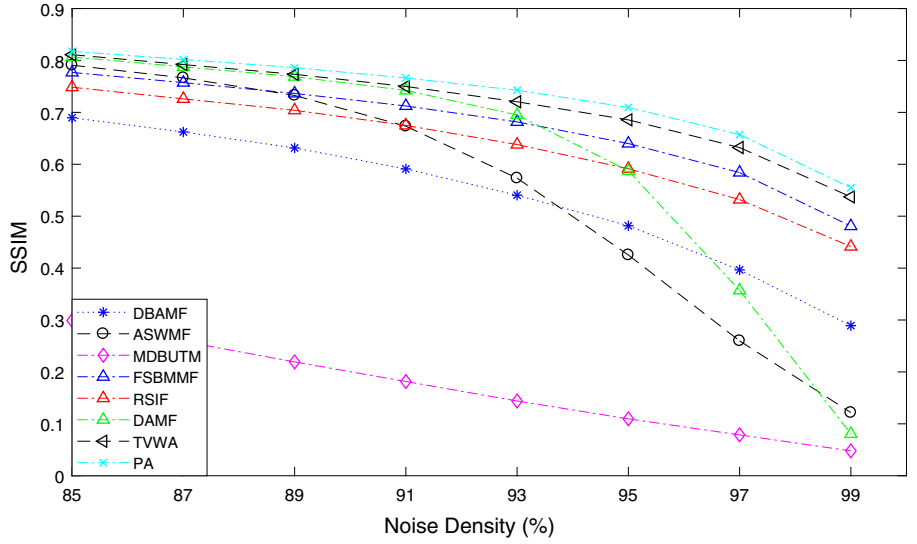
Metric	Noise (%)	[13]	[15]	[5]	[12]	[14]	[24]	[23]	Prop
PSNR	85	23.07	25.70	16.34	25.57	24.61	26.44	26.64	26.83
	87	22.68	25.11	15.50	25.25	24.20	26.04	26.22	26.49
	89	22.21	24.26	14.69	24.77	23.82	25.57	25.80	26.14
	91	21.70	22.86	13.89	24.38	23.36	24.79	25.29	25.75
	93	21.02	20.95	13.08	23.79	22.82	23.26	24.70	25.30
	95	20.31	18.45	12.29	23.04	21.98	19.74	23.92	24.66
	97	19.26	15.75	11.54	22.03	21.0	14.63	22.88	23.61
	99	17.74	12.87	10.78	19.51	18.96	8.29	20.62	20.84
	Avg	21.00	20.74	13.51	23.54	22.59	21.10	24.51	24.95
SSIM	85	0.6896	0.791	0.2990	0.7769	0.7484	0.8067	0.8111	0.8177
	87	0.6622	0.7665	0.2587	0.7573	0.7260	0.7877	0.7920	0.8019
	89	0.6315	0.7326	0.2193	0.7365	0.7040	0.7688	0.7734	0.7862
	91	0.5910	0.6734	0.1815	0.7124	0.6749	0.7423	0.7499	0.7663
	93	0.5404	0.5733	0.1440	0.6816	0.6380	0.6950	0.7204	0.7426
	95	0.4816	0.4252	0.1096	0.6400	0.5914	0.5871	0.6856	0.7095
	97	0.3965	0.2599	0.0787	0.5841	0.5321	0.3571	0.6322	0.6571
	99	0.2891	0.1216	0.0478	0.4808	0.4414	0.0805	0.5372	0.5547
	Avg	0.5352	0.5429	0.1673	0.6712	0.6320	0.6031	0.7127	0.7295

4.4 Analysis on Medical Images

This subsection includes the simulation results on a medical image. Lungs image of size 425×425 is considered for the demonstration and the results of PSNR and SSIM values are shown in Table 4. Further, the variation in noise density considered is from 85 to 99%. The proposed filter has excellent results in comparison to other filters when this specific X-ray image is considered. The average improvement in PSNR values is about 10%.



(a) PSNR



(b) SSIM

Fig. 7 Plot of average PSNR and SSIM for seven grayscale images each of size 512 × 512

Figure 8 show the graphical representation of the PSNR value. From the figure, it can be seen that proposed filter provides excellent and consistent performance.

Figure 9 shows the restored Lungs image when the image is corrupted with 95% ND and passed through proposed and other existing filters. The proposed algorithm provides best quality of restored image with less blurring, lower streaking and better edge detection.

Table 4 Quality metrics of filtered X-ray images using proposed and exiting filters

Metric	Noise (%)	[13]	[15]	[5]	[12]	[14]	[24]	[23]	Prop
PSNR	85	31.19	34.49	14.80	34.60	31.09	37.94	37.34	38.48
	87	30.18	32.88	13.85	33.65	30.58	36.95	36.88	38.03
	89	29.06	29.72	13.06	32.79	29.58	36.18	35.42	37.51
	91	27.75	25.83	12.16	31.65	28.67	32.40	35.25	36.94
	93	25.67	21.81	11.20	29.92	27.17	27.84	32.10	35.94
	95	23.54	17.98	10.45	27.33	25.94	21.40	30.54	34.75
	97	20.25	14.18	9.63	23.90	23.07	14.90	28.26	31.73
	99	14.67	10.56	8.87	16.64	15.92	7.77	21.10	24.53
	Avg	25.29	23.43	11.75	28.81	26.50	26.92	32.11	34.74
SSIM	85	0.9035	0.9590	0.2017	0.9476	0.9391	0.9700	0.9710	0.9719
	87	0.8829	0.9395	0.1681	0.9397	0.9294	0.9649	0.9662	0.9682
	89	0.8642	0.9019	0.1382	0.9302	0.9177	0.9587	0.9605	0.9631
	91	0.8250	0.8040	0.1084	0.9160	0.8985	0.9411	0.9534	0.9577
	93	0.7694	0.6424	0.0878	0.8954	0.8744	0.8924	0.9406	0.9467
	95	0.6898	0.4138	0.0681	0.8620	0.8284	0.7417	0.9235	0.9319
	97	0.5679	0.2017	0.0513	0.8126	0.7570	0.3963	0.8895	0.8970
	99	0.3144	0.0929	0.0416	0.6600	0.5151	0.0607	0.7928	0.8110
	Avg	0.7271	0.6194	0.1082	0.8704	0.8325	0.7407	0.9247	0.9309

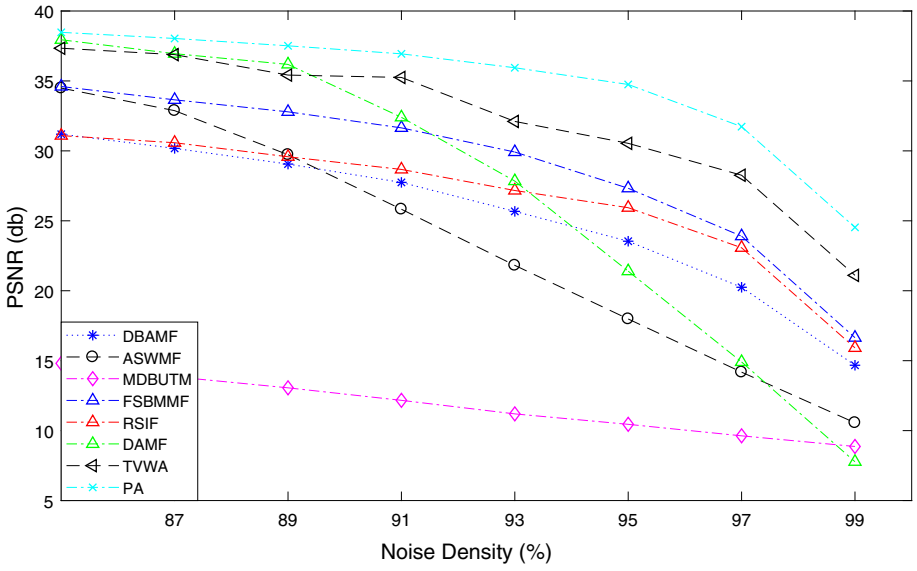


Fig. 8 Plot of PSNR for Lungs image of size 512 × 512

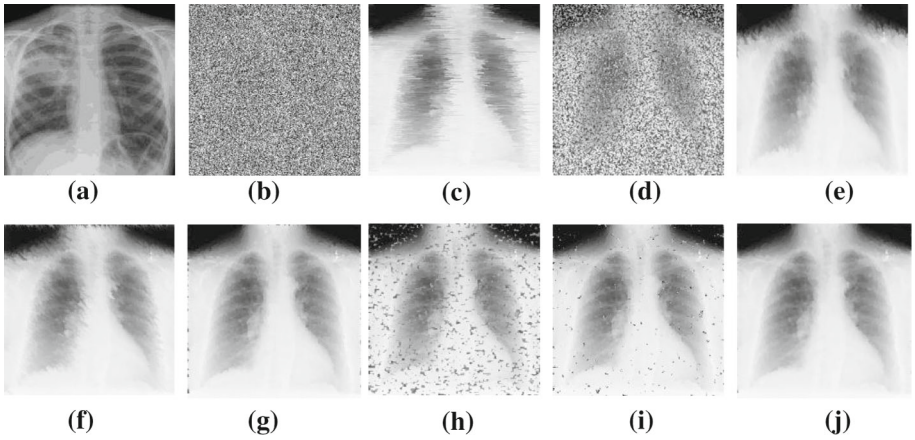


Fig. 9 X-ray (Chest) images **a** original **b** with 95% ND, and filtered images using: **c** DBAMF, **d** MDBUTM, **e** FSBMM, **f** RSIF, **g** TVWA, **h** ASWMF, **i** DAMF, and **j** Proposed median filters

5 Conclusion

This paper presented a new two-stage weighted filter that utilizes adaptive size window for very high impulse noise removal. The proposed filter identifies the maximum, minimum and middle values and then computes weights based on the number of NFPs close to these values. The window size is increased if the number of NFPs in the given window are less than predefined threshold. The proposed filter can efficiently retain the edges and details of the image even at noise densities as high as 97% and 99% resulting in less blurring and low streaking. The proposed filter has stable performance on large set of grayscale and coloured images. Especially, in case of medical images, the performance of the proposed filter is excellent with 10% average improvement in PSNR values. So, the proposed filter, upon being tested, could be used in the medical field.

References

1. Astola, J., & Kuosmaneen, P. (1997). *Fundamentals of nonlinear digital filtering*. Boca Raton, FL: CRC.
2. Pitas, I., & Venetsanopoulos, A. N. (2013). *Nonlinear digital filters: Principles and applications* (Vol. 84). Springer.
3. Zhang, S., & Karim, M. A. (2002). A new impulse detector for switching median filters. *IEEE Signal Processing Letters*, *9*(11), 360–363.
4. Ng, P.-E., & Ma, K.-K. (2006). A switching median filter with boundary discriminative noise detection for extremely corrupted images. *IEEE Transactions on Image Processing*, *15*(6), 1506–1516.
5. Esakkirajan, S., Veerakumar, T., Subramanyam, A. N., & PremChand, C. (2011). Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter. *IEEE Signal Processing Letters*, *18*(5), 287–290.
6. Li, Z., Liu, G., Xu, Y., & Cheng, Y. (2014). Modified directional weighted filter for removal of salt and pepper noise. *Pattern Recognition Letters*, *40*, 113–120.
7. Arora, S., Hanmandlu, M., & Gupta, G. (2018). Filtering impulse noise in medical images using information sets. *Pattern Recognition Letters*.
8. Ramachandran, V., & Kishorebabu, V. (2019). A tri-state filter for the removal of salt and pepper noise in mammogram images. *Journal of Medical Systems*, *43*(2), 40.
9. Murugan, K., Arunachalam, V., & Karthik, S. (2019). Hybrid filtering approach for retrieval of MRI image. *Journal of Medical Systems*, *43*(1), 9.

10. Hwang, H., & Haddad, R. A. (1995). Adaptive median filters: New algorithms and results. *IEEE Transactions on Image Processing*, *4*(4), 499–502.
11. Bhadouria, V. S., Ghoshal, D., & Siddiqi, A. H. (2014). A new approach for high density saturated impulse noise removal using decision-based coupled window median filter. *Signal, Image and Video Processing*, *8*(1), 71–84.
12. Vijaykumar, V., Mari, G. S., & Ebenezer, D. (2014). Fast switching based median-mean filter for high density salt and pepper noise removal. *AEU-International Journal of Electronics and Communications*, *68*(12), 1145–1155.
13. Aiswarya, K., Jayaraj, V., & Ebenezer, D. (2010). A new and efficient algorithm for the removal of high density salt and pepper noise in images and videos. In *2010 second international conference on computer modeling and simulation* (Vol. 4). IEEE, pp. 409–413.
14. Veerakumar, T., Esakkirajan, S., & Vennila, I. (2014). Recursive cubic spline interpolation filter approach for the removal of high density salt-and-pepper noise. *Signal, Image and Video Processing*, *8*(1), 159–168.
15. Faragallah, O. S., & Ibrahem, H. M. (2016). Adaptive switching weighted median filter framework for suppressing salt-and-pepper noise. *AEU-International Journal of Electronics and Communications*, *70*(8), 1034–1040.
16. Thanh, D. N. H., Hien, N. N., & Prasath, S. (2020). Adaptive total variation L1 regularization for salt and pepper image denoising. *Optik*, *208*, 163677.
17. Chen, J., & Li, F. (2019). Denoising convolutional neural network with mask for salt and pepper noise. *IET Image Processing*, *13*(13), 2604–2613.
18. Sohi, P. J. S., Sharma, N., Garg, B., & Arya, K. Noise density range sensitive mean-median filter for impulse noise removal. In *Innovations in computational intelligence and computer vision* (pp. 150–162). Springer.
19. Garg, B., & Arya, K. V. (2020). Four stage median-average filter for healing high density salt and pepper noise corrupted images. *Multimedia Tools and Applications*, *79*(43), 32305–32329.
20. Garg, B. (2020). Restoration of highly salt-and-pepper-noise-corrupted images using novel adaptive trimmed median filter. *Signal, Image and Video Processing*, *14*, 1555–1563.
21. Satti, P., Sharma, N., & Garg, B. (2020). Min–Max average pooling based filter for impulse noise removal. *IEEE Signal Processing Letters*, *27*, 1475–1479.
22. Garg, B. (2020). An adaptive minimum–maximum value-based weighted median filter for removing high density salt and pepper noise in medical images. *International Journal of Ad Hoc and Ubiquitous Computing*, *35*(2), 84–95.
23. Lu, C.-T., Chen, Y.-Y., Wang, L.-L., & Chang, C.-F. (2016). Removal of salt-and-pepper noise in corrupted image using three-values-weighted approach with variable-size window. *Pattern Recognition Letters*, *80*, 188–199.
24. Erkan, U.(2018).Different applied median filter in salt and pepper noise. *Computers and Electrical Engineering*, *70*, 789–798.

Nikhil Sharma was born on 21st August, 1999. He is currently an intern at Oracle India Private limited, Bangalore and pursuing his final year of Bachelor of Engineering degree in Electronics and Computer from Thapar Institute of Engineering and Technology, Patiala, Punjab, India. In May of 2019, he started doing his research in second year under Assistant Professor Bharat Garg in the field of Image Processing and developed image de-noising algorithms for salt and pepper noise removal. In summer of 2020, he has done a research internship with MIS Research group in the field of Image Processing and computer vision at ABV-Indian Institute of Information Technology and Management (IIITM), Gwalior under the supervision of Professor Karm Veer Arya. His research interest includes Image Processing, Biomedical Image Analysis, Digital Signal Processing and Machine Learning.

Prateek Jeet Singh Sohi was born on 12th January, 2000. He is currently an intern at Micron Technology Operations India, Hyderabad. He is pursuing his final year of Bachelor of Engineering degree in Electronics and Computer from Thapar Institute of Engineering and Technology, Patiala, Punjab, India. In May of 2019, he started doing his research in second year under Assistant Professor Bharat Garg in the field of Image Processing and developed image de-noising algorithms for salt and pepper noise removal. In summer of 2020, he has done a research internship with MIS Research group in the field of Image Processing and computer vision at ABV-Indian Institute of Information Technology and Management (IIITM), Gwalior under the supervision of Professor Karm Veer Arya. His research interest includes Image Processing, Biomedical Image Analysis, Signal Processing and Deep Learning.

Bharat Garg received the B.E. degree in Electronics from Rajiv Gandhi Prodhyogiki Vishwavidhyalaya, Bhopal, in 2001, and M. Tech. (VLSI Design) and Ph.D. from ABV-Indian Institute of Information Technology and Management Gwalior, India, in 2007, and 2017 respectively. Currently, he is working as Assistant Professor in Electronics and Communication Engineering Department in Thapar Institute of Engineering and Technology, Patiala. His research interest includes design and development of Energy Efficient VLSI Architectures and Hardware Security. He is having more than three years of experience in semiconductor industries and more than six years in academic institutions. He has authored/co-authored more than twenty five research papers in peer reviewed international Journals and conferences.