

Optimal fuzzy system design for additive noise removal in color images

Snekha¹, Bandaru Vasu Mani Kumar²

¹(ECE Department, Delhi Technological University, India)

²(ECE Department, Delhi Technological University, India)

Abstract: This paper presents a new approach to remove the additive noise using fuzzy logic and modified bacterial foraging in color images. In the first step the bacterial foraging technique is used to find the edges using a set of bacteria's that are randomly initialized to image pixel then they will try to search their nutrients which are edges in this case. The second step is to remove the gaussian and salt & pepper noise denoising we will leave edge pixel detected in step 1 as it is and rest of pixels in image will be candidate pixels for Denoising. We will use pixel similarity based concept. The Gaussian noise will be removed by using weighted mean filtering of peer group members. We tested the proposed method on color images that are corrupted with mixed noise both Gaussian and Salt & Pepper noise.

Keywords: BFOA, Noise Removal, Fuzzy Way

I. Introduction

Noise Filtering is a fundamental pre-processing step before further image processing techniques like image segmentation, image compression, and texture analysis can be performed. Noise often occurs in Images during Image acquisition, transmission/ reception and image storage or retrieval processes which affects the visual quality of image and results in poor and unpleasant visual image. So it is essential to remove noise. Linear filtering techniques like averaging filter, weighted mean filter available for image de-noising tend to blur the edges.[1] In Images, Edges contains essential information. Edges are the sharp features. Generally noises like Gaussian and salt & pepper are very common. Gaussian noise affects every pixel of the image. On the other hand, salt & pepper noise does not affect every pixel. Only the affected pixels intensity gets 0(pepper) or 255(salt).

Van De Ville et al. [6], the effective fuzzy derivatives are used for differentiating the noise and edge pixels in images corrupted with Gaussian noise. In Russo [7], FIRE (Fuzzy inference ruled by else sections) operators are used. Tuan-Anh Nguyen et al. [8] proposed spatially adaptive De-noising algorithm. This algorithm is consists of two stages; first noise detection and then noise removal filtering. Local weighted mean, local maximum and local weighted activity is defined to use local statistics of the image into De-noising process. Depending on local statistics, constraint for noise detection is specified. K. Rank et al. [9] proposed an adaptive 2-D recursive low-pass filter with some coefficients for restoring images corrupted with Gaussian Noise.

O.P Verma et al. [10], proposed an efficient fuzzy filter for edge preservation for color images. The algorithm is able to reduce salt & pepper noise with satisfactory results. Kenny Kal Vin Toh et al. [11] proposed a "Cluster-Based Adaptive Fuzzy Switching Median Filter for Universal Impulse Noise Reduction". In this filter, an easy to implement impulse detector and a detail preserving noise filter are cascaded.

In this proposed approach in the first step we are detecting the edges using bacteria foraging and in the second stage using the results from the first stage we are removing the noise using fuzzy rules by ignoring the edges. The paper is organized as follows. In Section 2, First bacterium foraging algorithm is discussed. Then an algorithm to calculate edge map using modified bacterium foraging algorithm is discussed after that a 2- stage filter to de-noise a colour image corrupted with Gaussian and Salt & pepper noise. In Section 3, we explain our proposed method for determining the components of interest. Section 4 describes the details of the simulations. Finally, in Section 5, we present the results of applying the proposed method.

II. Bacterial Foraging Algorithm

BFA is an optimization method developed by the (Passino, 2000; Passino, 2002)[2] based on the Foraging strategy of E. coli bacteria, which live in the human Intestine. Foraging means- The way of searching for food and provisions or we can generalize it foraging is an activity of looking thoroughly in order to find something. Foraging strategy is a method of animals for locating, handling and ingesting their food. It can be modeled as an optimization process, where bacteria search for and obtain nutrients in a way that maximize their energy E obtained per unit time spent T during foraging. In this process, the nutrient function is defined and each bacterium tries to maximize the amount of nutrient while minimizing time

spent T and energy cost E by following four stages: 1) Chemo taxis, 2) Swarming, 3) Reproduction, and 4) Elimination & Dispersal. In the beginning, there are as many solutions as the number of bacteria. So, each bacterium produces a solution for set of optimal values of parameters iteratively, and gradually all the bacteria converge on the global optimum.

II a. Presented method for edge detection

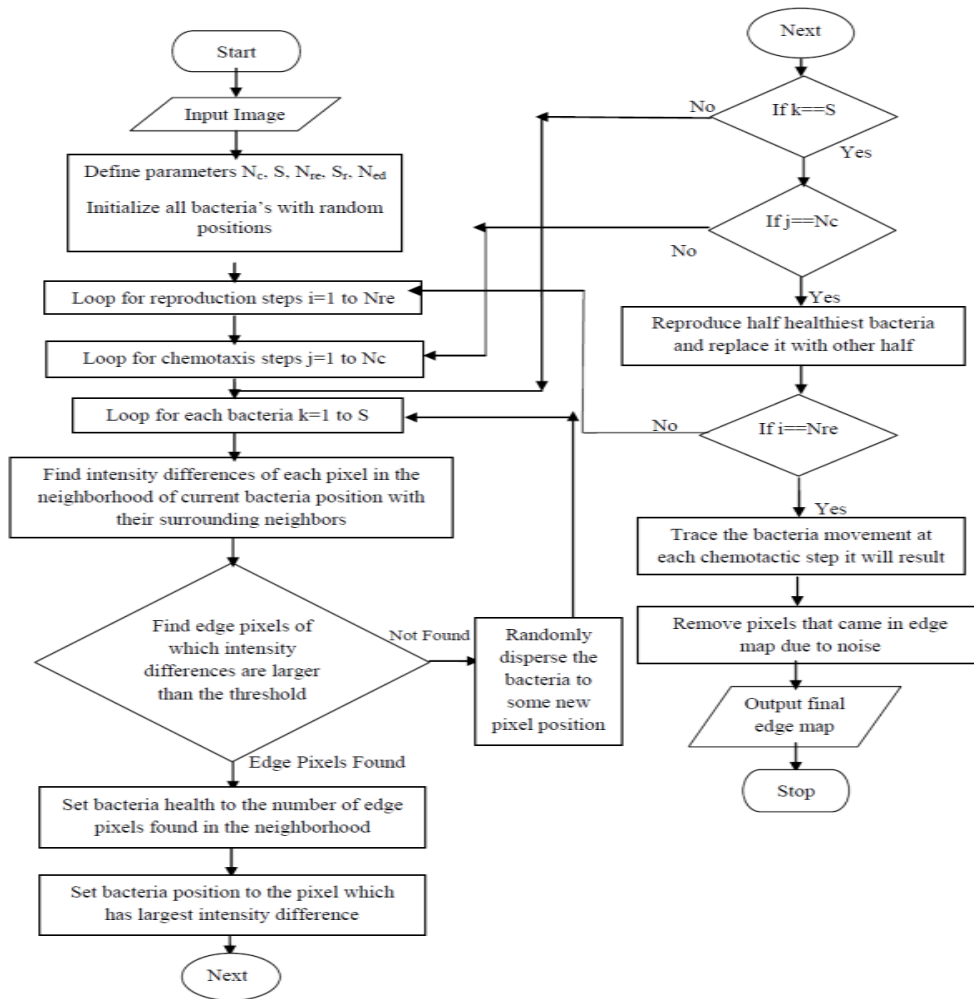


Fig.1 Flow chart of proposed edge detection method

Edge detection is the process of identifying and then locating sharp discontinuities or sharp changes in brightness in an image. The discontinuities are random changes in pixel intensity which characterize objects boundaries in a scene.

Bacterial Foraging Algorithm [Passino 2002], briefly described above is modified here to extract edges from noisy image. In this modified bacterial foraging algorithm, a set of bacteria's are randomly initialized to image pixel then they will try to search their nutrients which are edges in this case. This whole process will be carried out in 3 steps: Chemotaxis, Reproduction and elimination & dispersal step. Swarming step is avoided to make the algorithm simpler and to reduce the complexity of the algorithm. Following are the steps to be followed to extract edges using modified bacterial foraging algorithm.

Search Space

First we need to define the search space where the bacteria will forage. In this case, since we are searching for edges in image so whole image is the search space. Image is 2-dimensional search space consisting of x-y coordinates of pixels. Image size is limited by its x y dimension so the search space is finite in our case.

Defining the nutrient function

Now we need to define, what will be the nutrients for the bacteria. So by definition, edges are sharp changes in image brightness. So the nutrient function is based on the difference in intensities of a bacteria placed at a pixel position with respect to pixels in the surroundings. Here we have considered neighbourhood pixels which are coming in window size of 3*3. Each bacterium will try to find out edges in these neighbourhood pixels depending on the difference of those with their surrounding pixels.

Criteria for choosing the edge pixel

For a pixel to be considered as edge pixel, it should have average intensity difference greater than a threshold value with its surrounding pixels that are 8- neighborhoods in our case. Intensity difference will be taken as component wise difference in color images. This is explained with the following example:

P1	P2	P3
P4	P	P5
P6	P7	P8

A pixel with its neighborhood pixel in window size 3*3. Suppose we want to find whether pixel P with coordinates (x, is edge pixel or not, we will calculate component wise intensity difference with its neighborhood pixels in following way. These component wise differences will be added say in total variable. Now if this total is greater than a threshold value, then the pixel in interest that is P will be called as edge pixel otherwise not.

Choosing the threshold value

The threshold value is being used to differentiate edge pixels from non-edge pixels. It decides number of edge pixels to be detected. In case of high deviation of Gaussian Noise, a high threshold value is used. Following table is the optimal suggested setting to choose proper threshold value:

Standard Deviation	Threshold Value
Low in [0,10]	20
Medium [10,20]	35
High [20,30]	45

Table 1 for Threshold Value according to amount of Standard Deviation

III. Image noise removal using fuzzy way

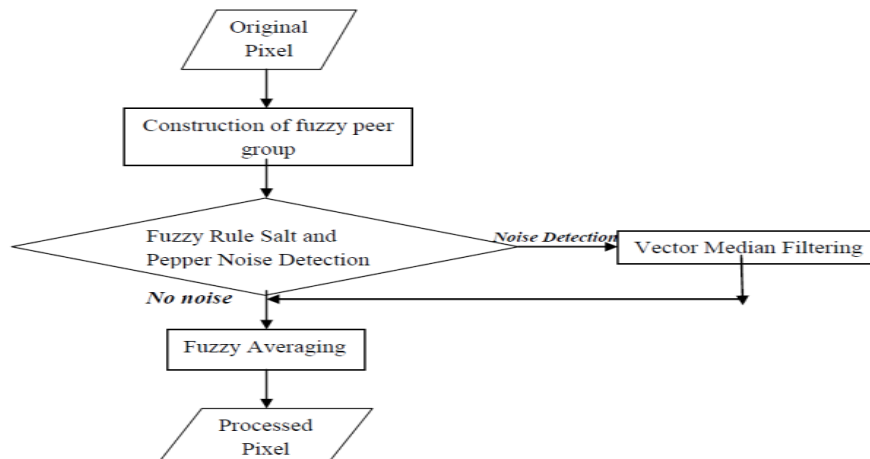


Fig.2 Flow chart of the proposed Filter

Now we will denoise the color image which is corrupted with additive noise i.e. Gaussian noise and salt & pepper noise. In Denoising we will leave edge pixel detected in step 1 as it is and rest of pixels in image will be candidate pixels for Denoising. We will be used to decide that the current pixel is corrupted

with salt and pepper noise or not. If it corrupted then using peer group members filtering will be applied. Then the Gaussian noise will be removed by using weighted mean filtering of peer group members. This algorithm is able to remove mixed noise with satisfactory results.

III a. Peer Group

The peer group of a given pixel is a set constituted by this pixel and those of its neighbors which are similar to it. The cardinality of peer group set will be used to decide the pixel in interest is free of noise or not. There are basically two ways to determine this peer group set i.e. Crisp way and Fuzzy way. Here we used fuzzy way. If some fuzzy metrics are used to define Peer Group set then it is called Fuzzy Peer Group. We will define membership of each pixel $P_i \in W$ in fuzzy way.

Construction of Fuzzy Peer Group

- We use a **fuzzy similarity function M** defined as:

$$M(P_0, P_i) = e^{-\|P_0 - P_i\|/F_q} \quad (1)$$

where $\| \cdot \|$ denotes Euclidean norm and $F_q > 0$ is a parameter. M will take values in [0, 1]. $M(P_0, P_i) = 1$ if and only if $P_i = P_0$. After calculating M, the color vectors or pixels $P_i \in W$ are sorted in descending order with respect to its similarity to P_0 which results in ordered set W' .

- Now we will define a **fuzzy set C^P_0** considering the proposition that P_i is similar to P_0 on the pixels P_i present in the window W' where membership function is given by:

$$\mu(C^P_0(P_i)) = M(P_0, P_i) \text{ for } i=0,1,2,\dots,n^2-1 \quad (2)$$

- Now we will define **accumulated similarity AS** for each pixel P_i .

$$AS^P_0(P_i) = \sum_{k=0:i} M(P_0, P_k), i=0,1,2,\dots,n^2 \quad (3)$$

Clearly $AS^P_0(P_0) = 1$ and $AS^P_0(P_i)$ takes $i+1$ if $P_0 = P_1 = \dots = P_i$. Largest value AS^P_0 can take is n^2

- Now we will create a **fuzzy set $L^P_0(P_i)$** by considering the vague proposition that $AS^P_0(P_i)$ is large. Clearly the memberships value of minimum value that $AS^P_0(P_0) = 0$ will be zero. Since it is the minimum value that AS^P_0 can take and memberships value of maximum value that is $AS^P_0(P_8) = 9$ will be 1. So the membership function for the fuzzy set $L^P_0(P_i)$ is $\mu(AS^P_0(P_i))$ is defined as follows:

$$L^P_0(P_i) = \mu(AS^P_0(P_i)) = \frac{AS^P_0(P_i) - 1}{AS^P_0(P_8) - 1} \text{ where } i=0, 1, 2, \dots, n^2-1 \quad (4)$$

- Now we will select **best number m** of members among window pixels W' . We aim to determine m number of members such that fuzzy peer group is the largest set that contain only similar pixels. The best number m where m can be 0, 1, 2, ..., n^2-1 is chosen such that it maximizes the certainty of following fuzzy rule.

Fuzzy Rule 1: If “ P_m is similar to P_0 ” and “ $AS^P_0(P_m)$ is large” then “the certainty of m to be the best number of members is high”.

“ P_m is similar to P_0 ” is given by fuzzy set $C^P_0(P_m)$, “ $AS^P_0(P_m)$ is large” is given by $L^P_0(P_m)$.

The certainty of Fuzzy rule 1 is computed for each value of m and the value which maximizes this certainty is selected as the best number m of members for peer group on window W'

We use the product t-norm as the conjunction operator, so no de-fuzzification is needed.

$$CFR1(m) = C^P_0(P_m) L^P_0(P_m) \quad (5)$$

So m is chosen such that $CFR1(m)$ is maximum. Now we will define fuzzy peer group $FP^P_0(m)$ with first m members from W' and their membership function will be defined by $M(P_0, P_i)$.

Filtering with help of fuzzy peer group:

This filter works in 2-steps.

- Salt and pepper noise detection and correction
- Gaussian noise smoothing

To reduce Salt and Pepper Noise, a fuzzy rule based procedure is used. First with the help of fuzzy rule based procedure, it is decided that the pixel in interest is corrupted with Salt & pepper noise or not. A threshold value is used to detect corrupted pixel. If a pixel is found to be corrupted, then Vector median filtering is applied to correct those noisy pixels. For Gaussian noise smoothing, fuzzy averaging is done among the members of fuzzy

peer group of the pixel under processing. In Fuzzy averaging, weights are given by the membership function which defines the similarity or how strong they are similar to the peer group of the pixel in interest.

III b.Salt and pepper noise detection and correction

A salt and pepper noise pixel can be defined as a pixel which is significantly different in intensity from its neighborhood pixels. Conversely, a salt and pepper noise-free pixel should have some neighbors quite similar to it. According to the above, we can model this condition in terms of fuzzy peer groups as follows:

A pixel P_0 is free of salt and pepper noise if for the fuzzy peer group $FP_0^P(m)$ it is satisfied that $AS_0^P(P_m)$ is large and P_m is similar to P_0 . Determining the certainty of the pixel P_0 to be free of salt and pepper noise.

The following fuzzy rule 2 represents this condition:

Fuzzy Rule 2: If “ $AS_0^P(P_m)$ is large” and “ P_m is similar to P_0 ” then P_0 is free of salt and pepper noise. “ P_m is similar to P_0 ” is given by fuzzy set $C_0^P(P_m)$, “ $AS_0^P(P_m)$ is large” is given by fuzzy set $L_0^P(P_m)$. We use the product t-norm as the conjunction operator.

$$C_{FR2}(P_0) = L_0^P(P_m) C_0^P(P_m) \tag{6}$$

It is already calculated in C_{FR1} . So no more computations are required. We will use C_{FR2} to detect and replace salt and pepper noisy pixel according to threshold-based rule. If $C_{FR2}(P_0) \geq F_t$ then P_0 is free of salt and pepper noise. Else P_0 is a noisy pixel and replace P_0 with VMF_{out} (Vector median Filtering) where F_t is a parameter.

Gaussian Noise smoothing procedure:

To remove Gaussian noise, a weighted averaging operation is performed among color vectors. So, to smooth the pixel P_0 , we use the members of fuzzy peer group $FP_0^P(m)$, where the weighting coefficient for each color vector is its membership degree to the fuzzy peer group $FP_0^P(m)$. Every pixel is smoothed by weighted averaging operation in following manner:

$$P_{out} = \frac{\sum_{i=0}^m FP_m^{P_0}(P_i) * (P_i)}{\sum_{i=0}^m FP_m^{P_0}(P_i)} \tag{7}$$

Suggested setting for parameters used

Following are the suggestion for proper setting of parameters that are being used in the filter that is F_q and F_t .

Table 2 :Suggested setting for F_t parameter

Percentage of Salt and Pepper Noise	F_t
Low [in (0,10)]	0.05
Medium [in (10,20)]	0.15
High [in (20,30)]	0.25

Table 3: Suggested setting for F_q parameter

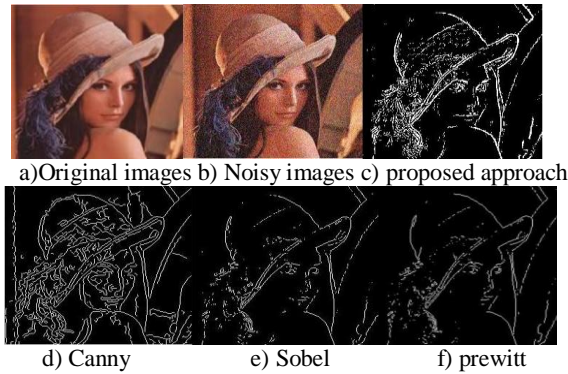
Standard Deviation of Gaussian Noise	F_q
Low [in (0,10)]	50
Medium [in (10,20)]	100
High [in (20,30)]	175

IV. Results

Table 4 represents the values of entropy measure for Majority image and for the results of various approaches: Sobel, Prewitt, Canny and proposed approach. The entropy for results of Sobel and Prewitt is comes out to be smaller than the proposed approach for all the four test images, because they provide less edge information. The canny method gives very thin edges and it does not work on the color images thus there will be information loss in the result, therefore the entropy value obtained using this methods is less than the proposed method.

Table 4: Entropy of different methods

	Canny	Sobel	Prewitt	Proposed method
Lena	2.8616	2.2515	1.6566	2.9818
Penguins	2.6962	1.6316	1.6502	2.7085
Peppers	2.9926	1.8438	1.7062	2.6610
House	2.0682	1.4315	1.4999	2.1286



The colour images of Lena Peppers of size 256 * 256 corrupted with the Gaussian and Impulse noise is considered as test images. The performance of the proposed approach is evaluated on test images with density of salt & pepper noise 10% , 15%, 20% and with standard deviation 10, 20, 30. The results of the proposed approach are compared in terms of PSNR with methods “Restoration of images corrupted by mixed Gaussian-impulse noise via 11-10minimization” (RICMG)[12], “Spatially Adaptive Denoising Algorithm for a Single Image Corrupted by Gaussian Noise” (SADA) [12]. Comparison of PSNR values resulting from applying the proposed filter and the other methods is shown in table 5. Looking at the table we can see the proposed filter is superior in case of PSNR with other method shown in the table. Now we will show denoised image of Lena and Peppers with the outputs of other filters.



a)Original Lena Image

b) Lena Image Corrupted with $\sigma =10$ and 10% salt and pepper noise



c) Output of RICMG

d)Output of SADA

e)Output of Proposed Filter

f)Original image

g) Lena Image Corrupted with $\sigma =20$ and 20% salt and pepper noise



h) Output of RICMG

i)Output of SADA

j) Output of Proposed Filter

Table 5: Comparison of PSNR of Proposed filter with other filter

Standard Deviation	Density of Salt and Pepper Noise	Noisy	RICMG	SADA	Proposed Method
Lena Image					
$\sigma = 10$	10%	19.1908	31.0342	30.2316	31.5431
$\sigma = 20$	15%	17.2428	29.1864	28.9976	29.7025
$\sigma = 30$	20%	16.2325	27.4152	27.6519	28.6229

V. Conclusions

For de-noising an image, first edge detection using modified bacterial foraging algorithm is done on noisy image. Edge map is calculated to keep image edges and sharp features intact. Experimental results showed that our calculated edge map is better in terms of quality as well as visual aspects comparing with any other technique. The entropy value of the presented method is better than any other method. In second step, we have presented a fuzzy peer group technique to denoise the image. To remove Gaussian noise fuzzy weighted averaging is done where weights are given by the membership function used to calculate pixel membership to the peer group. Experimental results showed that our method to denoise the image is better both in qualitative and quantitative measures. We are able to get a better PSNR with other techniques with good visual quality output images. The proposed filter is also able to work on high density of Gaussian and Salt & pepper noise.

References

- [1]. A.K.Jain, Fundamentals of digital image processing (Prentice Hall, Englewood Cliffs, 1989)
- [2]. Kevin M. Passino, "Biomimicry of Bacterial Foraging", IEEE Control Systems Magazine IEEE, 2002.
- [3]. K. N. Plataniotis and A. N. Venetsanopoulos, "Color Image Processing and Applications", Berlin, Germany: Springer, 2000.
- [4]. Om Prakash Verma, Madasu Hanmandlu, Anil Singh Parihar and Vamsi Krishna Madasu, "Fuzzy Filters for Noise Reduction in Color Images", ICGST-GVIPJournal, Volume 9, Issue 5, September 2009, ISSN: 1687-398X.
- [5]. Om Prakash Verma, Madasu Hanmandlu, Puneet Kumar, Sidharth Chhabra, Akhil Jindal, "A novel bacterial foraging technique for edge detection", ScienceDirect Pattern Recognition Letters 32 (2011) 1187–1196.
- [6]. D. Van de Ville, M. Nachtgael, D. Van der Weken, W. Philips, I.Lemahieu, and E. E. Kerre, "Noise reduction by fuzzy image filtering," IEEE Transactions on Image Processing, vol. 11, no. 4, pp. 429–436, Apr. 2001.
- [7]. F. Russo and G. Ramponi, "A noise smoother using cascaded FIRE filters", Proc. of 4th Intl. Conf. on Fuzzy Systems, Vol.1, pp. 351- 358, 1995.
- [8]. Tuan-Anh Nguyen, Won-Seon Song, Min-Cheol Hong, "Spatially adaptive Denoising algorithm for a single image corrupted by Gaussian noise", Consumer Electronics, IEEE Transactions on Volume:56, Issue: 3, 2010, Page(s): 1610 – 1615.
- [9]. Rank, K., Unbehauen, R., "An adaptive recursive 2-D filter for removal of Gaussian noise in images", Image Processing, IEEE Transactions on Volume: 1, Issue: 3, 1992, Page 431 – 436.
- [10]. O.P. Verma, A.S. Parihar, M. Hanmandlu, "An Edge Preserving Fuzzy Filter for Color Images", Computational Intelligence and Communication Networks (CICN), 2010 International Conference, on page(s): 122, Print ISBN: 978-1-4244-8653-3, INSPEC Accession Number: 11776685.
- [11]. Kenny Kal Vin Toh, Student Member, IEEE, and Nor Ashidi Mat Isa, Member, IEEE, "Cluster- Based Adaptive Fuzzy Switching Median Filter for Universal Impulse Noise Reduction", IEEE Transactions on Consumer Electronics, Vol. 56, No. 4, November 2010.
- [12]. Yu Xiao, Tiejong Zeng, JianYu, Michael K.Ng, "Restoration of images corrupted by mixed Gaussian-impulse noise via 11–10 minimization", Pattern Recognition 4