An Adaptive Grayscale Image De-noising Technique by Fuzzy Inference System

Ashik Mostafa Alvi, Sheikh Faishal Basher, Ahsan Habib Himel,
Tonmoy Sikder, Mashrikul Islam, Rashedur M Rahman
Department of Electrical and Computer Engineering, North South University
Plot – 15, Block – B, Bashundhara, Dhaka 1229, Bangladesh
{ashik.mostafa, faishal.sheikh, ahsan.himel, sikder.tonmoy, mashrekul.islam, rashedur.rahman}@ northsouth.edu

Abstract—**Noise is an issue for image in terms of corruption. Some noises can be automatically added with the image while capturing the image. Camera sensors and lighting factors are responsible for those. Noises that can be added with the image is known as additive noise. To get a perfect image noise reduction is very important. Edge detection, image enhancing, object detection etc. are directly related with noise reduction. Noises need to be redacted before image processing. There are many well established noise reduction techniques available. Linear noise reducing algorithms are not efficient to reduce impulse noise as they are mainly focused on smoothing that makes the edges of an image blur. On the contrary, nonlinear algorithms do a great job for reducing impulse noises. In this paper, we have made a proposal which is an efficient algorithm for digital grayscale image de-noising using fuzzy logic. Our proposed algorithm uses the Fuzzy Inference System (FIS) for the membership value calculation. The noise affected pixels are determined by the FIS from the 3 × 3 neighborhood. Our method works very fine with impulse noises. The proposed algorithm returns almost the original noise free image which has a significant difference in the Peak Signal to Noise Ratio (PNSR) compared to the existing Median and Average filtering technique.**

Keywords- Fuzzy Logic; Image Filtering; Impulse Noise; Fuzzy Inference System.

I. Introduction

Noise is an error of an image that makes the image different from the real scene. Real intensity values are being manipulated by the noise. These noises can be additive or come from the sensor of that image capturing camera or the lighting factors. When a sensor captures an image, it filters the image with different smoothing algorithms. Digital image capturing devices have better support for noise reduction. Nowadays, those devices have a good image capturing sensor. All recording devices, both analog and digital, have traits which make them susceptible to noise [1]. An image with corrupted pixels can be represented by,

$$f(x, y) = i(x, y) + \eta(x, y) \tag{1}$$

here, f (x, y) represents the corrupted image, i (x, y) stands for the noise free original image and η (x, y) is the noise which is mixed the original image.

The very first thing that every image processing task requires is that noise reduction. There are several types of noises. The most commonly occurred noises are i) Multiplicative noise, ii) Additive noise, iii) Impulse noise. Additive noises are i) Gaussian Noise, ii) Locavore Noise, iii) Poisson noise.

When a certain distribution generates a value and then added with the image pixels is known as additive noise. Impulse noise is usually characterized by some portion of image pixels that are corrupted, leaving the remaining pixels unchanged [2]. It is the most common noise that occurs frequently. Examples of impulse noise can be fixed valued, randomly generated and salt-and-pepper noises. It can be added within the image while transferring the image for data communication or capturing. It does not affect all the pixels of an image like Gaussian or Poisson noise. Impulse noise exchanges some pixel values of an image which can change the gray value of the image. If an image is corrupted with salt-and-pepper noise, then the pixels that are corrupted will have the gray value in the range 0 to 255. Among all the additive noise Multiplicative noise is more sensitive regarding noise removal. Multiplicative noise is generally more difficult to remove from images than additive noise because the intensity of the noise varies from the signal intensity [3-4]. Linear noise reduction algorithms do almost the same instead of checking the variation of the signal.

There are many well established noise removal algorithms. Tukey [5], Astola et al. [6], and Pitas et al. [7] have used the median filtering for de-noising. The median filter basically tries to replace the original corrupted pixel with the lowest noise, but it is not sufficient for high rate of noise.

In recent years, many upgrades have come for median filter to remove the impulsive noise. Adaptive median filter [8], The center weighted median filter [9], The Tri-state median filter [10], Weighted median filter [11], noise adaptive soft-switching median filter [12, 13] are the recent developed works of median filter. These algorithms are basically non-linear and work fine in removing noises but they are not so good in preserving the fine details and they blur the edges. So, we have tried to design algorithm which can deal with uncertain information like edges and shapes in the different portion of the image.

Some research works have been proposed using fuzzy logic which has the ability to process the uncertain information. Fuzzy filters have the features of restoring almost the original image and suitable for real applications. Russo and Ramponi [14, 15] proposed a fuzzy rule based filtering for image smoothing. This method used heuristic knowledge for de-nosing.

978-1-5386-2165-3/17/$31.00 ©2017 IEEE

Most of the fuzzy logic based algorithms are not suitable for impulse noise and not all the time produce eye pleasing results. When the noise ratio is very high, those algorithms fail to give a satisfactory result. So, we have proposed an adaptive fuzzy inference system (FIS) based filtering technique which de-noises the impulse noise, very particularly the salt-and-pepper noise. While other fuzzy filters use the gray level intensity value for finding noise and de-noising techniques, we used the intensity difference of the targeted pixels from the minimum, maximum, average and median in a 3×3 sliding window including the targeted pixel at the center of that window. The differences have been passed in the FIS. There are 95 If then rules used in the FIS for generating the noise rate's membership. This algorithm has proved better in PSNR and MSE results than other existing algorithms, because we prioritize the deviation within the window. From the deviation, we measure the noise ratio which helps us to adapt the best suited filter for the current window.

II. RELATED WORKS

Chowdhury, Gao and Islam [16] have proposed a method which utilizes fuzzy membership functions using mask filtering. They have discussed several impulse filtering methods. Mean filter is one of them. It is used for smoothing or blurring which in other word is "low pass" filtering [17]. Another one is median filter- it tries to de-noise the image and preserves the edges at the same time [9]. They also mentioned different fuzzy rule based filtering techniques.

Fuzzy Filtering Technique works by using membership function and gray pixels charted into fuzzy surface [18]. Fuzzy membership function distinguishes among properties, for example, textural properties, darkness etc. The filter algorithm they have proposed starts with a defined square mask and ascertains the level of intensity using fuzzy rules. The process of replacing the noisy or corrupted pixels starts with computing fuzzy membership functions. To remove the noisy pixels the algorithm calculates the fuzzy membership function within the boundary of a window or mask. The advantages of mean and median filter are shown in their proposed technique. Their proposed technique works as follows- it takes NxN window and then find- average, minimum and maximum intensity levels. After that fuzzy membership is assigned using a function and lastly the mask center position is replaced by the largest valued intensity level. The proposed techniques were tested using several corrupted images with different noise level. The method was compared with various filters, such as-Adaptive Median Filter (AMF), Standard Median Filter(SM), Standard Average Filter (SAF). It shows the best peak-signal-to-noise ratio value indicating the advantage of the methods proposed by the authors. The proposed filtering method is capable of removing high density impulse noise from both gray-scale and color images.

Preety D. Swami and Alok Jain [24] have proposed a method to restore the curves of an image after denoising. Because most of the time after filtering an image the curves or edges becomes blur. Basically, the authors have worked with the wavelet transform. They have divided an image into two parts one is noisy texture image and the other part is noisy cartoon image. They have denoised the first part using wavelet transform and the second one is denoised by wave atoms. Then they have combined two parts and produced an denoised image preserving the edges and curves.

Motta, Ordentlich, Ramírez, Seroussi and Weinberger [25] have modified the discrete universal denoising algorithm by adding statistical modeling tools. The algorithm uses two passes over the image matrix. First pass determines the conditional probability from the given corrupted neighborhood. In the final pass, there is decision making part for image denoising. The decision making is done by the conditional distribution which is determined in the first pass. Their proposed worked proved better in salt and pepper, M-ary symmetric and Gaussian noises.

Hussain, Masood and Jaffar [19] have proposed a fuzzy based image filtering The authors discussed about two image restoration techniques category reviewed by Liu and Li [20]- conventional image restoration and blind image restoration. Noise removal falls into the second category and more emphasized study area in recent days. They have proposed a filtering method dominated by fuzzy logic for noise removing and details preservation. They have mentioned several other techniques. Median filtering, center weighted median filters [9], noise adaptive soft-switching median filters [12, 13] are few of them. But, all of these works well only when noise probability is low. Couples of machine learning based techniques were also mentioned. Considering several drawbacks and failings of other techniques the proposed technique will keep the fine details and nullify lengthy training process. The authors claim that their method is better in impulse noise filtering from digital gray scale images with salt-pepper and long tailed impulse noise. They have utilized Histogram based Fuzzy Filter (HFF) and Novel Fuzzy Filter (NFF) to calculate fuzzy sets. They performed the experiment on two types of noises- salt and pepper noise, additive noise having long tailed impulses. The proposed method that includes the following modules: noise detection, intelligent fuzzy controller and details preservation module. The authors compared their experiment result (FBINR- Fuzzy based Image Noise Reduction) with the median filter (MF), HFF, NFF, DPFF and MHFF. Performance was tested considering four factors. Whereas, the error measures they have fixed SSIM (Structural Similarity Index Measure), MSE (Mean Square Error) and PSNR (peak-signal-to-noise ratio). From all the test cases, they have found that their proposed technique gives superior performance compared to other mentioned techniques considering those mentioned error measures. They have concluded that to preserve gray scale image and in impulse noise reduction their technique is more efficient.

Jampour, Ziari, Zadeh and Ashourzadeh [21] have proposed another method for detecting and reducing impulse noise which utilizes fuzzy logic and median heuristic filter. The process is divided into two parts- detecting noise and replacing nosy pixels with heuristic median filter. Referring to FIDRM [22] they have presented that their Fuzzy Logic and Median Heuristic Filter (FMHF) outperforms considering time and better peak-signal-to-noise ratio. To detect impulse noise using fuzzy logic they have done a series of task- started with finding fuzzy effective value then constructing fuzzy if-then rules resulting in fuzzy system construction using Fuzzy Inference Engine following formula mentioned in cited paper [23]. Finally, using heuristic median filter where they excluded 0 and 255 valued pixels and doing average of the rest in a mask.

III. EXISTING FILTERS

There are several existing filters available for noise reduction. Most common of them are the Mean filter and Median filter.

A. Standard Average Filter (SAF)

Mean filter is commonly known as average filter, and it is one of the most basic filters. For Mean filter, we define a window of size N×N, where N is an odd number. Usually the filter size is taken as 3×3, 5×5, 7×7, 9×9 etc. the most common window size for mean filter is 3×3. We add all the pixel values (intensity values of pixels) of the window, which gives us the sum and then divide the sum with total number of pixels present in the window. By this we get an average. Then we replace the center pixel value with the average value. In average filter, we always target the center pixel and modify it. In a 3×3 window our targeted pixel is (2,2) and in a 5×5 window our targeted pixel is (3,3). If the center pixel has a very different value than its neighboring pixels, that value is replaced by the average value. So, there is no out layer value in a neighborhood.

$$\frac{1}{25} \times \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & \mathbf{1} & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 \\ \hline \end{array}$$

Here we designed a 5×5 window. So, total number of pixel is 25 and we need to divide the sum of intensity of these 25 pixels by 25 to get the average. Then we replace the value of the center pixel with the average value. Here, the weight of all pixels is same. In weighted average filter, different pixels might have different weights assigned to them based on their position or users' choice. This produces different average. In that case, all the pixel weights are not uniform. We sum up all the weights and then multiply the pixel value with its assigned weight and sum those pixel values. Then divide the sum with the sum of the weights.

This method gives us best result when the neighboring pixel values are very close. But this method is also not free from the common draw backs of all the other filters of this type. It blurs the edges of the original image, as it does not have any edge preservation method. So, we need to sacrifice the edge information to get a smooth and less noisy image.

B. Standard Median Filter (SMF)

Median filter is another most common basic filter. For Median filter, we also need to define a window of size N×N, where N is an odd number. Usually the filter size is taken as 3×3, 5×5, 7×7, 9×9 etc. The most common window size for median filter is 3×3 too. We sort all the pixel values (intensity values of pixels) of the window and then take the median value from that window which basically is the middle pixel of the sorted value. For 3×3 we take the value of 5th position and for 5×5 we take the intensity value of 13th position. Here we also target the center pixel. We replace the center pixel value with the median value. In a 3×3 window our targeted pixel is (2,2) and in a 5×5 window our targeted pixel is (3,3).

$$f(x,y) = median[\sum_{i=-2}^{2} \times \sum_{j=-2}^{2} window(x-i, y-j)] \quad (2)$$

Here we defined the function for a 5×5 window size. Median filter works best when there is one or two out layer values in the window. It preserves the edge information better than the average filter. But it also makes the image blur when there are many sharp edges in the image.

C. Histogram based Fuzzy Filter (HFF)

Histogram based fuzzy filter (HFF) is an image filtering technique which is used for image filtering to reduce the noise in a noisy image. In this method, we estimate the histogram of the original image from the histogram of the noisy image. This method runs a window of 3×3 through the whole image and checks the pixel of the window to estimate the original histogram of the image. This estimated histogram of the original image is used to construct a fuzzy set. We get a fuzzy set for each window. These fuzzy sets and estimated histograms are used in filtering process. Five variables are used, i.e., very dark, dark, medium, bright, very bright, which are based on the intensity values.

$$\left.\begin{array}{r} vdk = \left[a_{vdk}\ b_{vdk}\ c_{vdk}\ d_{vdk}\right] \\ dk = \left[a_{dk}\ b_{dk}\ c_{dk}\ d_{dk}\right] \\ md = \left[a_{md}\ b_{md}\ c_{md}\ d_{md}\right] \\ br = \left[a_{br}\ b_{br}\ c_{br}\ d_{br}\right] \\ vbr = \left[a_{vbr}\ b_{vbr}\ c_{vbr}\ d_{vbr}\right] \end{array}\right\}$$

This method contains a major drawback which is to identify significant peaks in the image.

D. Neural Fuzzy Filter (NFF)

Neural fuzzy filter is mainly used for removing impulse noise from corrupted images. This de-noising is done using two main process, which are (a) fuzzy number construction process and (b) neural fuzzy filtering process. Information set called image knowledge base is also required for this process. Image knowledge base is constructed from the noisy image information before applying neural fuzzy filtering process.

For image with high percentage of impulse noise this method gives us a significant result. But when the noise is low, it cannot outperform the very basic image filtering techniques. Also for clustered noisy pixels this method fails to provide significant output.

IV. METHODOLOGY

In this section, we present our proposed filtering technique for image de-noising using fuzzy logic. The proposed method assigns fuzzy membership value to the inputs derived from the image. Using fuzzy if-then-else rules we calculated the fuzzy membership value of noisy pixel and the using that value for two different calculations which include the average filter and median filter noise membership value, we decide which one of this two filter will be more appropriate for the current window pixel value. Then we replace targeted pixel value with either of the two values generated from these two filters.

A. Dataset

We have used MATLAB's standard grayscale images (cameraman.tif, pout.tif) and some other kodak's database for grayscale images.

B. Algorithm

a) We design a Fuzzy Inference System (fuzzySystem.fis file)
b) Select the pixel with in a musk of N×N, where N is an odd number and greater than or equal 3. [N = 3,5,7,…..]
c) We calculate the difference between targeted pixel intensity with maximum intensity, minimum intensity and median intensity of the window
d) We pass the maximum difference, minimum difference and average difference as input and get output_1 which is a fuzzy value
e) Then we pass the maximum difference, minimum difference and median difference as input and get output_2 which is a fuzzy value
f) Compare the two outputs and check which one gives the lower value and use the best fitted filter (average value or median value) for the current window
g) For simplicity and reducing the time complexity replace the boundary pixel intensity with the median gray level value of a musk of N×N, where N is an odd number and greater than or equal 3. [N = 3,5, 7,…..]
h) Repeat steps 3-7 until all the pixels are visited

C. Designing Fuzzy Inference System

Here we designed a fuzzy inference system with three input variables each having 5 membership value [Very Small, Small, Medium, Large, Very Large] and one output variable having five membership value [Very Low, Low, Medium, High, Very High]. The FIS, fuzzy inputs and outputs are given in figure [1, 2, 3].

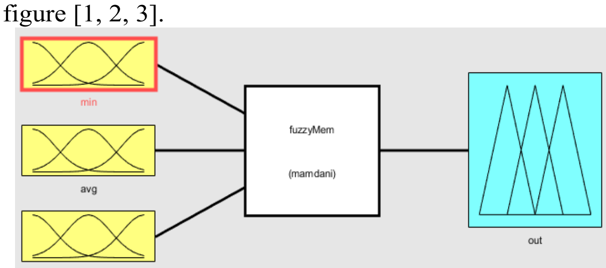

Figure 1. Adaptive FIS

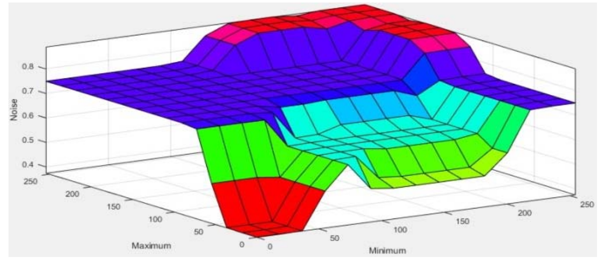

Figure 2. Surface Map of the FIS

a) Fuzzy Input (Minimum / Maximum)

We used Gaussian membership function for both minimum and maximum input values. The function is given below:

$$f(x;\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{3}$$

We used a modified range for our five fuzzy input values (very small, small, medium, large and very large). For distance value between 0 and 20 we considered the distance to be very small. For distance value between 20 and 50 we considered the distance to be small. For distance value between 20 and 75 we considered the distance to be medium. For distance value between 20 and 150 we considered the distance to be large. For distance value between 20 and 255 we considered the distance to be very large. Using Zadeh Max-Min we took the value with highest corresponding membership value.

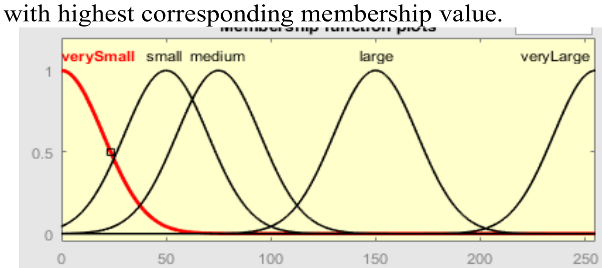

Figure 3. Fuzzy inputs (minimum/maximum)

b) Fuzzy Input (Average / Median)

We used Gaussian membership function also for both average and median input values. The function is given below:

$$f(x;\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{4}$$

We used a modified range for our five fuzzy input values (very small, small, medium, large and very large). For distance value between 0 and 15 we considered the distance to be very small. For distance value between 15 and 30 we considered the distance to be small. For distance value between 15 and 55 we considered the distance to be medium. For distance value between 15 and 150 we considered the distance to be large. For distance value between 15 and 255 we considered the distance to be very large. Using Zadeh Max-Min we took the value with highest corresponding membership value.

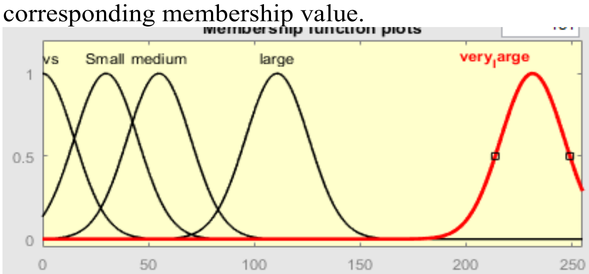

Figure 4. Fuzzy inputs (average/median)

c) *Fuzzy Output (Noise)*

We used Triangular membership function also for both average and median input values. The function is given below:

$$f(x;a,b,c) = \begin{Bmatrix} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{Bmatrix} \quad (5)$$

We used a modified range for our five fuzzy output values (very small, small, medium, large and very large).

$$very\ low = 0 \le 0.25$$
$$low = 0 \le 0.25 \le 0.5$$
$$medium = 0.25 \le 0.5 \le 0.75$$
$$high = 0.5 \le 0.75 \le 1$$
$$very\ high = 0.75 \le 1$$

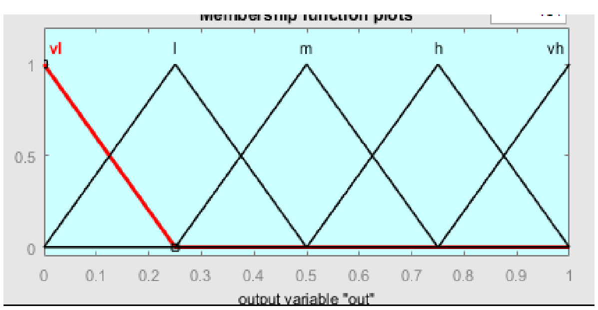

Figure 5. Fuzzy output (noise)

D. *Fuzzy if-then rules*

Using the generated distances from the pixels in the current window, we calculate the two values later to be compared to decide which filtering technique is suitable for that window pixel. Here all our three input fuzzy variables have 5 membership values. We control the fuzziness of the noise by 95 if-then rules. Some of the rules are defined below:

a) If minimum distance is Very Small and average/median distance is very Small and maximum distance is Very Small, then noise is Very Low
b) If minimum distance is Small and average/median distance is very Small and maximum distance is Small, then noise is Very Low.
c) If minimum distance is Small and average/median distance is Small and maximum distance is Large, then noise is Low.
d) If minimum distance is Medium and average/median distance is Small and maximum distance is Medium, then noise is Low.
e) If minimum distance is Large and average/median distance is Large and maximum distance is Very Small, then noise is Medium.
f) If minimum distance is Very Large and average/median distance is Small and maximum distance is Small, then noise is Medium.
g) If minimum distance is Large and average/median distance is Large and maximum distance is Large, then noise is High.
h) If minimum distance is Very Large distance and average/median is Large and maximum distance is Medium, then noise is High.
i) If minimum distance is Very Large and average/ median distance is Small and maximum distance is Large, then noise is Very High.
j) If minimum distance is Very Large and average/median distance is Very Large and maximum distance is Very Large, then noise is Very High

V. FINDINGS

Our proposed adaptive fuzzy rule-based filtering method is very much effective in impulse noise removing, also in very high rate of noise. The effectiveness has been verified from MATLAB implementation and analysis of result. We run our algorithm on Cameraman and Pout standard grayscale images. Our algorithm is proved better than existing algorithms. We analyze the effectiveness by three parameters such as:

i. Pick Signal Noise Ratio (PSNR)
ii. Mean Square Error (MSE)
iii. Visual Comparison of Final Images

We have calculated the MSE as below:

$$MSE = \sqrt[2]{\frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\, f\,(x,y) - i\,(x,y)]^2} \quad (6)$$

Here, f (x, y) is the final output image filtered from our proposed algorithm, i (x, y) is the original noise free image and M, N are the height and width of the grayscale image.

TABLE I. 75% SALT-AND-PEPPER NOISE MIXED CAMERAMAN.TIF IMAGE FILTERING (MSE COMPARISON)

Filters	cameraman.tif
Our Proposed Filter	14.8352
NFF	16.0248
SMF	16.5238
SAF	16.0248

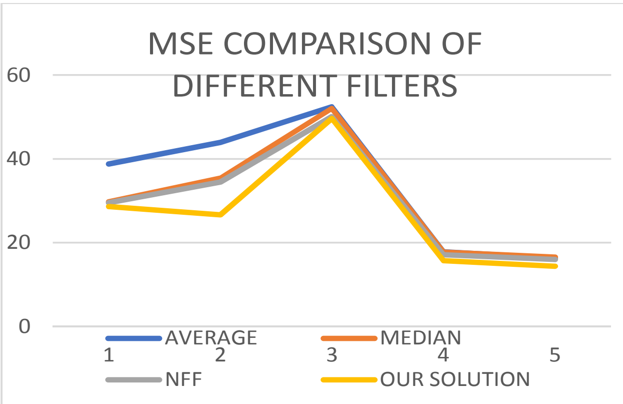

Figure 6. MSE comparison of different type of filters including our filter.

$$PSNR = 20 \times log_{10} \left(\frac{255}{MSE}\right) \quad (7)$$

TABLE II. 70% SALT-AND-PEPPER NOISE MIXED POUT.TIF IMAGE FILTERING (PSNR COMPARISON)

Filters	pout.tif
Our Proposed Filter	7.9529
HFF	7.3812
SMF	7.3386
SAF	7.3292

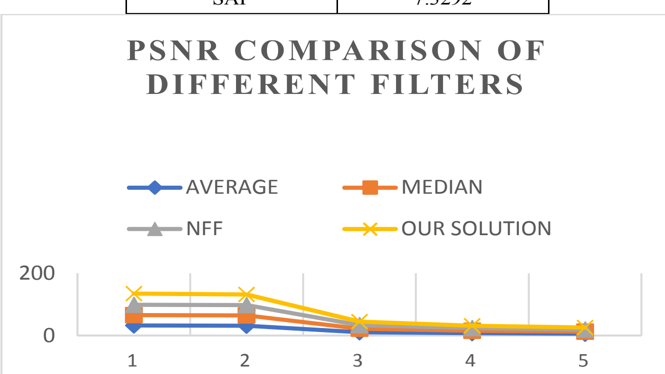

Figure 7. PSNR comparison of different type of filters including our filter.

VI. CONCLUSION

From the above findings and in figure 4, 5, we have proved our proposed algorithm works fine with impulse noise not only in low noise rate but also in very high noise rate. This is the main advantage our algorithm. All other fuzzy based filtering techniques also give very fine de-noised image but they fail when the noise ratio is very high. Most of the algorithms do not preserve the minor details. In our proposed algorithm, we have considered these features. Our proposed method works fine at very high noise and preserve minor details.

VII. SOFTWARE USED

While working on the project we had to use some software for our work. The main software that is used is MATLAB, specially the Fuzzy and Image Processing Toolbox. Other software includes Microsoft Excel 2016, notepad++.

REFERENCES

[1] Mrs. C. Mythili, V. Kavitha. Efficient Technique for Color Image Noise Reduction. *The Research Bulletin of Jordan ACM*. 2011, Vol. II (III). 41-44.

[2] M. Emre Celebi, Hassan A. Kingravi, Y. Alp Aslandog. A Nonlinear vector filtering for impulsive noise removal from color images. *Journal of Electronic Imaging*. 2007, 1-21.Vol.11, No.12.

[3] V.R. Vijay Kumar, S. Manikandan, D. Ebenezer, P.T. Vanathi and P. Kanagasabapathy.2007.High Density Impulse Noise Removal in Color Images Using Median Controlled Adaptive Recursive Weighted Median Filter. *IAENG International Journal of Computer Science* 1-10.

[4] Dimitri Van De Ville, Member, IEEE, Mike Nachtegael, Dietrich Van der Weken, Etienne E. Kerre. 2003. Noise Reduction by Fuzzy Image Filtering. *IEEE Transactions on Fuzzy systems*, Vol. 11, No. 4, 429-436

[5] Tukey JW (1971) *Exploratory data analysis*. Addison- Wesley.

[6] Astola J, Kuosmanen P (1997) Fundamentals of nonlinear digital filtering. CRC, Boca Raton .

[7] Pitas I, Venetsanopoulos A (1990) Nonlinear digital filters: principles and application. Kluwer, Norwell.

[8] M. Mozammel Hoque Chowdhury. A Robust De-Noising Model for Image Enhancement with Adaptive Median Filtering. *American Journal of Modeling and Optimization*, Vol. 2, No. 3, pp. 69-72, United States, 2014.

[9] S. J. Ko and Y.H Lee. Center weighted median filters and their applications to image enhancement. *IEEE Transactions on Circuits and Systems*, Vol. 38, pp. 984-993, 1991.

[10] T. C. Chen, K. K. Ma and L. H. Chen. Tri-state median filter for image denoising. *IEEE Transactions on Image Processing*, Vol. 8, No. 12, pp. 1834-1838, Dec. 1999.

[11] T. Loupas, W. N. McDicken, and P. L. Allan. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. IEEE Transactions on Circuits and Systems, Vol. 36, Jan. 1989.

[12] H. L. Eng and K. K. Ma. Noise adaptive soft-switching median filter. *IEEE Transactions on Image Processing*, Vol. 10, No. 2, pp. 242 –251, Feb. 2001.

[13] W. Ping, L. Junli, L. Dongming, and C. Gang. A fast and reliable switching median filter for highly corrupted images by impulse noise. *IEEE International Symposium on Circuits and Systems*, pp. 3427-3430, May 2007.

[14] F. Russo and G. Ramponi. A noise smoother using cascaded FIRE filters. *Proc. of 4th Intl. Conf. on Fuzzy Systems*, Vol.1, pp. 351- 358, 1995.

[15] F. Russo and G. Ramponi. 1996. A fuzzy filter for image corrupted by impulse noise. *IEEE Signal Processing Letter*, vol. 3, pp. 168 -170

[16] Chowdhury, M., Gao, J., & Islam, R. (2016, 07). Fuzzy logic based filtering for image de-noising. *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. doi:10.1109/fuzz-ieee.2016.7737990.

[17] Chowdhury, M. M., Islam, M. E., Begum, N., & Bhuiyan, M. A. (2007, 12). Digital image enhancement with fuzzy rule-based filtering. *2007 10th International Conference on Computer and Information Technology*. doi:10.1109/iccitechn.2007.4579400.

[18] M. Mozammel Hoque Chowdhury. Stereo Imaging for 3D Scene Reconstruction: A Novel Approach. ISBN: 978-3-659-45179-9, LAMBERT Academic Publishing, Germany, 2013.

[19] Hussain, A., Bhatti, S. M., & Jaffar, M. A. (2011). Fuzzy based Impulse Noise Reduction Method. *Multimedia Tools and Applications,60*(3), 551-571. doi:10.1007/s11042-011-0829-7.

[20]] Liu P, Li H (2004) Fuzzy techniques in image restoration research – a survey (invited paper*).International Journal of Computational Cognition* 2(2):131–149.

[21] Jampour, M., Ziari, M., Zadeh, R. E., & Ashourzadeh, M. (2010). Impulse noise detection and reduction using fuzzy logic and median heuristic filter. 2010 *International Conference on Networking and Information Technology*. doi:10.1109/icnit.2010.5508565.

[22] Pei-Yin Chen; Chih-Yuan Lien; Chang-Yan Tsai. An Effective Impulse Noise Detector of Switching Median Filter using Min-Max Working Window. *International Conference on Innovative Computing, Information and Control*. Volume I, PP: 234- 234. 2007

[23] Yuan, S.-Q.; Tan, Y.-H. Difference-type noise detector for adaptive median filter. Letters Electronics Volume 42, Issue 8, PP: 454 - 455. 2006.

[24] Preety D. Swami & Alok Jain (2012). Image denoising by supervised adaptive fusion of decomposed images restored using wave atom, curvelet and wavelet transform. *Springer-Verlag London Limited* 2012. doi: 10.1007/s11760-012-0343.

[25] Motta, G., Ordentlich, E., Ramirez, I., Seroussi, G., & Weinberger, M. J. (2011). The iDUDE Framework for Grayscale Image Denoising. *IEEE Transactions on Image Processing,* 20(1), 1-21. doi:10.1109/tip.2010.2053939

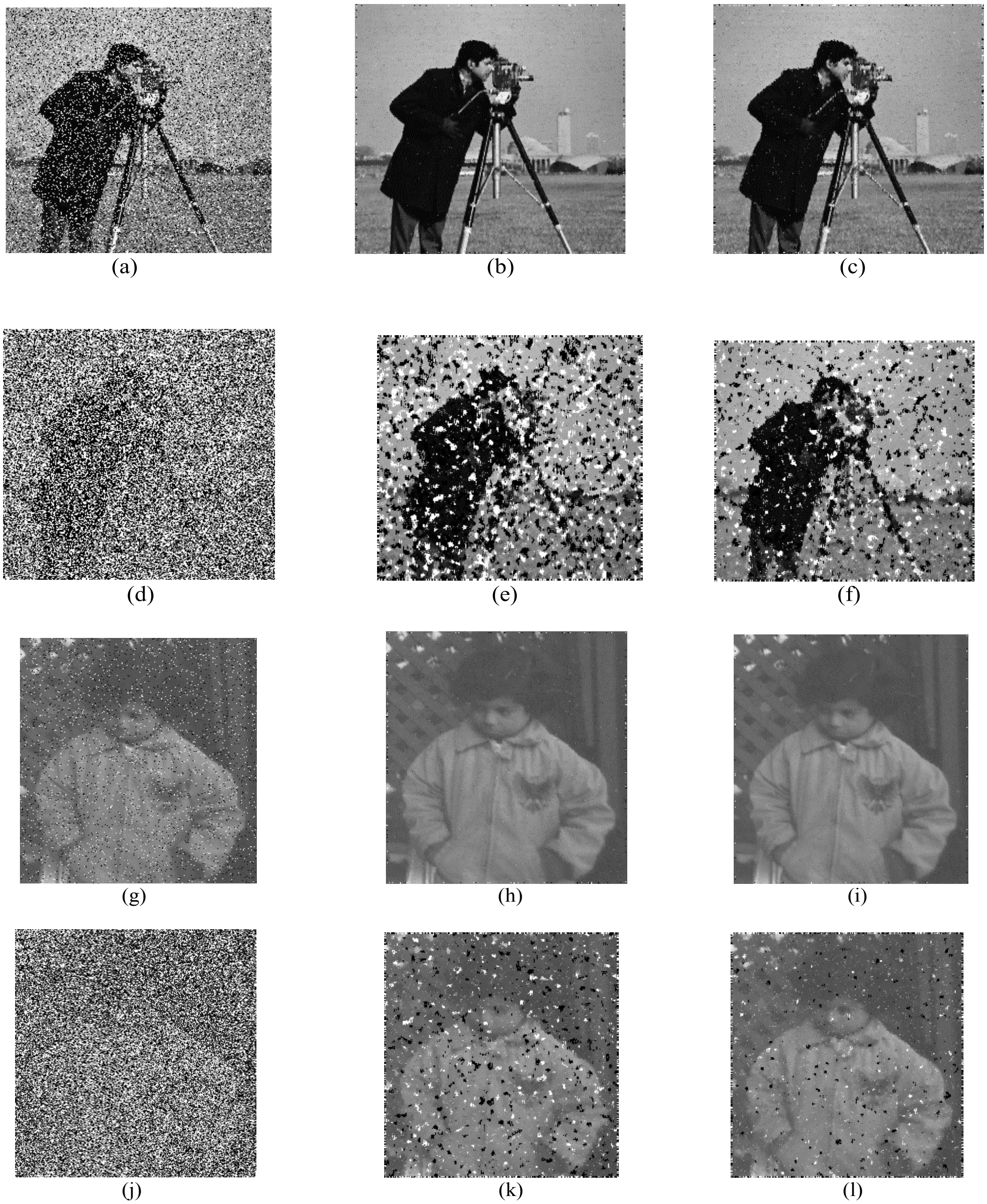

Figure 8. cameraman.tif image of (a) with 5% salt-and-pepper noise, (b) de-noised image of 5% noise by NFF, (c) de-noised image of 5% noise by our algorithm (d) with 75% salt-and-pepper noise, (e) de-noised image of 75% noise by NFF, (f) de-noised image of 75% noise by our algorithm. pout.tif image of (g) with 5% salt-and-pepper noise, (h) de-noised image of 5% noise by HFF, (i) de-noised image of 5% noise by our algorithm (j) with 70% salt-and-pepper noise, (k) de-noised image of 70% noise by HFF, (l) de-noised image of 75% noise by our algorithm.