Volume 2, Issue 3, March 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Introduction of Image Restoration using Fuzzy Filtering

Sandeep Mehan*
Research Scholar,RIEIT, Railmajra (Punjab), India
sandeep_3714@hotmail.com

Mrs.Neeru Singla
Asst. Professor,RIEIT, Railmajra (Punjab), India

Abstract— ***Image restoration or reduction of noise in image is very active research area in image processing. In this approach a fuzzy filter is described which is used restore the images corrupted with impulse noise. This paper consists of two modules detection module and filtering module to remove impulse noise from corrupted images. Both modules use fuzzy rules to determine whether a pixel is corrupted with impulse noise or not. Then after this fuzzy filtering technique focuses on only those pixels which are detected by fuzzy detection module. So the filtering module is concentrated on the real impulse noise pixels.***

Keywords— ***Image processing, fuzzy filter, membership functions.***

I. INTRODUCTION

One of the most important stages in image processing applications is removing of noise i.e. image restoration. Image restoration is used as a preprocessing stage in many applications including image encoding, pattern recognition, image compression. Images can become corrupted during any of the acquisition, preprocessing, compression, transmission phases of processing. Images are often corrupted with impulse noise due to noisy sensor or channel transmission error. Fuzzy techniques have already been applied in several fields of image processing e.g. filtering, interpolation and morphology. A number of non linear approaches have been already developed for impulse noise removal, for example the well known fuzzy inference rule by else-action filter (FIRE), weighted fuzzy mean filter, iterative fuzzy control based filter, histogram adaptive fuzzy filter (HAF), adaptive fuzzy switching filter (AFSF) etc. These fuzzy filters are mainly used for images corrupted with fat tailed noise like impulse noise. In this paper we have new fuzzy filter which consists of two different modules: detection and filtering module.

The detection module consists of two fuzzy algorithms, which are complementary to each other and which are then combined together to get most robust detection which will determine whether a pixel is corrupted with impulse noise or not.

The filtering module focuses only on the pixels detected by detection module. So the filtering is concentrated on the real impulse noise pixels.

II. DETECTION MODULE

This detection module is composed of two subunits that are used to define corrupted impulse noise pixels. The first subunit investigates the neighborhood around a pixel to conclude if the pixel can be considered as impulse noise or not. The second subunit uses fuzzy gradient values to determine the degree in which a pixel can be considered as impulse noise and the degree in which a pixel can be considered as noise free.

A. First Detection Unit

The first detection unit is used to determine if a certain image pixel A (i, j) is corrupted with impulse noise or not where A is the original image. Then we calculate the mean differences in the window denoted as g(i, j):

$$g(i,j) = \frac{\sum_{k=-K}^{K} \sum_{l=-K}^{K} \left| A(i-k, j-l) - A(i,j) \right|}{(2K+1)^2 - 1}$$

Pixels corrupted with impulse noise cause large g(i, j) values, because impulse noise pixels normally occur as outliners in a small neighborhood (consider K = 1) around the pixel. The g(i, j) value could be relatively large in case of an edge pixel. So we have considered the following two values denoted as obs1(i, j) and obs2(i, j):

$$obs1(i,j) = \frac{\sum_{k=-K}^{K} \sum_{l=-K}^{K} g(i+k, j+l)}{(2K+1)^2}$$

$$obs2(i,j) = g(i,j)$$

If both values (obs1(i, j) and obs2(i, j)) are large, then the pixel can be considered as an edge pixel instead of a noisy one. So when the two values (obs1(i, j) and obs2(i, j)) are very similar we conclude that the pixel is noise free. Otherwise, if the difference between obs1(i, j) and obs2(i, j) is large then we consider the pixel as noisy. This can be implemented by the following fuzzy rule:

Fuzzy Rule 1: Defining when a central pixel A(i, j) is corrupted with impulse noise:
IF |*obs1(i, j) − obs2(i, j)*| *is large*
THEN *the central pixel A(i, j) is an impulse noise pixel.*

In this rule, large can be represented as a fuzzy set. A fuzzy set in turn can be represented by a membership function. An example of a membership function LARGE ,which is entitled as large, is pictured in Fig. 1. If the difference |obs1(i, j) − obs2(i, j)| for example has a membership degree one (or zero) in the fuzzy set large, it means that this difference is considered as large (or not large) for sure. Membership degrees between zero and one indicate that we do not know for sure if such difference is large or not, so that the difference is large to a certain degree. In Fig.1 we see that we have to determine two important parameters a and b. The parameter a is equal to the lowest g(i + k, j + l) value in the (2K + 1) × (2K + 1) window around the central pixel, i.e.

a(i, j) = $\min_{k,l \in \{-K.....+K\}} (g(i+k, j+l))$

So, a(i, j) corresponds to the g(i +k, j +l) coming from the most homogeneous region around A(i, j), which should correspond to the region with the smallest amount of impulse noise pixels. The parameter b(i, j) is b(i, j) = a(i, j) + 0.2 a(i, j), i.e. the larger the parameter a, the larger the uncertainty interval [a, b] should be.

Membership functions

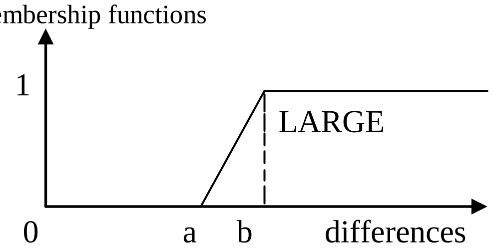

Fig.1 The membership function LARGE denoted as $\mu large$

Where the membership degree μimpulse, i.e. μimpulse (A(i, j)) = μlarge(|obs1(i, j) − obs2(i, j |))

B. Second Detection Unit

This unit is used to calculate the degree in which a certain pixel A (i, j) can be considered as impulse noise. Both detection units are complementary to each other, i.e. by combining them we receive a more robust detection method. Each of the eight neighbors of A(i, j) corresponds to one direction {North West (NW), North (N), North East (NE), East (E), South East (SE), South (S), South West (SW), West (W)}. Next, we define the gradient value $\nabla_D A$ (i, j) of pixel position (i, j) in direction D, which corresponds to a certain position as $\nabla_{(k,l)}$ A(i, j) = A(i + k, j + l) − A(i, j) with k, l ∈ {−K, . . . , K}

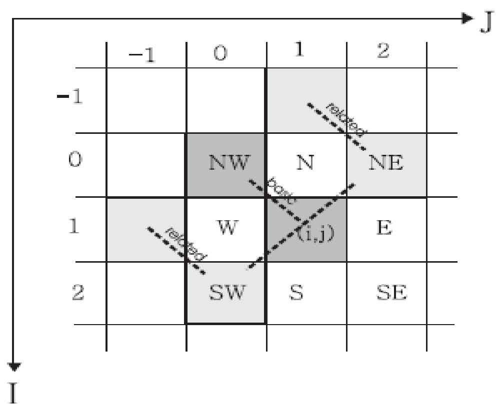

Fig. 2 Involved centers for the calculation of the related gradient values in the NW-direction

Where (k, l) corresponds to one of the eight directions and (i, j) is centre of the gradient. The eight gradient values are called the basic gradient values. There are two cases where large gradient values occur: (1) if one of the two pixels is corrupted with impulse noise or (2) when an edge is presented. Here we only want to detect the first case and therefore we use not only one basic gradient value for each direction but also two related gradient values. These two related gradient values in the same direction as the basic gradient are determined by the centres making a right-angle with the direction of the corresponding basic gradient in fig. 2. For the NW-direction (i.e. for (k, l) (−1,−1)). The basic gradient and the two related gradient values at position (i, j) are defined as $\nabla_{(-1,-1)}$ A(i, j), $\nabla_{(-1,-1)}$ A(i −1, j +1) and $\nabla_{(-1,-1)}$ A(i + 1, j − 1), respectively. In Table 1 we give an overview of the involved gradient values: each direction D (column 1) corresponds to a position w.r.t. a central position. Column 2 states the basic gradient for each direction, column 3 lists the two related gradients.

Table 1 Involved gradient values to calculator the fuzzy gradient

D	Basic gradient	Related gradient
NW	$\nabla_{NW}A(i,j)$	$\nabla_{NW}A(i+1,j-1),\nabla_{NW}A(i-1,j+1)$
N	$\nabla_{N}A(i,j)$	$\nabla_{N}A(i,j-1),\nabla_{N}A(i,j+1)$
NE	$\nabla_{NE}A(i,j)$	$\nabla_{NE}A(i-1,j-1),\nabla_{NE}A(i+1,j+1)$
E	$\nabla_{E}A(i,j)$	$\nabla_{E}A(i-1,j),\nabla_{E}A(i+1,j)$
SE	$\nabla_{SE}A(i,j)$	$\nabla_{SE}A(i-1,j+1),\nabla_{SE}A(i+1,j-1)$
S	$\nabla_{S}A(i,j)$	$\nabla_{S}A(i,j-1),\nabla_{S}A(i,j+1)$
SW	$\nabla_{SW}A(i,j)$	$\nabla_{SW}A(i-1,j-1),\nabla_{SW}A(i+1,j+1)$
W	$\nabla_{W}A(i,j)$	$\nabla_{W}A(i-1,j),\nabla_{W}A(i+1,j)$

For each direction we will finally calculate a membership degree in the fuzzy set impulse noise (denoted as $\gamma^{D}_{impulse}$ for direction D) and the membership degree in the fuzzy set impulse noise free (denoted as $\gamma^{D}_{impulsefree}$ for direction D). This is realized by the following Fuzzy Rules 2 and 3.

Fuzzy Rule 2: Defining when a central pixel A(i, j) is corrupted with impulse noise for a certain direction D:

IF (| $\nabla_D A$ (i, j)| is **not** large) AND (| $\nabla'_D A$ (i, j)| is large) AND (| $\nabla''_D A$ (i, j)| is large

OR

IF (| $\nabla_D A$ (i, j)| is large) AND (| $\nabla'_D A$ (i, j)| is **not** large) AND (| $\nabla''_D A$ (i, j)| is **not** large

OR

IF (| $\nabla_D A$ (i, j)| is large) AND (| $\nabla'_D A$ (i, j)| is large) AND (| $\nabla''_D A$ (i, j)| is **not** large

OR

IF (| $\nabla_D A$ (i, j)| is large) AND (| $\nabla'_D A$ (i, j)| is **not** large) AND (| $\nabla''_D A$ (i, j)| is large

THEN *the central pixel A(i, j) is an impulse noise pixel in direction D*

Fuzzy Rule 3: Defining when a central pixel A (i, j) is not corrupted with impulse noise for a certain direction D:

IF (| $\nabla_D A$ (i, j)| is large) AND (| $\nabla'_D A$ (i, j)| is large) AND (| $\nabla''_D A$ (i, j)| is large

OR

IF (| $\nabla_D A$ (i, j)| is not large) AND (| $\nabla'_D A$ (i, j)| is not large) AND (| $\nabla''_D A$ (i, j)| is not large

THEN *the central pixel A(i, j) is impulse noise free in direction D*

We have denoted the basic gradient as $\nabla_D A$ (i, j), while the two related gradient values were entitled as $\nabla'_D A$ (i, j) and $\nabla''_D A$ (i, j), respectively. "IF (| $\nabla_D A$ (i, j)| is not large) AND $\nabla'_D A$ (i, j)| is large) AND (| $\nabla''_D A$ (i, j)| is large'' is calculated by (1−μlarge(| $\nabla_D A$ (i, j)|)) · μlarge(| $\nabla'_D A$ (i, j)|) · μlarge(| $\nabla''_D A$ (i, j)|), where we used the "product'' triangular norm and where μlarge has the same graph as in Fig.1 using the following parameters a and b:

$$a(i,j)=\frac{\sum_{k=-K}^{K}\sum_{l=-K}^{K} g(i+k,j+l)-g(i,j)}{(2K+1)^2-1},$$

b(i, j) = a(i, j) + a(i, j) *0.2

So, a(i, j) corresponds to the average of the g(i + k, j + l) values. By using these parameters we have managed the incorporation of regions containing edges, because these non-homogeneous regions will cause higher parameters so that our detection method is adapted to such situations. Gradient values in non-homogeneous regions are labeled as large if they are large in comparison to their neighbors. In homogeneous regions we will get smaller values of a(i, j) and b(i, j), which will cause a much stronger detection method. The outputs of the second detection unit are the eight membership degrees in the fuzzy set impulse noise for the eight directions around a certain position (i, j), i.e. the degrees $\gamma^D_{impulse}$A (i, j) and the eight membership degrees in the fuzzy set impulse noise free for the eight directions around a certain position (i, j) , i.e. the degrees $\gamma^D_{impulsefree} A(i, j)$ The degree $\gamma^D_{impulse}$A(i, j) and $\gamma^D_{impulsefree}$ A(i, j) are calculated using Fuzzy Rule 2 and 3, respectively.

III. FILTERING MODULE

In this module we will apply the filtering method on pixels that are determined as noisy in both detection units. Pixels having a non-zero membership degree in the fuzzy set impulse noise for the first detection unit are considered as noisy, i.e. $\mu_{impulse}$ A(i, j)) > 0. In the second detection unit we consider two fuzzy sets namely impulse noise and impulse noise free in order to decide if a pixel is considered as noisy or not. Here we have decided that if $\sum_{D\in\{N,\ldots S\}} \gamma^D_{impulse}$A(i,j)≥ $\sum_{D\in\{N,\ldots S\}} \gamma^D_{impulsefree}$ A(i,j) then impulse noise is considered at A(i, j). So, the filtering method will be applied to pixels where both restrictions are satisfied, i.e.

$$\mu_{impulse}\ (A(i, j)) > 0,$$

$$\sum_{D\in\{N,\ldots<S\}} \gamma^D_{impulse}A(i,j) \geq \sum_{D\in\{N,\ldots<S\}} \gamma^D_{impulsefree}A(i,j)$$

The output of the filtering method for the input pixel A(i, j) is denoted as F(i, j) and is calculated as follows:

$$F(i,j) = (1-\lambda(i,j))\frac{\sum_{k=-L}^{L}\sum_{l=-L}^{L} A(i+k,j+l)w(i+k,j+l)}{\sum_{k=-L}^{L}\sum_{l=-L}^{L} w(i+l,j+l)} + \lambda(i,j)A(i,j)$$

The filtering method uses a (2L+1)×(2L+1) (not necessary equal to K) neighbourhood around A(i, j). Each A(i +k, j +l) is multiplied by a corresponding weight w(i +k, j +l) indicating in which degree the pixel should be used to filter the central pixel. The parameter λ (i, j) is finally used to control the amount of correction.

To determine the weights, we have used the pixels from the (2K + 1) × (2K + 1) neighborhood around A(i, j).Those pixels are then sorted so that the window is denoted as [$x_1, x_2, \ldots\ldots, x_{(2K+1)^2}$] with x_1 and $x_{(2K+1)^2}$ the lowest and highest intensity value from the corresponding window, respectively.

We have constructed a fuzzy set called similar to calculate the corresponding weights w(i+k, j+l). This fuzzy set is represented by the membership function SIMILAR, denoted as μsim. When the (2K + 1) × (2K + 1) neighbourhood is very homogeneous we know that median based algorithms are working very well, but when we have non-homogeneous neighbourhoods it is better to incorporate the knowledge of the other pixels as well to improve the filtering performance. The membership value indicates in which degree a certain intensity value can be observed as similar to the observed neighbourhood. Pixels having a degree of one (zero) are (not) similar for sure to the corresponding neighbourhood. So the weight w (i +k, j +l) is defined as w(i +k, j +l) = µsim(A(i + k, j + l)).

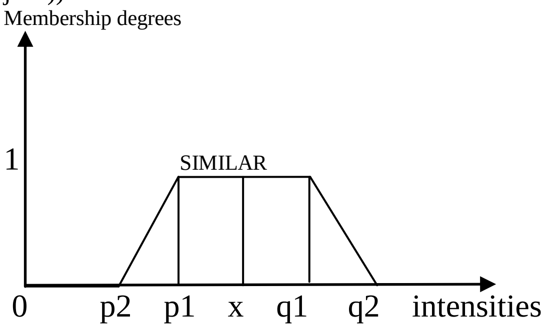

Fig.3 The membership function SIMILAR denoted as *µsim*

The membership function is determined by means of four parameters. To define these parameters (p1, p2, q1 and q2) we first have calculated the mean differences ρ , i.e.

$$\rho\ (i, j) = \frac{\sum_{k=2}^{(2K+1)^2} (x_k - x_{k-1})}{(2K+1)^2 - 1}$$

Using this (i, j) we have determined the parameters as follows:

$$x(i, j) = \underbrace{median}_{k,l\in\{-K,\ldots\ldots,+K\}} (A(i+k, j+l))$$

$p1(i, j) = x(i, j) - \rho\ (i, j)$,

$p2(i, j) = x(i, j) - 1.1\,\rho\ (i, j)$,

$q1(i, j) = x(i, j) + \rho\ (i, j)$,

$q2(i, j) = x(i, j) + 1.1\,\rho\ (i, j)$

Homogeneous regions will have very similar pixels, which will cause very low (i,j) values, because the differences between the pixel intensity values will be low. In these situations we will receive parameters (p1(i, j), p2(i, j), q1(i, j) and q2(i, j)) that are very close to the median and therefore the filtering performance will be similar to median based filters. On the other hand we will get an improvement in cases of non homogeneous windows, because we still incorporate the information of all the neighbouring pixels around A(i, j) as well. Finally, we will reduce the correction process for the central pixels A(i, j) having a very high weight w(i, j) because if the corresponding weight is very high, it means that these pixels seem to be similar to their neighbourhood unless the detection. Therefore we define the parameter λ (i, j) to be equal to the weight of the centre, i.e. λ (i, j) = w(i, j).

IV. SIMULATION RESULTS

This filter is applied to 256*256 Lena image (8-bit, L=255), after adding impulse noise of different levels i.e. 10%-50%. As a measure of objective dissimilarity between a filtered image and the original one, we use the MSE and PSNR defined by

$$MSE(F,O) = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}[O(i,j) - F(i,j)]^2,$$

$$PSNR(F,O) = 10\log_{10}\frac{S^2}{MSE(F,O)}$$

Where O is the original image, F is the restored image of size NM and S is the maximum possible intensity value.

Fig. 4 Process of Restoration of noisy Lena image (256*256) after 30% of impulse noise (salt and pepper noise).

V. CONCLUSION

This filter consists of two fuzzy detection methods and fuzzy filtering algorithm which is used to reduce the noise from the image corrupted with impulse noise. The main feature of this method is that it leaves the pixels unchanged which are noise free and preserves the some amount of thin lines. finally ,this filter is very easy to implement and has a very low execution time.

REFERENCES

1) Stefan Schulte, Valerie De Witte, Mike Nachtegael, Dietrich Van Des Wekan and Etienne E. Kerre "Fuzzy random impulse noise reduction method" Fuzzy sets and systems, pages: 270-283, 2007

2) Hamed Vahdat Nejad, Hameed Reza Pourreza and Hasan Ebrehimi "A Noval fuzzy technique for image noise reduction" World academy of science, engg. and technology, 2006

3) Dimitri Van De Ville, Dietrich Van Des Wekan and Wilfried Philips "Noise reduction by fuzzy image filtering" IEEE transactions in fuzzy systems, Vol.11, pages:429-436, 2007
4) Ayyaz Hussain, Sihail Masood Bhatti and M. Arfan Jaffar "Multimedia tools applications DOI 10.1007/s11042-011-0829-7" Springerlink.com, 2007
5) Mike Nachtegael, E.E. Kerre "Decomposing and construting fuzzy morphology operations over alpha-cuts: continuous and discrete case" IEEE Transactions fuzzy system, pages:615-626, 2000
6) K. Arakawa, E.E. Kerre and M. Nachtegael "Fuzzy rule based image processing with optimization" Fuzzy technique in image processing, springer, pages:222-247, 2000
7) Aneesh Agrawal, Abha Choubey and Kapil Kumar Nagwanshi "Development of adaptive fuzzy based image filtering techniques for efficient noise reduction in medical images" International journal of computer science and information technologies Vol.2(4), pages: 1457-1461, 2008
8) Nidhi Khrab and Amanpreet Kaur "Analysis of advanced fuzzy filters for image denoising" International journal of advances in electronics engg. Pages:178-181
9) Mahesh T R, Prabhanjan S and Vinaybabu "Noise reduction using fuzzy image filtering" journal of theoretical and applied information technology,Pakistan,Vol.15(2), pages:115-120, 2007
10) Ayyaz Hussain, M. Arfan Jaffar, Zia Ul- Qayyum and Anwar M.Mirza "Directional weighted median based fuzzy filter for Random-valued impulse noise removal" ICIC international Express leters, pages:2185-2766, 2007
11) K. Ratna Babu, Dr. K.V.N. Sunitha "A new fuzzy Gaussian noise removal method for gray-scale images" international journal of computer science and information technology, Vol.2(1), pages: 504-511, 2007
12) Jagdish H. Pujar "Robust fuzzy median filter for impulse noise reduction of gray scale images" World academy of science, engg. and technology, pages:630-634, 2007
13) Mahmoud Saeidi, Khadijeh Saeidi and Mahmoud Khaleghi "Noise reduction in image sequences using an effective fuzzy algorithm "World academy of science, Engg. and technology, pages:351-355, 2007
14) H.Xu, G.Zhu, D.Wang and H.Peng "adaptive fuzzy switching filter for images corrupted by impulse noise" Pattern recognition Vol.25, pages: 1657-1663, 2004
15) F.Russo "Fire operators for image processing" Fuzzy sets and system Vol.103(2), pages: 265-275, 1999
16) F.Russo and G. Ramponi "A noise smoother using fire filter" IEEE conference on fuzzy systems" pages: 351-358,1995
17) F.Russo and G. Ramponi "A fuzzy filter for images corrupted by inputs noise" IEEE signal process, pages: 168-170, 1996

