High Density Impulse Noise Removed Depending on Nearest Interpolation and Median Algorithm

Hana' H. kareem
AL – Mustansiriyah University
Physics Department, College of Education

Abstract

In this paper, a Nearest Interpolation and Median (NIM) algorithm is proposed to remove high density salt & pepper noise from digital images. First stage in this algorithm the noisy pixels are detected and in the second stage is calculated the absolute deferent between median and mean value in the kernel, after noise values are eliminated, if the region is homogenous, the center of the kernel replaced by the nearest interpolation value or not it replaced by the median value, according to absolute deferent. The proposed algorithm shows significantly better image quality than a simple median filter (SMF), Adapted Mean Filter (AMF), Decision Based Algorithm (DBA) and Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF). The proposed algorithm is tested with different gray scale image and it gives better Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF).

1. Introduction

Impulse noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel [1]. There are two models of impulsive noise, namely, salt, and pepper noise and random valued impulse noise. Salt and pepper noise is sometimes called fixed valued impulse noise producing two gray level values 0 and 255. Since, linear filtering techniques are not effective in removing impulse noise, non -linear filtering techniques are greatly used in the restoration process. The best-known and most widely used non-linear digital filters, based on order statistics are median filters. Median filters are known for their capability to remove impulse noise without damaging the edges. Median filters are known for their capability to remove impulse noise as well as preserve the edges. The main drawback of a simple median filter (SMF) [2], at high noise densities, SMFs often exhibit blurring for large window sizes and insufficient noise suppression for small window sizes. However, most of the median filters operate uniformly across the image and thus tend to modify both noise and noise-free pixels. Consequently, the effective removal of impulse often leads to images with blurred and distorted features. Adaptive Median is a “decision-based” or “switching” filter that first identifies possible noisy pixels and then replaces them using the median filter or its variants, while leaving all other pixels unchanged. This filter is good at detecting noise even at a high noise level. The adaptive structure of this filter ensures that most of the impulse noises are detected even at a high noise level provided that the window size is large enough. The performance of AMF is good at lower noise density levels, due to the fact that there are only fewer corrupted pixels that are replaced by the median values [3], [4].many techniques first detect the impulse locations and then filter the noisy pixels without processing the uncorrupted ones as DBA [5] and MDBUTMF [6] algorithms. The Decision Based Algorithm (DBA) is one of the fastest methods and it is an efficient algorithm capable of impulse noise removal at noise densities as high as 80% [7]. the MDBUTMF is eliminate any value of noise in the kernel and then find the median value , but the drawback of these algorithms are streaking at very higher noise densities. The proposed algorithm re-moves this drawback at very high noise density depending on absolute deferent between median and mean value in the kernel, and gives better PSNR and IEF values than the existing algorithm.

2. Salt and pepper impulsive noise

In the salt and pepper impulsive noise, it is usually assumed that the salt noise is the maximum gray level (255) and the pepper noise is the minimum gray level (0) [7], thus each pixel in an image has probability p/2 ($0<p<1$) to be corrupted into either a white dot (salt) or a black dot (pepper) where:

$$I = \begin{cases} 255 & with\,probability \quad p/2 \\ 0 & with\,probability \quad p/2 \\ X & with\,probability \quad 1-p \end{cases} \tag{1}$$

Where *I* is the noisy-free image, *X* is the noisy image and *xy* is the size of image.

3. Review of Image Denoising Algorithms

There are many algorithm used to remove salt and pepper noise from the image at deferent noise density level we can discuss as following:

3.1. Median Filter

It is a based order statistics filter [8], simple Median Filter (SMF) used to reduce a noise in the image by replacing the value of the pixel by the median of gray levels in the neighborhood of that pixel or window. If the *I(i,j)* and *Ie (i,j)* be the input and output respectively , then the median filter is :

$$Ie(i,j) = median\{I(i-r,j-c)\} \tag{2}$$

Where $r \times c$ represent the size of the window. The advantage of a median filter is that it is an excellent choice for the removal of especially salt and pepper noise with low density level whereas the drawback of SMF is unacceptable in the case of signal dependent noise.

3.2. Adaptive Median Filter

The Adaptive Median Filter performs spatial processing to determine which pixels in an image have been affected by impulse noise [7]. It is classifies pixels as noise by comparing each pixel in the image to its surrounding neighbor pixels. The size of the neighborhood is adjustable, as well as the threshold for the comparison. A pixel that is different from a majority of its neighbors, as well as being not structurally aligned with those pixels to which it is similar, is labeled as impulse noise. These noise pixels are then replaced by the median pixel value of the pixels in the neighborhood that have passed the noise labeling test.This filter also smoothens out other types of noise, thus, giving a much better output image than the standard median filter. The algorithm of AMF is [9]:

1.Input I(i,j) is the gray image
i=1 to width
j=1 to high
2. Level A: A1 = Imed - Imin
A2 = Imed - Imax
if A1 > 0 AND A2 < 0, go to level B
else increase the window size
if window size < Smax, repeat level A
else output I(i,j)
3.Level B: B1 = I(i,j) - Imin
B2 = I(i,j) - Imax
if B1 > 0 AND B2 < 0, output I(i,j)
else output Imed Repeat Steps 1 to 3 until all the pixels in the entire image are processed.

Where the AMF changes size of *Sxy* (the size of the neighborhood) during operation and , *Imin* is minimum gray level value in *Sxy*, *Imax* is maximum gray level value in *Sxy*,Imed is median of gray levels in *Sxy*, *Ixy* is gray level at coordinates *(i, j)* ,and *Smax* is the maximum allowed size of *Sxy*

3.3 Decision-Based Algorithm

It's simple and fast method, The DBA processes the corrupted image by first detecting the impulse noise. If the value of the pixel processed is within the range (0-255), then it is an uncorrupted pixel and left unchanged. If the value does not lie within this range, then it is a noisy pixel and is replaced by the median value of the window or by its neighborhood values. If the noise density is high, there is a possibility that the median value is also a noise value. In the latter case, the pixel processed is replaced by the previously processed adjacent neighborhood pixel value in place of the median value. The PA is as follows [5]:

1. Select 2-D window of size 3 3. Assume that the pixel being processed is I(i,j).
2. The pixel values inside the window are sorted, I_{min} , I_{mid} and I_{max} are determined as follows:
   a) The rows of the window are arranged in ascending order.
   b) The columns of the window are arranged in ascending order.
   c) The right diagonal of the window is now ar-ranged in ascending order. now the first element of the window is the minimum value I_{min} , the last element of the window is the maximum value I_{max} , and the middle element of the window is the median value I_{mid} .
3. The I(i,j)is an uncorrupted pixel if I_{min}< I(i,j) < I_{max} , I_{min}>0 and I_{max} 255 the pixel being processed is left unchanged. Otherwise, I(i,j) is a corrupted pixel.
4. If I(i,j) is a corrupted pixel, it is replaced by its median value if I_{min}< I_{mid} < I_{max} and 0< I_{mid} < 255.
5. If I_{min}< I_{mid} < I_{max} is not satisfied or 255< I_{mid}=0 then the I_{mid} is a noisy pixel. In this case, the I(i,j) is replaced by the value of neighborhood pixel value.
6. Repeat steps 1 to 3 until all the pixels in the entire image are processed.

3.4 MDBUTMF A LGORITHM

The MDBUTMF algorithm processes the noisy images by first detecting the impulse noise, if the processing pixel lies between maximum and minimum gray level values then it is noise free pixel, it is left unchanged. If the processing pixel takes the maximum or minimum gray level then it is noisy pixel which is processed by MDBUTMF, the steps of the MDBUTMF are elucidated as follows[6]:

1. Select 2-D window of size 3 3. Assume that the pixel being processed is I(i,j).
2. If 0 <I(i,j)<255 then is an uncorrupted pixel and its value is left unchanged.
3. If I(i,j)=0 or I(i,j)=255 then is a corrupted pixel then two cases are possible as given in Case:
   a) If the selected window contain all the elements as 0's and 255's. Then replace I(i,j) with the mean of the element of window.
   b) If the selected window contains no tall elements as 0's and 255's. Then eliminate 255's and 0's and find the median value of the remaining elements. replace I(i,j) with the median value.

4. Repeat steps 1 to 3 until all the pixels in the entire image are processed.

4. Nearest Interpolation and Median (NIM) Algorithm

This algorithm depending on absolute deferent between median and mean value in the kernel, after noise values are eliminated, the data in the gray image can by classified in homogenous and edge region. Generally, in the local region, the median value nearest from mean (or interpolation) value if the region is homogenous, whereas in the edge region the median value not equals the mean value, we can benefit from this fact for noise removed from the image in the impulse noise depending on the nearest interpolation value. Figure (1) illustrated the kernel with size 3×3. The nearest interpolation for the center value of the kernel (I_{22}) is calculated from two vectors:

I_{11}	I_{12}	I_{13}
I_{21}	I_{22}	I_{23}
I_{31}	I_{32}	I_{33}

Figure 1: the [illegible] 3×3).

$$V_1 = [\, I_{12} \quad I_{21} \quad I_{32} \,] \qquad (3)$$

$$V_2 = [\, I_{11} \quad I_{31} \,] \qquad (4)$$

V_1 is the first neighborhood; V_2 is the second neighborhood,
And V_{1e}, V_{2e} are eliminated the (0 and 255) from the V_1 and V_2 respectively where:

$$\{V_{1e}\,,\, V_{2e}\} \neq \{0,255\} \qquad (5)$$

The nearest interpolation (I_{ni}) is existence should be V_{1e}, V_{2e} include at least one value this mean:

$$\{V_{1e}\,,\, V_{2e}\} \neq \{\emptyset\} \qquad (6)$$

$$If\ V_{1e} \neq \{\emptyset\}\ \ then\ \ I_{ni} = V_{1e}(1) \qquad (7)$$

V_{1e} is existence (includes one or two or three value), the interpolation value is the first value in the vector V_{1e}.

$$If\ V_{2e} \neq \{\emptyset\}\ \ then\ \ I_{ni} = V_{2e}(1) \qquad (8)$$

V_{2e} is existence (includes one or two value) ,the interpolation value is the first value in the vector V_{2e}. Finally, the I_{22} was replaced by the I_{ni}

$$I_{22} = I_{ni} \qquad (9)$$

Figure (2) shows flowchart of this algorithm.

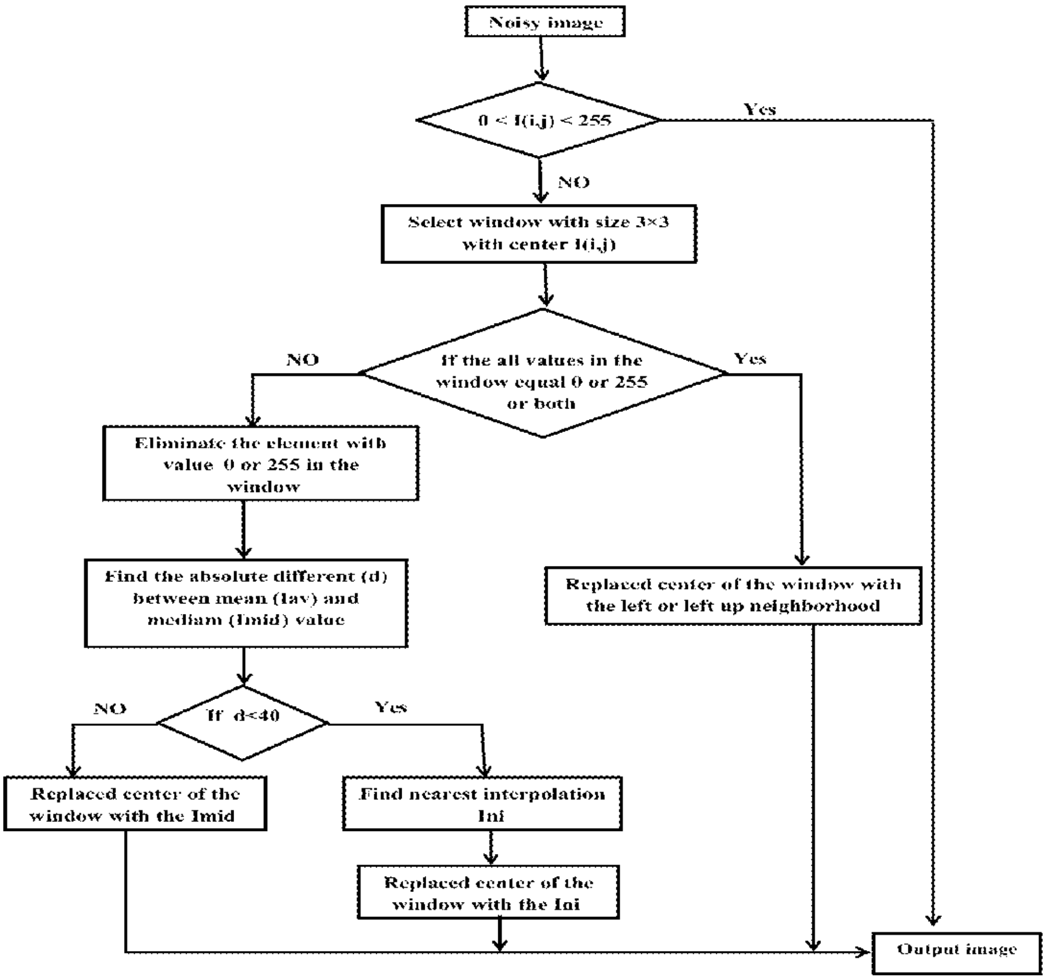


Figure 2: Flow chart of NIM.

The steps of the algorithm are elucidated as follows:

1. A window of size 3x3 is selected. The pixel to be processed is I(i,j).
2. If 0 < I(i,j) < 255 .The I(i,j) is an uncorrupted and it remains without change.
3. If the all values in the window equal 0 or 255 or both, Replaced I(i,j) with the I(i-1,j).
4. If the all values in the window not equal 0 or 255 or both, Eliminate the element with values 0's and 255's in the window
5. Find the absolute difference between the average and median value Iav , Imid .
6. If d > 40 then Replaced the center value I(i,j) with the nearest interpolated value Ini .and If d ≤ 40 then Replaced the center value I(i,j) with the value Imid.
7. Repeat steps 1 to 6 until all the pixels in the entire image are processed.

5. Results and Discussion

Gray-scale images such as peppers and elephants of size 512×512 with type bmp have been used to test the performance of the algorithm. Images will be corrupted by salt-and-pepper noise at different high noise densities, such as noise (80%), (85%) and very high noise densities as (90%) and (95%). Then the NIM algorithm is applied to the corrupted image to remove the noise, yielding the restored gray-scale image. The performance of the restoration process is quantified using PSNR and IEF. The results show improved performance of the proposed algorithm in terms of these measures. All the filters are implemented in MATLAB R2008. The metrics for comparison are defined as follows:

$$PSNR = 10\log_{10}\left(\frac{255^2}{\sqrt{MSE}}\right) \quad (10)$$

$$MSE = \frac{1}{xy}\sum_{i=1}^{x}\sum_{j=1}^{y}(I_{ij} - Ie_{ij})^2 \quad (11)$$

$$IEF = \frac{\sum_{i=1}^{x}\sum_{j=1}^{y}(I_{ij} - In_{ij})^2}{\sum_{i=1}^{x}\sum_{j=1}^{y}(I_{ij} - Ie_{ij})^2} \quad (12)$$

Where I_{ij} is the original image, Ie_{ij} is the restored image and In_{ij} is the corrupted image. Figures (3&5) show the original and corrupted cameraman images. and restored images obtained by the various filters such as SMF, which uses 11×11 for high noise density (80%), and 13×13 window size for very high noise density (85%, 90% and 95%), AMF which uses 13×13 for high noise density and 15×15 window size for very high noise densities, in DBA, MDBUTMF and MF which uses a small fixed size 3×3 window for all noise levels. Figures (4&6) show the comparison of different filters, performed on Lena and Boat gray-scale image at various noise densities, the first column represents the processed image using SMF from 80% to 95% noise densities. Subsequent columns represent the processed images for AMF, DBA, MDBUTMF and DBMM. From the Figures (4&6) , it is possible to observe that the quality of the restored image using proposed algorithm (DBMM) is better than the quality of the restored image using other algorithms. The PSNR and IEF values of the proposed algorithm are compared against the existing algorithms by varying the noise density from 80% to 95% and are shown in Table 1&2 for Lena and Boat, From the Tables it is observed that the performance of the proposed algorithm (DBMM) is better than the existence algorithms at both high and very high noise densities, these data in the table are display in the figures (7) & (8) that illustrated the IEF and PSNR versus the noise density for processing images at different noise densities.

6. Conclusion

In this paper, an algorithm (INM) is proposed which gives better performance in comparison with SMF, AMF, DBA and MDBUTMF for impulse salt & pepper noise removal, depending on the IEF and PSNR. The suggested algorithm has been tested at high and very high noise densities levels, the INM gives better results in comparison with other existing algorithms depending on visual and quantitative results.

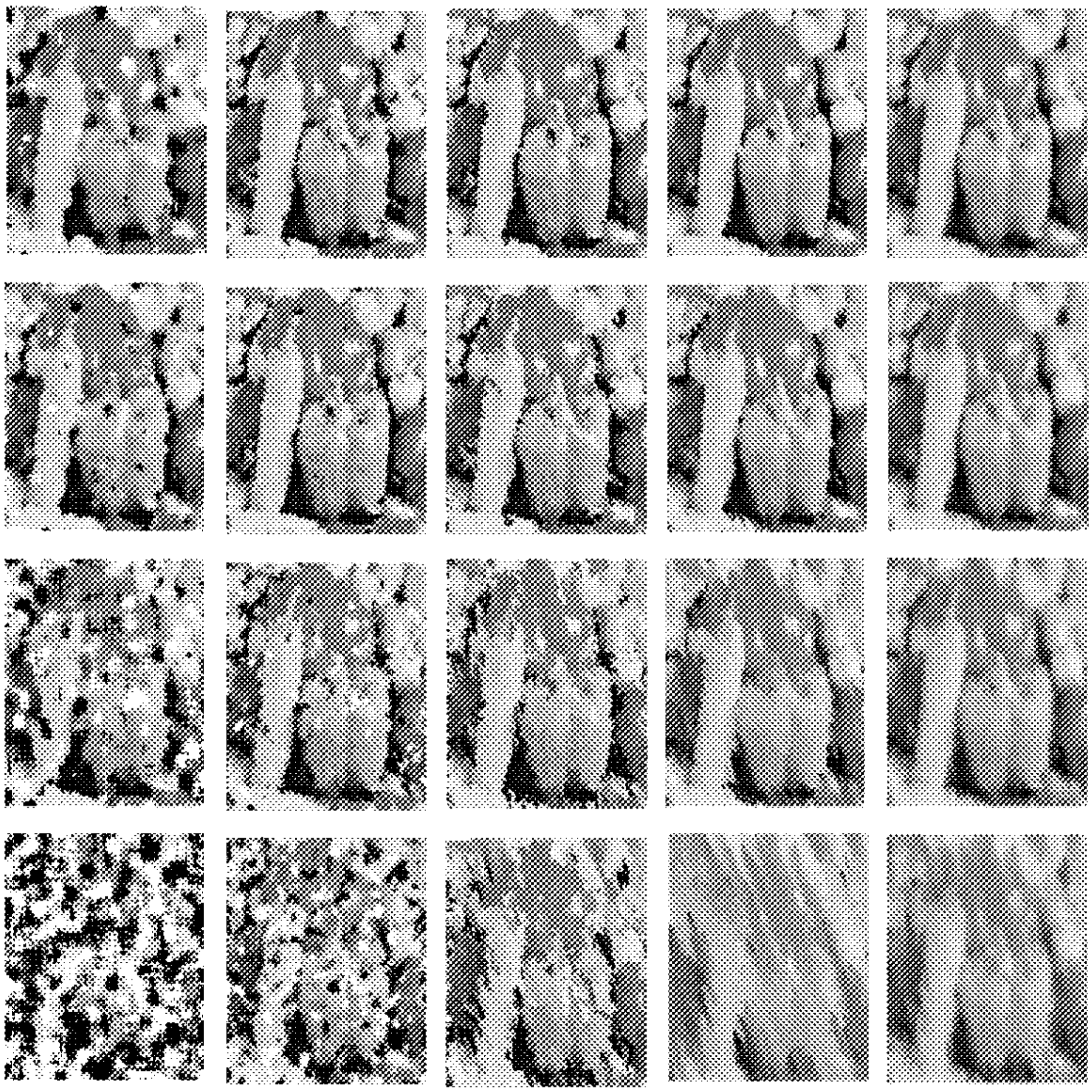
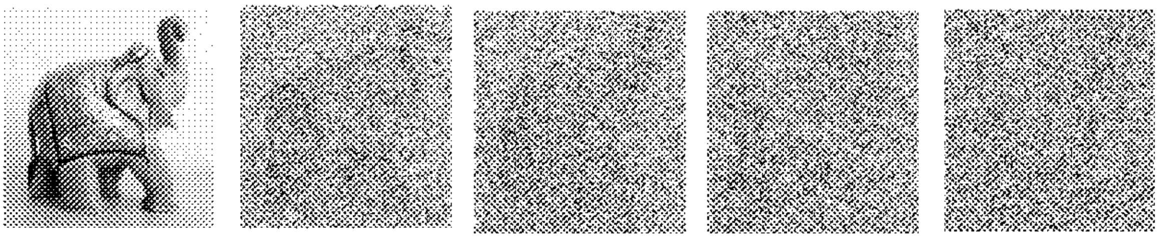
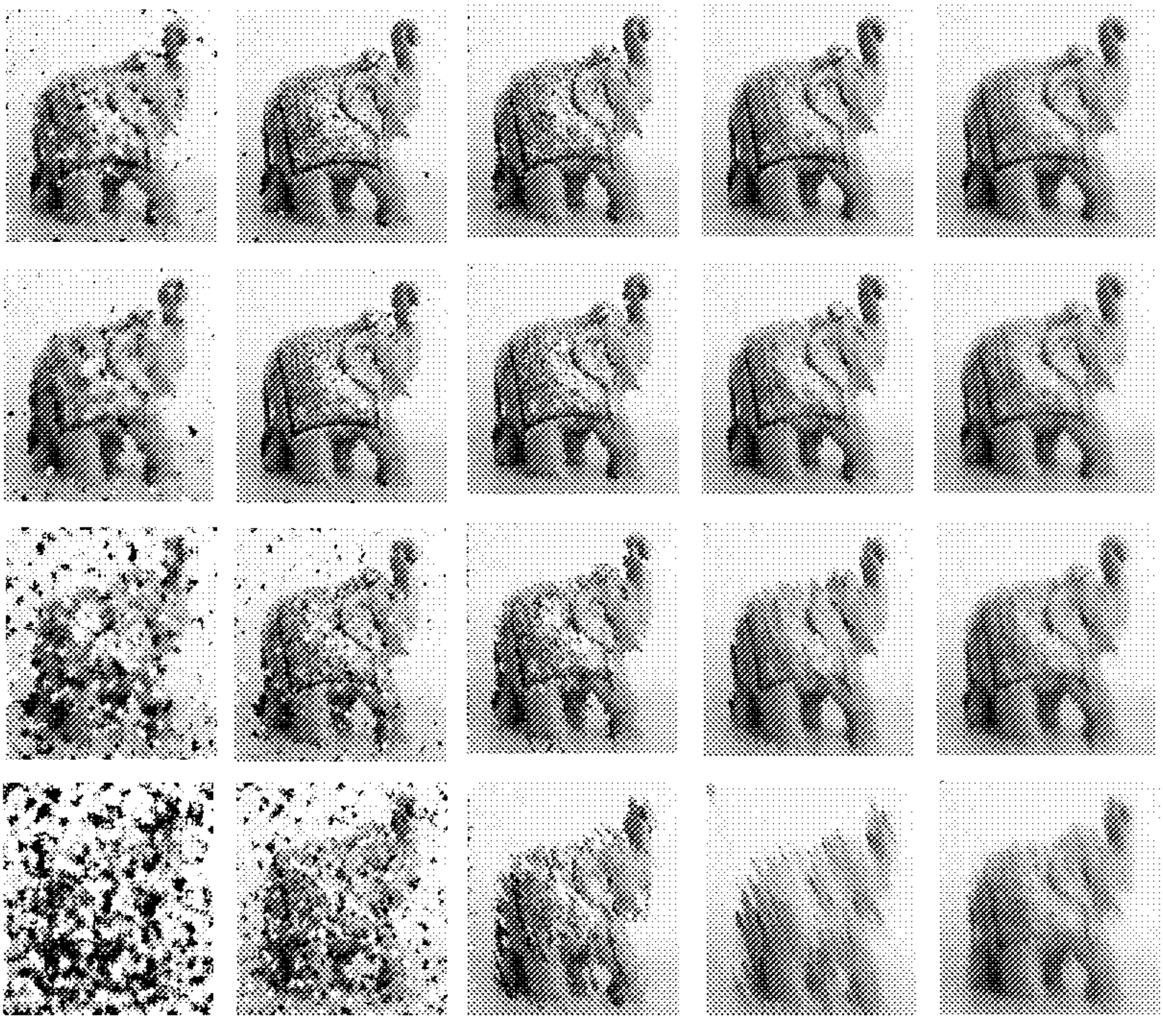


Figure 4: Results of different algorithms for peppers an image by using (a) SMF. (b) AMF. (c) DBA. (d) MDBUTMF. (e) INM. Every column shows the denoising images that corrupted with densities 0.8, 0.85, 0.9 and 0.95 respectively.



(a) (b) (c) (d) (e)

Figure 5: elephant in (a) Original image and its images distorted by salt and pepper noise in (b, c, d and e) at densities 0.8, 0.85, 0.9 and 0.95 respectively.



images that corrupted with densities 0.8, 0.85, 0.9 and 0.95 respectively.

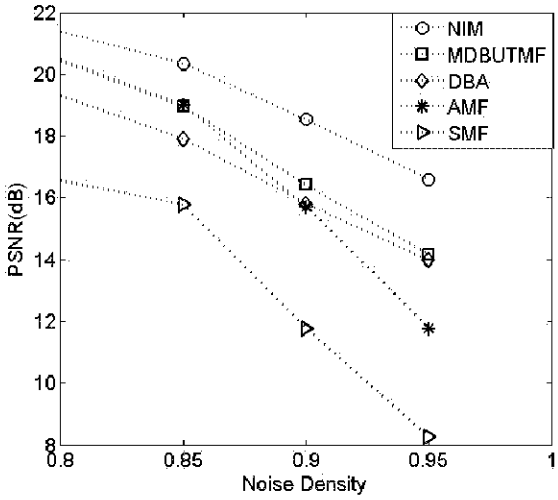
Table 1: PSNR and IEF for various filters for peppers image at different noise densities.

PSNR in dB					
Noise %	**SMF**	**AMF**	**DBA**	**MDBUTMF**	**DBMM**
80	16.56	20.502	19.31	20.439	21.371
85	15.787	19.006	17.914	18.956	20.351
90	11.761	15.698	15.807	16.407	18.543
95	8.25	11.767	13.98	14.168	16.597
IEF					
Noise %	**SMF**	**AMF**	**DBA**	**MDBUTMF**	**DBMM**
80	10.697	25.653	18.674	25.614	31.195
85	9.509	19.292	14.208	19.163	25.986
90	4.29	9.742	9.314	11.075	17.986
95	2.367	4.471	6.394	6.606	11.788

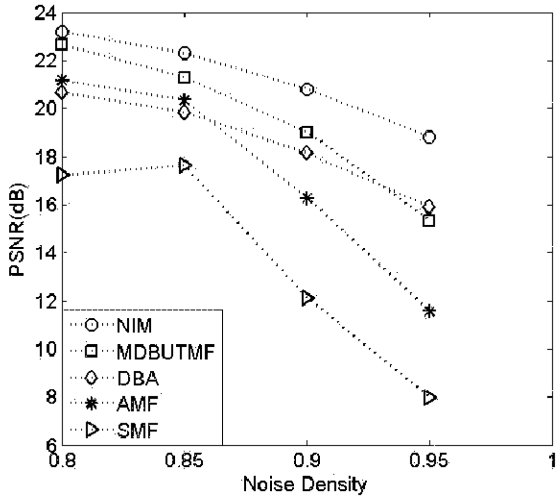
Table 2: PSNR and IEF for various filters for elephant image at different noise densities.

PSNR in dB					
Noise %	**SMF**	**AMF**	**DBA**	**MDBUTMF**	**DBMM**
80	17.212	21.168	20.654	22.648	23.201
85	17.642	20.34	19.84	21.308	22.282
90	12.128	16.273	18.15	19.01	20.805
95	7.985	11.571	15.918	15.343	18.822
IEF					
Noise %	**SMF**	**AMF**	**DBA**	**MDBUTMF**	**DBMM**
80	13.798	33.561	29.105	47.855	53.988
85	16.063	29.34	25.444	37.155	46.077
90	5.067	12.337	18.408	23.268	34.803
95	2.341	4.622	11.643	10.541	23.083

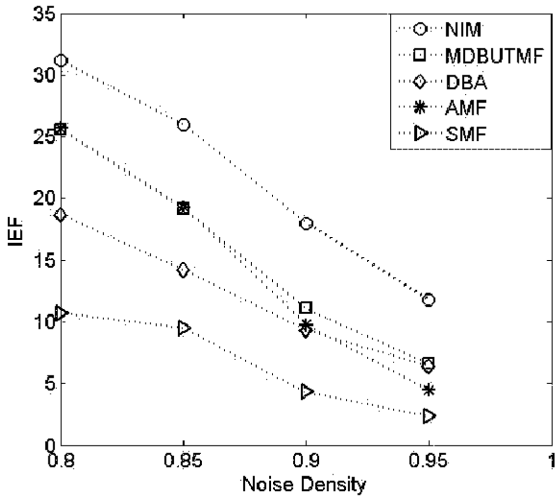
Figure 7: Noise density versus PSNR for (a) peppers and (b) elephant images.



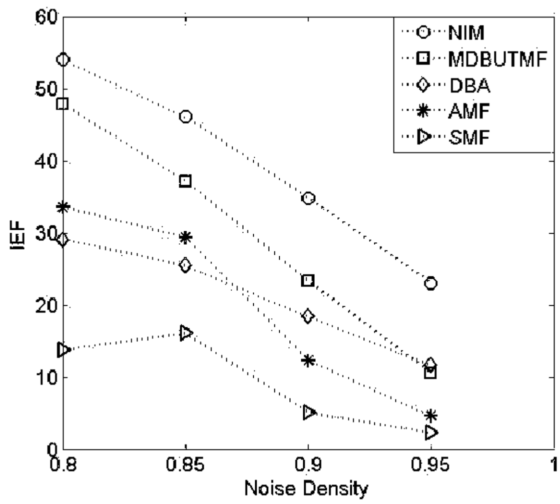
(a)



(b)



(a)



(b)

Figure 8: Noise density versus IEF for (a) peppers and (b) elephant images.

Referance

1.A. Bovik, Handbook of Image and Video Processing. NewYork: Aca- demic, 2000.
2. N.C.Gallagher, Jr. and G.L.Wise, "A Theoretical Analysis of the Properties of Median Filters," IEEE Trans. Acoust., Speech and Signal Processing, vol.ASSP-29, pp.1136-1141, Dec.1981.
3. S.Manikandan, O.Uma Maheswari, D.Ebenezer, "An Adaptive Recursive Weighted Median Filter withImproved Performance in Impulsive Noisy Environment",WSEAS Transactions on Electronics, Issue 3, Vol.1, July2004.
4. 7. Ho-Ming Lin, "Median Filters with Adaptive Length", IEEE Transactions on Circuits and Systems, Vol. 35, No. 6, June 1988.
5. K. S. Srinivasan and D. Ebenezer. A new fast and efficient decisionbased algorithm for removal of high density impulse noise, IEEE Signal Process. Lett., Vol. 14, No. 3, pp. 189–192, Mar. 2007.
6.S. Esakkirajan, T. Veerakumar, A. N. Subramanyam and C. H. Prem Chand, Removal of High Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter, IEEE Signal Processing Letters, Vol.18, No.5, 2011, pp.287-290.
7. R. H. Chan, C .-W. Ho, and M. Ni kolova, "Salt-and-pepper noise removal by median-type noise detectors and detail preserving regularization," IEEE Transact ions on Image Processing , vol. 14, no. 10, pp. 1479–1485, 2005.
8. Gonzalez and Woods, Digital image processing, 2nd Edition, prentice hall, 2002.
9. D.Dhanasekaran, K. Bagan, High Speed Pipeline Architecture for Adaptive Median Filter, European Journal of Scientific Research, Vol.29 ,No.4, PP.454-460. , 2009.

ازالة الضوضاء النبضية العالية الكثافة بالاعتماد على خوارزمية الاستكمال لاقرب جوار والوسيط

هناء حسن كريم
الجامعة المستنصرية /كلية التربية- قسم الفيزياء

الخلاصة

في هذا البحث تم اقتراح خوارزمية الاستكمال لاقربة جوار والوسيط (NIM) لازالة ضوضاء الملح والبهار العالية الكثافة من الصور الرقمية . الخطوة الاولى في هذه الخوارزمية هي استخلاص عناصر الصورة التي تحتوي على الضوضاء ، اما الثانية فهي حساب قيمة الخطأ المطلق بين المعدل والوسيط في النافذة ، فيما اذا كانت المنطقة متجانسة يتم استبدال القيمة المركزية داخل النافذة بالمعدل، بخلاف هذا يتم استبدالها بالوسيط بالاعتماد على قيمة الخطأ المطلق. ان الخوارزمية المقترحة هي افضل بكثير من مرشح الوسيط البسيط (SMF) و مرشح الوسيط المطور(AMF) وخوارزمية التحكم التلقائي(AMF) وخوارزمية التحكم التلقائي لقيم الوسيط المنتقاة (MDBUTMF) . الخوارزمية المقترحة تم تجريبها على مختلف الصور ذات التدرجات الرمادية واظهرت نتائج افضل للقيمة العظمى لنسبة الاشارة والضوضاء (PSNR) ومعامل التحسين للصورة (IEF) .