ISSN 2229-5518

Enhanced Image Filtering Method

Saksham Sinha, Hifzur Rahman, Mahesh Thorat

Abstract— We have developed a new fuzzy filter for the noise reduction in images having the additive noise. Our proposed filter works in two stages. The first stage of our filter computes the fuzzy derivatives for eight different directions. And the second stage of our filter utilizes these fuzzy derivatives to do fuzzy smoothing by taking weightage of the contributions of neighboring pixels. Both of the stages uses fuzzy rules which takes help of membership functions. This filter could be used many times iteratively to give more better results. While filtering, the membership functions are altered according to the amount of noise present in an image after every iteration. This is done by using the homogeneity in the image.

Index Terms— Additive noise, fuzzy filter, noise detection and removal, homogeneity.

1 Introduction

The fuzzy techniques application in image processing has been a good research field [1]. In many domains fuzzy techniques are already in use like for example filtering, interpolation [2], and morphing [3], etc. It also has many applications in industrial and medical images and their processing [5], [6].

Here, we are focusing on techniques of image filtration using fuzzy technique. Although many fuzzy filters for noise removal has been implemented like for example the FIRE-filter [7], [8], [9], the weighted fuzzy mean filter [10], [11], and the iterative fuzzy control based filter [12]. Here most the implemented fuzzy filters works on fat-tailed noises like impulse noise. These filters are capable to outperform filters like median filter but yet these are not particularly designed for Gaussian noises.

Hence, in this paper, we are presenting a technique for filtering noises like narrow tailed and medium narrow tailed noises by the use of fuzzy filter. We have introduced two features, first one is that the filter calculates a "fuzzy derivative" so that it can be less delicate to local variations which happens due to structures such as an edge, etc. The second feature is that the membership functions that we are using are changed according to the amount of noise present to apply "fuzzy smoothing".

The development of our fuzzy filter is explained later. For every pixel in the image being processed, the first part calculates the fuzzy derivate and then a set of 16 fuzzy rules are applied to find a correction term. These can use fuzzy derivative as input. After that Fuzzy sets are used to depict the properties small, positive and negative. We have fixed membership functions for positive and negative, but for small, the membership function is changed after every iteration. This scheme is also explained later in the paper. The results obtained are then compared to few already existing filters and conclusions were drawn.

2 Fuzzy Filter

The idea behind our filter is that to average a pixel's value using other pixel's values from its neighbors, and at the same time to take care of structures like edges. The main focus of our proposed filter is to differentiate between local variations because of noise and because of image structures like edge.

For doing this, for every pixel we find a value that shows the amount in which the derivative in a direction is small. That value is found for every direction corresponding to neighbor pixels of the pixel in focus by a fuzzy rule.

The later construction of the filter is based upon the observation which is that a small fuzzy derivative is very much likely to happen due to noise and a large fuzzy derivative is likely to happen due to an edge present in an image. Therefore, for every direction we apply two fuzzy rules that uses this remark, and which signifies the contribution of neighbor pixels.

The result of these 16 rules is defuzzified and a "correction term" is acquired for the pixel being processed.

2.1 Fuzzy Derivative Estimation

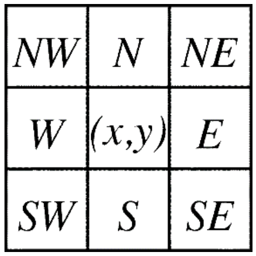
Finding derivatives and filtering are dependent on each other. For filtering we need to find edges and to detect edges we need good filtering.

We start by checking for edges in the images. We have tried to get a strong estimate by using fuzzy rules.

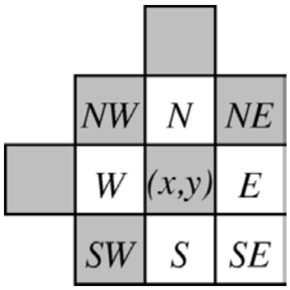
Taking the 3x3 neighbor of a pixel say (x, y) as shown in Fig. 1(a). A simple derivative at the center pixel (x, y) in the direction of D (D comprises of {NW, W, SW, S, SE, E, NE, N}) is explained as the difference between the pixel (x, y) and its neighbor in the direction D. This derivative is denoted by $\nabla D(x, y)$. For example

$$\nabla N(x, y) = I(x, y-1) - I(x, y)$$
$$\nabla NW(x, y) = I(x-1, y-1) - I(x, y). \qquad (1)$$

Then, the principle of this fuzzy derivative is established on the following observation. Let an edge be passing through the pixel (x, y) in the SW – NE direction. The value of the derivative $\nabla NW(x, y)$ will be high, but derivative values of neighbor pixels which are perpendicular to the edge's direction will be large too. For example, in the direction of NW, we can find the values $\nabla NW(x, y)$, $\nabla NW(x-1, y+1)$ and $\nabla NW(x+1, y-1)$. Our idea is to eliminate the effect of one derivative value which can turn out to be high due to presence of the noise. Hence, if two out of the three derivatives are small, then we can assume that no edge is there in the considered direction. This study will be considered when we devise the fuzzy rule to find values of fuzzy derivative.



(a)



(b)

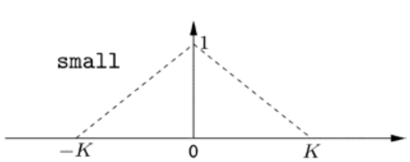
Fig. 1. (a) Neighbour of a central pixel (x, y). (b) Pixel values shown in grey are used to calculate the "fuzzy derivative" of the centre pixel (x, y) for the NW-direction.

TABLE I
Pixels required to find the fuzzy derivative in each direction

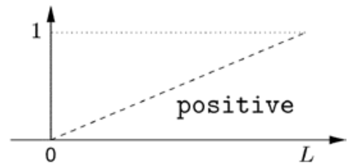
direction	position	set w.r.t. (x, y)
NW	$(x-1, y-1)$	{(-1,1),(0,0),(1,-1)}
W	$(x-1, y)$	{(0,1),(0,0),(0,-1)}
SW	$(x-1, y+1)$	{(1,1),(0,0),(-1,-1)}
S	$(x, y+1)$	{(1,0),(0,0),(-1,0)}
SE	$(x+1, y+1)$	{(1,-1),(0,0),(-1,1)}
E	$(x+1, y)$	{(0,-1),(0,0),(0,1)}
NE	$(x+1, y-1)$	{(-1,-1),(0,0),(1,1)}
N	$(x, y-1)$	{(-1,0),(0,0),(1,0)}

IJSER

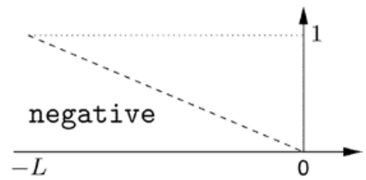
In Table I, we have shown an overview of the pixels that we used to find the fuzzy derivative for every direction. Every direction in column 1 corresponds to a fixed position in column 2. The sets shown in column 3 specifies which pixels are to be taken with respect to center pixel (x, y).
To find the value that could express the extent to which the fuzzy derivative in a particular direction is small, then we will make use of small fuzzy set.
The membership function being denoted by mk(u) for the property small as shown in Fig. 2(a) is given below-



(a)



(b)



(c)

Fig. 2. Membership functions (a) small, (b) positive, and (c) negative.

The membership function being denoted by mk(u) for the property small as shown in Fig. 2(a) is given below-

$$m_k(u) = \begin{cases} 1 - \frac{|u|}{k}, 0 \leq |u| \leq K \\ 0, \ |u| > K \end{cases} \qquad (2)$$

Where k is an adaptive parameter.
For example, the fuzzy derivative value ∇FNW(x, y) for the pixel (x, y) in the direction of NW is found by applying the rule show below-

If (∇NW(x, y) is small and ∇NW(x-1, y+1) is small) or
(∇NW(x, y) is small and ∇NW(x+1, y-1) is small) or
(∇NW(x-1, y+1) is small and ∇NW(x+1, y-1) is small)
Then ∇FNW(x, y) is small. (3)

Eight rules like above are applied, each one of them computing the extent of membership of the fuzzy derivatives ∇FD(x, y), D ∈ dir, to the set small. All these rules are applied by using the minimum to represent the AND-operator, and the maximum is used for the OR-operator. A defuzzification is not required here as the second stage which is the "fuzzy smoothing", directly utilizes the membership degree to small.
The robustness that we have tried to achieve by this fuzzy derivative is by integrating multiple derivatives around a pixel and by making the K parameter as adaptive. The proper choice of parameter K is explained later.

2.2 Fuzzy Smoothing

For computation of correction term for each processed pixel, fuzzy rules are applied in each direction corresponding to the pixel. The general idea used is; if in a certain direction there

ISSN 2229-5518

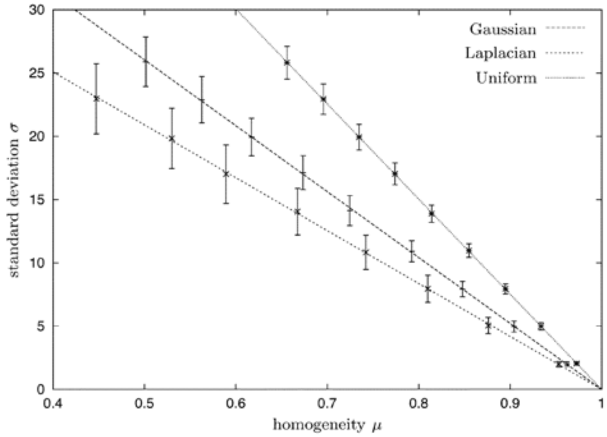


Fig. 3. Relationship between the homogeneity μ and the noise level σ empirically measured by patches of size 9 x 9 (N = 9). The accuracy of σ is shown by the standard deviation of σ itself.

is no edge (assumption) then derivative value is used to compute correction term. The edge assumption is accomplished using fuzzy derivative value. For correction term computation we need to determine positive and negative values.

For example, for direction say NW, using fuzzy derivative value for that direction; ∇FNW(x, y) and derivative value ∇NW(x, y) following two rule are applied and henceforth their truthiness λ+NW and λ-NW.

λ+NW : If ∇FNW(x, y) is small and ∇NW(x, y) is positive then c is positive

λ-NW : If ∇FNW(x, y) is small and ∇NW(x, y) is negative then c is negative.

Next for each direction, linear membership functions (fig 2(b) and 2(c)) are used for positive and negative properties to implement the AND–operator and OR-operator by respectively the minimum and maximum.

Final step was defuzzification because we aimed at finding correction term Δ which can be added to the respective pixel. Hence the truthiness of rules λ+D and λ-D, D is direction are aggregated by computing and rescaling the mean truthiness as follows:

$$\Delta = \frac{L}{\,}\sum_{D \in dir}(\lambda_D^+ - \lambda_D^-) \qquad (4)$$

where D represents all directions and L ,the number of gray levels. As a result each directions contribute to correction term Δ.

3 Adaptive Threshold Selection

Instead of using lager windows for heavier noise to obtain desired result, we preferred applying filter iteratively. The shape of membership function small is acquired after each iteration depending on the estimate of remaining noise. The method assumes that a percentage p of image is homogeneous and can be used to get noise density.

At start we divide the whole image into N X N blocks and then for each block a rough measure for homogeneity is considered by using maximum and minimum pixel value.

$$\mu = 1 - \frac{max_{(x,y)\in B} I(x,y) - min_{(x,y)} I(x,y)}{L} \qquad (5)$$

(a)

(b)

Fig. 4. Original test images. (a) “Cameraman.” (b) “Boats.”

This measure is generally used for fuzzy image processing [13]. Next, homogeneity value is computed and the presumption: percentile p of the most homogeneous block is computed. From statistical model it can be deduced that there is a linear relationship between homogeneity and standard deviation.

Assume M independent identically distributed noise samples with fX (x; σ) as probability density function and FX(x; σ) as cumulative density function. As PDF rescales with change in standard deviation, the maximum minimum of these samples are scaled same way. This helps in establishing linear relationship between homogeneity and standard deviation. This can also be derived formally; assume expectation value E[X] be zero, and variance E[X2] to be σ2. If PDF is scaled with a factor α, results:

$$f_X(x; \alpha\sigma) = \frac{1}{\alpha} f_X\left(\frac{x}{\alpha}; \sigma\right) \qquad (6)$$

$$F_X(x; \alpha\sigma) = \frac{1}{\alpha} F_X\left(\frac{x}{\alpha}; \sigma\right) \qquad (7)$$

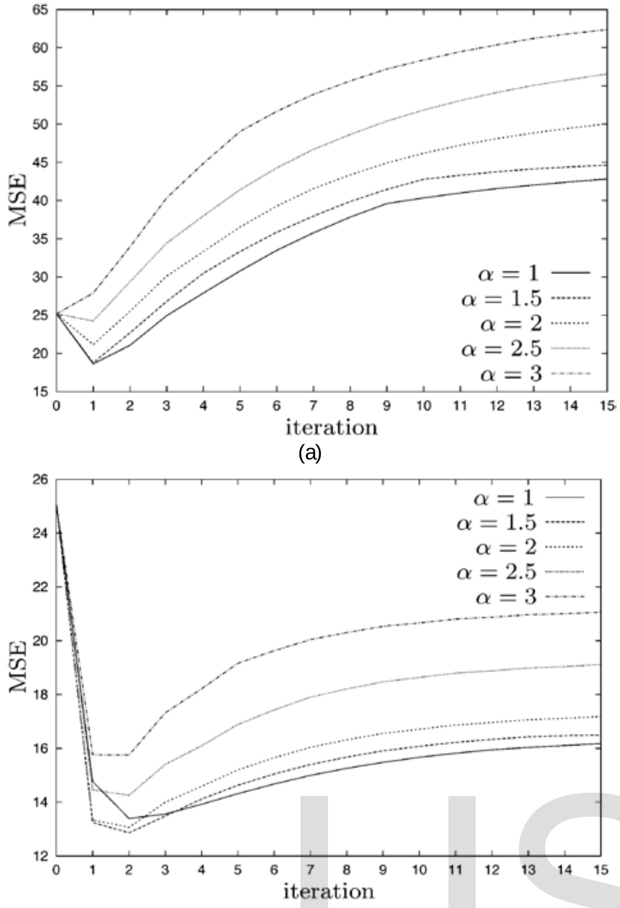
Minimum and maximum of these M samples can be defined as:

$$X_{max} = \max X_i$$

$$X_{min} = \min X_i$$

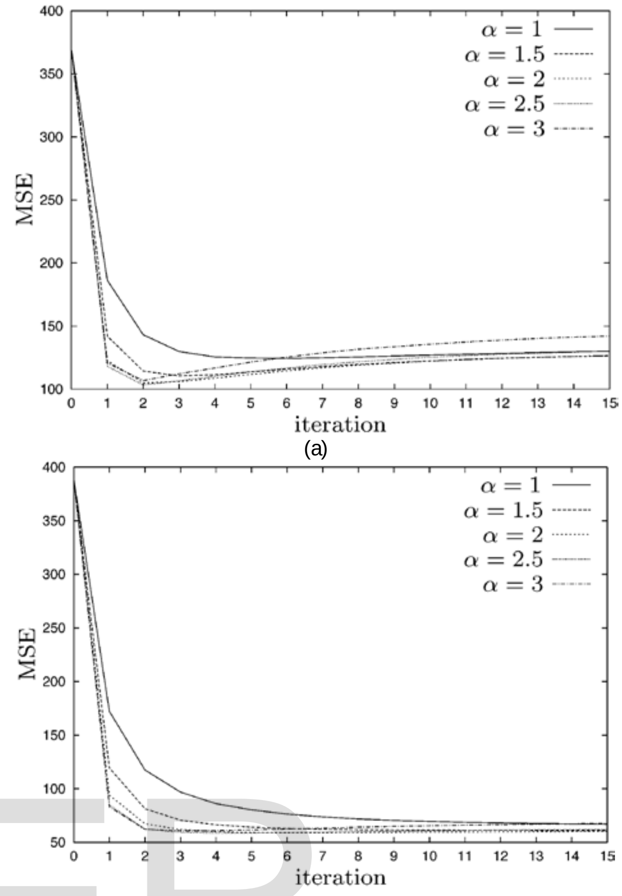
IJSER © 2015
http://www.ijser.org

ISSN 2229-5518



(b)

Fig. 6. MSE (mean squared error) for (a) “cameraman” and (b) “boats.” (σ = 5).



(b)

Fig. 7. MSE (mean squared error) for (a) “cameraman” and (b) “boats.” (σ = 20.)

IJSER

for which CDFs can be derived as

$$F_{X_{max}}(x;\sigma) = F_X^M(x;\sigma)$$
$$F_{X_{min}}(x;\sigma) = 1-(1-F_X(x;\sigma))^M$$

From (6) and (7) scaling of E[Xmax] and E[Xmin] according to α can be show as

$$E[Xmax;\alpha\sigma] = \alpha E[Xmax;\sigma]$$
$$E[Xmin;\alpha\sigma] = \alpha E[Xmin;\sigma]$$

Therefore, relationship between homogeneity can be expressed as

$$\sigma = (1-\mu)\text{ɣM} \qquad (8)$$

Where ɣM is slope.

The value of ɣM is computed empirically. A large number of blocks are generated and each block consist of expected distribution. The effective noise level for each block is computed. The mean value and standard deviation are computed for entire test set. This activity is conducted for several noise levels. Fig 3 shows the outcome for N=9 and 200 experiments. The errors bar shows standard deviation on noise level. This experiment is done for Gaussian noise, Laplacian noise, and uniform noise.

Next using the assumption that at least a percentage p blocks were originally homogeneous. The histogram of homogeneity of every block is computed. The value µP of this is related to noise variance σ using the linear relationship. A final amplification factor (discussed later) is used to get the parameter.

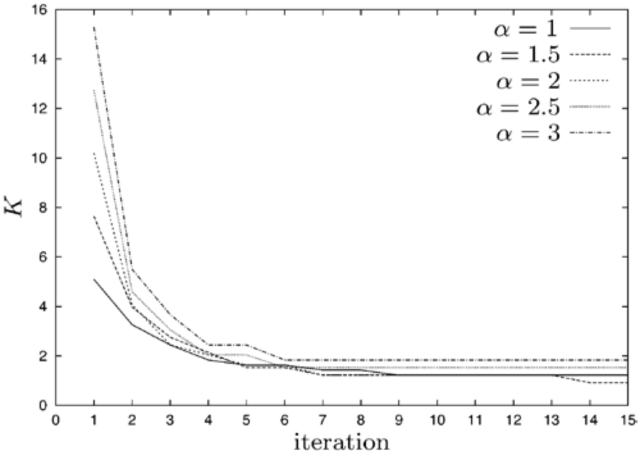
$$K = \alpha\sigma = \alpha(1-\mu)\ \text{ɣN2} \qquad (9)$$

This is applied every time before each iteration to obtain parameter K, to determine shape of membership function small. This helps in differentiating between blocks containing both image and blocks containing only noise. This is achieved by sorting histogram on homogeneity values. As a result, noise variance is on smooth blocks only, if initial hypothesis remains true.

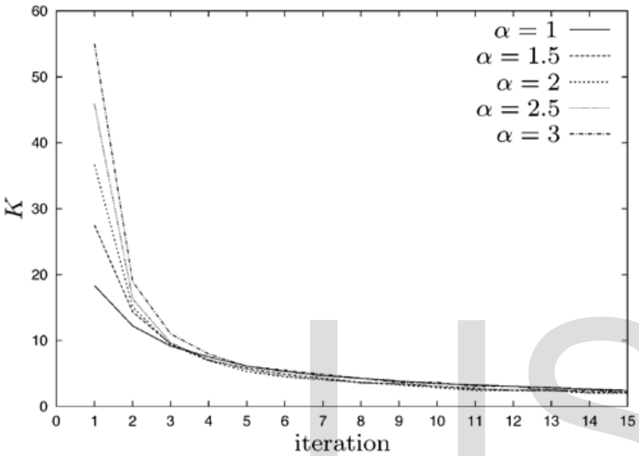
4 Results

After adding Gaussian noise of different levels to PGM image, the proposed filter is applied to it. This helps in comparing and evaluating filtered image against original image. Fig. 4 shows test images: “cameraman” and “boats”.

ISSN 2229-5518



(a)

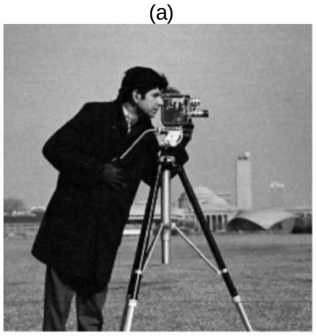


(b)

Fig. 8. Parameter K for "boats." (a) σ = 5. (b) σ = 20.

(a)

(b)



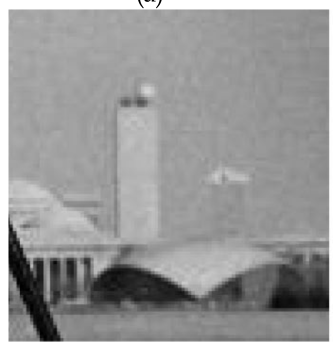
(c)

(d)

Fig. 9. (a) "Cameraman" with additive gaussian noise (σ = 5). (b) After Wiener filtering (3x3). (c) After fuzzy mean (FM). (d) After proposed fuzzy filter (σ = 1).

(a)

(b)



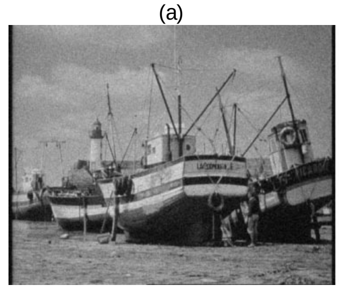
(c)

(d)

Fig. 10. Detail images of the results of Fig. 9.

(a)

(b)



(c)

(d)

Fig. 11. (a) "Boats" with additive gaussian noise (σ = 20). (b) After Wiener filtering (3x3). (c) After AWFM2. (d) After proposed fuzzy filter (σ = 2).

Fig. 5 shows normalized histogram of homogeneity of "cameraman" for both original and corrupted image with different noise levels, i.e., σ =5, σ =10 and σ =20. Using 20 % percentile and (8), the noise levels are respectively, 5.2, 9.4, and 17.7. Our filter is applied to these noise levels using different values of amplification factor α (α=1.0-3.0). To evaluate results, for both original and filtered image, the mean squared error (MSE) is computed.

Fig. 6 and 7 depicts plot of MSE as function of number of iterations for image containing noise with σ=5 and σ=20.

ISSN 2229-5518

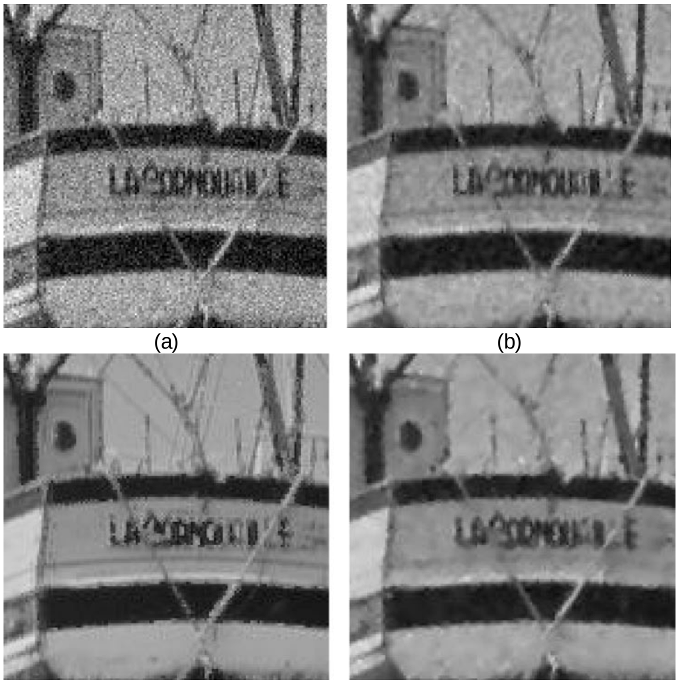


Fig. 12. Detail images of Fig. 11.

It can be noticed that one iteration is sufficient to remove low noise levels (Fig. 6). Also, low amplification factor α gives better results. The MSE of "cameraman" unexpectedly increases with increasing number of iterations, this is mainly due to its content. i.e.; the grass is close to noise and gets filtered. This increase is not observed in other image such as "boats". So, image with low noise levels needs to be treated carefully.

For high noise levels (Fig. 7), results are much more stable. A low number of iteration only efficiently removes noise. Also higher value α produces better results.

Fig. 8 shows parameter K that depends on remaining noise level σ for "boats" test image. Based on application's acceptable noise level, we could use σ estimate as stop criterion. Also, a change K with respect to previous iteration is small. Smoothing is affected by change in parameter α. MSE-curves shows α can be computed using σ estimate. Noise levels are directly proportional to factor α, i.e., a high value of noise levels gives high value of α.

Results of different filters: the mean filter, the adaptive Weiner filter [14], fuzzy median (FM) [15], the adaptive weighted fuzzy mean filter (AWFM1 and AWFM2) [10], [11], the iterative fuzzy filter (MIFC), and extended iterative fuzzy filter (EIFC) [12]. Observing Fig. 9 and 10 shows that our filter preserves certain details better like grass, background. It can be noted that our filter preserves grass better than fuzzy mean filter.

The image of "boats" gives a different result. For low levels of noise (σ=5), our proposed filter performs well, but for higher levels of noise, AWFM2 filter gives better results. Fig.11 describes the filtered images. The detailed images of Fig.12 shows that the filter AWFM2 did preserved small minute details such as narrow ropes where our proposed filter gives more natural image without having the patches of the adaptive Wiener filter.

5 Conclusion

This paper showed a fuzzy filter for additive noise removal. Using the fuzzy derivative estimation it distinguishes between local variations and structures like edges. Fuzzy rules are used to consider all directions around the pixel which is being processed. Also, membership function's shape changes and adapts according to amount of noise present after each iteration. Experimental results describes how much our filter is feasible. Numerical measure like MSE and observing visually we have obtained good results. Hence, our proposed filter is sufficiently simple to enable fast implementations.

References

[1] E. Kerre and M. Nachtegael, Eds., "Fuzzy Techniques in Image Processing." New York: Springer-Verlag, 2000, vol. 52, Studies in Fuzziness and Soft Computing.

[2] D. Van De Ville, W. Philips, and I. Lemahieu, "Fuzzy Techniques in Image Processing." New York: Springer-Verlag, 2000, vol. 52, Studies in Fuzziness and Soft Computing, ch. Fuzzy-based motion detection and its application to de-interlacing, pp. 337–369.

[3] M. Nachtegael and E. E.Kerre, "Connections between binary, gray-scale and fuzzy mathematical morphologies," Fuzzy Sets Syst., to be published.

[4] "Decomposing and constructing fuzzy morphological operations over α-cuts: Continuous and discrete case," IEEE Trans. Fuzzy Syst., vol. 8, pp. 615–626, Oct. 2000.

[5] B. Reusch, M. Fathi, and L. Hildebrand, "Soft Computing, Multimedia and Image Processing—Proceedings of the World Automation Congress." Albuquerque, NM: TSI Press, 1998, ch. Fuzzy Color Processing for Quality Improvement, pp. 841–848.

[6] S. Bothorel, B. Bouchon, and S. Muller, "A fuzzy logic-based approach for semiological analysis of microcalcification in mammographic images" Int. J. Intell. Syst., vol. 12, pp. 819–843, 1997.

[7] 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, Aug. 1995.

[8] "A fuzzy filter for images corrupted by impulse noise." IEEE Signal Processing Lett., vol. 3, pp. 168–170, June 1996.

[9] F. Russo, "Fire operators for image processing." Fuzzy Sets Syst., vol. 103, no. 2, pp. 265–275, 1999.

[10] 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.

[11] C.-S. Lee and Y.-H. Kuo, "Fuzzy Techniques in Image Processing." New York: Springer-Verlag, 2000, vol. 52, Studies in Fuzziness and Soft Computing, ch. Adaptive fuzzy filter and its application to image enhancement, pp. 172–193.

[12] 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.

[13] H. Haussecker and H. Tizhoosh, "Handbook of Computer Vision and Applications." New York: Academic, 1999, vol. 2, ch. Fuzzy Image Processing, pp. 708–753.

[14] J. S. Lim, "Two-Dimensional Signal and Image Processing." Upper Saddle River, NJ: Prentice-Hall, 1990, ch. Image Restoration, pp. 524–588.

[15] K. Arakawa, "Median filter based on fuzzy rules and its application to image restoration." Fuzzy Sets Syst., pp. 3–13, 1992.

IJSER

IJSER