

DOI: <http://dx.doi.org/10.21123/bsj.2021.18.3.0614>

Optimum Median Filter Based on Crow Optimization Algorithm

Basma Jumaa Saleh

*Ahmed Yousif Falih Saedi
Lamees abdalhasan Salman*

Ali Talib Qasim al-Aqbi

Computer Engineering Department, College of Engineering, Al-Mustansiriyah University, Baghdad, Iraq.

*Corresponding author: eng.basmaj@uomustansiriyah.edu.iq, ahmed.yousif@uomustansiriyah.edu.iq, lali.al_aqbi@uomustansiriyah.edu.iq, lameesiteng2013@gmail.com.

*ORCID ID: <https://orcid.org/0000-0003-0952-0910>, <https://orcid.org/0000-0003-2049-8371>, <https://orcid.org/0000-0002-5204-5636>, <https://orcid.org/0000-0001-7904-3334>

Received 2/12/2019, Accepted 25/6/2020, Published Online First 21/2/2021, Published 1/9/2021



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

Abstract:

A novel median filter based on crow optimization algorithms (OMF) is suggested to reduce the random salt and pepper noise and improve the quality of the RGB-colored and gray images. The fundamental idea of the approach is that first, the crow optimization algorithm detects noise pixels, and that replacing them with an optimum median value depending on a criterion of maximization fitness function. Finally, the standard measure peak signal-to-noise ratio (PSNR), Structural Similarity, absolute square error and mean square error have been used to test the performance of suggested filters (original and improved median filter) used to removed noise from images. It achieves the simulation based on MATLAB R2019b and the results present that the improved median filter with crow optimization algorithm is more effective than the original median filter algorithm and some recently methods; they show that the suggested process is robust to reduce the error problem and remove noise because of a candidate of the median filter; the results will show by the minimized mean square error to equal or less than (1.38), absolute error to equal or less than (0.22), Structural Similarity (SSIM) to equal (0.9856) and getting PSNR more than (46 dB). Thus, the percentage of improvement in work is (25%).

Key words: Crow optimization algorithm, Image de-noising, Median filter, Peak Signal to Noise Ratio (PSNR), Salt and pepper noise.

Introduction:

Due to errors in noisy signals and channels of communication, images are often distorted by impulse noise. The resulting image processes such as edge-preserving, image segmentation, and object detection may be poor if the input image involves noise. It is therefore an important issue in image processing to detect noise pixels and then replace them with acceptable (1).

Impulse noise distribution can typically be divided into two types, fixed-value, and random-value types. A noise pixel takes either a max value of 255 or min value of 0 in impulse noise with a fixed value. The salt and pepper noise is also another name of impulse noise. The noise magnitude is uniformly distributed in the range [0,255] in salt and peppers noise. The papers suggested the salt and peppers noise filters aimed solely at the impulse noise of fixed pixel. The above filters outweigh any other impulse noise filters regardless of whether they are for salt and peppers

noise value or many noises. Hence, this paper emphasizes only on the random value salt and pepper noise (2).

For a noise-corrupted image, linear or nonlinear filter algorithm may be used to remove noise. The details are high-frequency picture features in the frequency domain, which can easily be confused with high-frequency noises. Hence, the key to image filtering processing is how to keep the image information and remove random noises. The median filter is a nonlinear filter and is widely used in digital image processing for its good edge-preserving features and decreased impulse noise capability. The median filter is a class order filter. It's removing noise effects are based on the size and shape of the filter mask and the difficulty of its techniques vary based mainly on how the median pixel prevails. Further, to improve the removing-noise performance of the median filter, scholars have suggested several changed methods for the

conventional median filter. To increase the scanning velocity of the median pixel, several rapid strategies are suggested which are centered on the splitting-conquering technique and simplified the complexities of the conventional median filter compare (3).

For restoring images contaminated with salt and pepper noise, multiple non-linear filters were suggested. Some of them require iterative filtering processes for many pulses filtering techniques, such as (4). The escape method involves repeating several times the same process based on an impulse noise size detection. And effective window sizes have also been tested for various percentages of impulsive noises (5).

For higher density impulsive noise detection, Vijaykumar et al. suggested a method (6) using a robust 17X17 variable window estimate. A new strategy to adaptive median filtering of images was developed which uses small-size mask iterative image processing and post-processing of median filter result (7).

Erkan et al. (8) introduced the findings in distinct images, showing that the approach suggested by the DAMF is good and better compared with other methodologies. Many successful methods of color image denotation were built based on quaternion representation (9). An image enhancement element is projected in (10) in a modified cascaded filter for restore color, highly polluted by impulsive noise, and randomly valued pulsation noise.

The optimization is to find a solution in engineering problems that can reduce or optimize a cost function. The stochastic approach is more widely used today to solve the problem of optimization. The nature-inspired algorithm has recently proved the ability to solve more numerical optimization problems. It improves such methods of optimizing to solve complex problems such as flow shop planning, high-dimensional task optimization, accuracy, image processing, evolutionary computing, computer engineering, and other technical issues. Several other algorithms, such as Artificial Bee Colony (ABC), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Harmony Search (HS) and Crow Optimization Algorithm (COA) have used recently (11).

Crow Optimization Algorithm is a new natural inspired algorithm proposed by Askarzadeh in 2016 (12) based on the intelligent behavior that crows exhibit. In CSA, the population emulates a flock of crows which behave based on the thievery behaviors of crows. For global optimization, CSA is a simple and important alternative. It has only a few

criteria to set for having an excellent result compared to similar approaches.

Based on research on prophase, this paper suggested two modifications to the median filtering technique based on the crow optimization algorithm:

(1) To optimize noise reduction efficiency, the mask can be adaptive by a fitness function.

(2) Depending to the median filter crow optimization principle, it is only important to quickly find the median optimum value of the filtering mask and not to rank all mask pixels. The crow optimization algorithm is therefore used in the search process to speed up the search process with the optimal median value.

The remainder of the article is structured accordingly: Section II describes the median filtering algorithm works Section III offers the proposed optimum median filter. The simulation experiment and performance estimation of our experiment are explained in section IV. Section V eventually draws our conclusions.

Median Filtering Algorithms

The median filter is a statistical-based, signal processing of nonlinear technology. The digital image or sequence's noisy value is replaced by median value of mask. The components of the mask are numbered in the gray's sequence scales, and the med pixel of the mask is stocked to substitute the pixel of noise (13).

The output of a median filter is $\tilde{y}(i,j) = \text{med} \{ S(i,j), i,j \in W \}$, at which $y(i,j)$ is the noisy image, W is the 2-mask dimensionally : the length of mask is $Q \times Q$ (at which Q is odd) like 3×3 , 5×5 , 7×7 , etc; the form of median mask can be rectangular, circle, square, loop, etc..

Median Scaled Filter

For a window mask, the output of a median scaled filter (MSF) is defined as $W = \{ S_1, S_2, S_3, \dots, S_Q \}$ where Q is the window length and is always odd.

$$\tilde{y}_{med} = \underset{i \in W}{\text{argmin}} \sum_{j=1}^Q ||S_i - S_j|| \quad (1)$$

Where $i, j = 1, 2, 3, \dots, Q (i \neq j)$

Use MSF, the window's main pixel is changed with a value that minimizes the sum of the distances to all the other values in the odd-window (14).

Median Vector Filter:

Median vector filter (MVF) is a nonlinear

filter for digital image processing to eliminate impulse noise. The filtering technique is the following. $W(Q)$ is a determine size Q window, and $S_i=(S_i^R, S_i^G, S_i^B)$, ($i=1, 2, 3, \dots, Q$) shows a component vector in the window. Next, determine the ranges of the vectors (L1 norm or L2 norm) in the window from each pixel to others. The sum of the lengths of the vector can then be estimated as (14):

$$D_i = \sum_{j=1}^Q \rho_{i,j}(S_i, S_j) = \sum_{j=1}^Q \|S_i - S_j\|_\rho \quad (2)$$

Where $i, j=1, 2, 3, \dots, Q (i \neq j)$, $\rho_{i,j}$ indicates the norm for each (i,j) and $\|\cdot\|_\rho$ indicates of the norm. The filter used D_2 norm of Euclidian distance. At last, find the lowest of all the pixels (13). By sort configuration is $S_1, S_2, S_3, \dots, S_Q$ and the associated value vector is P_1, P_2, \dots, P_Q . The above description can be provided by:

$$\tilde{y}_{VMF} = \operatorname{argmin}_{i \in W} \sum_{i=1}^Q \|S_i - S_j\|_\rho \quad (3)$$

Adaptive Median Filter

The adaptive median filter (AMF), using an adaptively changed window length distinguish noise pixels, relieves the contradiction; however, the noise pixels also allow sort pixels in the window while filtering (5).

The adaptive median filter (AMF) can be used in noise detections for salt and pepper noise and substitute each $y(i, j)$ with a pixel median on a specific window and confirm that most of the noise is detectable even at a high noise level given that the window is sufficiently large. Let $S(i, j)$ be a $W \times W$ window pixel with a focus on (i,j), and allow $W\text{-max}(Q)$ be the maximum window. Initialize $W=3$ and compute $S(i,j)^{\min}$, $S(i,j)^{\text{med}}$, $S(i,j)^{\max}$ which denote the minimum, median, and maximum of the pixel values in $S(i,j)$, respectively. When $S(i,j)^{\text{med}} \in (S(i,j)^{\min}, S(i,j)^{\max})$, impulse noise does not dominate the window, so $y(i,j)$ is not a noise candidate if $y(i,j) \in (S(i,j)^{\min}, S(i,j)^{\max})$, else $y(i,j)$ is replaced by $S(i,j)^{\text{med}}$ (Median Filtering);

$$\tilde{y}(i,j) = \begin{cases} y(i,j) & \text{if } y(i,j) \in (S(i,j)^{\min}, S(i,j)^{\max}) \\ S(i,j)^{\text{med}}, & \text{else} \end{cases} \quad (4)$$

When $S(i,j)^{\text{med}} \notin (S(i,j)^{\min}, S(i,j)^{\max})$, impulse noise dominates this window, then set $w = w+2$ (Adaptive) and repeat the above steps. When $w \geq$

$w\text{-max}$, (i,j) is replaced by $S(i,j)^{\text{med}}$ (the algorithm is terminated when the maximum window size is reached). Notice that except for the noise candidates that are replaced by the median $(i,j)^{\text{med}}$, the remaining pixels are left unchanged (15).

Proposed Optimum Median Filter The Noise-Reducing

Since the median filter is a statistical-based, its mathematical model is relatively complex for the random noise image. The noise variance of the median filtering is approximately of zero noise average under standard deviation for an image:

$$\sigma_{med}^2 = \frac{1}{4Qf^2(\bar{Q})} \approx \frac{\sigma_i^2}{Q + \frac{\pi}{2} - 1} \cdot \frac{\pi}{2} \quad (5)$$

Where σ_i^2 is the standard deviation noise, the mask length of filter is Q , $f^2(\bar{Q})$ is the noise distribution variable. And the average filtering of variance noise is:

$$\sigma_0^2 = \frac{1}{Q} \sigma_i^2 \quad (6)$$

From equation (5) and equation (6), the results of the median filter are based on two principles: the length of mask and the noise distribution. The median processing efficiency of noise cancellation is better than mean filtering operation, but to the salt and pepper noise, particularly broad pulses are more distant and the length of signal is smaller than $Q/2$; the median filter is very efficient. The output of the median filter must be enhanced if the median filtering object recognition can be modified to the noise density of the mask besides the mean filtering process. A developed median filtering technique was proposed on the basis of this work (3).

Crow Optimization Algorithms

Crow search algorithm (CSA) is a modern meta-heuristic technique suggested by Alireza (12) to solve problems of optimization. It is motivated by crow's cleverness in seeking sources of food. The CSA is made up of three sequential stages. First, each crow's hiding place location is generated randomly and each crow F 's memory is initialized as the best experience with this location. Second, according to the fitness function, crow determines the value of its location. Eventually, crow randomly populates and leads one of the flock crows to determine the location of the foods that this crow hides. If the location of the food is good, the crow can change its location. Otherwise, the crow will remain in the current location and will not switch to

the created location. CSA's approach is explained as follows (16).

Step 1: Initialize a crow population within the H-dimensional search space, where the method defines a random variable by:

$$C_i^T = (C_{i,1}^T, C_{i,2}^T, C_{i,j}^T, \dots, C_{i,H}^T) \quad (7)$$

For the i th crow, $i = 1, 2, \dots, H$, where C is Random crow variable C_i^T is the j -th potential position of crow i , at iteration T . However, the crows of the swarm are described by its memory $M_i = (M_{i,1}, M_{i,2}, \dots, M_{i,H})$, where the crows are unaware of the food sources.

Step 2: Each crow is tested based on its location quality, which has been assigned to the fitness function required.

Step 3: Crows generate new locations in the search space: Crow i randomly assign one of the bird crows, i.e. crow k , and then discovers the crow safe food location where the new crow i location at next iteration ($T+1$) is created by:

$$C_{i,j}^{T+1} = \begin{cases} C_{i,j}^T + \text{rand}_i \times Fl \times (M_{k,j}^i - C_{i,j}^i) & \text{rand}_i > AP \\ \text{RandomPosition} & \text{rand}_i \leq AP \end{cases} \quad (8)$$

Where rand_i are random values with a uniform array selection in the range $[0,1]$, AP reflects the probability of knowledge of crow k at iteration T and the likelihood of crow k at iteration. $M_{k,j}^i$

Describes is the location of the hiding place of crow i at iteration T and Fl is the flight length of crow i at iteration T .

Step 4: After producing the locations of the crow, the new locations are checked, and each crow improves their memory by:

$$C_j^{T+1} = \begin{cases} M_{k,j}^{T+1} & f(C_{i,j}^{T+1}) > f(M_{k,j}^T) \\ M_{k,j}^T & \text{otherwise} \end{cases} \quad (9)$$

Where $f(\cdot)$ indicates the fitness function, $>$ indicates better than others. Crow's nature activity is described by memorizing and retrieving the location of secret food places over episodes. In this context, that crow is expected to store the location of secret spots in a memory indicated by M . Thus, in $Iter$ iteration, the location of the secret spot of crow i is indicated by $M_{k,j}$. The memory M , iteration of crow i is start with its preliminary location C_i , iteration in the initialize stage, then with each iteration memory is updated. (9) Achieve the best food source of secret location. Equation

(9) works by filling the crow's memory with its new location if it's better than the previous population.

Step 5: Finish the method if the largest no. of epochs is reached and the memory's optimum position is stated as the solution of the optimization problem in terms of the fitness function; otherwise, go repeat step 3.

Optimum Median Filter Based on Optimization Algorithms

The algorithm's complexity is determined primarily by measuring the median filtering value; the article presents the crow optimization algorithm to increase probability of finding the median value by replacing it with an optimal median value based on the fitness function. The method provides the following steps:

Step1: begin.

Step2: read the image from Matlab workspace, "Lena" is colored image (JPEG-format) with 174×221 sizes, "Peppers" is colored Vibrant image (JPEG-format) with 255×255 sizes, "Retinal_artery_occlusion" is colored medical image (PNG-format) with 433×458 sizes from (17), while "Cameraman" is a gray-scale image (JPEG-format) with 256×256 size.

Step3: adding salt and pepper noise (with different variance δ).

Step4: convert an image to double.

Step5: if image input is gray (one layer) goto step7. Else if an image is color (three-layer) goto step6.

Step6: separate image to R-layer, G-layer, and B-layer using the RGB space process.

Step7: determine (problem dimension, flack size, awareness probability, and flight length).

Step8: for each epoch .

Step9: Randomly generate a crow position $c(i,j)$.

Step 10: initialize $W = 3$.

Step 11: compute s^{\min} , s^{med} and s^{\max} , in $s(i,j)$.

Step 12: if $s^{\min} < s^{\text{med}} < s^{\max}$, then go to step 13. Otherwise, set $W = W+2$ until the W -max (Q) is reached.

Step 13: if $W \leq W\text{-max}(Q)$, go to step 11. Otherwise, choose pixels in the window such that $s^{\min} < s(i,j) < s^{\max}$ and go to step 15.

Step 14: if $s^{\min} < y(i,j) < s^{\max}$, then $\tilde{y}(i,j) = y(i,j)$, else choose pixels in the window such that $\tilde{y}(i,j) = s^{\text{med}}(i,j) + c(i,j)$ and go to step 15.

Step 15: evaluate of maximization fitness function (peak signal to noise ratio (PSNR)).

Step 16: memory initialization and fitness of memory position.

Step 17: generation of random candidate crows for chasing.

Step 18: a new solution is evaluated by the fitness function.

Step 19: update position and memory.

Step 20: best found value is used as optimum weight and medium value to reach for optimum filter.

Step21: until it finishes epochs, if gray image go to step24.

Step22: Else if colored image go to next step.

Step23: restored colored image from RGB layers.

Step24: display images.

Step25: end.

Simulation Experiment and Performance Estimation

The simulation experiments are performed on a classical computer with an AMD A8-4500M APU with Radeon™ HD graphics 1.9 GHz 6.00GB RAM, 64-bit operating system, and MATLAB 2019b to compare the results from the original median filtering technique and optimum median filter. The well-known “Peppers”, “Retinal_artery_occlusion”, “Cameraman” and “Lena” pictures have been taken as the benchmark images with (10%, 20%, 30%, 40%... 80%) value of salt and peppers noise. “Lena”, “Retinal_artery_occlusion” and “Peppers” are colored images, while “Cameraman” is a gray-scale im-age. The results of the comparative experiment are shown as (Fig. 1) for "Lena", (Fig. 2) for "Cameraman", (Fig.3) for "Retinal_artery_occlusion" and (Fig. 4) for "Peppers".

The visual influence of the subjective or the method of estimating the purpose may estimate the effect of the image noise reduction. The work takes the mean absolute error (MAE), Peak Signal-to-Noise Ratio (PSNR) and the mean square error (MSE) between the filtered image $\tilde{y}(i,j)$ with the rows and columns is $N \times R$ and the original image $y(i,j)$ with the rows and columns is $N \times R$, at which $i=1,2,3,\dots N$ and $j=1,2,3,\dots R$. The definition of PSNR, MAE, and MSE are at equations (10), (11) and (12), respectively (4), SSIM at equation (13).

$$MSE = \frac{\sum_{i=1}^M \sum_{j=1}^N \|y_{i,j} - \tilde{y}_{i,j}\|^2}{3 * M * N} \quad (10)$$

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \quad (11)$$

$$MAE = \frac{\sum_{i=1}^M \sum_{j=1}^N \|y_{i,j} - \tilde{y}_{i,j}\|^1}{3 * M * N} \quad (12)$$

$$SSIM(y, \tilde{y}) = \frac{(2\mu_y \mu_{\tilde{y}} + S_1)(2\sigma_{y\tilde{y}} + S_2)}{(\mu_y^2 + \mu_{\tilde{y}}^2 + S_1)(\sigma_y^2 + \sigma_{\tilde{y}}^2 + S_2)} \quad (13)$$

Where μ_y and $\mu_{\tilde{y}}$ represent the mean of the original and restored images. σ_y^2 and $\sigma_{\tilde{y}}^2$ represent the standard deviation of the original and restored

images. $\sigma_{y\tilde{y}}$ nd restored image. S_1 and S_2 represent small constant are added to avoid instability (7).

Tables 1-4 and Figs. 5-8 show the performance comparison of our proposed method (OMF), an original method with very recently methods. They are Standard Median Filter (SMF) (18), Decision-based Median Filter (DBMF) (19), Alpha-Trimmed Mean Filter (ATMF) (20), Adaptive Vector Median Filter (AVMF) (21), Adaptive Center-Weighted Vector Median Filter (ACWVMF) (22), Adaptive Rank Weighted Switching Filter (ARWSF) (23), Two-Stage Switching Vector Median Filter (TSVMF) (24), Quaternion-Based Switching Filter (QSF) (25), Adaptive Quaternion Color Distance (AQCDF) (26), Modified Cascaded Filter (MCF) (10), Progressive Switching Median Filter (PSMF) (4), Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) (27), Different Applied Median Filter (DAMF) [8], Local Pixel Grouping (LPG) (28), Switching Vector Median Filter (SVMF) (29), Quaternion Switching Vector Filter (QSVF) (30), Two Stage Quaternion Vector Median Filter (TSQVMF) (31), Improved Median Filter (IMF) (32) and Adaptive Median Filter (AMF) (5), in terms of Structural similarity index (SSIM), PSNR, AE, MSE for the test image corrupted by 10%, 20%, 30%, 40%, 50%, 60%, 70% and 80% noise density, respectively. This work test how the improved median filter is better than standard filtering technique and other works comparisons, based on an optimizing approach. The enhanced method has major advantages, especially with high peak-signal-to-noise parameter, high structural similarity metric, and low absolute error. These results show that it works with a very high probability of impulse noise efficiently and that the proposed algorithm provides higher subjective quality in terms of impulse noise elimination and image information preservation. By summarizing and analyzing the numerical results listed in Tables 1–4, some conclusions can be drawn. In the SMF, the value of PNSR and SSIM is smaller, for example, in Table 1 (Lena image); the value of PNSR is 36.61 and 91% respectively under 20% impulse noise. The value of MAE and MSE is larger; it is 1.83 and 14.21 respectively under 20% impulse noise. It blurs image details severely. OMF has a better performance than the classical solutions at low noise density. For example, in Table 1, the value of PNSR is 35.59 and 80% respectively up to 50% impulse noise. The value of AE and MSE is larger; it is 2.44 and 17.96 respectively up to 50% impulse noise. When the density of impulse noise is increased, OMF has best performance under lower noise or higher noise.

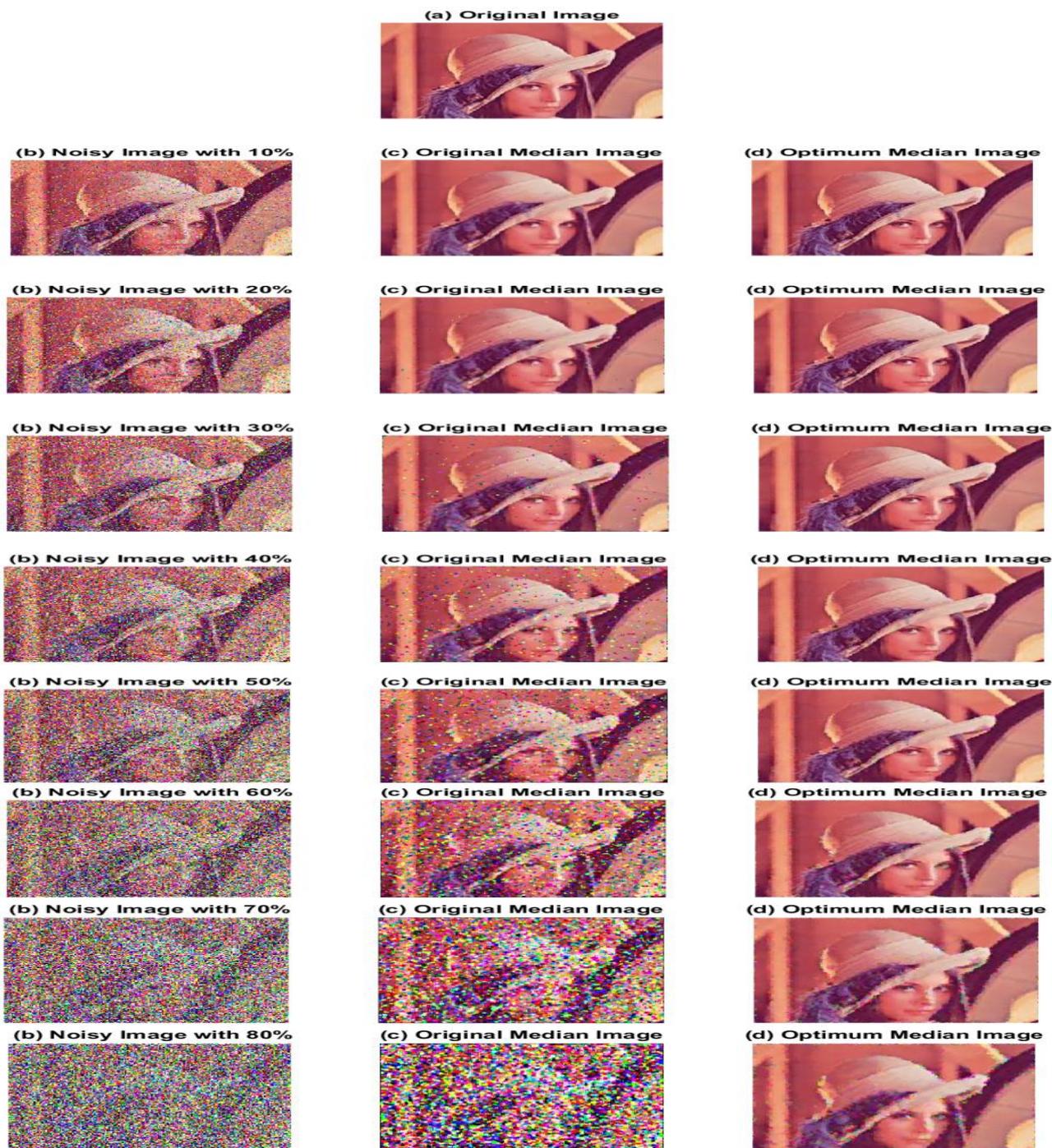


Figure 1. (a) The benchmark “Lena” image; (b) The noised “Lena” with salt and peppers noise (10%,20%,30%,40%,50%, 60%, 70% and 80%); (c) the standard median filter; (d) The suggested algorithm (OMF).

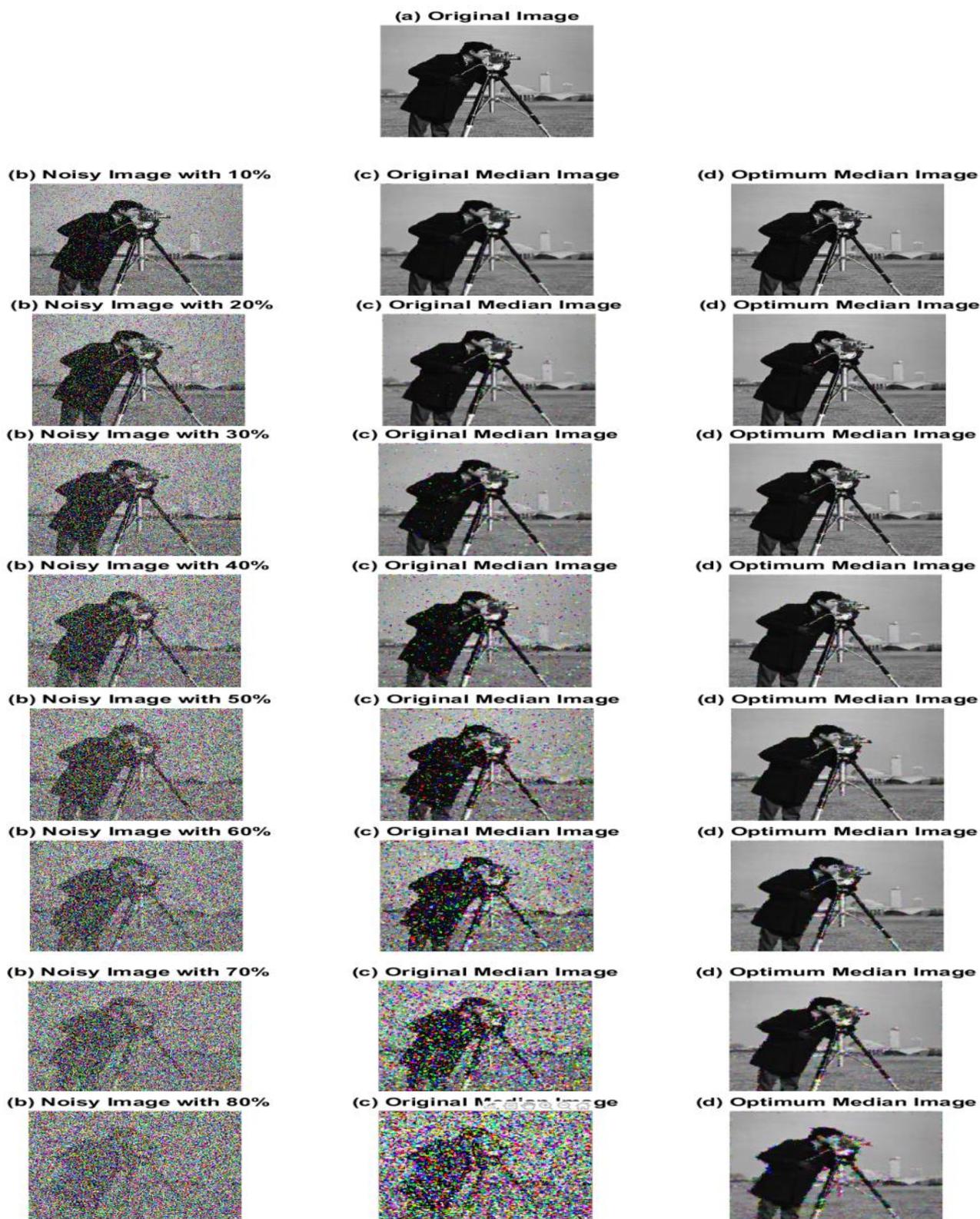


Figure 2. (a) The benchmark “Cameraman” image; (b) The noised “Cameraman” with salt and peppers noise (10%,20%,30%,40%,50%, 60%, 70% and 80%); (c) the standard median filter; (d) The suggested algorithm (OMF).

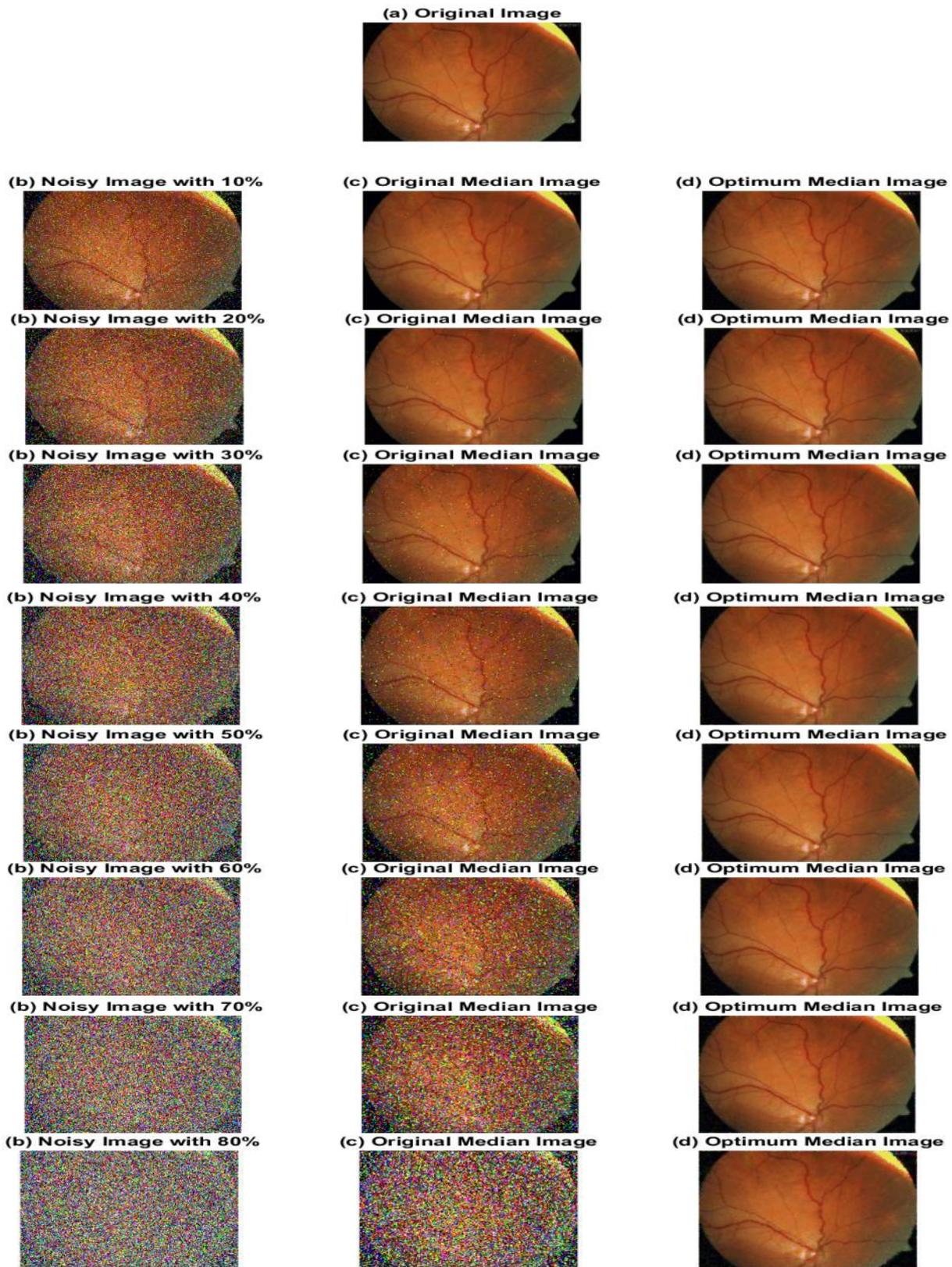


Figure 3. (a) The benchmark "Retinal_artery_occlusion" image; (b) The noised "Retinal_artery_occlusion" with salt and peppers noise (10%,20%,30%,40%,50%, 60%, 70% and 80%); (c) the standard median filter; (d) The suggested algorithm (OMF).

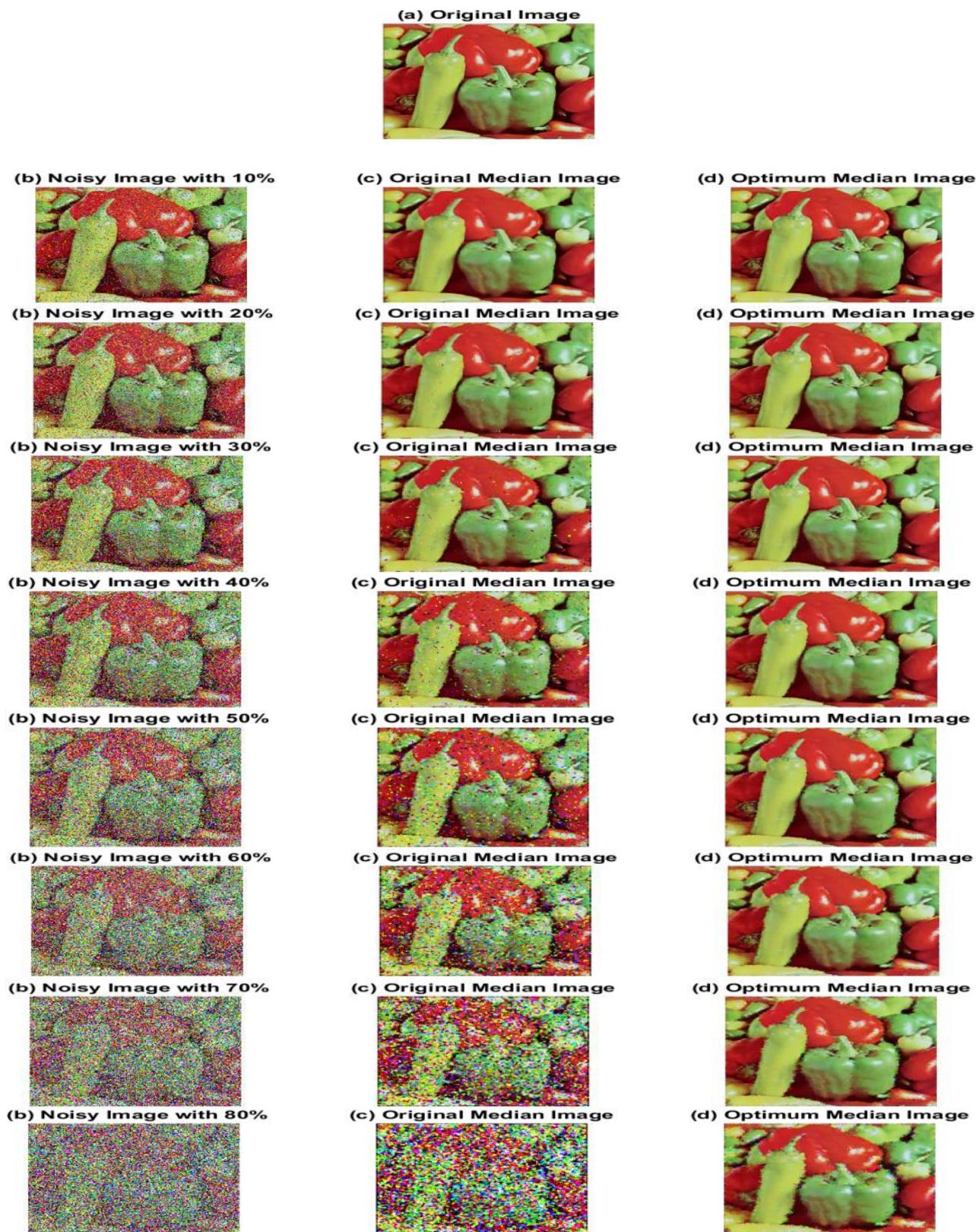


Figure 4. (a) The benchmark “Peppers” image; (b) The noised “Peppers” with salt and peppers noise (10%,20%,30%,40%,50%, 60%, 70% and 80%); (c) the standard median filter; (d) The suggested algorithm (OMF).

Table 1. PSNR value of the various noises density 10% to 80% using different algorithms and OMF (proposed) for different test images (Best result shown in bold and underline).

Image	Filters	10%	20%	30%	40%	50%	60%	70%	80%
Lena	SMF[18]	36.61	35.80	34.79	33.76	32.52	31.36	30.12	28.92
	DBMF[19]	36.35	28.73	25.35	20.43	16.72	13.31	10.89	8.67
	ATMF[20]	29.12	31.41	21.78	19.29	17.32	15.46	14.35	13.11
	AVMF [21]	34.95	38.35	32.67	31.47	30.33	28.98	27.69	-
	ACWVMF [22]	36.19	34.85	33.57	32.15	30.92	29.34	28.00	-
	ARWSF [23]	38.61	35.16	31.56	29.71	27.10	24.55	22.11	-
	QSF [25]	35.73	33.35	31.25	31.30	29.48	26.51	25.03	-
	TSVMF[24]	38.93	36.23	34.19	32.54	31.56	29.61	28.37	-
	AQCDF[26]	34.61	30.21	26.62	-	-	-	-	-
	MCF[27]	32.44	24.89	25.82	31.49	31.03	30.53	29.79	28.58
OMF- Proposed	<u>45.28</u>	<u>41.98</u>	<u>39.82</u>	<u>38.01</u>	<u>36.69</u>	<u>35.59</u>	<u>34.39</u>	<u>33.24</u>	
Cameraman	SMF[18]	35.26	34.85	34.30	33.49	32.62	31.47	30.37	29.23
	PSMF[4]	35.49	31.38	28.27	24.75	20.97	12.01	9.67	7.81
	DBMF[19]	37.67	32.42	29.28	25.96	23.35	20.86	18.41	15.68
	MDBUTMF[28]	35.14	31.01	29.44	29.74	30.28	29.36	27.86	23.71
	DAMF[8]	43.94	39.37	36.83	34.72	33.00	31.27	29.62	27.75
	OMF- Proposed	<u>44.89</u>	<u>41.84</u>	<u>39.81</u>	<u>38.15</u>	<u>37.05</u>	<u>35.31</u>	<u>35.16</u>	<u>34.26</u>
Retinal_artery_occlusion	SMF[18]	36.22	35.97	35.48	34.66	33.52	32.12	30.75	29.51
	AVMF[21]	34.95	33.85	32.67	31.47	30.33	28.98	27.69	-
	LPG[29]	33.41	33.16	32.41	31.22	29.87	29.26	26.64	-
	ARWSF [23]	36.19	34.85	33.57	32.15	30.92	29.43	28.00	-
	SVMF [30]	38.61	35.16	32.36	29.71	27.10	24.55	22.11	-
	QSVF [31]	35.73	33.55	31.56	29.83	28.26	26.51	25.03	-
	TSQVMF[32]	38.93	36.23	34.19	32.54	31.56	29.61	28.37	-
OMF- Proposed	<u>46.07</u>	<u>43.68</u>	<u>40.94</u>	<u>39.52</u>	<u>38.97</u>	<u>37.86</u>	<u>39.99</u>	<u>36.16</u>	
Peppers	SMF[18]	34.54	32.42	29.82	27.07	23.83	20.36	16.53	12.82
	AMF[5]	35.09	29.99	24.17	19.00	15.13	12.29	9.87	8.03
	DBMF[19]	32.72	31.73	30.78	29.50	28.20	26.95	25.62	23.25
	IMF[33]	42.94	38.50	35.36	33.18	30.66	28.34	25.45	23.49
	OMF- Proposed	<u>45.59</u>	<u>41.80</u>	<u>39.57</u>	<u>37.85</u>	<u>36.57</u>	<u>35.35</u>	<u>34.34</u>	<u>33.04</u>

Table 2. MSE value of the various noises density 10% to 80% using different algorithms and OMF (proposed) for different test images (Best result shown in bold and underline).

Image	Filters	10%	20%	30%	40%	50%	60%	70%	80%
Lena	SMF[18]	14.21	17.09	21.55	7.36	36.38	47.59	63.30	83.37
	OMF-Proposed	<u>1.94</u>	<u>4.13</u>	<u>6.69</u>	<u>10.07</u>	<u>13.93</u>	<u>17.96</u>	<u>23.71</u>	<u>30.93</u>
Camera-man	SMF[18]	19.41	21.28	24.51	29.07	35.54	46.33	59.71	77.56
	OMF-Proposed	<u>2.11</u>	<u>4.27</u>	<u>6.81</u>	<u>9.97</u>	<u>12.85</u>	<u>17.09</u>	<u>19.84</u>	<u>24.42</u>
Retinal_artery_occlusion	SMF[18]	15.73	16.66	18.60	22.38	29.00	40.03	54.75	72.82
	OMF-Proposed	<u>1.36</u>	<u>2.58</u>	<u>5.32</u>	<u>7.38</u>	<u>8.52</u>	<u>10.71</u>	<u>13.09</u>	<u>15.78</u>
Peppers	SMF[18]	23.66	37.98	65.99	119.1	253.4	592.3	1.4e+3	3.4e+3
	AMF[5]	21.06	66.59	267.6	298.7	2.0+3	3.9+3	6.6e+3	1.0e+3
	DBMF[19]	34.43	46.51	55.47	73.28	93.32	129.4	175.4	270.9
	IMF[33]	3.50	9.67	18.28	34.35	56.71	93.46	170.3	343.7
	OMF-Proposed	<u>1.79</u>	<u>4.29</u>	<u>7.81</u>	<u>10.70</u>	<u>14.31</u>	<u>19.02</u>	<u>23.97</u>	<u>32.35</u>

Table 3. SSIM value of the various noises density 10% to 80% using different algorithms and OMF (proposed) for different test images (Best result shown in bold and underline).

Image	Filters	10%	20%	30%	40%	50%	60%	70%	80%
Lena	SMF[18]	.9111	.8586	.7280	.5069	.2886	.1468	.0693	.0335
	AVMF[21]	.8412	.5902	.3310	-	-	-	-	-
	AWQCDF[26]	.9509	.8855	.7744	-	-	-	-	-
	OMF-Proposed	<u>.9856</u>	<u>.9646</u>	<u>.9401</u>	<u>.8984</u>	<u>.8561</u>	<u>.8022</u>	<u>.7330</u>	<u>.6430</u>
Camera-man	SMF[18]	.8520	.8030	.6742	.4644	.2635	.1271	.0663	.0322
	OMF-Proposed	<u>.9795</u>	<u>.9501</u>	<u>.9185</u>	<u>.8787</u>	<u>.8347</u>	<u>.7925</u>	<u>.7292</u>	<u>.6546</u>
Retinal_artery_occlusion	SMF[18]	.7076	.7657	.5685	.3765	.1872	.0753	.0285	.0126
	OMF-Proposed	<u>.9552</u>	<u>.9148</u>	<u>.8808</u>	<u>.8375</u>	<u>.7890</u>	<u>.7435</u>	<u>.6989</u>	<u>.6507</u>
Peppers	SMF[18]	.8948	.8439	.7258	.5109	.3129	.1574	.0834	.0432
	OMF-Proposed	<u>.9791</u>	<u>.9561</u>	<u>.9248</u>	<u>.8902</u>	<u>.8478</u>	<u>.7904</u>	<u>.7234</u>	<u>.6277</u>

Table 4. MAE value of the various noises density 10% to 80% using different algorithms and OMF (proposed) for different test images (Best result shown in bold and underline).

Image	Filters	10%	20%	30%	40%	50%	60%	70%	80%
Lena	SMF[18]	1.83	2.43	3.59	5.89	10.26	16.36	25.59	37.69
	AVMF [21]	1.00	1.47	1.98	2.55	3.16	3.87	4.69	-
	ACWVMF [22]	0.76	1.22	1.72	2.29	2.89	3.61	4.44	-
	ARWSF [23]	0.52	1.06	1.70	2.52	3.65	5.29	7.72	-
	QSF [25]	0.83	1.47	2.23	3.11	4.15	5.50	7.08	-
	SVMF[30]	0.52	1.16	1.41	2.23	2.90	3.67	4.07	-
	OMF- Proposed	<u>0.23</u>	<u>0.49</u>	<u>0.83</u>	<u>1.25</u>	<u>1.82</u>	<u>2.44</u>	<u>3.33</u>	<u>4.54</u>
Camera-man	SMF[18]	2.71	3.31	4.29	6.59	10.08	15.89	23.84	34.64
	OMF-Proposed	<u>0.29</u>	<u>0.62</u>	<u>1.01</u>	<u>1.54</u>	<u>2.06</u>	<u>2.81</u>	<u>3.42</u>	<u>4.57</u>
Retinal_artery_occlusion	SMF[18]	1.74	1.89	2.35	3.48	5.71	9.38	14.85	21.54
	AVMF[21]	1.00	1.47	1.98	2.55	3.16	3.87	4.69	-
	LPG[29]	2.57	2.54	2.69	3.00	3.45	4.10	4.97	-
	ARWSF [23]	0.76	1.22	1.72	2.29	2.89	3.61	4.44	-
	SVMF [30]	0.52	1.06	1.70	2.52	3.65	5.29	7.72	-
	QSVF [31]	0.83	1.47	2.23	3.11	4.15	5.50	7.08	-
	TSQVMF[32]	0.52	1.16	1.41	2.23	2.90	3.67	4.07	-
	OMF- Proposed	<u>0.19</u>	<u>0.35</u>	<u>0.61</u>	<u>0.48</u>	<u>1.00</u>	<u>1.26</u>	<u>1.53</u>	<u>1.86</u>
Peppers	SMF[18]	2.09	2.73	3.83	5.89	9.30	14.66	22.37	32.82
	OMF-Proposed	<u>0.22</u>	<u>0.54</u>	<u>0.93</u>	<u>1.43</u>	<u>1.96</u>	<u>2.69</u>	<u>3.45</u>	<u>5.06</u>

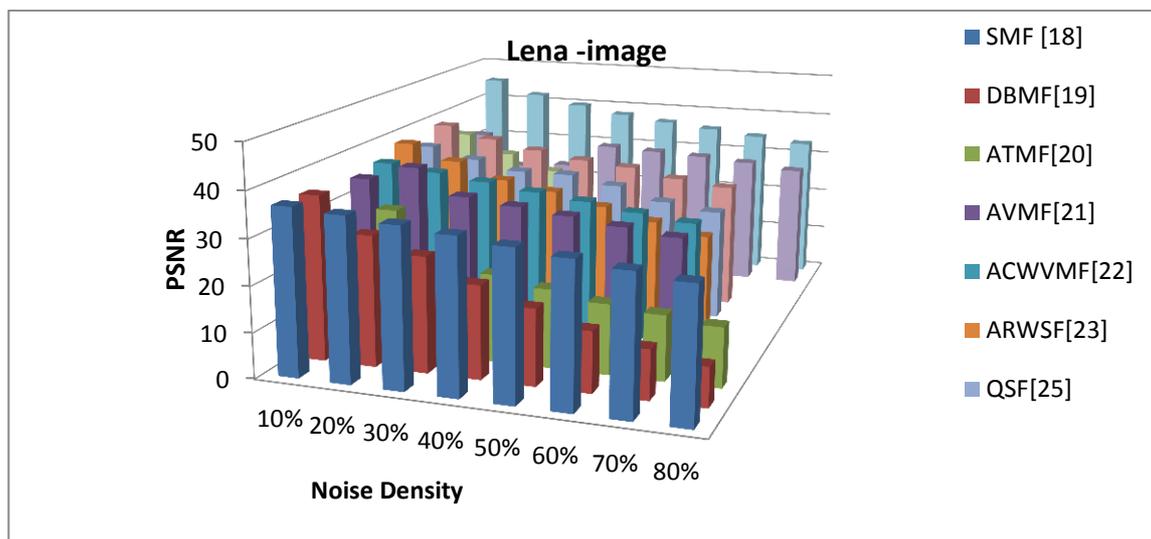


Figure 5. Noise density versus PSNR for Lena image.

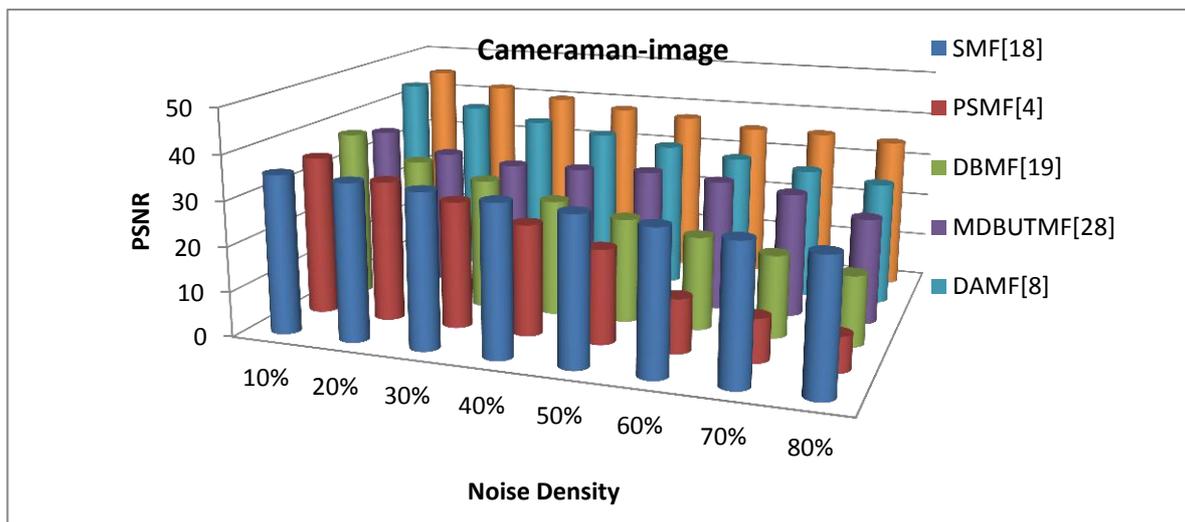


Figure 6. Noise density versus PSNR for Cameraman image.

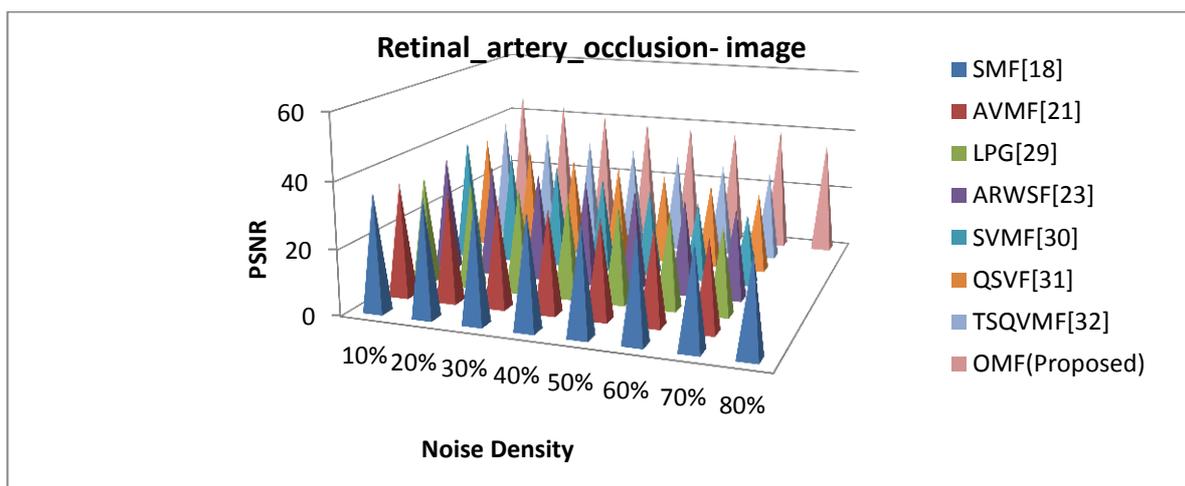


Figure 7. Noise density versus PSNR for Retinal_artery_occlusion- image.

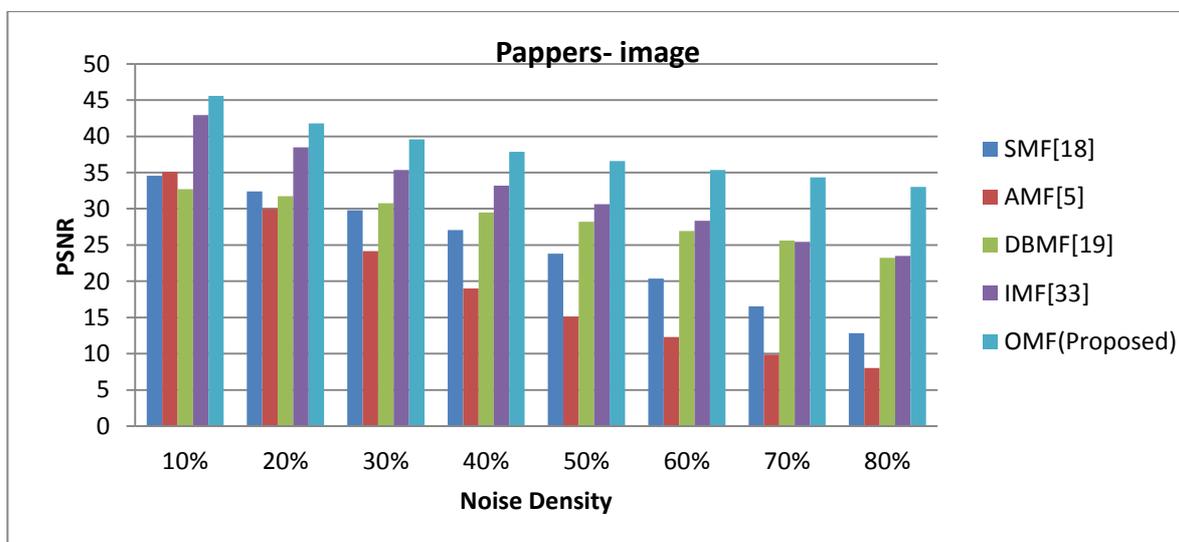


Figure 8. Noise density versus PSNR for Peppers image.

Conclusion:

The improved median filter based on the optimization algorithm has presented for gray and color images with salt and peppers in this work.

The suggested process examined by MATLAB (R2019b) set on the cameraman, peppers, retinal_artery_occlusion and Lena images. The simulation and experimental results show evidently

the capability of improved the suggested process which has an optimum median value of mask. The modified architecture increase the computational time required for finding the median, but this increases the efficiency of the system with minimum error. The algorithm removes noise even at higher noise densities (up to 50%) and preserves the edges and fine details. The performance of the algorithm is better when compared to the other architecture of this type because the optimum median filter has a capability to find optimum candidate median value with less complexity process and high structural similarity index (up to 95%) for color images and (up to 97%) for gray image with percentage of improvement (25%).

Acknowledgments:

The authors would like to thank Mustansiriyaah University (www.uomustansiriyaah.edu.iq) Baghdad- Iraq for its support in the present work.

Authors' declaration:

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are mine ours. Besides, the Figures and images, which are not mine ours, have been given the permission for republication attached with the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee in Al-Mustansiriyaah University.

References:

1. Gonzalez R C. Digital_Image_Processing. 2nd ed; 2002. 793 p.
2. Kang C, Wang W. Fuzzy reasoning-based directional median filter design. Signal Process. 2009; 89 (3): 344–351.
3. Zhu Y, Huang C. An Improved Median Filtering Algorithm for Image Noise Reduction. Phys. Procedia. 2012; 25: 609–616.
4. Wang Z, Zhang D. Progressive switching median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Circuits Syst. 1999; 46 (1): 78–80.
5. Hwang H, Haddad RA. Adaptive median filters: new algorithms and results. IEEE Trans Image Process. 1995; 4 (4): 499–502.
6. Vijaykumar VR, Ebenezer D. High Density Impulse Noise Removal Using Robust Estimation Based Filter. IAENG Int. J. Comput. Sci. 2008; 35 (3).
7. Lyakhov PA, Orazayev AR, Chervyakov NI, Kaplun DI. A New Method for Adaptive Median Filtering of Images. In 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus) 2019 Jan 28 (pp. 1197-1201). IEEE.
8. Uğur E, Gökrem L, Engino S. Different applied median filter in salt and pepper noise. Comput. Electr. Eng. 2018; 70: 789-798.
9. Jin L, Zhu Z, Song E, Xu X. An effective vector filter for impulse noise reduction based on adaptive quaternion color distance mechanism. Signal Process. 2018; 147: 173–189.
10. Karthik B, Kumar TK, Vijayaragavan SP, Sriram M. Removal of high density salt and pepper noise in color image through modified cascaded filter. J. Ambient Intell Human Comput. 2020 Feb 1:1-8.
11. Saleh BJ, Al-Aqbi ATQ, Saedi A YF. A novel biogeography inspired trajectory-following controller for national instrument robot. CCIS. 2018; 938: 171–189.
12. Askarzadeh A. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. Comput. Struct. 2016; 169: 1–12.
13. Ma Z, Wu RH, Feng D. Partition-based vector filtering technique for suppression of noise in digital color images. IEEE Trans. Image Process. 2006; 15 (8): 2324–2342.
14. Xu J, Wang L, Shi Z. A switching weighted vector median filter based on edge detection. Signal Process. 2014; 98: 359–369.
15. Guo D, Xiaobo Q, Xiaofeng D, Keshou W, Xuhui Ch. Salt and Pepper Noise Removal with Noise Detection and a Patch-Based Sparse Representation. Adv. Multimedia. 2014: 1-14.
16. Huang KW, Girsang AS, Wu ZX, Chuang YW. A hybrid crow search algorithm for solving permutation flow shop scheduling problems. Appl. Sci. 2019; 9 (7): 1–15.
17. MedPix Medical Image Database. <https://library.tmc.edu/website/medpix-medical-image-database>, 2019.
18. Gallagher N, Wise G. A Theoretical Analysis of the Properties of Median Filters. IEEE Trans. Acoust, Speech, Signal Process. 1981; 29: 1136-1141.
19. Pok G, Jyh-Charn L. Decision based median filter improved by predictions. ICIP. 1999; 2: 410-413.
20. Luo W. An efficient detail-preserving approach for removing impulse noise in images. IEEE Signal Process. Lett. 2006; 13 (7): 413–416.
21. Lukac R. Adaptive vector median filtering. Pattern Recogn. Lett. 2003; 24 (12):1889–1899.
22. Lukac R, Smolka B. Application of adaptive center-weighted vector median framework for the enhancement of cdna microarray images. Int J Appl Math Comput Sci. 2003; 13 (3):369–383.
23. Smolka B, Malik K, Malik D. Adaptive rank weighted switching filter for impulsive noise removal in color images. J Real-Time Image Proc. 2012; 10(2): 289–311.
24. Chanu, P, Singh KH. A two-stage switching vector median filter based on quaternion for removing impulse noise in color images. Multimed Tools Appl. 2019; 78: 15375–15401.
25. Wang G, Liu Y, Zhao T. A quaternion-based switching filter for colour image denoising. Signal Process. 2014; 102: 216–225.

26. Jin L, Zhu Z, Song E, Xu X. An effective vector filter for impulse noise reduction based on adaptive quaternion color distance mechanism. Signal Process. 2018; 147: 173–189.
27. Esakkirajan S, Veerakumar T, Subramanyam AN, Premchand CH. Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter. IEEE Signal Process. Lett. 2011; 18(5): 287- 290.
28. Rambabu TG, Kishore KK. Medical image denoising using KPCA with local pixel grouping. In2015 International Conference on Computer Communication and Informatics (ICCCI) 2015 Jan 8 (pp. 1-5). IEEE.
29. Chanu PR, Singh KM. A Switching vector median filter for impulse noise removal from color images. Proceedings of TENCON. 2017, 5-8th Nov., Penang, Malaysia: 2819–2824.
30. Jin L, Zhu Z, Xu X, Xiang L. Two-stage quaternion switching vector filter for color impulse noise removal. Sign. Process. 2016; 128:171–185.
31. Chanu PR, Singh KM. Impulse Noise Removal from Medical Images by Two Stage Quaternion Vector Median Filter. J. Med. Syst. 2018; 42: 197.
32. Yadav AK, Roy R, Kumar R, Kumar CS, Kumar AP. Algorithm for de-noising of color images based on median filter. In2015 third International Conference on Image Information Processing (ICIIP). 2015 Dec 21 (pp. 428-432). IEEE.

المرشح المثالي المتوسط المستند على خوارزمية الغراب للتحسين

علي طالب قاسم العقابي

أحمد يوسف فالح الساعدي
لميس عبد الحسن سلمان

بسمة جمعة صالح

قسم هندسة الحاسوب، كلية الهندسة، الجامعة المستنصرية، بغداد، العراق.

الخلاصة:

يُقترح مرشح متوسط جديد يعتمد على خوارزميات تحسين الغراب (OMF) لتقليل ضوضاء الملح واللفل العشوائية وتحسين جودة الصور ذات اللون الرمادي والملونة. الفكرة الرئيسية لهذا النهج هي أن أولاً، تقوم خوارزمية تحسين الأداء بالكشف عن وحدات البكسل الخاصة بالضوضاء، واستبدالها بقيمة بسيطة مثالية تبعاً لدالة الأداء. أخيراً، تم استخدام نسبة القياس القسوى لنسبة الإشارة إلى الضوضاء (PSNR)، والتشابه الهيكلي والخطأ المربع المطلق والخطأ التربيعي المتوسط لاختبار أداء المرشحات المقترحة (المرشح الوسيط الأصلي والمحسن) المستخدمة في الكشف عن الضوضاء وإزالتها من الصور. يحقق المحاكاة استناداً إلى MATLAB R2019b والنتائج الحالية التي تفيد بأن المرشح المتوسط المحسن مع خوارزمية تحسين الغراب أكثر فعالية من خوارزمية المرشح المتوسط الأصلية ومرشحات لطرق حديثة؛ أنها تبين أن العملية المقترحة قوية للحد من مشكلة الخطأ وإزالة الضوضاء بسبب مرشح عامل التصفية المتوسط؛ سنظهر النتائج عن طريق تقليل الخطأ التربيعي المتوسط إلى أدنى أو أقل من (1.5)، والخطأ المطلق للتساوي (0.22)، والتشابه الهيكلي أكثر من (95%) والحصول على PSNR أكثر من 45 (dB) ونسبة تحسين (25%).

الكلمات المفتاحية: إزالة الضوضاء، خوارزمية تحسين الغراب، ضوضاء الاندفاع، معالجة الصور، مرشح متوسط المثالي.