ISSN 2229-5518

Using Neuro Fuzzy and Genetic Algorithm for Image Denoising

Shaymaa Rashid Saleh Raidah S. Khaudeyer

Abstract—Noise does not only cause loss of image quality but it also distorts the information storing in the image and converted it into another values. So image denoising is often a necessary and the first step to be taken befor the images data is analyzed where it is not only used to improve image quality but is also used as preprocessing before most image processing operations such as encoding, recognition, tracking and etc. In other words, without this preprocessing, the other processing would have inappropriate or even false results and it is necessary to apply an efficient denoising technique to compensate for such data corruption.
This research attempts to denoise images by using hybrid filters (neuro fuzzy filter and Genetic algorithm with neuro fuzzy filter) from several types of noise such as:
1. Additive noise (Gaussian noise, Poisson noise)
2. Impulse noise (Salt and pepper noise)
3. Mixed noise (Gaussian with salt and pepper, Poisson with salt and pepper)

This work is programmed using Matlab 7.7 and for evaluating the performance of hybrid filters, PSNR (Peak Signal to Noise Ratio) is used and all the results of hybrid filters will comparing with the known image filters that used in hybridization which are mean median filters. The experimental results show that hybrid filters give the good results for all types of noise but genetic algorithm gives the best result in PSNR and architecture.

Keywords: Image denoising, Genetic algorithms and Neuro fuzzy.

1 Introduction

THE main purpose of digital image processing is to allow human beings to obtain an image of high quality or descriptive characteristics of the original image [1]. Elimination of noise is one of the major works to be done in computer vision and image processing, as noise leads to the error in the image. Noise presence is manifested by undesirable information, not related to the scene under study, which perturbs the information relative to the form observable in the image. It is translated to more or less severe values, which are added or subtracted to the original values on a number of pixels [2]. There are many techniques to remove the noise from the image and produce the clear visual of the image such as mean and median filter [3].

Fuzzy Logic is a mathematical tool for dealing with uncertainty. It provides a technique to deal with imprecision and information granularity. The fuzzy theory provides a mechanism for representing linguistic constructs such as "many," "few, "low" [4].

Artificial neural networks (ANN) are computational networks which attempt to simulate the networks of nerve cell (neurons) of the biological (human or animal) central nervous system. ANN consists of a large number of computational units or neurons. These neurons are connected with each other through *network weights* that specify the strength of the connection. Neural networks combine with fuzzy logic lead to a ***neural-fuzzy system (NFS)*** which is designed to realize the process of fuzzy reasoning, where the connection weights of the network correspond to the parameters of fuzzy reasoning. Using the backpropagation-type learning algorithms, the NFS can identify fuzzy rules and learn membership functions of the fuzzy reasoning.

Genetic Algorithms (GAs): GAs are derivative-free optimization techniques, which can evolve through procedures analogous to biological evolution: natural selection, crossover, and mutation [5].

Different filters such as ***mean*** and ***median*** filter are used to remove noise from image. Noise [6] may be:

- Additive such as Gaussian: is characterized by adding to each image pixel a value from Gaussian distribution.
- Multiplicative noise such as Speckle noise: a few pixels are noisy.
- Fixed-value impulse noise such as salt and pepper: the corrupted pixels are either set to the maximum value, which is something like a snow in image or have single bits flipped over.
- Mixed noise: occurs when images are corrupted by two or more noise.

2 Rank-Order Absolute Difference (EC-ROAD) Method

It can statistics based on Extremum Compression an impulse or effectively characterize each image pixel as either unaffected pixel [7]. This method will be used in this work to detect salt and pepper noise and algorithm (1) will be described also.

Let $x=(x_1, x_2)$ be the location of the pixel under consideration, and let Ω_x denote a r X r neighborhood centered at x the EC-ROAD will computed as follows

1- The pixel values in Ω_x, except for central pixel are rearranged in a vectorized format. Let V_x denote the sorted vector in ascending order.

2- Compress the maximum or the minimum in V_x, and only reserve one maximum or minimum (denoted by V_x^{EC}).

3- Calculate $d'(x, y) = | v(x) - V_x^{EC}(y') |$, y' is the index of reserved pixel, let $Dist(d')$ denote the sorted $d'(x, y')$ in ascending order.

4- Let $m' = Round(Dim(V_x^{EC})/2)$ denote the index of V_x^{EC}, $Dim(\cdot)$ function gets the dimension of the vector V_x^{EC}, $Round(\cdot)$ represents the round function.

5- Compute EC_ROAD as follows

$EC\_ROAD_{m'}(X)=\sum_{k=1}^{m'} b_k(x)$

and $b_k(x)$=kth smallest $d'(x, y')$ in $Dist(d')$.

Algorithm 1: Computing EC_ ROAD value

3 Image Quality Measurement Unit [8]

To assess the performance of the proposed filters for removal noise, Peak to Signal Noise (PSN) is used. PSNR is the ratio between the reference signal and the distorted signal in an image, given in decibels and it is defined as:

$$PSNR(db) = 10 * \log_{10}(255^2 / MSE) \qquad \text{............. 1}$$

And to assess the performance of the proposed MSE (mean square error) is used, MSE is the average squared difference between an original image and the restored images and is defined as

$$MSE = \frac{1}{m*n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(y(i,j)-d(i,j))^2 \qquad \text{............. 2}$$

Where d and y are the original and the restored images respectively, m and n are the number of pixels in both images (dimensions of the images). The higher the PSNR in the restored image, the better is its quality.

4 Hybrid Filters

4.1 Neuro Fuzzy System as Denoise Filter

The architecture of the NF network that is used based on Takaj fuzzy inference system as shown in figure (1).

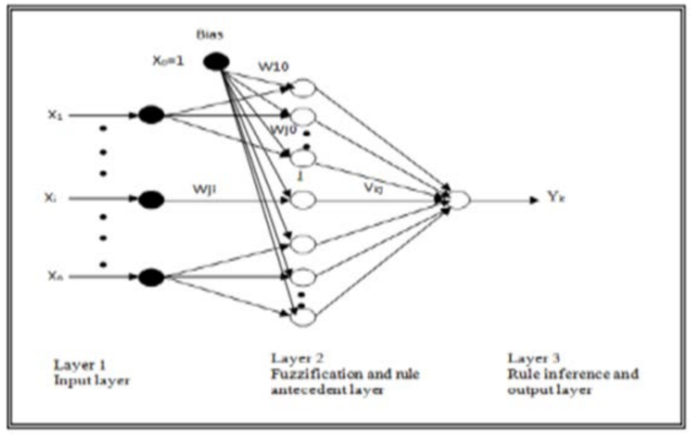

Figure (1) Structure of Neuro Fuzzy Network

The gradient descent based on BP algorithm is employed to adjust the parameters in NF network by using training pat- ... values (crisp ... tion. Thus,

$$Net_i = x_i \qquad \forall i = 1..N \qquad \text{.......... 3}$$

$Net_i = x_i$ $\forall i = 1..N$ 3

where N 1 is a number of neurons in the input layer

where $netj = w_{ji}x_i + w_{j0}$
w_{ji} weights from neuron j in hidden layer to neu
x_i input variable from neuron i in input layer to
w_{j0} weights from neuron j in hidden layer ti bias
$f(netj)$ output of neuron j in hidden layer

$f(netj) = \mu_i = e$ 4

where $netj = w_{ji}x_i + w_{j0}$

w_{ji} weights from neuron j in hidden layer to neuron i in input layer

x_i input variable from neuron i in input layer to bias neuron

w_{j0} weights from neuron j in hidden layer ti bias neuron

$f(netj)$ output of neuron j in hidden layer

Each neuron in output layer determines the output value by calculating output weights average for hidden layer as follow:

$$y_k = \frac{netk}{\sum \mu_j} \qquad \text{.......... 5}$$

The goal of backward phase is to minimize the error function:

where $netk = \sum_j \mu_j v_{kj}$

v_{kj} weights from neuron k inoutput layer to neuron j in hdden layer

$$E = \tfrac{1}{2}(y-d)^2 \qquad \text{.......... 6}$$

Where, d is the desired output. The learning algorithm in NF is realized by adjusting connection weights of the neurons in output layer and hidden layer as follow:

$$v_{kj}^{new} = v_{kj}^{old} + \Delta v_{kj} \qquad \text{.......... 7}$$

$$\Delta v_{kj} = -\eta \cdot \delta_k^{v} \qquad \text{.......... 8}$$

$$= -\eta \frac{\partial E}{\partial v_{kj}}$$

$$= -\eta \frac{\partial E}{\partial y_k} \cdot \frac{\partial f(netk)}{\partial netk} \cdot \frac{\partial netk}{\partial v_{kj}}$$

Then

$$\Delta v_{kj} = \eta (d_k - y_k) \cdot \frac{1}{\sum u_j} \mu_j \qquad \text{.......... 9}$$

While, adaptation of centers and widths of membership functions is as follows:

$$w_{ji}^{new} = w_{ji}^{old} + \Delta w_{ji} \qquad \text{.......... 10}$$

$$\Delta w_{ji} = -\eta \cdot \delta_{ji}^{w} \qquad \text{.......... 11}$$

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}}$$

$$= -\eta \Sigma \left[\frac{\partial E}{\partial y_k} \cdot \frac{\partial f(netk)}{\partial netk} \right] \cdot \frac{\partial netk}{\partial \mu_j} \frac{\partial f(netj)}{\partial netj} \frac{\partial netj}{\partial w_{ji}}$$

$$= -\eta (\Sigma \delta_k v_{kj}) f'(netj) x_i$$

Then

$$\Delta w_{ji} = -\eta (\Sigma \delta_k v_{kj})(-2 * netj * e^{-netj^2} * x_i) \qquad \text{.......... 12}$$

For Initialization of neuro fuzzy system , weights from neuron j in hidden layer to neuron I in input layer Input variable from neuron i in input layer to bias neuron Weights from neuron j in hidden layer to bias neuron instead of random process we initialize the weights of neuro fuzzy system from input layer to hidden layer by assign w_{j0}=1 and w_{ij} equal to first input pattern, while weights from neuron in hidden layer to neuron in output layer by assign v_{kj} with first training data set desired output.

4.2 Generating Training and Testing Patterns for NFSystem

To get best result in this work, different methods for generating training and testing patterns for each type of noise will be used and it is explained as follows:

1- Gaussian and Poisson noise

The training patterns for this type of noise consist of five inputs generated from the pixel values of 2x2 local neighbors of noisy image and the mean value of them.

In testing phase the value of each pixel and its 4-neighbors figure (3) form the five inputs for the neuro fuzzy network these are five inputs passed through the trained network to form a denoised pixel .The output of the neural fuzzy network is considered as the value of the denoised pixel located at *(i,j)*. If *i-1 or j-1* is less than 1, it is considered as 1. If *i+1* is greater than *N or j+1* is greater than *M*, *i+1* is considered as *N and j+1* is considered as *M*, where the NXN is dim of an image.

	i-1,j	
i,j-1	*i,j*	*i,j+1*
	i+1,j	

Figure (3): The pixel and its 4-neighbors

When we use this mechanism to denoise Gaussian noise, we named it as Neuro Fuzzy with Mean filter for Gaussian noise **(NF-Mean-G)** and for Poisson noise as Neuro Fuzzy with Mean filter for Poisson noise **(NF-Mean-P).**

2- Salt and pepper noise

The training patterns consist of five inputs generated from the pixel values of 2x2 local neighbors of the noisy image and the median value. The target value is the center of 2x2 local neighbors. Median value is computed as follows:

1- If the number of salt and pepper pixels in 2x2 local neighbors is less than a threshold then compute median value for 2x2 local neighbors.

2- If the number of salt and pepper pixels in 2x2 local neighbors is great than a threshold then compute median value for 3x3 local neighbors.

To insure processing all pixels, image is surrounded by a pad this pad and comes from replicating border pixels of image.

In this type of noise, Median value is computed depending on the number of noisy pixels (pixels that corrupted by salt and pepper noise) that found in 2x2 local neighbors where if the number of noisy pixels is great than a threshold the median of 3x3 local neighbors is computed and became the entry to NF network with the pixels value of 2x2 local neighbors else the median of 2x2 local neighbors will be computed.

To compute the number of salt and pepper noisy pixels we need to detect or locate salt and pepper noise position. The EC-road method is used to detect the pixels that corrupted by salt and pepper noise.

We use 3x3 local neighbors to compute EC-road value for each pixel. If the EC-road value for a pixel is less than a threshold then this pixel will be considered as noisy and get the value of 1 else its noisy free pixel and get the value of 0. After this, we have a binary array contains 0 and 1, 0 for free noisy pixel and 1 for noisy pixel.

In testing the same inputs for training pattern (2x2 local neighbors of the noisy image and the median value) will be passed through the trained NF. The median value is computed as described in training phase. The output of the NF network is considered as the value of the denoised pixel located at the center of 2x2 local neighbors. We call this as Neuro Fuzzy with median filter and Detection mechanism for denoising Salt and Pepper noise (NF-Median-D-SP).

3- Mixed noise

The training patterns consist of six inputs. Five inputs generated from the pixel value that processed and the values of its 4-neighbors; the sixth input is the mean value. The target value is the pixel that processed. Mean value is computed as follows:

1- If the number of salt and pepper noisy pixel in the pixel that processed and its 4- neighbors less than a threshold then Compute mean value for free salt and pepper noisy pixels in the Pixel and its 4-neighbors.

2- If the number of salt and pepper noisy pixel in the pixel that Processed and its 4- neighbors great than a threshold then compute mean value for free salt and pepper noise pixels found in the pixel and its 8- neighbors.

To insure processing all pixels image is surrounded by a pad that describes in salt and pepper noise. Also to detect salt and pepper noise pixels the EC-road method is used.

In testing the same inputs for training pattern (the pixel and its 4- neighbors with the mean value) is passed through the trained NF network and the mean value will be compute as described in training phase. This will be called as Neuro Fuzzy with Mean filter and Detection mechanism for denoising mixed noise of type Gaussian and Salt and Pepper noise (NF-Mean-D-GSP) and for mixed noise of type Poisson and Salt and Pepper noise as (NF-Mean-D-PSP).

Before generating Training patterns for each type of noise, the training image (Lena) is corrupted by specified ratio of noise according to the type of noise.

In this work, the NF network is trained with different numbers of fuzzy set and each input assigns the same number of fuzzy set for training the network. After training the NF network separately with each type of noise, the testing image is passed through the network then the *PSNR* value is computed for each image and compared with a specific filter mean or median or both depending on the type of noise. Algorithms (2) and (3) show the general algorithm to train and test NF that will be used for each type of noise where all follow the same steps but differ in the type of noise and the generating patterns of training and testing for each type of noise.

We use the following for training the neuro fuzzy network:

1- The number of fuzzy set (3-10) which is the same for each input.

2- The number of training iteration (max_cycle=100).

3- From trial and error we find best value for Momentum=0.1 and we use adaptive learning rate, where the learning rate (Lrate) is computed as in equation (13) and this value changes in each cycle.

$$Lrate = LR*(1-cycleno/5000) \quad .... 13$$

Where: the initial value for learning rate (LR) =0.8. Cycle no = 1….max_cycle.

Step 1: Choose type of noise
Step 2: Read the original training image and put in img-org
Step 3: Add noise to img-org
Step 4: Read the noisy image and put in img-noise
Step 5: Extract the dimensions of img-noise and put in *m* and *n*
Step 6:The pixel values of img-org and img-noise are normalized by dividing with 255
Step 7: cycle =1
Step 8: max_cycle=100
Step 9: Determine the number of fuzzy set
Step 10: Initialize the weights of NF
Step 11: *Etot* =0
Step 12: For i=1 to *m*
Step 13: For j=1 to *n*
Step 14: Depending on the type of noise extract the training pattern
Step 15: Compute the output of NF network (O_{NF}) as in equation (33)
Step 16: Compute the square error E^p between NF output and desired output where $E^p = (O_{NF}-O_d)^2$
Step 17: Update the weights of the NF using backpropagation algorithm as in equation (5) and equation (3)
Step18: Compute the total error *Etot* =
Step 19: End for
Step 20: End for
Step 21: Compute MSE or network error (where MSE= $Etot/m$)
Step 22: compute learning rate as in equation (3.9)
Step 23: cycle= cycle+1
Step 24: steps (11-23) until (cycle > max_cycle or MSE< desired_error)

Algorithm (2): the general algorithm to train NF system

Step 1: Read the testing image and put in img-org
Step 2: Repeat steps 3-6 in algorithm (2)
Step 3: For i=1 to m
Step 4: For j=1 to n
Step 5:Extract the testing pattern and pass through trained NF
Step 6:Compute the output of NF network as in equation (3)
Step 7: End for
Step 8: End for
Step 9: Compute PSNR value for denoising image

Algorithm 3: the general algorithm to test NF system

4.3 Genetic Algorithm with NF System

Algorithm (4) shows the genetic algorithm steps that are used.

Step1: Generate P (t) with pop size individuals
Step2: Evaluate P (t)
Step3: While (maximum number of generations not reached) do
    Select p1 and p2 from P (t)
    Perform crossover and mutation to generate new offspring
    Evaluate offspring
    Replace an individual from P (t) with one of off spring only if it is better t=t+1
End while

Algorithm 4: genetic algorithm steps that used in this work

In the following, the algorithm steps are described:

1- Encoding and initial population

The chromosome will take the representation that explains in figure (3). In the initial the population of individuals is generated randomly: the number of fuzzy set takes a random value between [1, 7], the input weights also take random value between [-0.5, 0.5] and the output weights between [-0.1, 0.1]. Different sizes (15, 20) of population are used .

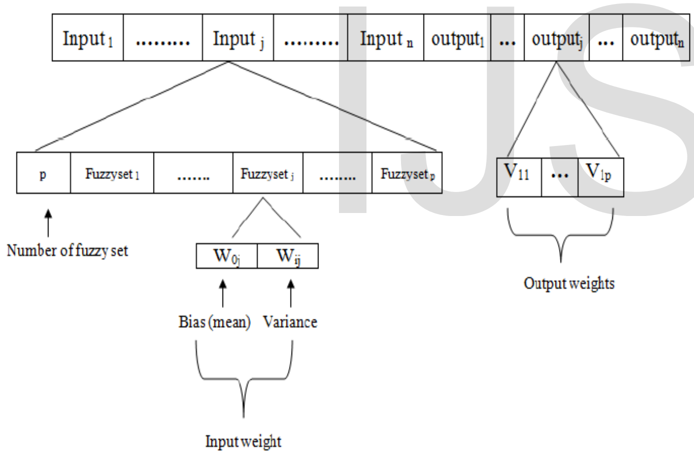

Figure (3): The chromosome and its representation

2- Fitness function

Fitness value is computed as follows:

$$fitness = \beta e + \alpha \left(\frac{\sum_{i=1}^{k} fset_i}{\max * k} \right) \quad \text{.......... 14}$$

Where α and β are a constant in the range [0, 1], e is the output error, k is the number of inputs, $fset_i$ is the number of fuzzy set assign for input i and max is a constant value equal to 7. The fitness function minimizes the BP error and also minimizes the number of fuzzy set that used. This means that high fitness value has less error and / or less rules.

*3- **Selection***

The tournament selection is used for crossover. In tournament selection, three individuals are selected randomly from population then two of them that have better fitness value are selected.

4- Crossover

Uniform crossover is used to recombine the parents (two parents) and to produce two new child where a two masks (one for each child) contain 0 and 1 is generated randomly and if the mask value is 0, the child inherits its feature from the first father and if the value is 1 the child inherits its feature from the second father . The mask is generated randomly and the size of mask depends on the minimum number of fuzzy set.

5- Mutation

We use uniform mutation where a mask of "0" and "1" are generated randomly and if the mask value=1 then perform mutation else do not perform. Mutation performs only on the weights and not on the number of fuzzy set. The mutation is performed by adding a random value from the normal distribution in the range [-0.8, 0.8] to input and output weights.

6- Replacement

For replacement, we use the tournament replacement where we select three individuals randomly and the worst of them is replaced with the new individual if the new individual has better fitness value.

7- Stopping criteria

GA will be stopped if the number of cycle reaches the maximum which is equal to 20. After GA stops, a number of chromosomes are generated the best of them which has the best fitness (lower fuzzy set and/or lower error) will be selected. This chromosome contains the parameters of NF which are the input and output weights and the number of fuzzy set for each input and this chromosome are used for training the neuro fuzzy network algorithm as described previously. After using genetic algorithm each filter proposed previously is be antecedent with G and as follows G-NF-Mean-G, G-NF-Mean-P, G-NF-Median-D-SP, G-NF-Mean-D-GSP and G-NF-Mean-D-PSP.

5 RESULTS

For training NF system gray scale image called "Lena" , while For testing, six images are used. Three of them are grayscale image and the other three are truecolor images (RGB image) as figure 4. All images are "jpg" and the size of each image is 255x255.

a- Cameraman b- Barbara c- Pepper
d- Parrot e- House f- Tower

Figure 4: Test images

For each type of noise the testing images are corrupted separately by different levels as in figure (5) and passed through the proposed hybrid neuro fuzzy then the PSNR value is computed for each image. In the following, the results for each type of noise are shown.

Lena corrupted by Gaussian noise and salt and pepper noise; Lena corrupted by salt and pepper noise with density 5%; Lena corrupted by Poisson noise; Lena corrupted by Gaussian noise with σ = 10; Lena corrupted by Poisson noise and salt and pepper noise

Figure (5): Test images with 5 cases of noise

1. Gaussian noise

Table (1-A) shows the number of fuzzy set that used for training neuro fuzzy system in case Gaussian noise with the net error of training when using (NFMean- G) while table (1.B) shows the number of fuzzy set that get from Genetic algorithm for five best run with the net error of training.

Table (2) shows the PSNR values for testing images that results of passing these images through (NF-Mean-G) and mean filter after corrupted these images separately by Gaussian noise with differt variances (∂=10, ∂=20, ∂=30). Table (3) shows the PSNR values when using (G-NF-Mean-G).

Table (1) A: the number of fuzzy set for training neuro fuzzy system in case Gaussian noise with the net error of training when using (NF-Mean-G). B. Architecture that get from Genetic algorithm for five best run with the net error of training.

Fuzzy_set.no	Error
(3,3,3,3,3)	0.0000013
(4,4,4,4,4)	0.0000012
(5,5,5,5,5)	0.0000007
(6,6,6,6,6)	0.0000005
(7,7,7,7,7)	0.0000004
(8,8,8,8,8)	0.0000004
(9,9,9,9,9)	0.0000004
(10,10,10,10,10)	0.0000005

A

Run no.	Architecture	Error
1	(1,1,6,2,1) (11)	0.000017
2	(2,1,1,1,1) (6)	0.000021
3	(1,2,5,1,3) (12)	0.000001
4	(1,3,1,3,1) (9)	0.0000008
5	(2,2,3,2,2) (11)	0.0000007

B

Table (2) PSNR values of testing images that corrupted by Gaussian noise of zero mean and different variances and denoised by NF-Mean-G with different architecture and mean filter

Var.		σ = 10						σ = 20						σ = 30					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3)	27.58	29.36	32.3	30.33	30.19	31.06	25.73	26.75	28.15	27.32	27.22	27.68	23.76	24.38	25.16	24.77	24.69	24.94
	(4,4,4,4,4)	27.53	29.34	32.24	30.33	30	30.97	25.67	26.72	28.13	27.31	27.11	27.62	23.7	24.35	25.17	24.75	24.62	24.91
	(5,5,5,5,5)	27.54	29.35	32.13	30.34	29.94	30.94	25.68	26.72	28.09	27.32	27.08	27.6	23.71	24.36	25.14	24.77	24.61	24.89
	(6,6,6,6,6)	27.6	29.41	32.21	30.39	30.04	31.01	25.75	26.78	28.14	27.37	27.16	27.65	23.78	24.42	25.18	24.83	24.67	24.93
	(7,7,7,7,7)	27.63	29.42	32.26	30.4	30.12	31.05	25.78	26.8	28.16	27.39	27.2	27.68	23.81	24.43	25.19	24.84	24.7	24.96
	(8,8,8,8,8)	27.65	29.41	32.29	30.41	30.17	31.08	25.79	26.8	28.18	27.39	27.23	27.7	23.82	24.44	25.2	24.85	24.72	24.97
	(9,9,9,9,9)	27.66	29.43	32.33	30.41	30.22	31.1	25.81	26.82	28.2	27.4	27.26	27.71	23.83	24.45	25.21	24.86	24.74	24.98
	(10,10,10,10,10)	27.67	29.44	32.35	30.41	30.25	31.12	25.81	26.81	28.2	27.4	27.27	27.73	23.84	24.45	25.21	24.86	24.74	24.99
Mean filter		24.92	26.66	28.46	28.84	27.73	28.38	24.25	25.72	27.09	27.42	26.53	27.07	23.31	24.5	25.51	25.8	25.1	25.52

Table (3) PSNR values of testing images that corrupted by Gaussian noise of zero mean and different variances and denoised by G-NF-Mean-G with different architecture and mean filter

Var.		σ = 10						σ = 20						σ = 30					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(1,1,6,2,1) (11)	27.38	29.06	32.11	30	29.92	30.78	25.38	26.32	27.74	26.78	26.7	27.34	23.35	23.86	24.59	24.16	24.08	24.49
	(2,1,1,1,1) (6)	27.67	29.65	32.4	30.37	30.14	31.01	25.81	26.91	28.17	27.34	27.18	27.6	23.83	24.47	25.12	24.79	24.67	24.82
	(1,2,5,1,3) (12)	27.43	29.2	32.07	30.2	29.87	30.88	25.55	26.54	27.89	27.07	26.85	27.49	23.56	24.14	24.84	24.49	24.31	24.71
	(1,3,1,3,1) (9)	27.66	29.36	32.33	30.26	30.21	31.04	25.71	26.67	27.99	27.19	27.13	27.57	23.69	24.25	24.92	24.63	24.56	24.76
	(2,2,3,2,2) (11)	27.49	29.07	32	30.08	29.83	30.94	25.47	26.35	27.61	26.95	26.75	27.42	23.39	23.93	24.53	24.37	24.19	24.6
Mean filter		24.92	26.66	28.46	28.84	27.73	28.38	24.25	25.72	27.09	27.42	26.53	27.07	23.31	24.5	25.51	25.8	25.1	25.52

From tables (2) and (3) we notice that**:** In case the variance is less than 30 *NF-Mean-G* filter gives the best result in PSNR for truecolor images. In case the variance is less than 30 *G-NF-Mean-G* filter gives the best result in PSNR for grayscale images. Mean filter gives the best result if the variance is equal to 30.

2. Poisson noise

ISSN 2229-5518

Table (4-A) shows the number of fuzzy set that used for training neuro fuzzy system with the net error of training when using (NF-Mean-P) while table (4-B) shows the number of fuzzy set that get from genetic algorithm for five best run with the net error of training when using (G-NF-Mean-P).

Table (5) shows the PSNR values that result of passing testing images through the trained (NF-Mean-P) and mean filter after corrupted these images by Poisson noise. While table (6) shows the PSNR values of these images when using genetic algorithm.

Table (4): A: The net error with the number of fuzzy set that used for training NF-Mean-P. B: The number of fuzzy set that get from Genetic algorithm for five best run with the net error of training when using (G-NF-Mean-P)

Fuzzy_set.no	Error
(3,3,3,3,3)	0.0000014
(4,4,4,4,4)	0.0000014
(5,5,5,5,5)	0.0000007
(6,6,6,6,6)	0.0000006
(7,7,7,7,7)	0.0000004
(8,8,8,8,8)	0.0000004
(9,9,9,9,9)	0.0000004
(10,10,10,10,10)	0.0000005

A

Run no.	Architecture	Error
1	(3,1,1,3,2) (10)	0.0000018
2	(1,4,3,3,1) (12)	0.0000011
3	(6,1,1,3,2) (13)	0.000001
4	(1,2,1,3,1) (8)	0.0000014
5	(2,1,1,2,1) (7)	0.0000013

B

Table (5) PSNR values of testing images that corrupted by Poisson noise and denoised by NF-Mean-P with different Architecture and mean filter.

Image name		Cameraman	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3)	27.44	29.15	31.63	30.22	29.94	30.7
	(4,4,4,4,4)	27.32	29.2	31.48	30.15	29.68	30.52
	(5,5,5,5,5)	27.37	29.18	31.36	30.18	29.65	30.52
	(6,6,6,6,6)	27.44	29.25	31.43	30.23	29.76	30.58
	(7,7,7,7,7)	27.48	29.25	31.48	30.25	29.84	30.62
	(8,8,8,8,8)	27.49	29.25	31.51	30.25	29.9	30.65
	(9,9,9,9,9)	27.5	29.26	31.53	30.26	29.92	30.67
	(10,10,10,10,10)	27.51	29.26	31.54	30.26	29.95	30.68
Mean filter		24.87	26.58	28.29	28.78	27.59	28.25

Table (6) PSNR value of testing images that corrupted by Poisson noise and denoised by G-NF-Mean-P with different Architecture and mean filter

Image name		Cameraman	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,1,1,3,2) (10)	27.55	29.38	31.64	30.26	30.08	30.72
	(1,4,3,3,1) (12)	27.47	29.49	31.58	30.14	30.17	30.74
	(6,1,1,3,2) (13)	27.52	29.44	31.49	30.24	29.87	30.58
	(1,2,1,3,1) (8)	27.42	29.22	31.42	30.02	29.94	30.61
	(2,1,1,2,1) (7)	27.24	28.97	31.12	29.8	29.61	30.4
Mean filter		24.87	26.58	28.29	28.78	27.59	28.25

From tables (5) and (6) we notice that: G-NF-Mean-P filter gives the best result in PSNR for all images in comparative with the result of NF-Mean-P and mean filter**.**

3. Salt and Pepper noise

Table (7-A) shows the number of fuzzy set that used for training neuro fuzzy system in case Salt and Pepper noise with the net error of training when using (NF-Median-D-SP) while table (7-B) shows the number of fuzzy set that get from Genetic algorithm for five best run with the net error of training (G-NF Median- D-SP). Table (8) shows the PSNR values that result from passing testing images through trained (NF-Median-D-Sp) and median filter after corrupted these images separately by Salt and Pepper noise with different density (10%,20%,30%). While table (9) shows the PSNR values of these images when using genetic algorithm (G-NF- Median-D-SP).

Table (7):A: The number of fuzzy set that used for training neuro fuzzy system in case Salt and Pepper noise with the net error of training when using (NF-Median-D-Sp).B: The number of fuzzy sets that get from Genetic algorithm for five best run with the net error of training when using (G-NF- Median-D-Sp)

Fuzzy_set.no	Error
(3,3,3,3,3)	0.00075
(4,4,4,4,4)	0.00061
(5,5,5,5,5)	0.00052
(6,6,6,6,6)	0.00048
(7,7,7,7,7)	0.00047
(8,8,8,8,8)	0.00047
(9,9,9,9,9)	0.00042
(10,10,10,10,10)	0.00041

A

Run	Architecture	Error
1	(2,3,2,1,4) (12)	0.000408
2	(2,6,7,2,5) (22)	0.000406
3	(3,2,1,1,3) (10)	0.000329
4	(2,1,3,3,4) (13)	0.000405
5	(2,1,1,4,3) (11)	0.000368

B

Table (8): PSNR values of testing images that corrupted by Salt and Pepper noise of different ratios and denoised by NF-Median-D-SP with different architecture and median filter

ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3)	24.5	24.05	26.42	27.52	20.74	26.19	22.2	22.33	23.52	24.47	19.12	23.81	19.84	20.36	20.81	21.74	17.51	21.54
	(4,4,4,4,4)	24.15	24.19	28.65	28.36	23.96	27.67	22.57	23.1	25.57	25.72	20.86	24.99	20.31	21.17	22.26	22.69	18.89	22.54
	(5,5,5,5,5)	25.46	24.93	29.59	29.78	24.5	28.59	23.14	23.45	26.08	26.25	22.07	25.76	20.62	21.43	22.7	23	19.76	23.07
	(6,6,6,6,6)	25.61	24.15	29.89	29.83	25.51	28.99	22.92	22.72	25.83	25.6	22.56	25.38	20.33	20.72	22.27	22.07	19.94	22.3
	(7,7,7,7,7)	25.36	24.64	30.35	30.45	18.71	27.43	23.12	23.43	26.76	26.46	17.81	24.81	20.01	20.87	22.01	21.49	16.43	21.16
	(8,8,8,8,8)	25.57	24.66	30.15	30.38	18.88	27.54	23.22	23.43	26.64	26.39	17.91	24.86	20.02	20.82	21.91	21.41	16.47	21.15
	(9,9,9,9,9)	25.85	24.84	30.59	31.04	17.69	27.02	23.38	23.52	26.68	26.61	16.9	24.44	19.99	20.71	21.66	21.21	15.64	20.7
	(10,10,10,10,10)	25.4	24.66	30.55	30.34	18.31	27.36	23.15	23.39	26.82	26.3	17.41	24.74	20	20.76	21.91	21.28	16.07	21.03
Median filter		26.01	29.48	30.65	29.55	29.49	30.35	23.53	25.87	27.41	26.64	26.2	27.07	20.54	21.98	22.29	22.45	22.11	22.84

Table (9): PSNR values of testing images that corrupted by Salt and Pepper noise of different ratios and denoised by G-NF-Median-D-SP with different architecture and median filter

ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(2,3,2,1,4) (12)	28.76	29.48	30.34	30.74	24.74	29.39	25.93	26.35	27.4	27.71	22.84	27.26	23.18	23.42	24.19	24.43	20.89	24.56
	(2,6,7,2,5) (22)	27.14	26.47	33.31	32.81	27.24	32.42	25.42	25.51	30.45	29.29	25.38	29.76	22.86	23.42	26.11	25	22.66	25.86
	(3,2,1,1,3) (10)	29.09	30.45	32.47	33.09	24.32	30.85	26.23	27.37	29.76	29.59	23.33	28.93	23.28	24.26	25.91	25.66	21.67	25.84
	(2,1,3,3,4) (13)	28.35	28.49	31.08	31.06	24.34	29.5	25.48	25.96	28.5	27.83	23.14	27.18	22.45	23.22	24.89	24.26	21.28	24.29
	(2,1,1,4,3) (11)	29.57	30.28	31.83	32.41	26.06	31.37	26.6	27.32	29.43	29.29	24.58	29.11	23.3	24.01	25.45	25.16	22.25	25.54
Median filter		26.01	29.48	30.65	29.55	29.49	30.35	23.53	25.87	27.41	26.64	26.2	27.07	20.54	21.98	22.29	22.45	22.11	22.84

From tables (8) and (9) we notice that:G-NF-Median-D-SP filter gives the best results in PSNR for gray and truecolor image in comparison with NF-Median-D-SP and median filter. G-NF-Median-D-SP filter is more effective for denoising high density of Salt and Pepper noise.

4. Mixed noise of type Gaussian with Salt and Pepper

Table (10-A) shows the number of fuzzy set that used for training neuro fuzzy system in case mixed noise of type Gaussian and Salt and Pepper noise with the net error of training when using (NF-Mean-D-GSP) while table (10-B) shows the number of fuzzy set that get from Genetic algorithm for five best run with the net error of training.

Tables (11), (12) and (13) show the PSNR values that result from passing testing images through trained (NF-Mean-D-GSP) and mean and median filter after corrupted these images separately by Gaussian noise with different variances (∂ =10, ∂ =20, ∂ =30) where each table referred to a variance and then each variance is corrupted by Salt and Pepper noise with different density (10%,20%,30%). While tables (14), (15) and (16) show the PSNR values of these images when using Genetic algorithm (G-NF-Mean-D-GSP).

Table (10): A:The number of fuzzy set that used for training neuro fuzzy system with the net error of training when using (NF-Mean-D-GSP). B: The number of fuzzy set that get from Genetic algorithm with the net error of training when using (G- NF-Mean-D-GSP).

Fuzzy_set.no	Error
(3,3,3,3,3,3)	0.00101
(4,4,4,4,4,4)	0.00098
(5,5,5,5,5,5)	0.00097
(6,6,6,6,6,6)	0.00096
(7,7,7,7,7,7)	0.00097
(8,8,8,8,8,8)	0.00095
(9,9,9,9,9,9)	0.00095
(10,10,10,10,10,10)	0.00099

A

Run	Architecture	Error
1	(1,1,3,2,3,3) (13)	0.00102
2	(2,2,6,1,2,2) (15)	0.00097
3	(1,3,2,4,5,3) (18)	0.001
4	(2,2,2,1,1,3) (11)	0.00099
5	(2,4,2,3,1,2) (14)	0.00101

B

Table (11): PSNR values of testing images that corrupted by ∂ = 10 Gaussian noise and Salt and Pepper noise of different ratios and denoised by NFD-Mean-GSP with different Architecture and also by mean and median filter.

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3,3)	27.46	28.3	30.38	30.09	22.57	28.4	26.44	27.33	29.09	29.22	22.3	27.81	24.98	25.58	27.41	27.74	21.89	26.62
	(4,4,4,4,4,4)	27.56	28.68	31.48	30.44	26.19	29.96	26.69	27.75	30.34	29.63	25.78	29.35	25.38	26.01	28.64	28.27	25.15	28.08
	(5,5,5,5,5,5)	27.67	28.67	31.45	30.47	26.8	30.13	26.78	27.73	30.34	29.65	26.35	29.5	25.45	26	28.64	28.27	25.65	28.19
	(6,6,6,6,6,6)	27.7	28.65	31.57	30.53	27.32	30.29	26.83	27.72	30.49	29.7	26.87	29.72	25.51	26	28.75	28.28	26.11	28.39
	(7,7,7,7,7,7)	27.77	28.73	31.55	30.54	27.21	30.28	26.86	27.75	30.37	29.68	26.69	29.63	25.48	25.96	28.59	28.23	25.9	28.26
	(8,8,8,8,8,8)	27.79	28.7	31.7	30.66	28.5	30.59	26.9	27.76	30.59	29.79	27.86	29.93	25.55	26.02	28.81	28.29	26.87	28.5
	(9,9,9,9,9,9)	27.91	28.71	31.73	30.67	28.98	30.71	27.01	27.76	30.62	29.77	28.22	30.03	25.65	26.02	28.83	28.25	27.1	28.55
	(10,10,10,10,10,10)	27.81	28.69	31.8	30.51	29.37	30.79	26.95	27.7	30.57	29.56	28.47	30.04	25.62	25.95	28.73	28.06	27.23	28.56
Mean filter		21.25	22.25	23.07	22.88	22.12	23.25	19	19.91	20.41	20.13	19.16	20.83	17.2	18.1	18.47	18.26	17.24	19.12
Median filter		25.34	28.3	29.64	28.22	28.53	28.79	23.47	25.73	26.32	25.69	25.36	26.2	20.3	21.56	22	22.25	21.75	22.49

Table (12): PSNR values of testing images that corrupted by ∂ = 10Gaussian noise and Salt and Pepper noise of different ratios and denoised by GNF-D-Mean-GSP with different Architecture and also by mean and median filter.

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(1,1,3,2,3,3) (13)	27.22	28.55	31.67	30.45	27.24	30.25	26.55	27.74	30.87	29.68	26.76	29.68	25.63	26.25	29.85	28.42	26.05	28.48
	(2,2,6,1,2,2) (15)	27.49	28.61	31.05	30.47	23.58	28.87	26.77	27.84	30.28	29.73	23.27	28.4	25.77	26.36	29.36	28.52	22.86	27.43
	(1,3,2,4,5,3) (18)	27.41	28.55	31.59	30.44	26.07	29.92	26.72	27.74	30.76	29.63	25.65	29.35	25.78	26.27	29.76	28.4	25.07	28.24
	(2,2,2,1,1,3) (11)	27.34	28.57	30.92	30.38	23.69	28.75	26.68	27.81	30.19	29.63	23.39	28.29	25.75	26.33	29.34	28.46	23	27.38
	(2,4,2,3,1,2) (14)	27.21	28.41	30.8	30.22	23.36	28.8	26.5	27.61	29.94	29.45	23.1	28.3	25.52	26.11	28.94	28.19	22.73	27.29
Mean filter		21.25	22.25	23.07	22.88	22.12	27.24	19	19.91	20.41	20.13	19.16	20.83	17.2	18.1	18.47	18.26	17.24	19.12
Median filter		25.34	28.3	29.64	28.22	28.53	28.79	23.47	25.73	26.32	25.69	25.36	26.2	20.3	21.56	22	22.25	21.75	22.49

Table (13): PSNR values of testing images that corrupted by ∂ = 20Gaussian noise and Salt and Pepper noise of different ratios and denoised by NF-D-Mean-GSP with different Architecture and also by mean and median filter

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3,3)	25.14	25.82	26.77	26.64	22.11	25.86	24.43	24.97	26.04	26.05	21.81	25.31	23.39	23.85	24.9	25.14	21.36	24.39
	(4,4,4,4,4,4)	25.23	26.06	27.28	26.81	24.58	26.57	24.66	25.26	26.66	26.29	24.27	26.07	23.72	24.18	25.59	25.47	23.7	25.23
	(5,5,5,5,5,5)	25.29	26.02	27.21	26.78	24.92	26.6	24.7	25.22	26.6	26.26	24.61	26.11	23.75	24.14	25.54	25.43	23.99	25.27
	(6,6,6,6,6,6)	25.3	26.01	27.27	26.81	25.24	26.67	24.73	25.23	26.67	26.29	24.94	26.22	23.77	24.13	25.57	25.43	24.28	25.38
	(7,7,7,7,7,7)	25.34	26.04	27.21	26.81	25.18	26.66	24.74	25.21	26.58	26.26	24.83	26.17	23.75	24.09	25.46	25.39	24.14	25.3
	(8,8,8,8,8,8)	25.34	26.01	27.25	26.82	25.85	26.73	24.75	25.22	26.63	26.28	25.47	26.25	23.79	24.1	25.52	25.39	24.69	25.38
	(9,9,9,9,9,9)	25.38	25.97	27.2	26.79	26.06	26.77	24.81	25.18	26.59	26.24	25.63	26.29	23.84	24.08	25.49	25.34	24.79	25.42
	(10,10,10,10,10,10)	25.42	26.04	27.3	26.82	26.32	26.95	24.84	25.19	26.62	26.22	25.81	26.43	23.85	24.04	25.45	25.27	24.88	25.52
Mean filter		20.86	21.99	22.56	22.48	21.73	22.83	18.83	19.73	20.23	19.92	19.04	20.55	17.11	17.95	18.44	18.03	17.08	18.9
Median filter		23.88	25.72	26.3	25.78	25.93	26.14	22.35	23.75	23.98	23.89	23.84	24.16	19.72	20.68	20.99	20.85	20.44	21.15

Table (14): PSNR values of testing images that corrupted by ∂ = 20Gaussian noise and Salt and Pepper noise of different ratios and denoised by GNF-D-Mean-GSP with different Architecture and also by mean and median filter

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(1,1,3,2,3,3) (13)	25.05	25.91	27.42	26.87	25.26	26.67	24.62	25.29	26.93	26.37	24.92	26.22	23.89	24.54	26.28	25.59	24.3	25.47
	(2,2,6,1,2,2) (15)	24.99	25.75	26.98	26.66	22.73	25.85	24.57	25.22	26.58	26.23	22.46	25.47	23.87	24.57	26.07	25.57	22.05	24.83
	(1,3,2,4,5,3) (18)	25.22	25.95	27.48	26.9	24.55	26.6	24.78	25.34	26.98	26.39	24.23	26.17	24.03	24.62	26.35	25.64	23.68	25.45
	(2,2,2,1,1,3) (11)	24.94	25.71	26.89	26.58	22.78	25.67	24.56	25.19	26.52	26.16	22.52	25.33	23.9	24.56	26.05	25.53	22.13	24.76
	(2,4,2,3,1,2) (14)	25.05	25.88	27.22	26.83	22.68	26.08	24.58	25.26	26.7	26.3	22.42	25.63	23.83	24.51	26.05	25.54	22.01	24.89
Mean filter		20.86	21.99	22.56	22.48	21.73	22.83	18.83	19.73	20.23	19.92	19.04	20.55	17.11	17.95	18.44	18.03	17.08	18.9
Median filter		23.88	25.72	26.3	25.78	25.93	26.14	22.35	23.75	23.98	23.89	23.84	24.16	19.72	20.68	20.99	20.85	20.44	21.15

Table (15) PSNR values of testing images that corrupted by ∂ = 30Gaussian noise and Salt and Pepper noise of different ratios and denoised by NFD-Mean-GSP with different Architecture and also by mean and median filter.

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3,3)	22.89	23.43	24.11	23.83	21.21	23.48	22.34	22.71	23.51	23.37	20.93	22.95	21.56	21.86	22.6	22.71	20.49	22.25
	(4,4,4,4,4,4)	22.98	23.62	24.38	23.96	22.8	23.86	22.54	22.95	23.85	23.55	22.51	23.4	21.83	22.11	22.98	22.96	22.06	22.79
	(5,5,5,5,5,5)	23	23.56	24.3	23.91	22.97	23.85	22.57	22.9	23.77	23.5	22.68	23.41	21.84	22.06	22.91	22.91	22.21	22.8
	(6,6,6,6,6,6)	23	23.55	24.32	23.93	23.18	23.88	22.56	22.89	23.79	23.5	22.88	23.46	21.82	22.05	22.89	22.9	22.39	22.87
	(7,7,7,7,7,7)	23.03	23.55	24.27	23.92	23.13	23.88	22.57	22.87	23.72	23.49	22.81	23.43	21.82	22	22.82	22.87	22.3	22.81
	(8,8,8,8,8,8)	22.99	23.53	24.26	23.91	23.5	23.85	22.56	22.86	23.71	23.47	23.16	23.42	21.81	22	22.79	22.84	22.61	22.81
	(9,9,9,9,9,9)	22.99	23.47	24.19	23.86	23.58	23.86	22.57	22.82	23.65	23.43	23.22	23.44	21.82	21.96	22.74	22.8	22.64	22.84
	(10,10,10,10,10,10)	23.12	23.58	24.3	23.95	23.76	24.11	22.66	22.87	23.7	23.47	23.35	23.64	21.89	21.97	22.72	22.8	22.72	23
Mean filter		20.46	21.44	22.11	21.98	21.09	22.2	18.49	19.33	19.86	19.61	18.69	20.21	16.87	17.84	18.41	17.85	16.84	18.67
Median filter		22.36	23.66	24.01	23.58	23.72	23.76	20.83	21.7	22.04	22.03	21.93	22.15	18.77	19.5	19.73	19.54	19.46	19.69

Table (16): PSNR values of testing images that corrupted by ∂ = 30Gaussian noise and Salt and Pepper noise of different ratios and denoised by GNF-D-Mean-GSP with different Architecture and also by mean and median filter.

Noise ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(1,1,3,2,3,3) (13)	22.87	23.55	24.45	24.03	23.22	23.92	22.53	23.01	24.02	23.63	22.91	23.51	21.94	22.38	23.43	23.08	22.45	22.96
	(2,2,6,1,2,2) (15)	22.7	23.33	24.12	23.78	21.52	23.35	22.41	22.88	23.8	23.45	21.28	23.01	21.89	22.36	23.37	23.01	20.93	22.54
	(1,3,2,4,5,3) (18)	23.02	23.61	24.55	24.07	22.81	23.94	22.69	23.07	24.12	23.67	22.51	23.54	22.09	22.47	23.56	23.13	22.08	23.02
	(2,2,2,1,1,3) (11)	22.7	23.3	24.05	23.71	21.52	23.22	22.45	22.86	23.76	23.4	21.3	22.93	21.96	22.36	23.36	22.98	20.97	22.51
	(2,4,2,3,1,2) (14)	22.89	23.58	24.49	24.04	21.62	23.65	22.52	23.01	24.04	23.62	21.36	23.21	21.93	22.41	23.49	23.07	20.98	22.66
Mean filter		20.46	21.44	22.11	21.98	21.09	22.2	18.49	19.33	19.86	19.61	18.69	20.21	16.87	17.84	18.41	17.85	16.84	18.67
Median filter		22.36	23.66	24.01	23.58	23.72	23.76	20.83	21.7	22.04	22.03	21.93	22.15	18.77	19.5	19.73	19.54	19.46	19.69

From tables (11), (12), (13) (14), (15) and (16) we notice that**:** G-NF-Mean-D-GSP and NF-Mean-D-GSP filters are more effective in denoising mixed noise of type Gaussian with Salt and Pepper than mean or median filter.

5. ***Mixed noise of type Poisson with Salt and Pepper***

Table (17-A) shows the number of fuzzy set that used for training neuro fuzzy system in case mixed noise of type Poisson and Salt and Pepper noise with the net error of training when using (NF-Mean-D-PSP) and table (17-B) shows the number of fuzzy set that get from Genetic algorithm for five best run with the net error of training.

Table (18) shows the PSNR values that result from passing testing images through trained (NF-Mean-D-PSP) and also through mean and median filters after corrupted these images separately by Poisson noise and then by Salt and Pepper noise with different density (10%,20%,30%) while table (19) shows the PSNR values of these images when using Genetic algorithm or (G-NF-Mean-DPSP).

Table (17) A: The number of fuzzy set that used for training neuro fuzzy system in case mixed noise of type Poisson and Salt and Pepper noise with the net error of training when using (NF-Mean-D-PSP). B: The number of fuzzy set that get from Genetic algorithm for five best run with the net error of training when using (G- NF-Mean-D-PSP)

Fuzzy_set.no	Error
(3,3,3,3,3,3)	0.00088
(4,4,4,4,4,4)	0.00084
(5,5,5,5,5,4)	0.00083
(6,6,6,6,6,6)	0.00082
(7,7,7,7,7,7)	0.00082
(8,8,8,8,8,8)	0.00082
(9,9,9,9,9,9)	0.00081
(10,10,10,10,10,10)	0.00081

A

Run	Architecture	Error
1	(2,1,2,7,1,2) (15)	0.00092
2	(2,3,2,3,1,2) (13)	0.00091
3	(1,3,2,3,3,2) (14)	0.00092
4	(4,2,1,3,2,5) (17)	0.00096
5	(1,1,3,1,2,2) (10)	0.0009

B

Table (18): PSNR values of testing images that corrupted by Poisson noise and Salt and Pepper noise of different ratios and denoised by NF-Mean-D-PSP with different architecture and also by mean and median filter

ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(3,3,3,3,3,3)	27.42	24.91	28.3	29.63	21.16	27.27	26.26	24.35	28.08	28.69	20.85	26.61	24.75	23.65	26.51	27.24	20.39	25.58
	(4,4,4,4,4,4)	27.60	25.1	30.24	30.12	24.50	28.93	26.47	24.5	28.95	29.14	24.05	28.12	24.07	23.74	27.1	27.50	23.28	26.77
	(5,5,5,5,5,5)	27.86	25.25	30.25	30.24	24.36	28.82	26.65	24.67	29.09	29.29	23.82	28.04	25.07	23.93	27.35	27.8	23.09	26.79
	(6,6,6,6,6,6)	27.85	25.17	30.27	30.32	24.6	28.93	26.64	24.6	29.1	29.36	24.05	28.17	25.05	23.87	27.32	27.83	23.29	26.88
	(7,7,7,7,7,7)	27.81	25.16	30.25	30.3	24.54	28.91	26.58	24.58	29.05	29.31	23.98	28.12	24.98	23.84	27.25	27.76	23.21	26.81
	(8,8,8,8,8,8)	27.96	25.37	30.29	30.32	24.53	28.94	26.8	24.79	29.22	29.42	23.99	28.2	25.24	24.07	27.52	27.98	23.26	27.02
	(9,9,9,9,9,9)	27.85	25.23	30.29	30.31	24.95	29.07	26.65	24.64	29.11	29.34	24.35	28.28	25.07	23.89	27.31	27.82	23.54	26.98
	(10,10,10,10,10,10)	27.85	25.23	30.34	30.33	25.32	29.19	26.65	24.64	29.15	29.34	24.66	28.38	25.06	23.89	27.34	27.81	23.79	27.05
Mean filter		21.3	22.15	23.03	22.83	22.04	23.31	18.98	19.85	20.33	20.08	19.2	20.71	19.08	18.01	18.43	18.27	17.22	19.08
Median filter		25.11	27.86	28.89	28.05	28.14	28.38	23.15	25.19	25.79	25.71	25.48	25.97	20.33	21.56	21.74	22.24	21.46	22.33

Table (19): PSNR values of testing images that corrupted by Poisson noise and Salt and Pepper noise of different ratios and denoised by G-NF-Mean-DPSP with different architecture and also by mean and median filter

ratio		10%						20%						30%					
Image name		Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower	Camera man	Pepper	House	Barbara	Parrot	Tower
Architecture	(2,1,2,7,1,2) (15)	27.53	24.73	30.38	29.8	25.78	29.43	26.77	24.5	29.78	29.2	25.2	28.84	25.63	23.98	28.79	28.08	24.4	27.78
	(2,3,2,3,1,2) (13)	27.61	24.69	30.38	29.82	26.04	29.49	26.78	24.44	29.66	29.15	25.33	28.8	25.59	23.88	28.63	27.95	24.43	27.64
	(1,3,2,3,3,2) (14)	27.54	24.91	30.34	29.88	24.92	29.08	26.79	24.67	29.63	29.18	24.3	28.45	25.72	24.15	28.79	28.1	23.54	27.5
	(4,2,1,3,2,5) (17)	27.31	24.84	30.31	29.71	25.71	29.15	26.56	24.52	29.42	28.91	24.92	28.45	25.48	23.96	28.48	27.78	24.02	27.49
	(1,1,3,1,2,2) (10)	27.72	25	30.48	30.04	25.58	29.2	26.91	24.75	29.8	29.34	24.87	28.6	25.8	24.23	28.9	28.25	24.03	27.66
Mean filter		21.3	22.15	23.03	22.83	22.04	23.31	18.98	19.85	20.33	20.08	19.2	20.71	19.08	18.01	18.48	18.27	17.22	19.08
Median filter		25.11	27.86	28.89	28.05	28.14	28.38	23.15	25.19	25.79	25.71	25.48	25.97	20.33	21.56	21.74	22.24	21.46	22.33

From tables (18) and (19), we notice that: G-NF-Mean-D-PSP and NF-Mean-D-PSP filters are more effective in denoising mixed noise of type Poisson with Salt and Pepper than mean or median filter. G-NF-Median-D-PSP filter is more effective for denoising high density of mixed noise of type Poisson with Salt and Pepper.

5 Conclusion

Removing the noise from image has an important role in many applications. In this work, several types of noise such as Gaussian, Poisson, Salt and Pepper and mixed noise are removed using hybrid filters. This hybrid filters consist from combining NF system and one of mean or median filter according to the type of noise that we want to remove and then Genetic algorithm is used to obtained best weights and architecture. For each type of noise, we suggest a structure for input and output patterns. In general, we conclude the following points:

1- The best choice of training and testing data structure affect the performance of the hybrid filters.

2- Hybrid filters (NF-Mean-G, G- NF-Mean-G) are more effective in denoising Gaussian noise if the variance is less than 30 while mean filter is more effective in denoising Gaussian noise if the variance equal to 30.

3- Hybrid filter (G-NF-Mean-P) gives the best result in PSNR in denoising Poisson noise.

4- Hybrid filter (G-NF-Median-D-SP) is more effective in denoising Salt and Pepper noise and gives the best result even if the density of noise is high.

5- Mean and median filters are not effective in denoising mixed noise.

6- G-NF-Mean-D-GSP and NF-Mean-D-GSP filters are more effective in denoising mixed noise of type Gaussian with Salt and Pepper even if the density of noise is high and also gives best result in PSNR.

7- G-NF-Mean-D-PSP and NF-Mean-D-PSP filters are more effective in denoising mixed noise of type Poisson with Salt and Pepper even if the density of noise is high and also gives best result in PSNR.

6 References

[1] H. Zhou, J. Wu, and J. Zhang."Digital Image Processing-Part one", Ventus publishing APS, 2010.

[2] C. Khare and K. K. Nagwanshi, "Image Restoration in Non-Linear Filtering Domain using MDB approach", World Academy of Science, Engineering and Technology, Vol.37, pp. 761-765, 2010.

[3] P. C. Khanzode and S. A. Ladhake, "Two Stage Impulse Noise Removal Technique based on Neural Network and Fuzzy Decisions ", 2nd National Conference on Information and Communication Technology (NCICT), 2011.

[4] S. N. Sivanandam, S. Sumathi and S. N. Deepa, "Introduction to Fuzzy Logic using MATLAB", Springer-Verlag Berlin Heidelberg, 2007.

[5] F. O. Karray and C. d. Silva,"Soft Computing and Intelligent Systems Design", Pearson Education Limited, 2004.

[6] S. Qureshi, "Embedded Image Processing on the TMS320C6000™ DSP", Springer Science+Business Media, Inc.2005.

[7] G. Xu, J. Tan and J. Zhong, "An Improved Trilateral Filter for Image Denoising using an Effective Impulse Detector", IEEE 4th International Congress on Image and Signal Processing, Vol.1, pp. 90-94, 2011.

[8] Y. A. Al-Sbou, "Artificial Neural Networks Evaluation as an Image Denoising Tool", World Applied Sciences Journal, Vol.17, pp. 218-227, 2012. pplications", InTech, 2011