ISSN:0975-9646

A New Fuzzy Gaussian Noise Removal Method for Gray-Scale Images

K.Ratna Babu[#1], Dr K.V.N.Sunitha[*2]

[#] *Associate professor, IT Department*
SIR CRR College of Engineering,Eluru,W.G.Dist
Andhra Pradesh,India

[*] *Professor & HOD CSE Dept.,*
G.Narayanamma Institute of Technology & Science
Shaikpet,Hyderabad
Andhra Pradesh,India

Abstract: A New Fuzzy Filter that adopts Fuzzy Logic is proposed in this paper which removes Gaussian Noise from the Corrupted Gray scale Images which is also good for Impulsive and multiplicative Noise. This Method Consists of Two Steps.1) Estimating the Noise 2) Smoothing according to the Noise Level. It uses Fuzzy concepts to decide whether a pixel in the Image is a Noisy one or not for conducting smoothing operation while preserving Image Detail. Many popular Algorithms consider Image Noise Level as an important quantity to adjust the Image Parameters for Image De-noising. i.e. they need an account for variations in image Noise levels. A stronger noise removal setting makes resultant image blurry. There are mainly two problems associated with these Algorithms: Edge Detection and Feature Preserving We Illustrate a new method of De-noising of Images which uses a correction term to de-noise the Images where above mentioned problems are minimized to an acceptable range and we obtained good results for different specified number of Iterations of the Algorithm. These Results illustrates that the proposed method can be used as an effective Noise removal method.

Keywords: Wiener filter, Mean Filter, Gaussian noise, Impulse noise, Multiplicative Noise, Correction term

I. INTRODUCTION

The Proposed System mainly aims at Gaussian Noise [5, 15] which is also good at removing other Noises like Impulsive [18, 19] and Multiplicative Noise [12]. Impulsive Noise consists of random occurrences of energy spikes having random amplitude and spectral content. Multiplicative noise is a type of signal-dependent noise where brighter areas of the images appear noisier. Gaussian noise [5] is a type of statistical noise in which the amplitude of the noise follows that of a Gaussian distribution.

When Images are Acquired and Transmitted over channels they are often corrupted by Impulse Noise [3,4] due to faulty communications and Noisy channels .i.e. Noise characterized by transient short-duration disturbances distributed essentially uniformly over the useful pass band of a transmission system. The Main aim of an Image De-noising Algorithm is to reduce the Noise Level, while keeping its fundamental structure like corners and edges [1]. Most of the Algorithms mainly deals with fat-tailed Noise like Impulsive Noise [18,19]and not specifically designed for Gaussian noise or do not produce convincing results when applied to handle this type of noise. Impulse noise [3,4] consists of very large positive or negative spikes of short duration. A positive spike has a value much larger than those of background signals and appears like a bright spot on the image. On the other hand, a negative spike has a value much smaller than those of background signals and appears like a dark spot on the image. They both are easily detected by the eyes and degrade the image quality.

Fat-tail distributed "impulsive" noise is sometimes called salt-and-pepper noise or spike noise. An image containing salt-and-pepper noise will have dark pixels in bright regions and bright pixels in dark regions. This type of noise can be caused by dead pixels, analog to digital converter errors, bit errors in transmission, etc.This can be eliminated in large part by using dark frame subtraction and by interpolating around dark/bright pixels.

Experimental results show the feasibility of the proposed approach.

II. LITERATURE REVIEW

There are several filters used for reduction of the noise. Some of the important types of filters are discussed below.

A. Median Filter

Images and image sequences are frequently corrupted by noise in the acquisition and transmission phases. This Problem can be defined as follows

$$T(x, y) = O(x, y) + N(x, y) \text{----------}\rightarrow \quad (1)$$

where T is the Noisy image, O is the original Image and N is the Noise.

The goal of de-noising is to remove the noise, both for aesthetic and compression reasons, while retaining as much as possible the important signal features. Very commonly, this is achieved by approaches such as Wiener filtering which is the optimal estimator (in the sense of mean squared error (MSE) for stationary Gaussian process [5,15]. Since natural images typically consist of smooth areas, textures, and edges, they are clearly not globally *s*tationary. Similarly, non-stationary in video may further be caused by inter-frame motion. However, image and video can be reasonably treated as being locally stationary and similar arguments can be made for motion-compensated video. These insights have motivated the design of adaptive wiener filter. However this filter suffers from annoying noise around edges, due to the assumption that all samples within a local window are from the same ensemble. This assumption is invalidated if there is a sharp edge [1] within the window, for example; in particular, the sample variance near an edge will be biased large because samples from two different ensembles are combined, and similarly the sample mean [16] will tend to smear. The main problem, then, is how to effectively estimate local statistics. More recently there has been considerable attention paid to wavelet-based de-noising because of its effectiveness and simplicity. Both wavelet shrinkage and wavelet Wiener methods have shown to be very effective in signal and image de-noising, although the latter Wiener approach is the one of interest in our context. It is well established that the wavelet transform is an effective decorrelator, and thus a reasonable approximation to the Karhuen-Loeve basis. Consequently a local wavelet Wiener filter should be more effective than its spatial counterpart; however the no stationary local second order statistics must still be estimated.

B. WFM Filter

The removal of heavy additive impulse noise [3,4,15] is done using the weighted fuzzy mea*n* (WFM) filter [7,8,9,10]. The WFM-filtered output signal is the mean value [16] of the corrupted signals in a sample matrix, and these signals are weighted by a membership grade of an associated fuzzy set stored in a knowledge base [1]. The knowledge base contains a number of fuzzy sets [20,21] decided by experts or derived from the histogram of a reference image. When noise probability exceeds 0.3, WFM gives very superior performance compared with conventional filters when evaluated by mean absolute error (MAE), mean square error (MSE) peak signal-to-noise-rate (PSNR) and subjective evaluation criteria. For dedicated hardware implementation, WFM is also much simpler than the conventional median filter [6, 13, 14, 17].

III PROPOSED FILTER

Fuzzy set theory [7, 8, 9] has been successfully applied to pattern recognition fields. It is suitable for dealing with problems containing high levels of uncertainty, to which class pattern recognition or image processing problems usually belong. Obviously, the recovery of heavily noise-corrupted images is a task with high uncertainty levels. The general idea behind the filter is to average a pixel using other pixel values from its neighborhood, but simultaneously to take care of important image structures such as edges. The main concern of the present filter is to distinguish between local variations due to noise and due to image structure. In order to accomplish this, for each pixel we derive a value that expresses the degree in which the derivative in a certain direction is small. Such a value is derived for each direction corresponding to the neighboring pixels of the processed pixel by a fuzzy rule [10,20,21]. The further construction of the filter is then based on the observation that a small fuzzy derivative [, 8, 9,] most likely is caused by noise, while a large fuzzy derivative most likely is caused by an edge in the image. Consequently, for each direction we will apply two fuzzy rules that take this observation into account (and thus distinguish between local variations due to noise and due to image structure), and that determine the contribution of the neighboring pixel values. The result of these rules (16 in total) is defuzzified and a "correction term" is obtained for the processed pixel value.

Two important features of the present filter are: first, the filter estimates a "derivative" in order to be less sensitive to local variations due to image structures such as edges; second, the membership functions are adapted accordingly to the noise level to perform smoothing operation.

Estimating derivatives and filtering are very complex as for filtering we need a good indication of the edges, while to find these edges we need filtering. To counter this problem, we start by looking for the edges. We try to provide a robust estimate by applying fuzzy rules.

PNW	PN	PNE
PW	P	PE
PSW	PS	PSE

Fig 1: Pixels around a central Pixel P

Consider the neighborhood of a pixel as displayed in Figure 1.A simple derivative at the central pixel position in the direction is defined as the difference between the pixel at and its neighbor in the direction. This derivative value is denoted by $\nabla_{dir}(x,y)$.

$$\nabla_{PS}(x,y) = f(x,y+1) - f(x,y) \dashrightarrow (2)$$

$$\nabla_{PSE}(x,y) = f(x+1,y+1) - f(x,y) \dashrightarrow (3)$$

Step 1: Noise Estimation

Consider a 3X3 neighborhood of a pixel (x, y). A simple derivative at the central pixel position (x, y) value in the direction D (D=NW, W, SW, S, SE, E, NE, N) is defined as the difference between the value of the central pixel (x, y) and its neighbor pixel in that direction D. This derivative value is denoted by Ñd (x, y).

For example, ∇**PN (x, y) = f(x, y-1)-f(x, y) ----->(4)**

The principle of the fuzzy derivative [7, 9, 10] is based on the following observation. Consider an edge passing through the neighborhood of a pixel (x, y) in the direction SW, NE. The derivative value Ñnw (x, y) will be large, but also derivative values of neighboring pixels perpendicular to the edge's direction can expected to be large. For example, in NW - direction we can calculate the derivative values Ñnw (x, y), Ñnw (x-1, y+1) and Ñnw (x + 1, y- 1).. The idea is to cancel out the effect of one derivative value which turns out to be high due to noise. Therefore, if two out of three derivative values are small, it is safe to assume that no edge is present in the considered direction. This observation will be taken into account when we formulate the fuzzy rule [20,21] to calculate the fuzzy derivative.

NW: (if Ñnw(x,y) is small and Ñnw(x-1,y+1) is small) or (if Ñnw(x,y) is small and Ñnw(x+1,y-1) small) or (ifÑnw(x-1,y+1) is small or Ñnw(x+1,y-1) is small then Ñnwf(x,y) is small.

Step 2: Smoothing

The fuzzy filtering [7, 8, 9, 10] of the image reduces noise components of pixels by means of correction of pixel values. We describe this in terms of a correction term D. To compute correction term D for the processed pixel value, we use a pair of fuzzy rules for each direction. The idea behind the rules is if no edge is assumed to be present in a certain direction the derivative value in that direction can and will be used to calculate the correction term. Edge assumption part can be realized by using the fuzzy derivative value, for filtering part we will have to distinguish between positive and negative values. Fuzzy rules to calculate positive and negative values are

1. NW = lnw(+) :if Ñnwf(x,y) is small and Ñnw(x,y) is positive then D positive

2. NW = lnw(-) :if Ñnwf(x,y) is small and Ñnw(x,y) is negative then D negative.

We are interested in obtaining a correction term D, which can be added to the pixel value of location (x, y). Therefore, the truth ness of the rules ld(+) and ld(-) are aggregated and rescaled. So, each direction contributes to the correction term D.

The fuzzy rules for smoothing are as follows:

1. If a pixel is darker than neighboring pixels then make it brighter

2. If a pixel is brighter than neighboring pixels then make it darker.

3. Else leave it unchanged.

		(PN)N √	
	PNW √	PN	PNE √
(PW)W √	PW	P	PE
	PSW √	PS	PSE

Fig 2 : Estimating the Noise

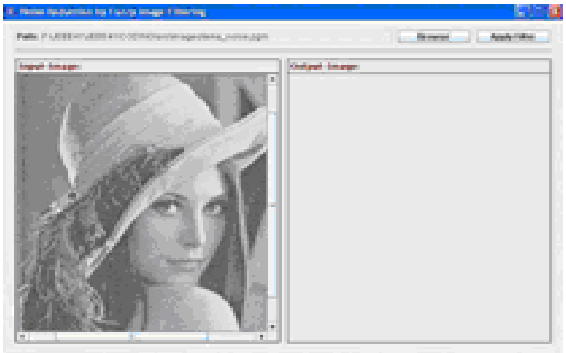

Fig 1 : Displaying the image window

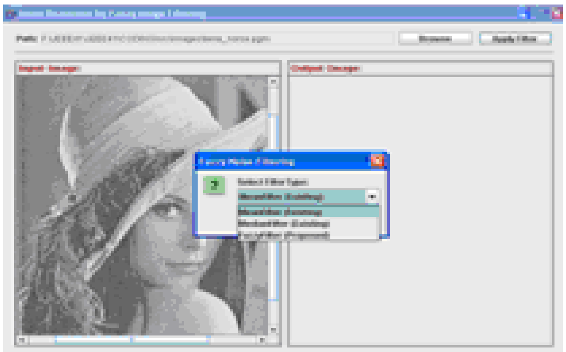

Fig 2: Selecting the filter

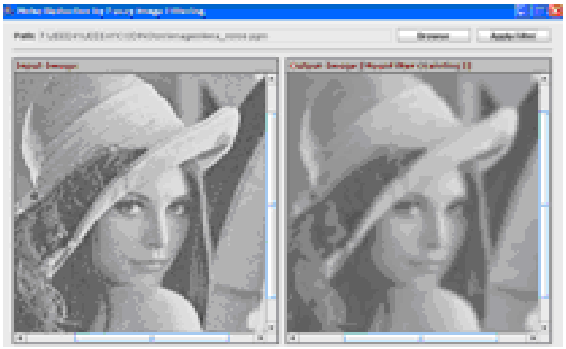

Fig 3: Filtered image with Mean Filter

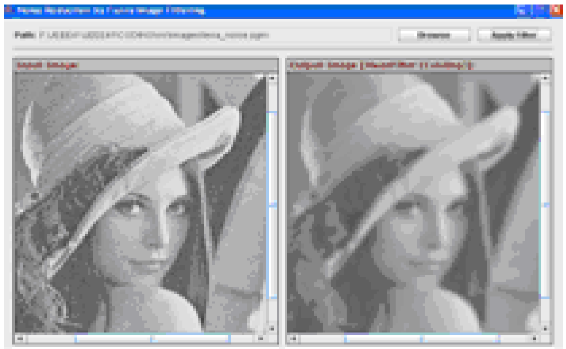

Fig 4: Filtered image with Median Filter

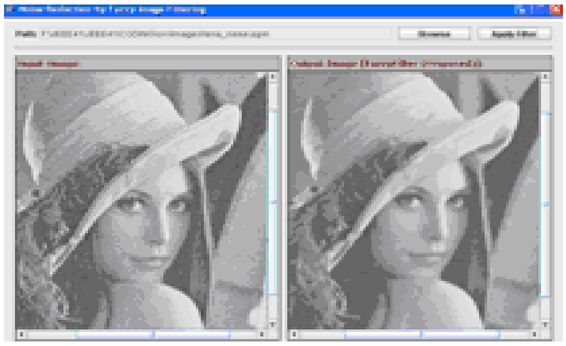

Fig 5: Filtered with proposed system

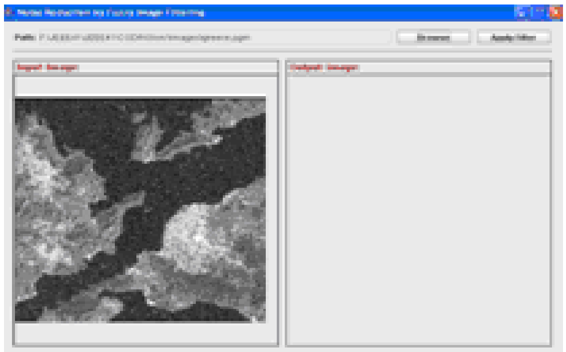

Fig 6: Displaying the image window

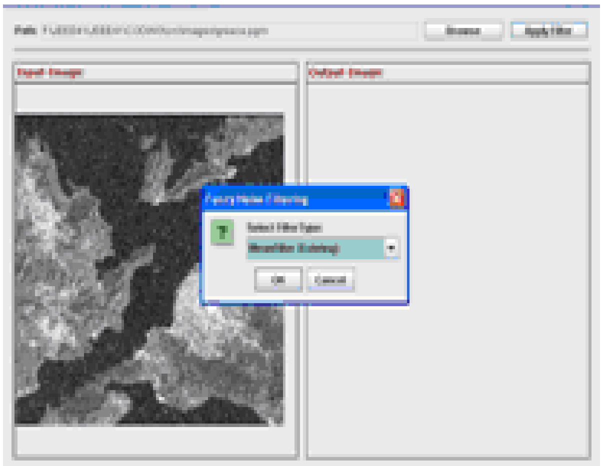

Fig 7: Filtering image with Mean Filter

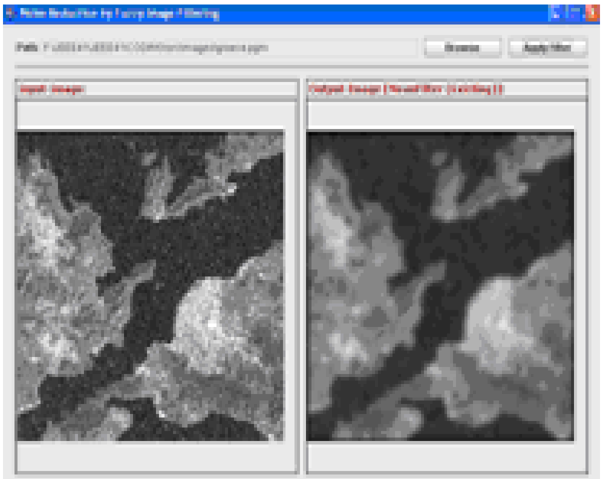

Fig 8: Image Filtered with Mean Filter

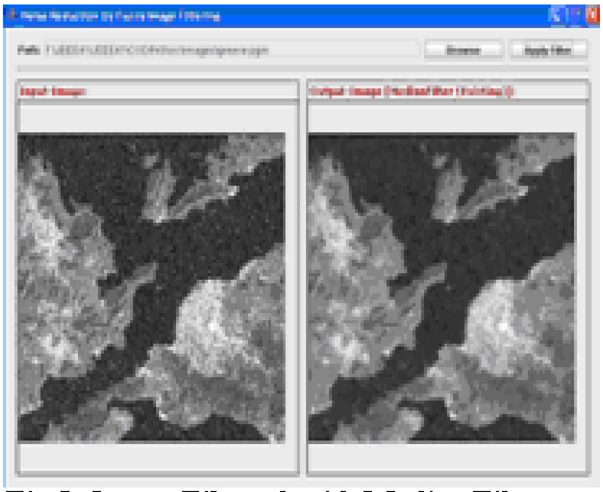

Fig 9: Image Filtered with Median Filter

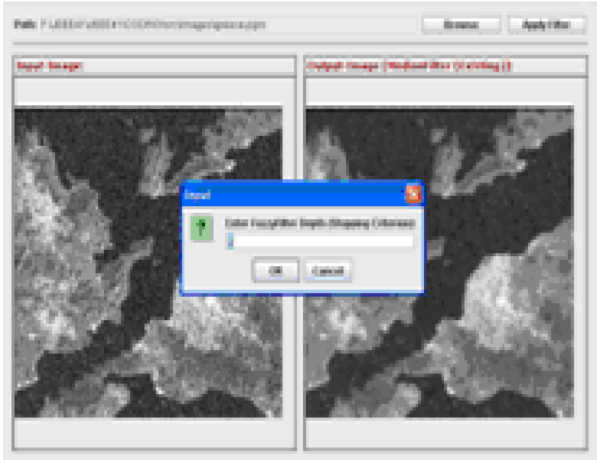

Fig10: Stopping Criteria for Proposed System

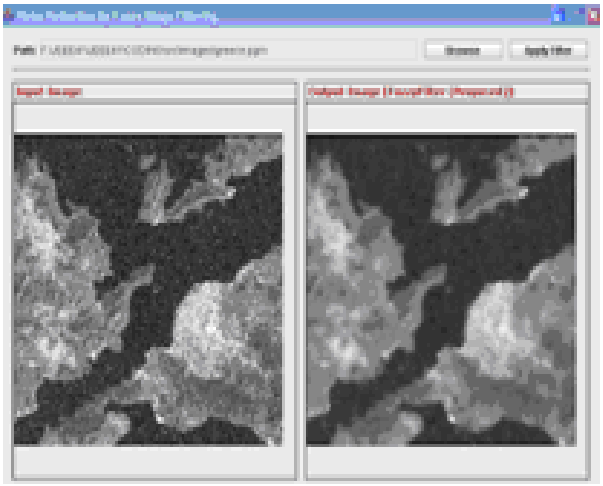

Fig11: Image obtained by applying Fuzzy Filter

For Example, to compute the derivative for the pixel P, we use PNW, PNE,(PN)N,PSW,(PW)W pixels as shown in the Following Table.

Pixel	set
PNW	{(-1,1),(0,0),(1,-1)}
PNE	{(-1,01),(0,0),(1,1)}
PSW	{(1,1),(0,0),(-1,-1)}
(PW)W	{(0,1),(0,0),(0,-1)}
(PN)N	{(-1,0),(0,0),(1,0)}

Fig 3: Pixel Positions

The Following is the Diagram which illustrates the total process involved in the Algorithm.

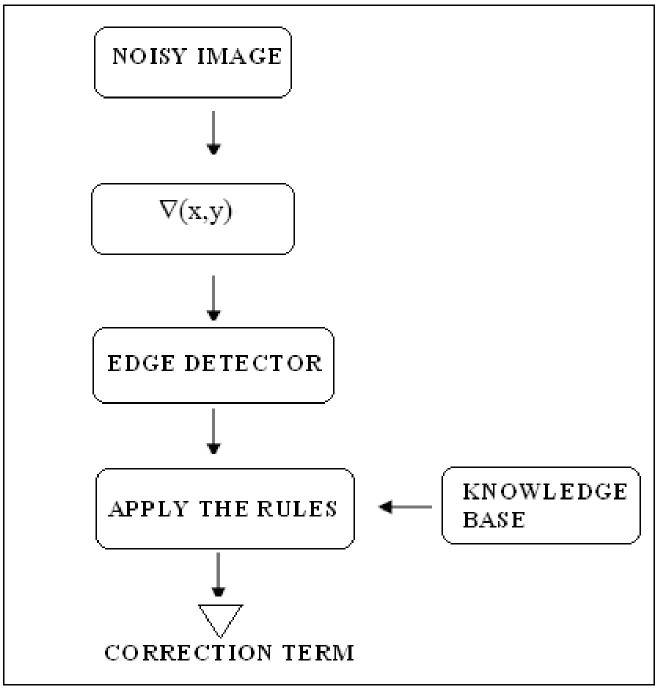

Fig 4: Synoptic Notation of proposed Method

IV. IMAGE TYPE

There are so many types of image files now days. Before applying our method to the images we must have to be particular with the types of image we use and the properties of that type of image also. Here in this project we use .pgm extension images. The name "PGM" is an acronym derived from "Portable Gray Map."

The PGM format is a lowest common denominator grayscale file format. It is designed to be extremely easy to learn and write programs for. (It's so simple that most people will simply reverse engineer it because it's easier than reading this specification). A PGM image represents a grayscale graphic image. There are many pseudo-PGM formats in use where everything is as specified herein except for the meaning of individual pixel values. For most purposes, a PGM image can just be thought of an array of arbitrary integers, and all the programs in the world that think they're processing a grayscale image can easily be tricked into processing something else.

A row of an image is horizontal. A column is vertical. The pixels in the image are square and contiguous. Each gray value is a number proportional to the intensity of the pixel, adjusted by the ITU-R Recommendation BT.709 gamma transfer function. (That transfer function specifies a gamma number of 2.2 and has a linear section for small intensities). A value of zero is therefore black. A value of Maxval represents CIE D65 white and the most intense value in the image and any other image to which the image might be compared.

A common variation on the PGM format is to have the gray value be "linear," i.e. as specified above except without the gamma adjustment. Pnmgamma takes such a PGM variant as input and produces a true PGM as output.

In the transparency mask variation on PGM, the value represents opaqueness. It is proportional to the fraction of intensity of a pixel that would show in place of an underlying pixel. So what normally means white represents total opaqueness and what normally means black represents total transparency. In between, you would compute the intensity of a composite pixel of an "under" and "over" pixel as under * (1-(alpha/alpha_maxval)) + over * (alpha/alpha_maxval). Note that there is no gamma transfer function in the transparency mask.

Each PGM image consists of the following:

1. A "magic number" for identifying the file type. A pgm image's magic number is the two characters "P5".
2. Whitespace (blanks, TABs, CRs, LFs).
3. A width, formatted as ASCII characters in decimal.
4. Whitespace.
5. A height, again in ASCII decimal.
6. Whitespace.
7. The maximum gray value (Maxval), again in ASCII decimal. Must be less than 65536, and more than zero.
8. A single whitespace character (usually a newline).
9. A raster of Height rows, in order from top to bottom. Each row consists of Width gray values, in order from left to right. Each gray value is a number from 0 through Maxval, with 0 being black and Maxval being white. Each gray value is represented in pure binary by either 1 or 2 bytes. If the Maxval is less than 256, it is 1

byte. Otherwise, it is 2 bytes. The most significant byte is first.

A row of an image is horizontal. A column is vertical. The pixels in the image are square and contiguous.

V. EXPERIMENTAL RESULTS

When this algorithm is compared with traditional effective algorithms it yields better results in terms of Visual Quality, high speed etc. To compare performance of the above approaches, these experiments have been repeated several times and the results are evaluated using MSE and SNR criteria. The results are shown in last page fig1-10.

VI. CONCLUSION

In this Paper we have introduced a new algorithm for de-noising images using correction Term Method. The implementation of our fuzzy Filter [7, 8, 9, 10] in the image restoration process was discussed. All the techniques and operations provide an efficient working and the output image is enhanced according to the user's requirements. This Framework is useful to enhance images of PGM format. We can extend it to support other formats. This system can be extended into other fields such as archeology, Medical Image processing, Remote sensing etc.

The existing system available for noise reduction deals with fat-tailed noise like impulse noise [3,4] . Median filter [6,14,17] and Low-pass filters are in job at present. Median Filters [6,14] mainly concentrate only on Impulsive Noise.Gausian Noise is not specifically concentrated. It does not distinguish local variation due to noise and due to image structure

Proposed Filter can clean an image completely of noise without making it blurry (Although that can produce an image that is very plasticy if depth level of the Tool is increased.).It presents a new technique for filtering narrow-tailed and medium narrow-tailed noise by a fuzzy filter [7, 8, 10].

REFERENCES

[1] X. Zheng and Q. Gao, "Image Noise Removal Using Perceptual Edge Features", GVIP Special Issue on Denoising, pp. 15-20, 2007.

[2] Ce Liu William T. Freeman, Richard Szeliski Sing Bing Kang , Noise Estimation from a Single Image, IEEE Conference on Computer Vision and Pattern Recognition, 2006

[3] Stefan Schulte, Mike Nachtegael, Valerie de Witte, Dietrich van der Weken and Etienne E.Kerre, "A fuzzy Impulse noise detection and reduction method", IEEE Trans. on Image Process, Vol 15,no 5,May 2006.

[4] M.Emin Yuksel, "A Hybrid Neuro-Fuzzy filter for Edge Preserving Restoration of Images Corrupted by Impulse Noise", IEEE Trans. on Image rocess,Vol 15,no 6, April 2006.

[5] F. Russo, "A Method Based on Piecewise Linear Models for Accurate Restoration of Images Corrupted by Gaussian Noise", IEEE Trans. on Instrumentation and Measurement,Vol. 55, No. 6, 2006

[6] P-E. Ng and K-K. Ma, "A Switching Median Filter with Boundary Discriminative Noise detection for Extremely Corrupted Images", IEEE Trans. on Image Processing, Vol. 15, No. 6, June 2006.

.[7] I. Nedeljkovic, « Image Classification Based On Fuzzy Logic». The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34, 2004.

[8] H.S. Kam, M. Hanmandlu, and W.H. Tan, "An adaptive fuzzy filter system for smoothing noisy images", Proc. of Intl. Conf. on Convergent Technologies for Asia-Pacific Region, Vol.4, pp. 1614 – 1617, 2003.

[9] Ville D., Nachtegael M., Weken D., Kerre E., Philips W., and Lemahieu I., **"***Noise Reduction by Fuzzy Image Filtering***"**, IEEE Transactions on Fuzzy system, Vol. II, No.4, August, 2003.

[10] D. Van De Ville, M. Nachtegael, D. Van der Weken, E. Kerre, W.Philips, and I. Lemahieu, Noise Reduction by Fuzzy Image Filtering, IEEE transactions on fuzzy systems, vol. 11, No. 4, August 2003.

[11] R. C. Gonzalez & R. E. Woods, « Digital Image Processing», Second Edition, Prentice Hall, 2002.

[12] Y. Hawwar and A. Reza, "Spatially adaptive multiplicative noise image denoising technique," *IEEETransactions on Image Processing*, vol. 11, pp. 1397-1404, Dec. 2002.

[13] F. Farbiz and M. B. Menhaj, Fuzzy Techniques in Image Processing.New York: Springer-Verlag, 2000, vol. 52, Studies in Fuzziness and Soft Computing, ch. A fuzzy logic control based approach for image filtering, pp. 194–221.

[14] J. Ishihara, M. Meguro and N. Hamada, "Adaptive weighted median filter utilizing impulsive noise detection," in Applications of Digital Image Processing '99, Proc SPIE 3808, 406-414 (1999)

[15] Y. Xu and E. M.–K. Lai, "Restoration of images contaminated by mixed Gaussian and impulse noise using a recursive minimum–maximum method," IEE
August 1998, Vol. 145 Issue 4 (p.264-270)

[16] C.-S. Lee, Y.-H. Kuo, and P.-T. Yu, Weighted fuzzy mean filters for image processing, Fuzzy Sets Syst., no. 89, pp. 157–180, 1997.

[17] Chen, C.T.[Chun-Te], Chen, L.G.[Liang-Gee], A Self-Adjusting Weighted Median Filter for Removing Impulse Noise in Images, ICIP96(16A9). BibRef 9600 Proceedings International Conference on Image Processing, or ICIP. Sponsored by IEEE Signal Processing

[18] Wang, Z.[Zhou], Zhang, D., Restoration of Impulse Noise-Corrupted Images Using Long-Range Correlation, IEEE Signal Processing Letters , (5), No. 1, January1998, pp. 4-7. 9802 [14] Russo, F., Ramponi, G., A Fuzzy Filter for Images Corrupted By Impulse Noise,IEEE Signal Processing Letters, (3), No. 6, June 1996, pp. 168-170. 9607

[19] Abreu, E., Lightstone, M., Mitra, S.K., Arakawa, K., A New Efficient Approach for the Removal of Impulse Noise from Highly Corrupted Images, IEEE Trans. Image Processing, (5), No. 6, June 1996, pp. 1012-1025. 9607
[20]] K. Arakawa, Median filter based on fuzzy rules and its application to image restoration, Fuzzy Sets Syst., pp. 3–13, 1996

[21] F. Russo and G. Ramponi, A fuzzy operator for the enhancement of blurred and noisy images, IEEE Trans. Image Processing, vol. 4, pp.1169–1174, August 1995.