An Artificial Intelligent Technique for Image Enhancement

R.Pushpavalli[1] and *G.Sivarajde*[2]

[1]Research Scholar, Electronics and Communication Engineering,
Pondicherry Engineering College, Puducherry, Puducherry (State), India-605 014.
[2]Professor, Electronics and Communication Engineering,
Pondicherry Engineering College, Puducherry,
Puducherry (State), India-605 014.
[1]Pushpavalli11@pec.edu, [2]shivaradje@pec.edu

Abstract

A class of neural filter for image enhancement is proposed in this paper. The proposed intelligent filter is carried out in two stages. In first stage the corrupted image is filtered by applying two special classes of decision based filters. Filtered image outputs from decision based filters are suitably combined with a Feed forward neural network in the second stage. The internal parameters of the feed forward neural network are adaptively optimized by training for three well known images. This is quite effective in eliminating impulse noise. Extensive simulation results show that the proposed filter is superior in terms of eliminating impulse noise as well as preserving edges and the results are compared with other existing filters.

Keywords: *Feed Forward Neural Network, Image Denoising, Impulse Noise, Nonlinear Filter*

1. Introduction

Detection and removal of impulse noise from digital images have been of research interest in the last few years [1-3]. Majority of the existing filtering methods comprise order statistic filters utilizing the rank order information of an appropriate set of noisy input pixels. These filters are usually developed in the general framework of *rank selection filters*, which are nonlinear operators, constrained to output an order statistic filters from a set of input samples.

The standard median filter (MF) is a simple rank selection filter and attempts to remove impulse noise from the center pixel of the processing window by changing the luminance value of the center pixel with the median of the luminance values of the pixels contained within the window. This approach provides a reasonable noise removal performance with the cost of introducing undesirable blurring effects into image details even at low noise densities. Since its application to impulse noise removal, the median filter has been of research interest and a number of rank order-based filters trying to avoid the inherent drawbacks of the standard median filter have been investigated [4-7]. Weighted order statistic filters, namely *weighted median filter* (WMF) and the center-weighted median filter (CWMF) respectively, employ a mechanism for appropriately weighting pixels of the analysis window to control the tradeoff between the noise suppression and edges and fine detail preservation. These filters yield better edges and fine detail preservation performance than the median filter at the expense of reduced noise suppression.

Conventional order statistics filters usually distort the uncorrupted regions of the input image during restoration of the corrupted regions, introducing undesirable blurring effects into the image. In switching median filters, the noise detector aims to determine whether the center pixel of a given filtering window is corrupted or not. If the center pixel is identified by the noise detector as corrupted, then the output of the system is switched to the output of the noise filter, which has the restored value for the corrupted pixel. If the center pixel is identified as uncorrupted, which means that there is no need to perform filtering, the noise removal operator is bypassed and the output of the system is switched directly to the input. This approach has been employed to significantly improve the performance of conventional median filtering and a number of median based filters exploiting different impulse detection mechanisms have been investigated [8-14]. Multiple decision based switching median (MDSM) filtering scheme has been examined to eliminate impulse noise using global and local statistics [15]. Determining the right value of threshold for a given image is a challenging task in single threshold Switching Median Filtering. Besides, the single threshold value cannot be expected to yield optimal performance over the entire image since the images are nonstationary processes. In order to avoid computational complexity, Multiple threshold switching median filtering scheme (MTSMFS) has been investigated [16]. Existing switching-based median filters [17-27] are commonly found to be non-adaptive to noise density variations and prone to misclassifying pixel characteristics. This exposes the critical need to evolve a sophisticated switching scheme and median filter. In order to improve filtering performances, decision-based median filtering schemes had been investigated. These techniques aim to achieve optimal performance over the entire image. A good noise filter is required to satisfy two criteria, namely, suppressing the noise and preserving the useful information in the signal.

A good noise filter is required to satisfy two criteria of (1) suppressing the noise while at the same time (2) preserving the useful information in the signal. Unfortunately, a great majority of currently available noise filters cannot simultaneously satisfy both of these criteria. The existing filters either suppress the noise at the cost of reduced noise suppression performance.

In order to address these issues, many neural networks have been investigated for image denoising. In the last few years, there has been a growing interest in the applications of soft computing techniques, such as neural networks and fuzzy systems, to the problems in digital signal processing. Neural networks are low-level computational structures that perform well when dealing with raw data although neural networks can learn.

Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the network function is determined largely by the connections between elements. This type of training is used to perform a particular function by adjusting the values of the connections (weights) between elements. Commonly neural networks are adjusted or trained to a specific target output which is based on a comparison of the output and the target, until the network output matches the target. Typically many such input/target pairs are needed to train a network.

A feed forward neural architecture with back propagation learning algorithms have been investigated [28-37] to satisfy both noise elimination and edges and fine details preservation properties when digital images are contaminated by higher level of impulse noise. Back propagation is a common method of training artificial neural networks algorithm so as to minimize the objective function. It is a multi-stage dynamic system optimization method. It is a supervised learning method, and is a generalization of the delta rule. It requires a dataset of the desired output for many inputs, making up the training set. It is most useful for feed-forward networks (networks that have no feedback, or simply, that have no connections that loop). The term is an abbreviation for "backward propagation of errors". Back propagation requires that the activation function used by the artificial neurons (or "nodes") be differentiable.

In addition to these, the back-propagation learning algorithm is simple to implement and computationally efficient in which its complexity is linear in the synaptic weights of the network. The input-output relation of a Feed forward adaptive neural network (FANN) can be viewed as a powerful nonlinear mapping. Conceptually, a feed forward adaptive network is actually a static mapping between its input and output spaces. Even though, intelligent techniques required certain pattern of data to learn the input. This pattern is given through conventional filter for training of the input. Therefore, intelligent filter performance depends on conventional filters performance. This work aims to achieving good de-noising without compromising on the useful information of the signal.

In this paper, a novel structure is proposed to eliminate the impulse noise and preserves the edges and fine details of digital images. A feed forward neural architecture with back propagation learning algorithm is used and is referred as an Artificial Intelligent Technique (AIE) for digital image enhancement.

The proposed intelligent filtering operation is carried out in two stages. In first stage the corrupted image is filtered by applying two special classes of filtering techniques. These filtered image outputs are suitably combined with a feed forward neural (FFN) network in the second stage. The internal parameters of the feed forward neural network are adaptively optimized by training of the feed forward back propagation algorithm.

The rest of the paper is organized as follows. Section 2 describes the noise model. Section 2 explains the structure of the proposed filter and its building blocks. Section 3 discusses the result of the proposed filter to the test images. Results of the experiments conducted to evaluate the performance of the proposed hybrid filter and comparative discussion of these results are also presented in this Section. 4 is the final Section, presents the conclusions.

2. Noise Model

Fundamentally, there are three standard noise models, which model the types of noise encountered in most images, they are additive noise, multiplicative noise and impulse noise. Digital image are often corrupted by salt and pepper noise (or impulse noise). Impulse noise is considered for proposed work. For images corrupted by salt-and-pepper noise (respectively fixed-valued noise), the noisy pixels can take only the maximum and the minimum values (respectively any random value) in the dynamic range. In other words, an image containing salt-and-pepper noise will have dark pixels in bright region and bright pixels in dark regions. A digital image function is given by $f(i,j)$ where (i,j) is spatial coordinate and f is intensity at point (i,j). Let $f(i,j)$ be the original image, $g(i,j)$ be the noise image version and η be the noise function, which returns random values coming from an arbitrary distribution. Then the additive noise is given by the equation (1)

$$g(i,j) = f(i,j) + \eta(i,j) \tag{2.1}$$

Impulse noise is caused by malfunctioning pixels in camera sensors, dead pixels, faulty memory locations in hardware, erroneous transmission in a channel, analog to digital converter, malfunctioning CCD elements (*i.e.*, hot and dead pixels) and flecks of dust inside the camera most commonly cause the considered kind of noise *etc*. It also creeps into the images because of bit errors in transmission, faulty memory locations and erroneous switching during quick transients. Two common types of impulse noise are the salt and pepper noise and the random valued noise. The proposed filter first detects the Salt and pepper noise present in digital images in very efficient manner and then removes it. As the impulse noise is additive in nature, noise present in a region does not depend upon the intensities of pixels in that region. Image corrupted with impulse noise contain pixels affected by some probability. The intensity of grayscale pixel is stored as an 8-bit integer giving 256 possible different shades of gray going from black to white, which can be represented as a [0, *L*-1] (*L* is 255) integer interval. In this paper the impulse noise is considered. In case of images corrupted by this kind of salt and pepper noise, intensity of the pixel A_{ij} at location (i,j) is described by the probability density function given by the following equation (2)

$$f(A_{ij}) = \begin{cases} p_a & for \;\; A_{ij} = a \\ 1 - p & for \;\; A_{ij} = Y_{ij} \\ p_b & for \;\; A_{ij} = b \end{cases} \qquad (2.2)$$

where a is the minimum intensity (dark dot); b is the maximum intensity (light dot); p_a is the probability of intensity (a) generation; p_b is the probability of intensity (*b)* generation; p is the noise density, and Y_{ij} is the intensity of the pixel at location (i, j) in the uncorrupted image. If either p_a or p_b is zero the impulse noise is called unipolar noise. If neither probability is zero and especially if they are equal, impulse noise is called bipolar noise or salt-and-pepper noise. Figure 1 shows the histogram of salt & pepper noise pattern with equal probabilities and impulse noise pattern is illustrated in Figure 1.

3. Proposed Filter

A Feed forward neural network is a flexible system trained by heuristic learning techniques derived from neural networks can be viewed as a 3-layer neural network with weights and activation functions.

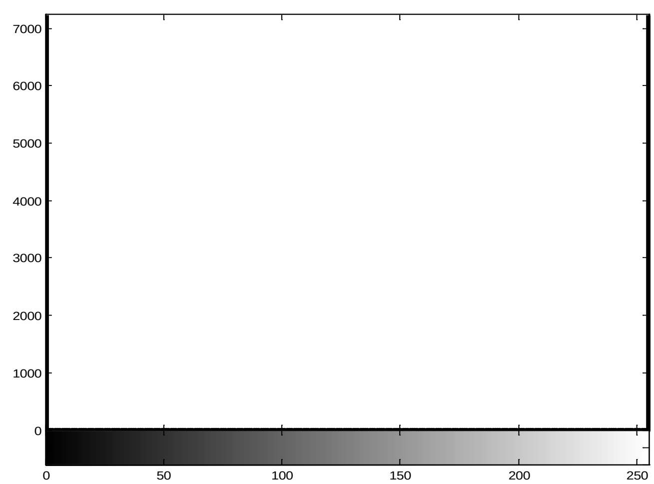

Figure 1. Salt and Pepper Noise Pattern using Histogram

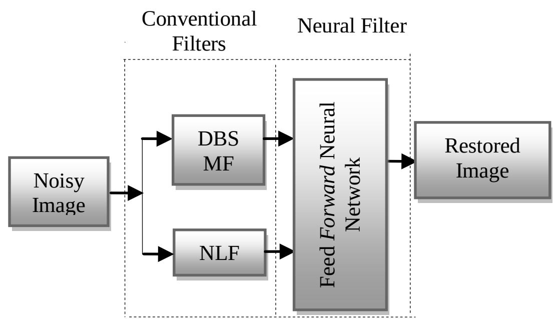

Figure 2. Block Diagram of Proposed Artificial Intelligent Technique

Figure 2 shows the structure of the proposed impulse noise removal filter. The proposed filter is obtained by appropriately combining output images from Decision based switching median filter (DBSMF) and Nonlinear filter with neural network. Learning and understanding aptitude of neural network congregate information from the two filters to compute output of the system which is equal to the restored value of noisy input pixel. The neural network learning procedure is used for the input-output mapping which is based on learning the proposed filter and the neural network utilizes back propagation algorithm. These two special classes of filters are described in Section 3.1 and 3.2.

3.1. Decision based Switching Median Filter (DBSMF)

This filtering technique has been investigated in the literature [22] for edge preserving properties of digital images. The filtering operation in this section is experimented by the way of edge preserving method to improve visual perception and then filtering is performed at the current pixel within the sliding window on digital image. Edge detection is a fundamental tool in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. Edges play an important role with these shapes because transition of one shape to another is controlled by the edge and its quality. The concept of hard, soft or lost edges control shape. All of this is conceptual with no basis in reality. In this paper, edges on the noisy image are identified using one of the properties of edge detection and preserved. Direction of orientation of edge is detected for filtering operation and this properties of edge detection and reduction of impulse noise is appropriately improves the learning and understanding skills of neural network.

In this filter, the central pixel is identified as corrupted one; it is replaced by the proposed edge preserving method. Therefore, edges on the image is detected by computing gradient value in the direction of horizontal, vertical, left diagonal and right diagonal within the filtering window respectively. Based on neighborhoods within the filtering window, the gradient value is obtained by determining the difference of two pixel intensities in direction of vertical (N and S), horizontal (W and E), left diagonal (SW and NE) and right diagonal (NW and SE) respectively. {NW = North West, N = north, NE = North East, W = west, E = east, SW = South West, S = south, SE = South East}. Each such direction with respect to (i,j) can also be linked to a certain position.

These four gradient values (according to the four different directions or neighbors) are considered for making the decision to eliminate impulse noise as well as preserve the edges of the image. If the gradient value is quite large, any one of the pixel is affected in the corresponding direction with minimum/maximum value of impulse noise. The minimum gradient value is a good indication that those pixels are noise free edge pixels in the direction of orientation. The minimum gradient value with respect to (i, j) can be used to determine direction of orientation of edge pixel. In order to preserve the edges, the corrupted central pixel is replaced by the average of two intensities which are obtained with respect to the direction of minimum gradient value. The filtering technique proposed in this paper employs a decision mechanism to detect the presence of impulse noise in the image as well as preserves the edges on digital images. This filtered output is a first input for neural network for training.

3.2. Nonlinear Filter (NF)

In this section, homogeneous region of image is preserved by applying decision based switching median filtering technique. The pixels inside the sliding window are classified as corrupted and uncorrupted pixels by comparing their values with the maximum (255) and minimum (0) values. Consider an image of size M×N having 8-bit gray scale pixel resolution. A two-dimensional square filtering window of size 3 x 3 is slid over the noisy image. As the window move over the noisy image, at each point the central pixel inside the window is checked whether it is a corrupted pixel or not. If the pixel is an uncorrupted one, it is left undisturbed and the window is moved to the next position.

On the other hand, if the pixel is detected as a corrupted one, the filtering procedure is performed by following the further steps described below. Separate the corrupted and uncorrupted pixels inside the filtering window at its current position. Check if the uncorrupted pixels inside the window add up to an odd number. If so, the median of the uncorrupted samples is set out as the filter output. If the uncorrupted samples sum up to an even number, then the Range estimator (RE) is determined for the uncorrupted samples. Range estimator for uncorrupted pixels is the difference between the last and the first of the sorted out uncorrupted pixels.

A suitable threshold value T is chosen for determining the presence of an edge at the central pixel. In this work, 125 is selected as suitable threshold. If RE is greater than the threshold value T, the central pixel is declared an edge and therefore, it left unaltered; otherwise, the central pixel is replaced by the arithmetic average of the uncorrupted pixels inside the filtering window. Then the window is moved to form a new set of values, with the next pixel to be processed at the centre of the window. This process is repeated until the last image pixel is processed. It may be noted that the filtering is performed by either taking the median or the arithmetic mean of uncorrupted pixels of the filtering window. Moreover, the mean filtering on the uncorrupted even number of sample is performed only if the central pixel is not an edge. Therefore the mean filtering is performed only in homogenous regions. As a result, the pixels in the filtered image do not cause any noticeable visual degradation. This filtered output is a second input for neural network for training.

3.4. Feed forward Neural Network

In feed forward neural network, back propagation algorithm is computationally effective and works well with optimization and adaptive techniques, which makes it very attractive in dynamic nonlinear systems. This network is popular general nonlinear modeling tool because it is very suitable for tuning by optimization and one to one mapping between input and output data. The input-output relationship of the network is as shown in Figure 3.

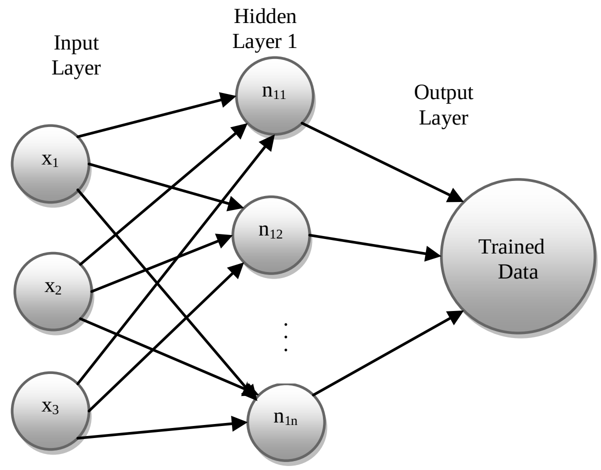

Figure 3. Feed Forward Neural Network Architecture

In Figure 3 x_m represents the total number of input image pixels as data, n_{kl} represents the number of neurons in the hidden unit, k represents the number hidden layer and l represents the number of neurons in each hidden layer. A feed forward back propagation neural network consists of two layers. The first layer or hidden layer, has a tan sigmoid (tan-sig) activation function is represented by

$$\phi(y_i) = \tanh(v_i) \tag{3.4.1}$$

This function is a hyperbolic tangent which ranges from -1 to 1, y_i is the output of the *i*th node (neuron) and v_i is the weighted sum of the input and the second layer or output layer, has a linear activation function. Thus, the first layer limits the output to a narrow range, from which the linear layer can produce all values. The output of each layer can be represented by

$$Y_{Nx1} = f(W_{NxM} X_{M,1} + b_{N,1}) \tag{3.4.2}$$

where Y is a vector containing the output from each of the N neurons in each given layer, W is a matrix containing the weights for each of the M inputs for all N neurons, **X** is a vector containing the inputs, b is a vector containing the biases and f(·) is the activation function for both hidden layer and output layer.

The trained network was created using the neural network toolbox from Matlab9b.0 release. In a back propagation network, there are two steps during training. The back propagation step calculates the error in the gradient descent and propagates it backwards to each neuron in the hidden layer. In the second step, depending upon the values of activation function from hidden layer, the weights and biases are then recomputed, and the output from the activated neurons is then propagated forward from the hidden layer to the output layer. The network is initialized with random weights and biases, and was then trained using the Levenberq-Marquardt algorithm (LM). The weights and biases are updated according to

$$Dn+1 = Dn - [J^T J + \mu I]^{-1} J^T e \tag{3.4.3}$$

where *Dn* is a matrix containing the current weights and biases, *Dn+1* is a matrix containing the new weights and biases, e is the network error, *J* is a Jacobian matrix containing the first derivative of e with respect to the current weights and biases, *I* is the identity matrix and μ is a variable that increases or decreases based on the performance function. The gradient of the error surface, g, is equal to *JTe*.

3.5. Training of the Feed Forward Neural Network

Feed forward neural network is trained using back propagation algorithm. There are two types of training or learning modes in back propagation algorithm namely sequential mode and batch mode respectively. In sequential learning, a given input pattern is propagated forward and error is determined and back propagated, and the weights are updated. Whereas, in Batch mode learning; weights are updated only after the entire set of training network has been presented to the network. Thus the weight update is only performed after every epoch. It is advantageous to accumulate the weight correction terms for several patterns.

For better understanding, the back propagation learning algorithm can be divided into two phases: propagation and weight update.

3.5.1. Propagation

Each propagation involves the following steps:

1. Forward propagation of a training pattern's input through the neural network in order to generate the propagation's output activations.
2. Backward propagation of the propagation's output activations through the neural network using the training pattern's target in order to generate the deltas of all output and hidden neurons.

3.5.2. Weight Update

For each weight-synapse follow the following steps:

1. Multiply its output delta and input activation to get the gradient of the weight.
2. Bring the weight in the opposite direction of the gradient by subtracting a ratio of it from the weight.

This ratio influences the speed and quality of learning; it is called the *learning rate*. The sign of the gradient of a weight indicates where the error is increasing; this is why the weight must be updated in the opposite direction. Repeat phase 1 and 2 until the performance of the network is satisfactory. In addition, neural network recognizes certain pattern of data only and also it entails difficulties to learn logically to identify the error data from the given input image. In order to improve the learning and understanding properties of neural network, different patterns are introduced to the neural network in terms of applying three different classes of nonlinear filtered output image data. Each filter has special preservation properties. First filter has better edge preserving properties. Second filter has superior performance in retaining the homogenous region of image. By combining these filters, the neural network has been trained for proposed filtering technique and also this will improve image quality.

Nonlinear filter is used to preserve the edges and fine details on digital images. Image edge is a fundamental feature of image, which contains abundant internal information such as direction, step characteristics, shape and etc. Decision based switching median filter is used to preserve the homogenous region of the images. Decision based switching median filter (DBSMF) and Nonlinear (NF) output images are considered as two inputs for neural network and noise free image is considered as a target image for training of the neural network. Back propagation is pertained as network training principle and the parameters of this network are then iteratively tuned. Once the training of the neural network is completed, its internal parameters are fixed and the network is combined with the nonlinear filters outputs to construct the proposed technique, as shown in Figure 4. While training a neural network, network structure is fixed and the unknown images are tested for given fixed network structure respectively. The performance evaluation is obtained through simulation results and shown to be superior performance to other existing filtering techniques in terms of impulse noise elimination and edges and fine detail preservation properties.

The feed forward neural network used in the structure of the proposed filter acts like a *mixture* operator and attempts to construct an enhanced output image by combining the information from the Decision based switching median filter (DBSMF) and Nonlinear filter. The rules of mixture are represented by the rules in the rule base of the neural network and the mixture process is implemented by the mechanism of the neural network. The feed forward neural network is trained by using back propagation algorithm and the parameters of the neural network are then iteratively tuned using the Levenberg–Marquardt optimization algorithm, so as to minimize the learning error, *e*. The neural network trained structure is optimized and the tuned parameters are fixed for testing the unknown images. The internal parameters of the neural network are optimized by training. Figure 4 represents the setup used for training and here, based on definition, the parameters of this network are iteratively optimized so that its output converges to original noise free image and completely removes the noise from its input image. The well known images are trained using this neural network and the network structure is optimized. The unknown images are tested using optimized neural network structure.

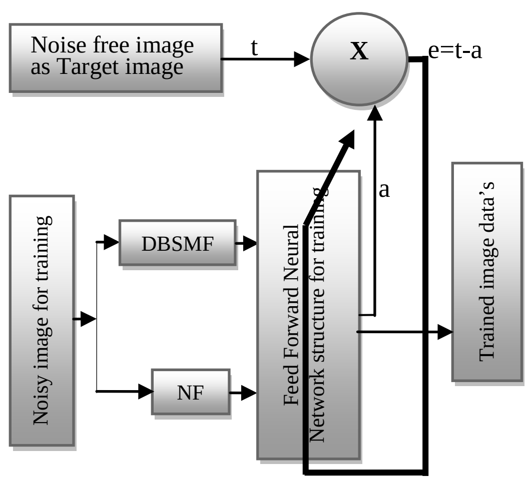

Figure 4. Training of the Feed forward Neural Network

In order to get effective filtering performance, already existing neural network filters are trained with image data and tested using equal noise density. But in practical situation, information about the noise density of the received signal is unpredictable one. Therefore; in this paper, the neural network architecture is trained using denoised three well known images which are corrupted by adding different noise density levels of 0.4, 0.45, 0.5 and 0.6 and also the network is trained for different hidden layers with different number of neurons. Noise density with 0.45 gave optimum solution for both lower and higher level noise corruption.

Therefore images are corrupted with 45% of noise is selected for training. Then the performance error of the given trained data and trained neural network structure are observed for each network. Among these neural network Structures, the trained neural network structure with the minimum error level is selected (10^{-3}) and this trained network structures are fixed for testing the received image signal.

Network is trained for 66 different architectures and corresponding network structure is fixed. PSNR is measured on Lena test image for all architectures with various noise densities. Among these, based on the maximum PSNR values; selected architectures is summarized in Table 4 for Lena image corrupted with 40% impulse noise. Finally, the maximum PSNR value with the neural network architecture of noise density 0.45 and single hidden layer with 3 neurons has been selected for training.

Figure 5 shows the images which are used for training. Three different images are used for network. This noise density level is well suited for testing the different noise level of unknown images in terms of quantitative and qualitative metrics. The image shown in Figure 5 ($a_{1,2\ \mathrm{and} 3}$) are the *noise free training image: cameraman Baboonlion and ship*. The size of an each training image is 256 x 256. The images in Figure 5 ($b_{1,2\ \mathrm{and}\ 3}$) are the *noisy training images* and is obtained by corrupting the noise free training image by impulse noise of 45% noise density. The image in Figure 5 ($c_{1,2\ \mathrm{and}\ 3}$) are the trained images by neural network. The images in Figure 5 (b) and (a) are employed as the *input* and the *target (desired)* images during training, respectively.

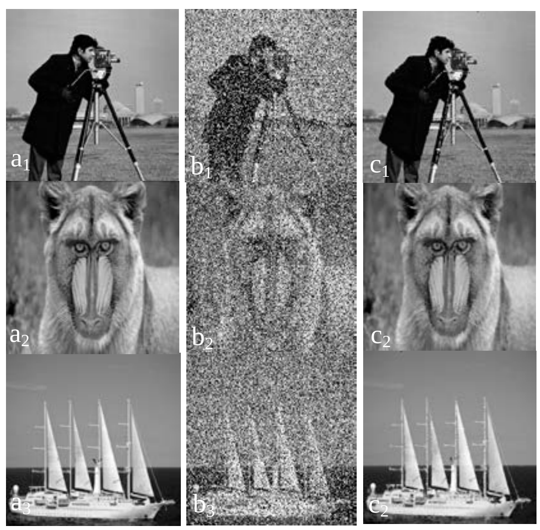

Figure 5. Performance of Training Image: ($a_{1,2\ \mathrm{and}\ 3}$) Original Images, ($b_{1,2\ \mathrm{and}\ 3}$) Images Corrupted with 45% of Noise and ($c_{1,\ 2\ \mathrm{and}\ 3}$) Trained Images

3.6. Testing of Unknown Images using Trained Structure of Neural Network

The optimized architecture that obtained the best performance for training with three images has 196608 data in the input layer, single hidden layer with 3 neurons and one output layer. The network trained with 45% impulse noise shows superior performance for testing under various noise levels. Also, to ensure faster processing, only the corrupted pixels from test images are identified and processed by the optimized neural network structure. As the uncorrupted pixels do not require further processing, they are directly taken as the output.

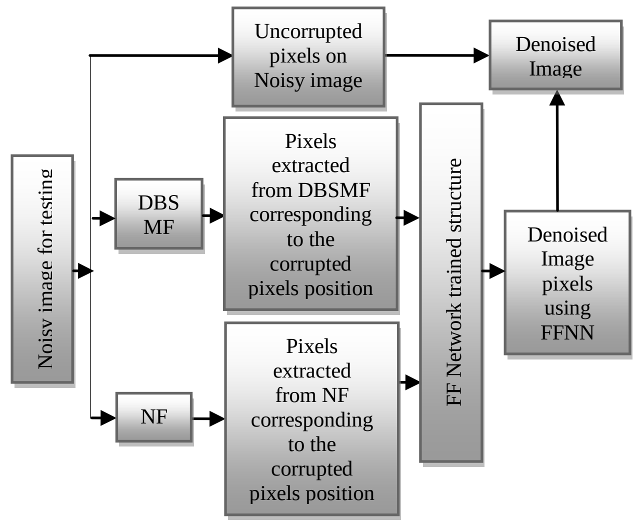

Figure 6. Testing of the Images using Optimized Feed Forward Adaptive Neural Network Structure

The chosen network has been extensively tested for several images with different level of impulse noise. Figure 6 shows the exact procedure for taking corrupted data for testing the received image signals for the proposed filter. In order to reduce the computation time in real time implementation; in the first stage, two different class of filters are applied on unknown images and then pixels (data) from the outputs of Decision based switching median filter and Nonlinear filter are obtained and applied as inputs for optimized neural network structure for testing; these pixels are corresponding to the pixel position of the corrupted pixels on noisy image. At the same time, noise free pixels from input are directly taken as output pixels. The tested pixels are replaced in the same location on corrupted image instead of noisy pixels. The most distinctive feature of the proposed filter offers excellent line, edge, and fine detail preservation performance and also effectively removes impulse noise from the image. Usually conventional filters give denoised image output and then these images are enhanced using these conventional outputs as input for neural filter while these outputs are combined with the network. Since, networks need certain pattern to learn and understand the given data.

3.7. Filtering of the Noisy Image

The noisy input image is processed by sliding the 3x3 filtering window on the image. This filtering window is considered for the two nonlinear filters. The window is started from the upper-left corner of the noisy input image, and moved rightwards and progressively downwards in a *raster scanning* fashion. For each filtering window, the nine pixels contained within the window of noisy image are first fed to the Decision based switching median filter and Nonlinear filter separately. Next, the center pixel of the filtering window on noisy image, the two outputs of the conventional filtered outputs are applied to the appropriate inputs for the neural network. Finally, the restored image is obtained at the output of this network.

4. Results and Discussion

The performance of the proposed filtering technique for image quality enhancement is tested for various level impulse noise densities. Four images are selected for testing with size of 256 x 256 including *Baboon, Lena, Pepper and Ship.* All test images are 8-bit gray level images. The experimental images used in the simulations are generated by contaminating the original images by impulse noise with different level of noise density. The experiments are especially designed to reveal the performances of the filters for different image properties and noise conditions. The performances of all filters are evaluated by using the *peak signal-to-noise ratio* (PSNR) criterion, which is defined as more objective *image* quality measurement and is given by the equation (4.1)

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \tag{4.1}$$

where

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left|(x(i,j)-y(i,j)\right|^2 \tag{4.2}$$

Here, M and N represents the number of rows and column of the image and $x(i, j)$ and $y(i, j)$ represents the original and the restored versions of a corrupted test image, respectively. Since all experiments are related with impulse noise.

The experimental procedure to evaluate the performance of a proposed filter is as follows: The noise density is varied from 10% to 90% with 10% increments. For each noise density step, the four test images are corrupted by impulse noise with that noise density. This generates four different experimental images, each having the same noise density. These images are restored by using the operator under experiment, and the PSNR values are calculated for the restored output images. By this method ten different PSNR values representing the filtering performance of that operator for different image properties, then this technique is separately repeated for all noise densities from 10% to 90% to obtain the variation of the average PSNR value of the proposed filter as a function of noise density. The entire input data are normalized in to the range of [0 1], whereas the output data is assigned to one for the highest probability and zero for the lowest probability.

Table 1. PSNR Obtained by Applying Proposed Filter on Lena Image Corrupted with 40 % of Impulse Noise

S.No	Neural network architecture				PSNR
	No. of hidden layers	No. of neuron in each hidden layer			
		Layer 1	Layer2	Layer3	
1	1	3	-	-	32.79
2	1	7	-	-	32.71
3	1	8	-	-	32.77
4	1	9	-	-	32.77
5	1	11	-	-	32.72
6	1	20	-	-	32.76
7	1	21	-	-	32.75
8	2	3	3	-	32.74
9	2	3	4	-	32.62
10	2	7	7	-	32.50

The architecture with single hidden layer and 3 neurons yielded the best performance.

The various parameters for the neural network training for all the patterns are summarized in Table 2 and 3. In Table 2, Performance error is nothing but Mean square error (MSE). It is a sum of the statistical bias and variance. The neural network performance can be improved by reducing both the statistical bias and the statistical variance. However there is a natural trade-off between the bias and variance. Learning Rate is a control parameter of training algorithms, which controls the step size when weights are iteratively adjusted. The learning rate is a constant in the algorithm of a neural network that affects the speed of learning. It will apply a smaller or larger proportion of the current adjustment to the previous weight If LR is low, network will learn all information from the given input data and it takes long time to learn. If it is high, network will skip some information from the given input data and it will make fast training. However lower learning rate gives better performance than higher learning rate. The learning time of a simple neural-network model is obtained through an analytic computation of the Eigen value spectrum for the Hessian matrix, which describes the second-order properties of the objective function in the space of coupling coefficients. The results are generic for symmetric matrices obtained by summing outer products of random vectors.

Table 2. Optimized Training Parameters for Feed Forward Neural Network

S.No	Parameters	Achieved
1	Performance error	0.00168
2	Learning Rate (LR)	0.01
3	No. of epochs taken to meet the performance goal	2500
4	Time taken to learn	1603 seconds

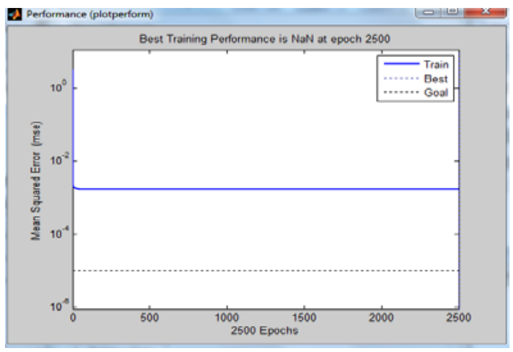

Figure 7. Performance Error Graph for Feed Forward Neural Network with Back Propagation Algorithm

Table 3. Bias and Weight Updation in Optimized Training Neural Network

Hidden layer		
	Weights	Bias
Weights from $x_{1,\ldots3}$ to n_1	-0.3353;0.2022	-0.4249
Weights from $x_{1,\ldots3}$ to n_2	0.6745;-0.4194	0.1020
Weights from $x_{1,\ldots3}$ to n_3	1.6673;-1.972	0.1169
Output layer		
Weights from n_1 to o	108.63	38.6822
Weights from n_2 to o	49.72	
Weights from n_3 to o	-1.470	

In Figure 7 and Figure 8 represent Performance error graph for error minimization and training state respectively. This Learning curves produced by networks using non-random (fixed-order) and random submission of training and also this shows the error goal and error achieved by the neural system. In order to prove the effectiveness of this filter, existing filtering techniques are experimented and compared with the proposed filter for visual perception and subjective evaluation on Lena image including the standard Median Filter (MF), the Weighted median filter (WMF), the Center weighted median filter (CWMF), the Tri state median filter (TSMF), a New impulse detector (NID), Multiple decision based median filter (MDBMF), Decision based median filter (DBSMF), Nonlinear filter (NF), Neural based post processing filtering techniques (NBPPFT), Modified Decision based Unsymmetric Trimmed Median Filter (MDBUTMF) and proposed filter in Figure 9.

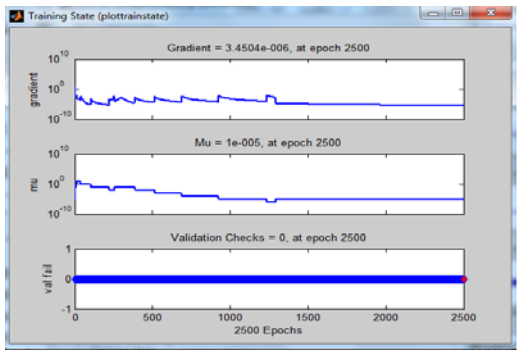

Figure 8. Performance of Gradient for Feed Forward Neural Network with Back Propagation Algorithm

Figure 9. Performance of Test Image: Lena (a) Noise Free Images, (b) Image Corrupted by 50% Impulse Noise (c) Images Restored by MF (d) Images Restored by WMF (e) Images Restored by CWMF (f) Images Restored by TSMF (g) Images Restored by MDBSMF (h) Images Restored by NID (i) Images Restored by DBSMF (j) Images Restored by NF and (k) Image Restored by NBPPFT,(l) Image Restored by MDBUTMF and (m) Image Restored by the Proposed Filter

Lena test image contaminated with the impulse noise of various densities are summarized in Table3 for quantitative metrics for different filtering techniques and compared with the proposed filtering technique and is graphically illustrated in Figure10. The summarized values for decision based switching median filter; nonlinear filter and the proposed filter are graphically illustrated in Figure 11 for the performance comparison of the proposed intelligent filter. This qualitative measurement proves that the proposed filtering technique outperforms the other filtering schemes for the noise densities up to 50%.

The PSNR performance explores the quantitative measurement. In order to check the performance of the feed forward neural network, percentage improvement (PI) in PSNR is also calculated for performance comparison between conventional filters and proposed neural filter for Lena image and is summarized in Table 4. This PI in PSNR is calculated by the following equation 8.

$$PI = \left[\frac{PSNR_{CF} - PSNR_{NF}}{PSNR_{CF}} x100 \right] \quad (4.3)$$

where PI represents percentage in PSNR, $PSNR_{CF}$ represents PSNR for conventional filter and $PSNR_{NF}$ represents PSNR values for the designed neural filter.

Table 3. Performance of PSNR for Different Filtering Techniques on Lena Image Corrupted with Various % of Impulse Noise

Filtering Techniques	Noise level				
	10	30	50	70	90
MF	31.74	23.20	15.28	9.98	6.58
WMF	23.97	22.58	20.11	15.73	8.83
CWMF	28.72	20.28	14.45	10.04	6.75
TSMF	31.89	23.96	15.82	10.33	6.58
MDBSMFS	34.83	24.79	16.99	11.28	6.97
NID	37.90	28.75	23.42	14.65	7.77
DBSMF	40.8	31.0	22.6	13.42	7.06
NF	38.42	30.47	24.92	18.84	10.03
MDBUTMF	37.91	32.29	28.18	24.30	18.40
NBPPFT	40.75	34.11	28.77	24.24	18.07
Proposed Filter	44.06	36.31	29.32	22.34	15.43

Here, the conventional filters are combined with neural network which gives the proposed filter, so that the performance of conventional filter is improved.

Table 4. Percentage Improvement in PSNR Obtained on Lena Image Corrupted with Different Level of Impulse Noise

Noise	Proposed	NF	PI for	DBSMF	PI

%	filter (PF)		PF		for PF
10	44.06	40.8	7.99	38.42	14.67
20	40.17	35.8	12.2	34.28	17.18
30	36.31	31.0	17.12	30.47	19.16
40	32.75	27.3	19.96	27.38	19.61
50	29.32	22.6	29.73	24.92	17.65
60	25.99	17.6	47.7	22.05	14.82
70	22.34	13.42	66.46	18.84	18.57
80	18.35	9.63	90.55	14.12	29.95
90	15.43	7.06	100	10.03	53.83

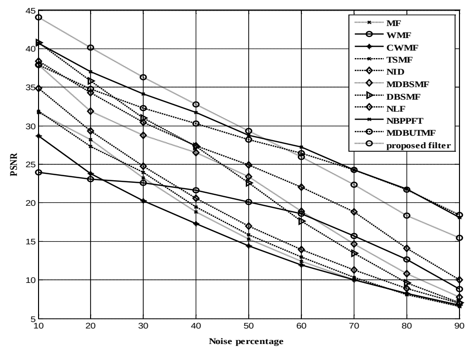

Figure 10. PSNR Obtained using Proposed Filter and Compared with Different Filtering Techniques on Lena Image Corrupted with Different Densities of Impulse Noise

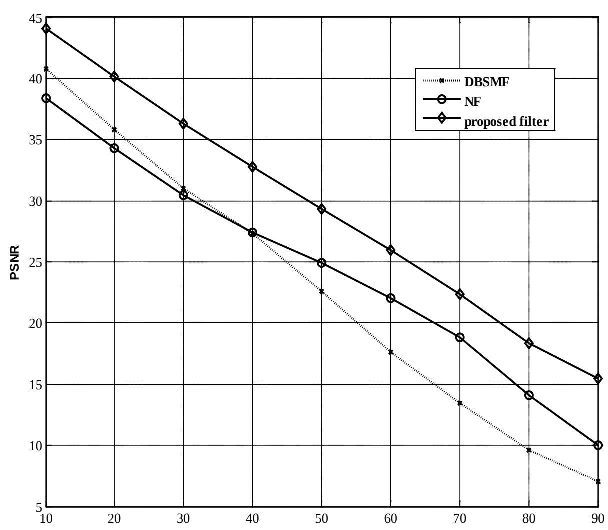

Figure 11. PSNR Obtained using proposed Filter and Compared with NF and DBSM Filtering Techniques on Lena Image Corrupted with Different Densities of Impulse Noise

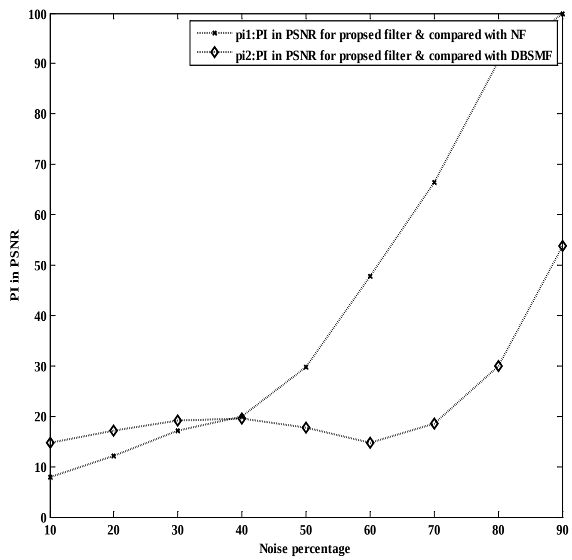

Figure 12. PI in PSNR Obtained on Lena Image for the Proposed Filter Corrupted with Various Densities of Mixed Impulse Noise

The summarized PSNR values in Table 4 for the proposed neural filter appears to perform well for human visual perception when images are corrupted up to 50% of impulse noise. These filters performance are better for quantitative measures when images are corrupted up to 70% of impulse noise. PI is graphically illustrated in Figure 12. Digital images are nonstationary process; therefore depends on properties of edges and homogenous region of the test images, each digital images having different quantitative measures.

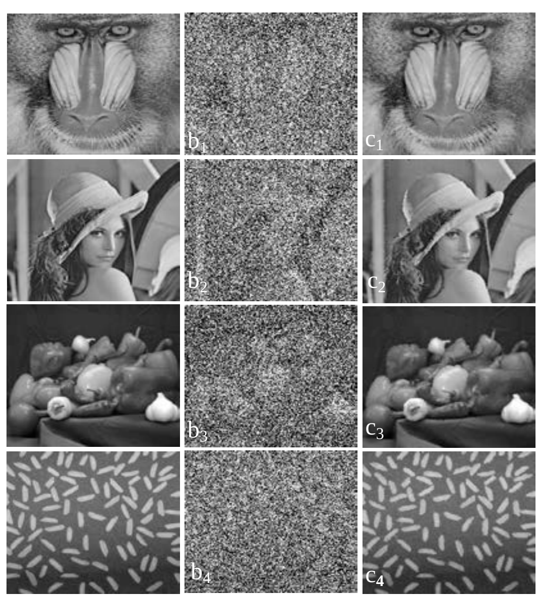

Figure 13. Performance of Test Images: ($a_{1,2 \text{ and } 3}$) Original Images,($b_{1,2 \text{ and } 3}$) Images Corrupted with 50% of Noise and ($d_{1, 2 \text{ and } 3}$) Images Enhanced by Proposed Filter

Figure 13 illustrate the subjective performance for proposed filtering Technique for Baboon, Lena, Pepper and Rice images: noise free image in first column, images corrupted with 50% impulse noise in second column, Images restored by proposed Filtering Technique in third column. This will felt out the properties of digital images. Performance of quantitative analysis is evaluated and is summarized in Table 5. This is graphically illustrated in Figure 14. The qualitative and quantitative performance of Pepper and Rice images are better than the other images for the noise levels ranging from 10% to 50%. But for higher noise levels, the Pepper image is better.

The Baboon image seems to perform poorly for higher noise levels. Based on the intensity level or brightness level of the image, it is concluded that the performance of the images like pepper, Lena, Baboon and Rice will change. Since digital images are nonstationary process. The proposed filtering technique is found to have eliminated the impulse noise completely while preserving the image features quite satisfactorily. This novel filter can be used as a powerful tool for efficient removal of impulse noise from digital images without distorting the useful information in the image and gives more pleasant for visual perception.

In addition, it can be observed that the proposed filter for image enhancement is better in preserving the edges and fine details than the other existing filtering algorithm. It is constructed by appropriately combining a two nonlinear filters and a neural network. This technique is simple in implementation and in training; the proposed operator may be used for efficiently filtering any image corrupted by impulse noise of virtually any noise density. Further, it can be observed that the proposed filter for image quality enhancement is better in preserving the edges and fine details than the other existing filtering algorithm.

The advantages of the newly proposed filter may be summarized as follows:

1) It has a very simple structure. It is constructed by appropriately combining a NF filter, modified canny edge detector and a NF network. The structure of the neural network is also very simple.

2) It does not require user-supplied heuristic tuning parameters. The internal parameter of the proposed operator is adaptively tuned by training.

3) The training is easily accomplished by using very simple images. However, contrary to its simplicity in implementation and convenience in training, the proposed operator may be used for efficiently filtering any image corrupted by impulse noise of virtually any noise density.

It is concluded that the proposed filtering technique can be used as a powerful tool for efficient removal of impulse noise from digital images without distorting the useful information within the image**.**

5. Conclusion

An Artificial Intelligent Technique (AIT) is described in this paper. This filter is seen to be quite effective in preserving image boundary and fine details of digital images while eliminating impulse noise. The efficiency of the proposed filter is illustrated applying the filter on various test images contaminated different levels of noise. This filter outperforms the existing median based filter in terms of objective and subjective measures. The corrupted pixels from input image alone are taken for testing the unknown digital images. As a result, misclassification of pixels is avoided. So that the proposed filter output images are found to be pleasant for visual perception.

References

[1] J. Astola and P. Kuosmanen, “Fundamental of Nonlinear Digital Filtering”, NewYork:CRC, **(1997)**.
[2] I. Pitasand and N. Venetsanooulos, “Nonlinear Digital Filters:Principles Applications”, Boston, MA: Kluwer, **(1990)**.
[3] W. K. Pratt, “Digital Image Processing”, Wiley, **(1978)**.
[4] T. Chen, K.-K. Ma and L.-H. Chen, “Tristate median filter for image denoising”, IEEE Trans. Image Process., vol. 8, no. 12, **(1991)**, pp. 1834-1838.
[5] Z. Zeng and J. Wang, “Advances in Neural Network Research and Applications”, Lecture notes springer, **(2010)**.
[6] M. Barni, V. Cappellini and A. Mecocci, “Fast vector median filter based on Euclidian norm approximation”, IEEE Signal Process. Lett., vol. 1, no. 6, **(1994)** June, pp. 92-94.
[7] S. Hoyos and Y. Li, “Weighted Median Filters Admitting Complex-Valued Weights and their Optimization”, IEEE transactions on Signal Processing, vol. 52, no. 10, **(2004)** October, pp. 2776-2787.
[8] E. Abreu, M. Lightstone, S. K.Mitra and K. Arakawa, “A new efficient approach for the removal of impulse noise from highly corrupted images”, IEEE Trans. Image Processing, vol. 5, no. 6, **(1996)**, pp. 1012-1025.
[9] T. Sun and Y. Neuvo, “Detail preserving median filters in image processing”, Pattern Recognition Lett., vol. 15, no. 4, **(1994)** April, pp. 341-347.
[10] Zhang and M.- A. Karim, “A new impulse detector for switching median filters”, IEEE Signal Process. Lett., vol. 9, no. 11, **(2002)** November, pp. 360-363.
[11] Z. Wang and D. Zhang, “Progressive Switching median filter for the removal of impulse noise from highly corrupted images”, IEEE Trans. Circuits Syst. II, vol. 46, no. 1, **(2002)** January, pp. 78-80.
[12] H.-L. Eng and K.-K. Ma, “Noise adaptive soft–switching median filter”, IEEE Trans. Image Processing, vol. 10, no. 2, **(2001)** February, pp. 242-25.
[13] P.-E. Ng and K-K. Ma, “A Switching median filter with boundary Discriminative noise detection for extremely corrupted images”, IEEE Transactions on image Processing, vol. 15, no. 6, **(2006)** June, pp. 1500-1516.
[14] T.-C. Lin and P.-T. Yu, “Salt-Pepper Impulse noise detection”, Journal of Information Science and Engineering, vol. 4, **(2007)** June, pp. 189-198.
[15] E. Srinivasan and R. Pushpavalli, “Multiple Thresholds Switching Median Filtering for Eliminating Impulse Noise in Images”, International conference on Signal Processing, CIT, **(2007)** August.
[16] R. Pushpavalli and E. Srinivasan, “Multiple Decision Based Switching Median Filtering for Eliminating Impulse Noise with Edge and Fine Detail preservation Properties”, International conference on Signal Processing, CIT, **(2007)** August.
[17] Y. Zhouand Quan-huanTang, “Adaptive Fuzzy Median Filter for Images Corrupted by Impulse Noise”, Congress on image and signal processing, vol. 3, **(2008)**, pp. 265-269.
[18] S. Kaisar and J. AI Mahmud, “Salt and Pepper Noise Detection and removal by Tolerance based selective Arithmetic Mean Filtering Technique for image restoration”, IJCSNS, vol. 8, no. 6, **(2008)** June, pp. 309-313.
[19] T. C. Lin and P. T. Yu, “Adaptive two-pass median filter based on support vector machine for image restoration”, Neural Computation, vol. 16, **(2004)**, pp. 333-354.
[20] M. S. Nair, K. Revathy and R. Tatavarti, “An Improved Decision Based Algorithm For Impulse Noise Removal”, Proceedings of International Congress on Image and Signal Processing-CISP 2008, IEEE Computer Society Press, Sanya, Hainan, China, vol. 1, **(2008)** May, pp. 426-431.
[21] V. Jayaraj and D. Ebenezer, “A New Adaptive Decision Based Robust Statistics Estimation Filter for High Density Impulse Noise in Images and Videos”, International conference on Control, Automation, Communication and Energy conversion, **(2009)** June, pp. 1-6.
[22] F. Duan and Y.-J. Zhang, “A Highly Effective Impulse Noise Detection Algorithm for Switching Median Filters”, IEEE Signal processing Letters, vol. 17, no. 7, **(2010)** July, pp. 647-650.
[23] R. Pushpavalli and G. Sivaradje, “Nonlinear Filtering Technique for Preserving Edges and Fine Details on Digital Image”, International Journal of Electronics and Communication Engineering and Technology, vol. 3, no. 1, **(2012)** January, pp. 29-40.
[24] R. Pushpavalli and E. Srinivasan, “Decision based Switching Median Filtering Technique for Image Denoising”, CiiT International journal of Digital Image Processing, vol. 2, no. 10, **(2010)** October, pp. 405-410.
[25] R. Pushpavalli, E. Srinivasan and S. Himavathi, “A New Nonlinear Filtering technique”, 2010 International Conference on Advances in Recent Technologies in Communication and Computing, ACEEE, **(2010)** October, pp. 1-4.
[26] R. Pushpavalli and G. Sivaradje, “New Tristate Switching Median Filter for Image Enhancement”, International Journal of Advanced research and Engineering Technology, vol. 3, no. 1, **(2012)** January-

June, pp. 55-65.
[27] A. Fabijanska and D. Sankowski, "Noise adaptive switching median-based filter for impulse noise removal from extremely corrupted images", IET image processing, vol. 5, no. 5, **(2010)** July, pp. 472-480.
[28] S. Esakkirajan, T. Veerakumar, A. N. Subramanyam and C. H. Premchand, "Removal of High Density Salt and pepper Noise through Modified Decision based Unsymmetric Trimmed Median Filter", IEEE Signal processing letters, vol. 18, no. 5, **(2011)** May, pp. 287-290.
[29] A. L. Betker, T. Szturm and Z. oussavi1, "Application of Feed forward Back propagation Neural Network to Center of Mass Estimation for Use in a Clinical Environment", IEEE Proceedings of Engineering in Medicine and Biology Society, vol. 3, **(2004)** April, pp. 2714-2717.
[30] C. Jindu and D. Runtao, "A Feed forward neural Network for Image processing", in IEEE proceedings of ICSP, **(1996)**, pp. 1477-1480.
[31] W. Qian, H. Li, M. Kallergi, D. Song and L. P. Clarke, "Adaptive Neural Network for Nuclear Medicine Image Restoration", Journal of VLSI Signal Processing, Kluwer Academic Publishers, **(1998)**, vol. 18, pp. 297-315.
[32] R. Pushpavalli, G. Shivaradje, E. Srinivasan and S. Himavathi, " Neural Based Post Processing Filtering Technique For Image Quality Enhancement", International Journal of Computer Applications, **(2012)** January.
[33] G. Panchal, A. Ganatra, Y. P. Kosta and D. Panchal, "Forecasting Employee Retention Probability using Back Propagation Neural Network Algorithm", Second International Conference on Machine Learning and Computing, **(2010)**, pp. 248-251.
[34] S. Kumar Misra, G. Panda and S. Mehar, "Cheshev Functional link artificial neural Networks for Denoising of Image Corrupted by Salt & Pepper Noise", International journal of recent Trends in Engineering, vol. 1, no. 1, **(2009)** May, pp. 413-417.
[35] W. Hong, W. Chen and R. Zhang, "The Application of Neural Network in the Technology of Image processing", Proceddings of the International Conference of Engineers and Computer Sciences, vol. 1, **(2009)**.
[36] "A new methods of denoising mixed noise using limited Grayscale Pulsed Coupled Neural Network", Cross Quad-Regional Radio Science and Wireless Technology Conference, **(2011)**, pp. 1411-1413.
[37] S. Tiwari, A. kumar Singh and V. P. Shukla, "Staistical Moments based Noise Classification using Feed Forward back Peopagation neural Network", International journal of Computer Applications, vol. 18, no. 2, **(2011)** March, pp. 36-40.
[38] A. Sharma, "Gradient Descent Feed Forward Neural Networks for Forecasting the Trajectories", International Journal of Advanced Science and Technology, vol. 34, **(2011)** September, pp. 83-88.

Authors

R. Pushpavalli received the B.E. degree in department of Electronic and Communication Engineering from Bharathiar University, Coimbatore, India in April 1997 and M.Tech degree in electronic and Communication engineering from Pondicherry Engineering College, Pondicherry University, India, in May 2007 respectively. In September 2000, she joined as lecturer in the department of ECE, Dr. Navalar Neduchezhiyan College of Engineering and Technology, Madras University, India. From June 2001 to August 2005 and from June 2007 to October she was a lecturer at SKP Engineering, Anna University. She got 2 years experience from Rajiv Gandhi College of Engineering, Puducherry, India. Currently she is a Full Time Research Scholar in the department of ECE, Pondicherry Engineering College, Puducherry, India. In 2007, she got a membership in ISTE.

G. Sivarajde received the B.E. degree in department of Electronic and Communication Engineering from Madras University, Chennai, India in April 1991 and M.Tech degree in electronic and Communication engineering from Pondicherry Engineering College, Pondicherry University, India, in February 1996 and he has obtained the Ph.D degree in electronic and Communication engineering from Pondicherry Engineering College, Pondicherry University, India, in December 2006 respectively. He is currently an Professor in the department of electronic and Communication engineering in Pondicherry Engineering College, Pondicherry University, India. He has published more than 65 papers in both International and National conferences and also he has published 14 national journals and 45 international journals. His research interests Mobile Computing, Convergence Networks. He is a senior member in IEEE, member in ISTE, CSI and IETE.