

A Survey on various Image Filtering Approaches to remove Impulse Noise

Aaditya Sharma, R. K. Pateriya
 Computer Science & Engineering Department
 Maulana Azad National Institute of Technology, Bhopal, India
 manit.aaditya@gmail.com,pateriyark@gmail.com

Abstract: Images gets corrupted either during acquisition or transmission. Frequently occurring noise that might occur in images is impulse noise, because of that various image processing operations such as image segmentation, object identification, and similarity matching etc. cannot be performed efficiently. This paper focus on various existing image filtering techniques and their improvements. Several median-based denoising methods tends to work well for low level impulse noise but perform poorly for high level impulse noise.

Index Terms- Noise models, threshold based switching median filter, operator based median filter, morphological based median filter, statistics based median filter.

1 INTRODUCTION

An image is a projection of three dimensional object to a two dimensional object. A grayscale image can be defined using two dimensional vector $X(i, j)$ where i and j are the spatial co-ordinates and value of $X(i, j)$ represent the intensity value of that pixel location [1]. Same thing is applicable for color images also except that in case of color images we need to consider the multi-color channel with their specific value. For color images intensity value of pixel $X(i, j)$ will be represented as: $X_{ij}^R X_{ij}^G X_{ij}^B$ Corresponding to red, green, blue channel respectively [2]. Image processing can be think of as combinations of three stages-

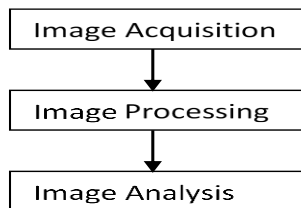


Fig.1. Various image processing operations

Let $X(i, j)$ be the original image and $\eta(i, j)$ be the noise component, (1.1) gives mathematical description of additive noise model.

$$g(i, j) = X(i, j) + \eta(i, j) \dots (1.1)$$

Where $g(i, j)$ is final noisy image. Rayleigh noise models is used in ultrasound images [4]. Impulse noise is statistically independent on the input signal. It follows bipolar density function. Impulse noise is neither considered as additive noise nor as multiplicative noise, rather it is just some random value picked from $[n_{min}, n_{max}]$, where n_{min} and n_{max} is the minimum and maximum number in some given range.(1.2) gives mathematical description of impulse noise.

$$g(i, j) = X(i, j) \text{ or } \eta(i, j) \dots (1.2)$$

Impulse noise can be categorized into two types-

- A. Random valued impulse noise or uniform impulse noise.
- B. Fixed valued impulse noise or salt and pepper impulse noise.

1.1 Random valued impulse noise

In this impulse noise pixel value can be replaced by some random number in some defined range. See (1.3) for mathematical description,

$$p(k) = \begin{cases} \frac{1}{b-a} & a \leq k \leq b \\ 0 & \text{else} \end{cases} \dots (1.3)$$

Here a and b are the maximum and minimum intensity values and k is some random variable showing the intensity value between the given range.(1.4) describes the mean of random valued impulse noise.

$$\mu = (a + b)/2 \dots (1.4)$$

μ is called the mean of the density function. Variance of the density function can be expressed as.

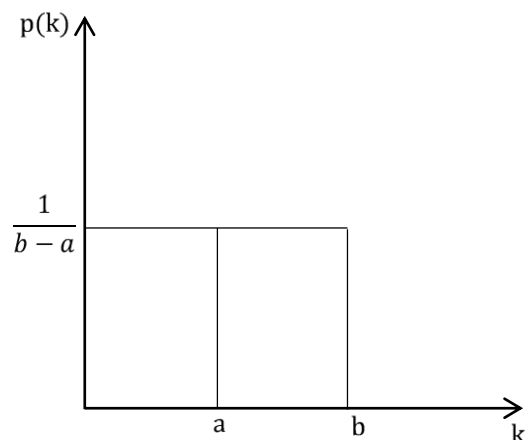


Fig. 2. probability density function of impulse noise

$$\sigma^2 = \frac{(b-a)^2}{12} \dots (1.5)$$

Fig 2 shows the PDF of uniform impulse noise.

1.2 Fixed valued impulse noise/salt and pepper noise

As the name implies in this noise image will contain some sprinkles of white or black dots. Such types of noise will contaminate only few pixels as compared to uniform impulse noise which corrupt many pixels [5, 6]. Probability density function of salt and pepper noise can be defined as in (1.6).

$$p(k) = \begin{cases} p(a) & \text{if } k = a \\ p(b) & \text{if } k = b \\ 0 & \text{else} \end{cases} \quad (1.6)$$

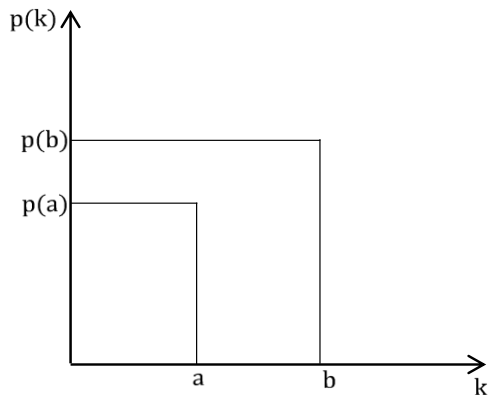


Fig.3. Probability density function of salt and pepper noise

Where gray level a is considered as black spots whereas gray level b is considered as white dots. (1.6) has been represented graphically in Fig. 3.

2 Classification of Impulse Removal Techniques

Several techniques have been implemented to remove impulse noise from the image without discerning between noisy and noiseless pixels. Several filters such as mean filter, alpha trimmed mean filter, rank ordered mean filter etc. have been used in past to remove impulse noise [7]. These filters can remove noise sufficiently but along with removal of noise some useful information also gets vanish from the image. These filters were unable to differentiate between the corrupted pixels and thin line edges due to which image gets blurred. Later researchers moved towards nonlinear filtering approaches. Noise removal in spatial domain can be classified into six classes.

- Methods based on threshold.
- Methods based on operators.
- Methods based on statistics.
- Methods based on fuzzy logic.

- Methods using morphology.
- Methods using ANFIS

Median filtering is nonlinear method of removing impulse noise, in median filtering a window of some size $2(L + 1) * 2(L + 1)$ is use where L is some positive integer. This window Is moved over the entire image and pixels which comes inside the window are processed to remove noise [8, 9]. See fig 4 which shows various image denoising methods.

2.1 Switching median filter using threshold

In this filter, impulse detector is used to find the corrupted pixel. Impulse detector finds the difference between the current pixel under consideration and the median of window, if the difference is greater than some suitable threshold then either median of window or weighted median is used to restore the noisy pixel [10 13]. Impulse detection operation can be defined mathematically by (1.7) which explains the functionality of TSM filter.

$$Y_{ij} = \begin{cases} X_{ij} & \text{if } T > d1 \\ Y_{ij}^{CWM} & \text{if } d2 \leq T < d1 \\ Y_{ij}^{SM} & \text{if } T < d2 \end{cases} \dots (1.7)$$

Where Y_{ij} is gray level of output pixel and Y_{ij}^{SM} is the output produced by standard median filter and Y_{ij}^{CWM} is the output of CWM filter. Here $d1 = \text{abs}(X_{ij} - Y_{ij}^{SM})$ and $d2 = \text{abs}(X_{ij} - Y_{ij}^{CWM})$ and T is suitably chosen threshold [14]. Multi-state median filter is more generalized form of TSM filter defined in [14]. MSM filtering scheme uses more than one CWM filters with increasing weights. In MSM filter a classifier has been used to decide which filter output should be used to decide the final filtering output [15]. This filtering scheme is given by (1.8). Where N specifies the number of pixels in window,

$$Y_{ij} = \begin{cases} X_{ij} & \text{if } d_1 < T \\ Y^{(N+1-w)}_{ij} & \text{if } d_w < T \leq d_{w-2}, 3 \leq w \leq N - 2 \\ Y^1_{ij} & \text{if } d_{N-2} \geq T \end{cases} \dots (1.8)$$

$$d_w = \text{abs}(X_{ij} - Y^w_{ij}), \quad w=1, 3, \dots, N-2.$$

To improve the performance of the impulse detection progressive switching median (PSM) has been proposed [16]. A technique based on median of absolute deviation from the median (MAD) has been used [16] to remove the problem of fixed threshold for the entire image. In this filter d_w is compared with dynamic updating threshold given by (1.9). Fig 5 shows calculation of median of absolute deviation.

$$T_w = s * MAD + \delta_w \quad \dots (1.9)$$

Here s is some constant that depends on the image and δ_w is some constant that changes over the iterations. Based on the condition $d_w > T_w$ where $w=0, 1, 2, 3$ for any w, pixel is Considered as noisy pixel. Simulations shows that $[\delta_0, \delta_1, \delta_2, \delta_3] = [40, 25, 10, 5]$ is good for removal of random valued impulse

noise and $[\delta_0, \delta_1, \delta_2, \delta_3] = [55, 40, 25, 15]$ is good for fixed valued impulse noise. By doing experiments on various images it has been observed that $0 \leq s \leq 0.6$ is good for noise suppression.

2.2 Operator based switching median filter

In operator based switching methods some operator is used to identify the noisy pixel. Some of the examples are Lapla-

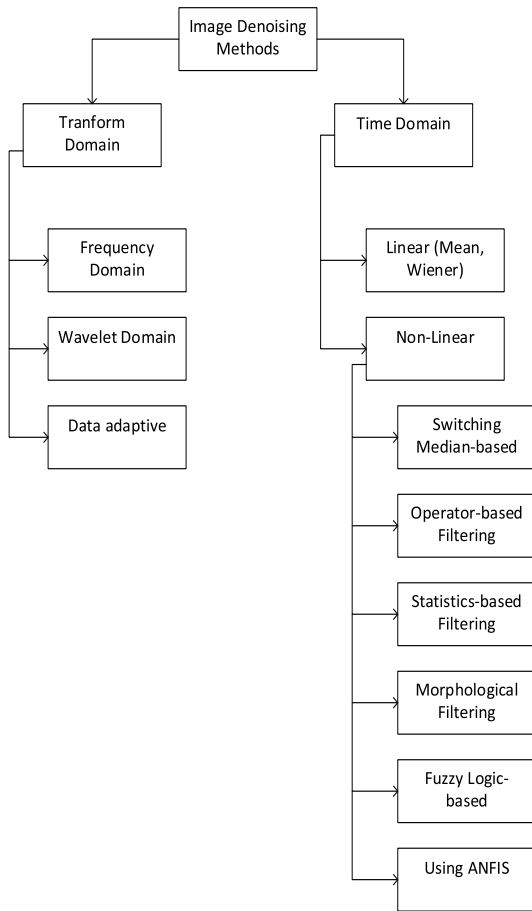


Fig. 4. Various image denoising methods

$$MAD = \text{median}\{|X_{(i-r,j-l)} - \text{median}(M_{1st}^X(i,j))|\} - 1 \leq (r,l) \leq 1$$

Fig. 5. Calculation of median of absolute deviation

cian, Lulu etc., more details can be found in [17-19]. By using the above four Laplacian kernels defined in Fig 6 convolution operation is performed over the selected window of pixels and for each kernel a value is computed, after that minimum of those four convolution is used to compute B_{ij} .

$$B_{ij} = \text{minimum}\{|X_{ij} \otimes K_c, 1 \leq c \leq 4|\} \dots (2.0)$$

B_{ij} is then compared with some threshold to identify whether current pixel is noisy or noise free.

2.3 Statistics based switching median filter

-1	0	0	0	0
0	-1	0	0	0
0	0	4	0	0
0	0	0	-1	0
0	0	0	0	-1

(a)

0	0	-1	0	0
0	0	-1	0	0
0	0	4	0	0
0	0	-1	0	0
0	0	-1	0	0

(b)

0	0	0	0	-1
0	0	0	-1	0
0	0	4	0	0
0	-1	0	0	0
-1	0	0	0	0

(c)

0	0	0	0	0
0	0	0	0	0
-1	-1	4	-1	-1
0	0	0	0	0
0	0	0	0	0

(d)

Fig. 6. Four different Laplacian kernels

2.3.1 Boundary discriminative noise detection (BDND)

Boundary discriminative noise detection method divides the pixels into three categories [20, 34].

- Pixels which are corrupted by impulse noise but of low intensity.
- Pixels which are not corrupted.
- Pixels which are corrupted with high impulse noise.

BDND algorithm works as follows-

1. First create a window of 21*21 and align it to the centre pixel.
2. Use some efficient sorting algorithm to sort the pixels inside the window to find the median, or we can find the median directly by using the algorithm [21] of the sorted vector V.
3. Then compute the difference of intensity of all pair of adjacent pixels and across the sorted vector V and find the difference V_d .
4. Pixels which are having lowest intensity in current window and median of window V, calculate the maximum intensity difference in V_d of the same range and mark the corresponding pixel in sorted vector V as boundary b1.
5. Another boundary b2 is calculated by taking difference maximum intensity in the window and median value of window.
6. If pixel currently being processed is found under middle cluster then classification process stops here and pixel is identified as noise free pixel, otherwise continue iteratively.
7. Execute the above steps from step-2 to step-5 iteratively around the pixel under consideration.

8. After processing all the pixels a map is created which contains two values 1 or 0, if some pixel is having value 1 it means that pixel is noisy and if some pixel is having 0 it is noise-free pixel.

9. Pixel restoration operation is applied over only noisy pixels, noise-free pixels are kept intact.

2.3.2 High performance detection filter (HPDF)

High performance detection filtering scheme is based on the observation that for every noise-free pixel there are some minimum number of similar neighbours [22]. This algorithm works in four stages, where each stage perform similar kind of work. Here centre pixel is subtracted from the other pixels of the window to find the absolute difference. Where N_k , the window size is in k 'th stage. Next step is to count the number of pixels in the current window for which M_j is lesser than the predefined intensity level C_k in k 'th stage. (2.1) shows that count () function returns the number of pixels lesser than intensity at the k 'th stage.

$$C = \text{count}(M_j \leq C_k) \quad \forall k=1, 2, 3, 4 \quad \dots (2.1)$$

C is compared with threshold to determine whether current pixel is noisy or not. This process is applied over all the pixels in the image and in i 'th stage $C_k = [40, 30, 20, 10]$ and $T_k = [7, 5, 3, 2]$ is applied. To restore the corrupted pixel mean of noise-free pixel is calculated and replaced at the centre of window. This algorithm considers each noisy pixel just once in the whole process.

2.3.3 Noise adaptive switching median based filter

This algorithm is based on finding the local extremes for current window, and a pixel is considered noisy if that pixel appears as local extreme N times, where N defines number of pixels in the filtering window. This process is applied over the entire image and after getting some local extremes median of current window is find out and that is used to restore the noisy pixel. This method is most suitable for fixed valued impulse noise [23].

2.3.4 Histogram based filtering scheme

This algorithm generates a binary map of 1's and 0's, which is used later to identify whether the considered pixel is noisy or noise-free [24, 35]. Algorithm has been defined below.

1. Take a window of 21*21 centred at the current pixel.
2. Generate the histogram of current window and find the values of maximum gray-level and minimum gray-level.
3. Calculate the average gray level and then for indexes between minimum gray-level and average gray-level calculate the difference non-zero indexes, out of all the differences find maximum difference and mark it as boundary $b1$.
4. Calculate the difference between non-zero indexes which comes between maximum gray-level and average gray-level,

then find the max value out of all computed differences and mark it $b2$.

5. After doing this computation pixels are categorized into three clusters, so some pixel X_{ij} is considered correct if it belongs to middle cluster otherwise it's corrupted.

6. This algorithm make sure that some noise-free pixel should not be considered as noisy pixels so to verify that one more time the above process is executed and if it is found that some pixel X_{ij} still belongs to middle cluster then finally it is declared as noise-free pixel.

7. To restore the corrupted pixels median of the current window is used to replace the current pixel.

e) Advanced boundary discriminative noise detection (ABDND)

This algorithm computes the histogram of the image from which range of gray-levels is computed which is used to find a suitable threshold for filtering operation. For current window centred at the current pixel the difference between the current pixel and the brightest pixel as well as current pixel and lowest intensity pixel is computed. Pixels which are found noisy in first time are sent to next stage using local statistics to avoid false detection [25]. To restore the noisy pixels adaptive switching median filter is used.

2.3.5 Morphology based switching median filtering

Morphology based approach uses opening and closing operations [3] to identify the noisy pixel. Closing operation is defined as dilation of image by structuring element b followed by erosion, and opening operation is defined as erosion of image by structuring element b followed by dilation. Morphological filtering scheme has been defined in [26]. Noisy image is fed to

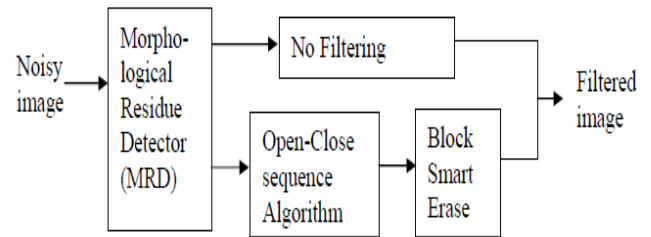


Fig.7. Morphological filtering scheme

the morphological residue detector (MRD), then two output files are created, one is fed to OCF/COF sequence algorithms and another is kept intact, later both are combined to produce the final filtered image. Fig: 7 shows graphical representation of this concept. Opening and Closing operations have been defined in (2.2) and (2.3) respectively.

$$E_o = X - X\Delta b \quad \dots(2.2)$$

$$E_c = X\forall b - X \quad \dots(2.3)$$

Where Δ is closing operation and \forall is opening operation, E_o and E_c are the opening and closing distance from the input signal

respectively, b is the structuring element. Pixels are identified as noisy or noise-less based on after comparing E_o and E_c with threshold. (2.4) entails more details.

$$g_{ij} = \begin{cases} 1 & \text{if } E_o \geq T \text{ and } E_c = 0 \\ -1 & \text{if } E_c \geq T \text{ and } E_o = 0 \\ 0 & \text{otherwise} \end{cases} \dots (2.4)$$

If $g_{ij}=1$, then X_{ij} is considered as salt noise, and if $g_{ij}= -1$ then current pixel is considered as pepper noise and if $g_{ij}=0$ then current pixel is considered as noise-free pixel. To remove the noise two filters are used in sequence of open and close. First one is called open close filter (OCF). (2.5) explains the working of OCF.

$$OCF(X) = (X \nabla b1) \Delta b2 \dots (2.5)$$

Where $b1$ and $b2$ are structuring element. To preserve the small details present in image size of $b1$ should be kept small and size of $b2$ should be made larger than $b1$. This filter removes pepper noise correctly, however the pepper noise whose size exceeds $b1$ cannot be eliminated, to remove remaining noise second filter close open filter (COF) is applied. Mathematical operation of COF has been explained in (2.6).

$$COF(X) = (X \Delta b1) \nabla b2 \dots (2.6)$$

Close open filter removes salt noise effectively. To restore the corrupted pixel this filter uses median of current window.

2.3.6 Conditional morphological detector

In this morphological system conditional opening and conditional closing operations are used to find the absolute deviation [27] which is compared with the current pixel using some threshold and if deviation is becoming more than the threshold then pixel is considered as noisy otherwise noise-less. Absolute deviation can be calculated by using following formula given in (2.7).

$$ad_{ij} = \left| \frac{(X \nabla b)_{ij}^c + (X \Delta b)_{ij}^c}{2} \right| - X_{ij} \dots (2.7)$$

Where ad_{ij} is absolute deviation and $(X \nabla b)_{ij}^c$, $(X \Delta b)_{ij}^c$ is called conditional closing and conditional opening operation respectively. Conditional opening and closing can be computed by using conditional erosion and dilation. Conditional dilation and conditional erosion can be computed by using noisy image, structuring element and erosion gradient, dilation gradient respectively.

2.3.7 Fuzzy logic based switching median filtering

It is difficult to set some parameter or some condition based on which some filter can be selected. It shows that filtering system possess some capability of uncertain information due to which we can incorporate fuzzy logic into image filtering. Fuzzy logic based methods can be categorized into two broad categories.

a) Fuzzy rule based technique for filtering.

b) Fuzzy techniques for identifying corrupted pixels.

To identify the corrupted pixel one such approach is defined in [28], it works as follows.

1. Let we have a window of size $N \times N$ centred at X_{ij} . First of all sort all the pixels in current window so that pixels come in ascending order. Corrupted pixels can be on the either end of this sorted sequence. To identify the current pixel as noisy let we define a function function as given in (2.8).

$$g_{ij} = \begin{cases} 1 & \text{if } X_{ij} \leq X_z \text{ or } X_{ij} \geq X_{N-z} \\ 0 & \text{otherwise} \end{cases} \dots (2.8)$$

(2.8) uses a constant named z .

2. Subtract the central pixel from all the pixels in the current window and find the average of absolute difference as given in (2.7). If average of absolute difference is high then it shows that pixel is noisy. A new statistics S_{ij} has been defined in (2.9).

$$S_{ij} = ad_{ij} * g_{ij} \dots (2.9)$$

3. Next a fuzzy impulse detector function is defined, (3.0) encompasses more details about fuzzy impulse detector.

$$fzz_{ij} = \begin{cases} 0 & \text{when } S_{ij} \leq W_1 \\ \frac{(S_{ij}-W_1)}{(W_2-W_1)} & \text{when } W_1 \leq S_{ij} \leq W_2 \\ 1 & \text{when } S_{ij} \geq W_2 \end{cases} (3.0)$$

Where W_1 and W_2 are some constant and $fzz_{ij} = 0$ shows that pixel is noise-free, $fzz_{ij} = 1$ shows that pixel is completely noisy whereas $0 < fzz_{ij} < 1$ indicates the extent up to which the pixel is noisy.

2.3.8 Adaptive Neuro –fuzzy inference based switching median filtering

In this filtering system after applying the noise filter output of that noise filter is fed to the sugeno-type fuzzy inference system. System parameters are tuned by computer generated artificial image and adaption of Neuro-fuzzy parameter is accomplished by using Levenberg-Merquardt optimization algorithm, details can be found here [29-30]. This filter takes two input, one is noisy image and other one is processed image, after applying median filter output is generated based on neuro-fuzzy system. Each of the input is having bell shaped membership function and its output function is linear [31]. Rules for the neuro-fuzzy system can be defined as given in (3.1), (3.2), (3.3) ... (3.9).

$$\text{If } (X_1 \text{ is } M_{11}) \text{ and } (X_2 \text{ is } M_{21}) \text{ then } L_1 = OP_1(X_1, X_2) (3.1)$$

$$\text{If } (X_1 \text{ is } M_{11}) \text{ and } (X_2 \text{ is } M_{22}) \text{ then } = OP_2(X_1, X_2) (3.2)$$

$$\text{If } (X_1 \text{ is } M_{11}) \text{ and } (X_2 \text{ is } M_{22}) \text{ then } L_3 = OP_3(X_1, X_2). (3.3)$$

$$\text{If } (X_1 \text{ is } M_{12}) \text{ and } (X_2 \text{ is } M_{21}) \text{ then } = OP_4(X_1, X_2). (3.4)$$

$$\text{If } (X_1 \text{ is } M_{12}) \text{ and } (X_2 \text{ is } M_{22}) \text{ then } L_5 = OP_5(X_1, X_2). (3.5)$$

If $(X_1 \text{ is } M_{12})$ and $(X_2 \text{ is } M_{23})$ then $= OP_6(X_1, X_2) \dots$ (3.6)

If $(X_1 \text{ is } M_{13})$ and $(X_2 \text{ is } M_{21})$ then $L_7 = OP_7(X_1, X_2)$. (3.7)

If $(X_1 \text{ is } M_{13})$ and $(X_2 \text{ is } M_{22})$ then $L_8 = OP_8(X_1, X_2)$. (3.8)

If $(X_1 \text{ is } M_{13})$ and $(X_2 \text{ is } M_{23})$ then $L_9 = OP_9(X_1, X_2)$. (3.9)

Where M_{ij} represent the j 'th membership function of i 'th input, L_k denotes the output of k 'th rule and OP_k denotes the k 'th output membership function. Input membership function can be defined as given in (4.0).

$$M_{ij} = \frac{1}{1 + \left| \frac{(u - p_{ij})}{q_{ij}} \right|^{2r_{ij}}}, i = 1, 2 \text{ and } j = 1, 2, 3 \dots (4.0)$$

Output membership function is given by (4.1).

$$OP_k(u_1, u_2) = t_{k1}u_1 + t_{k2}u_2 + t_{k3}, k = 1, 2, 3 \dots \dots 9 \dots (4.1)$$

In (4.1) parameters p, q, r, t are used to define the membership function. Neuro fuzzy system parameters are trained using suitable images [32, 33]. Weight factor associated with rules can be calculated as given in table 1. Output is given by (4.2).

$$Y_{OP} = \frac{(\sum_{k=1}^9 W_k L_k)}{\sum_{k=1}^9 W_k} \dots (4.2)$$

Some filtering approaches along with filtered pixel keep some other information also like whether the pixel is an edge pixel or not, see [33] for more details. Some approaches uses ANFIS to identify the corrupted pixel. From horizontal and vertical direction two neuro-fuzzy sub-detectors are used to find the corrupted pixel and then their output is fed to some decision making function which average out the inputs for final decision, for detail see [33]. Fig. 8. Is shown which explains the general structure of neuro-fuzzy impulse detector.

Table 1 Weight factor calculation rule

$W_1 = M_{11}(X_1) \cdot M_{21}(X_2)$
$W_2 = M_{11}(X_1) \cdot M_{22}(X_2)$
$W_3 = M_{11}(X_1) \cdot M_{23}(X_2)$
$W_4 = M_{12}(X_1) \cdot M_{21}(X_2)$
$W_5 = M_{12}(X_1) \cdot M_{22}(X_2)$
$W_6 = M_{12}(X_1) \cdot M_{23}(X_2)$
$W_7 = M_{13}(X_1) \cdot M_{21}(X_2)$
$W_8 = M_{13}(X_1) \cdot M_{22}(X_2)$
$W_9 = M_{13}(X_1) \cdot M_{23}(X_2)$

3 Conclusion

This paper focus on various median filtering based techniques

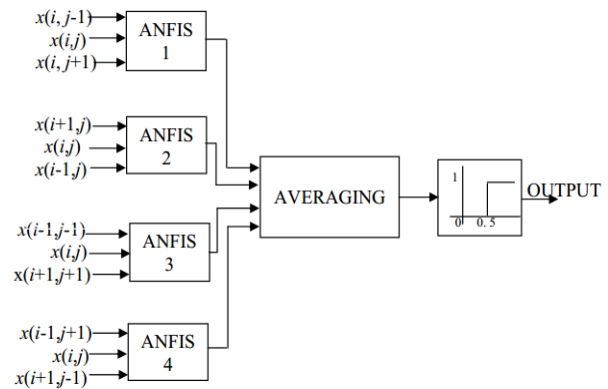


Fig. 8. General structure of neuro fuzzy impulse detector

to remove impulse noise from the images. Each technique has its own advantage along with limitation. Some techniques may be good for less noise density images however other methods might be highly computationally efficient. Some approaches are better for fixed valued impulse noise, other may be good for random valued impulse noise. After studying several techniques it is found that methods which uses impulse detector for noise identification, perform better than uniformly applied methods. After studying various approaches it is found that threshold based methods performance is poorer at high noise density because the choice of suitable threshold is critical factor however decision based approach is more suitable at high noise density.

4 REFERENCES

- [1] A. K. Jain, Fundamentals of Digital image processing, Prentice Hall of India, ISBN 0-13-336165-9, 1989.
- [2] R. Lukac and K. N. Plataniotis, Color Image Processing: Methods and Applications, CRC Press, 2007.
- [3] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Pearson Education, 2009.
- [4] S. Siltanen et al, "Statistical inversion for medical X-ray tomography with few radiographs I: General theory", Physics in Medicine and Biology, Vol. 48, pp. 1437-1463, 2003.
- [5] S. Sridhar, Digital Image Processing, Oxford University Press 2011.
- [6] E. Abreu et al, "A new efficient approach for the removal of impulse noise from highly corrupted images", IEEE Trans. Image Process., vol. 5, no. 6, pp.1012-1025, June 1996.
- [7] A. Mishra and M. Shandilya, "Fingerprint core point detection using gradient field mask", International Journal of Computer Applications, vol. 2, no.8, pp.19-23, June 2010.
- [8] M. Motwani, M. Gadiya, R. Motwani, and C. Frederick "A survey of image denoising techniques", Proc. Of GSPx 2004, Santa Clara Convention Center, Santa Clara, CA, pp. 1-7, Sept.2004.

- [9] T.A. Nodes and N. C. Gallagher, Jr., "Median filters: Some modifications and their properties", IEEE Trans. Acoust., Speech, Signal Process., vol. ASSP -30, pp. 739-746, May 1982.
- [10] A. Mehrotra, K.K. Singh, M.J. Nigam and K. Pal, "A novel algorithm for impulse noise removal and edge detection", International Journal of Computer Applications, vol.38, no. 7, pp.30-33, Jan.2012.
- [11] U. Ghanekar, A.K. Singh and R. Pandey, "A new scheme for impulse detection in switching median filters for image filtering", International conference on computational intelligence and multimedia Applications, IC-CIMA'07, Sivakasi, India, pp.442-446, Dec 2007.
- [12] N. Krishnan and J. Varghese, R. K. Selvakumar et al., "Selective switching median filter for the removal of salt and pepper impulse noise", in proc. IEEE Third International Conference on wireless and optical Communications Networks WOCN 2006, Bangalore, India, pp.1-5, April 2006.
- [13] A.B.Hamza, P.L.Escamilla, J.M.Aroza, R. R. Roldan, "Removing noise and preserving details with relaxed median filters", Journal of Mathematical Imaging and Vision, vol.11, pp.161-177, 1999.
- [14] T. Chen, K.-K. Ma, and L.-H. Chen, "Tri-state median filter for image denoising", IEEE Trans. Image Processing, vol.8, no. 12, pp. 1834-1838, Dec.1999.
- [15] T. Chen and H.R. Wu, "Space variant median filters for the restoration of impulse noise corrupted images", IEEE Trans. Circuits Syst. II, Vol.48, no. 8, pp. 784-789, Aug. 2001.
- [16] Z. Wang and D.Zhang, "Progressive switching median filter for removal of impulse noise from highly corrupted images", IEEE Trans. Circuits Syst. II, Vol. 46, no. 1, pp.78-80, 1999.
- [17] S. Zhang and A. Karim, "A new impulse detector for switching median filters", IEEE Signal Process. Lett. , vol. 9, no.11, pp. 360-363, Nov .2002.
- [18] O Kao, "Modification of the LULU operators for preservation of critical image details", International Conference on Imaging Science, Systems and Technology, Las Vegas, pp. 1-7, 2001.
- [19] Y. Dong and S. Xu, "A new Directional weighted median filter for removal of random valued impulse noise," IEEE Signal Process. Lett. , Vol. 14, no. 3, pp. 193-196, March 2007.
- [20] P.E. Ng, and K. K. Ma, "A switching median filter with boundary discriminative noise detection for extremely corrupted images," IEEE Trans. Image Process., Vol.15, no. 6, pp. 1506-1516, June 2006.
- [21] Manuel Blum, Robert W. Floyd, Vaughan Pratt, Ronald L. Rivest, Robert E. Tarjan, "Time bounds for selection," Journal of Computer and System Sciences, Vol. 7, Issue 4, pp. 448-461, Aug 1973.
- [22] A.S.Awad and H. Man, "High Performance Detection Filter for impulse noise removal in images," Electronics Letters, Vol. 44, no. 3, Jan 2008.
- [23] A. Fabijanska and D. Sankowski, "Noise adaptive switching median based-filter for impulse noise removal from extremely corrupted images," IET Image Process., Vol. 5, issue 5, pp. 472-480, 2011.
- [24] S. Indu and C. Ramesh, "A noise fading technique for images highly corrupted with impulse noise," International Conference on Computing: Theory and Applications, ICCTA'07, Kolkata, India, pp. 627-632, 2007.
- [25] A.K. Tripathi, U. Ghanekar and S. Mukhopadhyay, "Switching median filter: advanced boundary discriminative noise detection algorithm," IET Image Process., Vol. 5, Issue 7, pp. 598 -610, 2011.
- [26] D. Z. Feng, Y. Z. Ping, and X. Y. Lun, "High probability impulse noise removing algorithm based on mathematical morphology," IEEE Signal Process. Lett. , Vol. 14, no. 1, pp. 31-34, Jan. 2007.
- [27] X. M. Zhang, Z.P. Yin and Y.L.Xiong, "Adaptive switching mean filter using conditional morphological noise detector," IEEE Electron Lett. , Vol. 44, no. 06, March 2008.
- [28] W.Luo, "An efficient algorithm for the removal of impulse noise from corrupted images," IEEE Trans. On Consumer Electronics, Vol 61, Issue 8, pp. 551-553, Sept. 2007.
- [29] J.-S. R. Jang, C. -T. Sun, E. Mizutani, Neuro -Fuzzy and Soft Computing, PHI 2002.
- [30] R. Pandey and Ghanekar, "Fuzzy filtering algorithms for image processing: Performance evaluation of various approaches," proceedings of the International conference on Cognition and Recognition, PES college of Engg. Mandya, Karnataka, India, pp. 95-101, 2005.
- [31] M.E. Yuksel, "A simple neuro-fuzzy method for improving the performance of impulse noise filters for digital images," Int. J. Electron. Commun. (AEU), Vol. 59, Issue 8, pp. 463-472, 2006.
- [32] M.E. Yuksel, "A hybrid neuro-fuzzy filter for edge preserving restoration of image corrupted by impulse noise," IEEE Trans. Image Processing, Vol. 15, no. 4, pp. 928-936, April 2006.
- [33] M.E. Yuksel, and E. Besdok, "A simple neuro fuzzy impulse detection for efficient blur reduction of impulse noise removal operators for digital image," IEEE Trans. Fuzzy. Systems Vol. 12, no.6, pp. 854-856, Dec. 2004.
- [34] Hsien- Hsin Chou et. al., "A sparsity-ranking edge -preservation filter for removal of high-density impulse noise", International Journals of Electronics and Communications (AEU), Vol. 68, Issue. 11, pages 1129-1135, Nov-2014.
- [35] Hsien- Hsin Chou et.al, "A noise-ranking switching filter for images with general fixed-valued impulse noise", Signal Processing, Vol. 106, pages 198-208, Jan 2015.



Aaditya Sharma completed B.TECH (Hons.) in Computer Science & Engineering from Gautam Buddha Technical University Lucknow, India in 2012. He is pursuing M.TECH in Advance Computing from Maulana Azad National Institute of Technology, Bhopal, India. His area of research includes image processing and parallel algorithms.



Dr. R. K. Pateriya, an Associate professor in the Department of Computer Science & Engineering, at Maulana Azad National Institute of Technology, Bhopal, India. He is a member of the IEEE. His current research interests include image processing, cloud and Distributed computing, E-commerce, security etc. He has authored and published over 100 research papers in national and international journals.