

Removal of High Density Salt and Pepper Noise from Digital Images

Sruthi Ignatious¹

¹Mar Baselios College of Engineering and Technology,
Nalanchira, Trivandrum, Kerala, India
sruthieliza@gmail.com

Abstract: This paper compares efficient filters for the restoration of images that are corrupted by a high density of Salt and Pepper noise. There are several non linear filters for the restoration of the images. Among these, median filters are the most popular and powerful technique for the removal of salt and pepper noise. Different variations of median filters are now available. These variations outperform the standard median filter. These filtering techniques include two stages, such as noise detection stage and noise filtering stage. The noise detection stage will identify the noisy pixel within the image and noise filtering stage will filter the image. The performance qualities of different filters are measured using Peak-Signal to Noise Ratio (PSNR) and Image Enhancement Factor (IEF) value.

Keywords: Median filter, noise detection, noise filtering, noisy pixel.

1. Introduction

Digital images have wide applications in the field of military, engineering, medicine, astronomy etc. If these images are attacked by noise means it may cause worst effect. Noise is produced in the processes of storage, acquisition and transmission. This will degrade the quality of digital images. By applying filters we may be able to filter the image from the attack of noise. In this work salt and pepper noise is used for highlighting the performance of the filters. In salt and pepper noise model the noisy pixels are '0' and '255'. They are the minimum and maximum intensity pixel values within the image.

Before applying the filters directly into the image the noise density will be identified. The different approaches of identifying noise density are:- fuzzy approaches [1],[2], neural approaches [3] and boundary based approaches [4]. Among this boundary based approach is preferred due to simplicity.

Mainly there are two types filters for removing the salt and pepper noise. They are linear and non linear filters. Linear filters such as average filter produce blurring of images. Usage of non linear filter will reduce the blurring effect within the image. Median filters are the common type of non linear filter. Standard Median filter [5][6] will work effectively only when the noise density within the image is less. This filtering technique will not identify the noise density within the image. Instead directly apply the filtering process within the image. If the noise density is greater the edge details within the image will not be preserved. In Adaptive Median Filter [7], depending on the noise density present within the image, the filtering window will be selected. This filtering method also performs well when the noise density within the image is less. But when the noise density is high the size of the filtering window will be large. So the image will be blurred. To

enhance the capability of non-linear median filters, the switching concept was introduced.

In Switching Median Filter [8], the filtering will be done based on some threshold value. This threshold value selection depends on the noise density within the filtering window. The major drawback of this filtering technique is that it is difficult to define the threshold value. Also this technique will not preserve the edge details when the noise density is high.

The purpose of this paper is to compare different variations of median filter and to find the accurate filter by measuring the performance quality of the techniques.

2. Decision Based Unsymmetric Trimmed Median Filter

Decision Based Unsymmetric Trimmed Median Filter (DBUTMF) [9] reduces the blurring of the images to some extent. In this filtering technique, select a 3X3 window within the image and arrange the pixels within the window in either ascending or descending order. Then remove the noisy pixels within that window and replace the current pixel with the median of the remaining pixel values. In the case of high noise density the selected 3X3 window contains all the pixels as noisy. In such case, replace the current pixel with the mean of the pixels within the selected window.

2.1 Algorithm

This algorithm processes the image by checking whether the pixel is corrupted or uncorrupted. If the pixel is '0' or '255', then that pixels are corrupted, all the other pixels are uncorrupted pixels. All the uncorrupted pixels will be kept unchanged. The remaining corrupted pixels will be replaced by the filtering process.

Step 1: Impose a 3X3 window centered around the current pixel, P(i,j).

Step 2: If the pixel $P(i,j)$, is in the range $0 < P(i,j) < 255$ then the pixel $P(i,j)$ is an uncorrupted pixel. In this case no processing is required for $P(i,j)$.

Step 3: If $P(i,j) = '0'$ or $'255'$ then the pixel is corrupted. Here two cases are to be considered:-

Case 1: If the selected 3×3 window contain all the elements as noisy, that is, 0 & 255, then replace the central current pixel $P(i,j)$ with the mean of the elements of that window.

Case 2: If the selected 3×3 window contain not all the elements as noisy, that is, 0 & 255, then remove 0 & 255 from the selected window and replace $P(i,j)$ with the median of the remaining elements of that selected window.

Step 4: Repeat the above steps until all the pixels within the entire image are processed.

3. Iterative Average Estimation Filter using BDND algorithm

The Iterative Average Estimation Filter (IAEF) [10] using BDND algorithm consists of two stages, such as Noise Detection stage and Noise filtering stage. The noise detection stage uses Boundary Discriminative Noise Detection (BDND) algorithm for identifying the noisy pixels within the image. The noise filtering stage uses average estimated value of the uncorrupted pixels and replaces the corrupted pixel with the estimated value. This filtering technique consumes less time.

3.1 Noise Detection Stage

The noise detection stage identifies the corrupted pixel. The noise detection stage uses BDND [10] algorithm for identifying the corrupted pixel. After applying the BDND algorithm, a two dimensional binary detection map is created in which 0's indicates the corrupted pixels and 1's indicates the uncorrupted pixels. The steps involved in the BDND algorithm are given below.

Step 1: Select a 21×21 window around $P_{i,j}$, where $P_{i,j}$ is the current processing pixel in the image.

Step 2: Arrange the pixels within the above window range in ascending order and store the pixels into the vector V_0 , then find the median of the pixel values within V_0 and store the median value into the variable med .

Step 3: Calculate the difference of intensity values of the nearby pixels within the vector V_0 and store the result into the vector V_D .

Step 4: Find the pixels with maximum intensity difference from V_0 within the intervals of $[0, med]$ and $[med, 255]$.

Set these intensity values as the decision boundaries namely $b1$ and $b2$.

Step 5: If the pixel $P_{i,j}$ belongs to the middle cluster it is an uncorrupted pixel and the classification process can be stopped.

Otherwise consider the second iteration, in Step 6

Step 6: Select a 3×3 window around $P_{i,j}$ and repeat steps 2 to 4.

Step 7: If the current pixel $P_{i,j}$ belongs to the middle cluster range that pixel is classified as an "uncorrupted"

pixel. Otherwise that pixel is classified as "corrupted" pixel.

By applying this algorithm uncorrupted and corrupted set of pixel values are identified. So the positions of the uncorrupted pixels within the image are marked as one and the positions of the corrupted pixels within the image are marked as 0. Thus a two dimensional binary detection map is obtained.

3.2 Noise Filtering Stage

The noise filtering stage uses average estimation value to replace the corrupted pixel. Select a 3×3 window around each corrupted pixel within the image and detection map. Count the number of corrupted pixels within the selected window. If the count value is positive and central pixel within the selected window is noisy, then construct an array with uncorrupted pixels within the selected window. For replacing the central uncorrupted pixel, check whether there is minimum three uncorrupted or noise free pixels, if the condition is satisfying, replace the central corrupted or noisy pixel with the estimated value. Estimated value is the average of uncorrupted pixels within the selected window.

After this update the corresponding position of the central pixel within the detection map as 1. Thus the detection map can be updated.

3.3 Algorithm

The different steps involved in IAEF are given below

Step 1: Obtain the noisy image as input.

Step 2: Construct the detection map of the input image by using the above specified BDND algorithm.

Step 3: Check the detection map, for each uncorrupted pixels. If so do a)-e) until all the entries in the detection map becomes 0

- Consider each uncorrupted pixel $P_{i,j}$
- Impose a 3×3 window with central pixel $P_{i,j}$ as processing pixel within the image.
- Construct vector R that contains uncorrupted pixels.
- Compute the estimate value and replace the processing pixel with the estimated value, if the number of uncorrupted pixel is greater than or equal to 3 and then update the detection map.
- Process each pixel and also update the detection map.

Step 4: Display the new noise free image as output.

4. Iterative Switching Filter

The Iterative Switching Filter (ISF) [11] consists of three stages, such as noise detection and construction of detection map, noise density identification and noise filtering. This filtering technique will preserve edge details within the image.

4.1 Noise Detection and construction of detection map

In the case of salt and pepper noise, maximum and minimum intensity pixel values denote the uncorrupted pixels. So the corresponding positions of the pixel values within image will be treated as uncorrupted pixels. The positions of those pixels will be marked as 0 and the remaining pixels will be marked as 1 in the detection map. Thus a two dimensional binary

detection map will be created.

4.2 Noise density identification

The noise density is the ratio of sum of the uncorrupted pixels within the detection map to the total size of the detection map.

4.3 Noise filtering

In the filtering process there are two cases [11]. If the noise density is less than or equal to 40% then the uncorrupted central pixel within the selected 3X3 window will be replaced with the estimated value. Otherwise the central uncorrupted pixel within the selected window will be replaced with the weighted median value. The weight to be used is 3. This process will be continued until all pixels within the image is cleared from noise.

4.4 Algorithm

Step 1: Input the noisy image.

Step 2: Construct the detection map, D, binary matrix where zeros represent uncorrupted pixels and ones represent the corrupted pixels.

Step 3: Calculate the noise density within the image from the detection map.

Step 4: Find the uncorrupted pixels using the detection map, if there is an uncorrupted pixel, then do the following steps.

- Consider each uncorrupted pixel for processing from the image.
- Select a 3X3 neighborhood of the processing pixel and construct a vector which contains the uncorrupted pixels within the selected window.
- Compute the noise density. If the noise density within the selected window is less than or equal to 40% go to step d) otherwise go to step e).
- Compute the number of uncorrupted pixels within the selected window, if it is greater than or equal to 3 , then calculate the estimate value which is the average of values of vector and assign it to G(i,j). Otherwise go to step b)
- Compute the new value for the processing pixel as G(i,j). This value is the weighted median of the values within the vector.

Step 5: Assign the value of the processing pixel to G and update the detection map after the processing of the pixel.

Step 6: Check for any uncorrupted pixel, if so goto step 4 otherwise go to step 7

Step 7: Display the processed image as output.

Step 8: Stop

5. Modified Noise Adaptive Switching Median Filter

The Modified Noise Adaptive Switching Median Filter (MNASMF) [12] is an improvement over the Switching Median Filter [8]. MNASMF promises better performance quality than the other existing filters.

MNASMF consists of two stages. They are Noise detection stage and Noise Filtering stage. The noise detection stage uses the BDND algorithm for detecting the uncorrupted pixel. The different steps involved in the BDND algorithm are explained in the above section. The noise filtering stage uses the Modified Noise Adaptive Switching Filter for removing the uncorrupted pixels within the image.

5.1 Noise Filtering

Mainly two improvements are brought into the new filtering technique. First improvement is, first determine the filtering and maximum window size using the Table 1. Then estimate the noise density and also the number of pixels within the image (denoted by N_i). If number of uncorrupted pixels denoted as N_u , and $N_u < (1/2(1-P)N_i)$ and $W_F < W_D$, then extend the filtering window size by one pixel outside and repeat the same above process. If the above condition is not satisfied replace the central pixel with the median of the pixels within the filtering window.

Table 1: Suggested window size for noise density

Noise Density	$W_D \times W_D$
$0\% < p \leq 20\%$	3X3
$20\% < p \leq 40\%$	5X5
$>40\%$	7X7

The second improvement is the extension of the first improvement. It is done by determining the value of $Z_{i,j}$. In this instead of replacing the central pixel with median value, Y_{ij} , it will be replaced with $Z_{i,j}$. It is determined using the equation given below:-

$$Z_{ij} = Y_{ij} + \frac{1}{D} \sum_{k=1}^{N_u} \frac{V_u(k) - Y_{ij}}{d(k)} \quad (1)$$

where V_u is the vector containing the set of uncorrupted pixels. Y_{ij} is the original median value.

$$D = \sum_{m=1}^{N_u} \frac{1}{d(m)} \quad (2)$$

where N_u is the number of uncorrupted pixels within the filtering window. $d(m)$ is determined using the equation given below

$$d(k) = |S(k) - i| + |T(K) - j| \quad (3)$$

$d(k)$ is the distance between the k^{th} value in V_u and the central pixel. $S(k)$ and $T(k)$ are the row and column indices.

6. Performance Measures

The performance of the above algorithms is measured using PSNR (Peak-Signal to Noise Ratio) and Image Enhancement Factor (IEF) values. These values are defined as follows.

$$PSNR = 10 * 10 \log_{10} \left(\frac{255^2}{MSE} \right) \quad (4)$$

$$MSE = \sum_{m,n} \frac{[O(m,n) - R(m,n)]^2}{(M * N)} \quad (5)$$

$$IEF = \frac{\left(\sum_{m,n} (P(m,n) - O(m,n))^2 \right)}{\left(\sum_{m,n} (R(m,n) - O(m,n))^2 \right)} \quad (6)$$

where MSE means Mean Square error. O is the original image, R is the restored image, P is the corrupted image and MXN is the size of the original input image.

7. Results

The performances of the different filters are measured using the PSNR and IEF values. The measurement is carried out by giving the standard test Lena image of size 512X512. Figure 2 shows the results of different filters when the original image is corrupted with 90% of noise. Table 2 and 3 shows the PSNR and IEF values of different algorithms for different noise density.



Figure 1: Standard test image of LENA



Figure 2: Results of different filters when the image is corrupted with 90% of salt and pepper noise. a) Output of DBUTMF, b) Output of IAEF, c) Output of ISF d) Output of MNASMF.

Table 2: The PSNR values of different algorithms

Noise Density	DBUTMF	IAEF	ISF	MNASMF
20%	34.98	38.61	38.64	38.68
40%	30.45	34.72	33.47	33.75
60%	26.78	31.48	30.67	29.33
80%	22.11	27.46	27.55	27.34
90%	19.83	24.48	25.10	25.01

Table 3: The IEF values of different algorithms

Noise Density	DBUTMF	IAEF	ISF	MNASMF
20%	566.77	413.55	416	623.45
40%	422.89	341.66	254.33	503.13

Noise Density	DBUTMF	IAEF	ISF	MNASMF
60%	262.12	240.49	199.76	278.99
80%	101.23	127.32	125.93	133.51
90%	35.12	71.75	84.03	89.24

8. Conclusion

In this paper, different techniques for the removal of high density of salt and pepper noise are discussed. The performances of the different techniques are measured for high and low noise density.

References

- [1] Luo, W., "Efficient removal of impulse noise from digital images," IEEE Transaction on Consumer Electronics 52(2), 523–527 (2006)
- [2] Schulte, S., Nachtegaele, M., Witte, V.D., Weken, D.V., Kerre, E.E., "A Fuzzy Impulse Noise Detection and Reduction Method," IEEE Transactions on Image Processing 15(5), 1153–1162 (2006)
- [3] Schulte, S., Witte, V.D., Nachtegaele, M., Weken, D.V., Kerre, E.E., "Fuzzy Two-Step Filter for Impulse Noise Reduction From Color Images." IEEE Transactions on Image Processing 15(11), 3567–3578 (2006)
- [4] Nair, M.S., Raju, G., "A new fuzzy-based decision algorithm for high-density impulse noise removal," Signal Image and Video Processing, doi:10.1007/s11760-010-0186-4
- [5] J.Astola, P. Kuosmaneen, "Fundamentals of Nonlinear Digital Filtering", Boca Raton, FL: CRC, 1997.
- [6] G.Huang, T.Yang, G.Tang, "A Fast Two dimensional median Filtering", IEEE transactions on acoustics, speech, and signal processing, vol. Asp-27, no. 1, february 1979.
- [7] Abdul Majid, Muhammad Tariq Mahmood, "A Novel technique for removal of high density impulse noise from digital images", 2010 6th International Conference on Emerging Technologies(ICET)
- [8] Ng, P.-E., Ma, K.-K., "A Switching Median Filter with Boundary Discriminative Noise Detection for Extremely Corrupted Images," IEEE Transactions on Image Processing 15(6), 1506–1516 (2006)
- [9] S.Esakkirajan, T.Veerakumar, AdabalaN.Subramanyam, C.H.PremChand, "Removal of High density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter," IEEE Signal process. Lett., Vol. 18, no 5, May 2011
- [10] Sruthi Ignatious, Robin Joseph, "Iterative Average Estimation Filter Using BDND algorithm for the removal of high density Impulse noise," International Journal of Computer Science and Mobile Computing (IJCSMC) Vol.2 Issue 13, December 2013.
- [11] Jisha John, Ann Mary Jacob, Mekha Prasannan, Priyanka Suja Pradeep, Sruthi Ignatious, "Iterative Switching Filter for high density impulse noise," International Journal for Science and Engineering Research (IJSER) Vol.3 Issue 7, July 2012.

- [12] Iyad F.Jafar, Rami A.AINa'mneh and Khalid A.Darabkh, "Efficient Improvements on the BDND Filtering Algorithm for the Removal of High-Density Impulse Noise," IEEE Transactions on Image Processing, vol.22, No. 3, (1223-1232).March 2013.



Author Profile

Sruthi Ignatious received the B.Tech degree in Information Technology from Mar Baselios college of Engineering and Technology in 2012. She is currently doing M.Tech degree in Computer Science and Engineering at Mar Baselios college of Engineering and Technology. Her current research interests include image processing, signal processing and biomedical engineering.