Multimed Tools Appl (2018) 77:24405–24425
https://doi.org/10.1007/s11042-017-5592-y

# Jaya based functional link multilayer perceptron adaptive filter for Poisson noise suppression from X-ray images

**M. Kumar**[1] · **S. K. Mishra**

Received: 29 May 2017 / Revised: 27 October 2017 / Accepted: 27 December 2017 /
Published online: 14 February 2018


**Abstract** In this paper, a parameterless Jaya optimization based neural network filter named as Jaya-functional link multilayer perceptron (Jaya-FLMLP) is proposed for the elimination of Poisson noise from X-ray images. In this proposed adaptive filter, Jaya is applied for updating the weights of the FLMLP network. The proposed neural filter is a combination of a functional link artificial neural network (FLANN) and Multilayer Perceptron (MLP) network. The performance of Jaya-FLMLP is also compared with other five competitive networks such as Wiener, MLP, Least Mean Squares based Functional Link Artificial Neural Network (LMS-FLANN), Particle Swarm Optimization based Functional Link Artificial Neural Network (PSO-FLANN) and Cat Swarm Optimization based Functional Link Artificial Neural Network (CSO-FLANN). The comparison of performance is investigated by the Structural Similarity Index (SSIM), Peak Signal to Noise Ratio (PSNR) and Noise Reduction in Decibels (NRDB) values. The simulation results and non-parametric Friedman's test reveal the superiority of the Jaya-FLMLP filter over others.

**Keywords** Poisson noise · X-ray image · Adaptive filter · Artificial neural network · Optimization · Friedman's test

## 1 Introduction

The X-ray radiography is popularly used in non-invasive medical image modalities and provides distinctive information of internal musculoskeletal lesion. The acquired image depends heavily on randomly released photons from a source that passes through a patient's body towards a specified pixel on the detector. The entire process of image reconstruction follows the random probability procedure i.e. Poisson law, and due to this reason, the acquired medical images may get deteriorated by Poisson noise [14]. It degrades the contrast, and clarity

✉ M. Kumar
manish.guptasssss007@gmail.com; phdee10051.13@bitmesra.ac.in

[1] Birla Institute of Technology, Ranchi, India

and also weakens the clinical information from the acquired images. Enhancement techniques are required to accumulate significant information from the acquired noisy medical images. It assists the doctor in identifying pathological symptoms. Therefore, the denoised medical X-rays image is desirable not only for the diagnosis but also it can be further applied for post-processing tasks like segmentation, registration, analysis and telemedicine.

Moreover, to develop an effective and efficient denoising method is still a challenging task for researchers because most of the algorithms did not gain the appropriate applicability level and lack generality. The conventional fixed filters such as median, mean, rank order mean filter, weighted mean, etc. suppress a particular kind of noise and require prior knowledge of noise. These filtering techniques depend upon different noise conditions and nature of images. In fact, getting information about noise every time as well as selecting a filter accordingly is a time-consuming task for any manual or automated system. Hence, to avoid such limitations of conventional filters, adaptive filters are introduced. The parameters of an adaptive filter can be tuned by itself, according to the change in image local statistics as well as according to the noise characteristics. Such techniques are not only saving the valuable pre-processing operational time of the radiologist but are also very useful in situations like war and natural calamities where the number of patients is very high. The neural network filtering is a nonlinear adaptive filtering approach which substantially improves the overall filtering operation.

Over the past three decades, artificial neural network (ANN) techniques are being employed in image processing tasks, such as classification, recognition, segmentation and restoration purpose [2, 4, 5, 29]. The ANN's are efficient in performing the complex mapping between the input and output space, and thus, these techniques can form subjective nonlinear decision boundaries. Initially, the feed-forward multilayer perceptron (MLP) networks are mostly realized to get rid of the noise. However, designing MLP networks is a very complex task because of more than one number of hidden layers and many numbers of nodes present in their structure. The efficiency of any network depends upon controlling parameters like learning rate, which is again a complicated task and sometimes quite intuitive. To overcome such difficulties, the single layer FLANN was introduced by Pao [17] in the year 1989. It is a functionally expanded neural network that provides better results in terms of computation time with a faster convergence rate than the MLP. It has been effectively applied in the field of active noise control in less computational time [24]. Although different spurious noises are removed by the FLANN network, it does not ensure a refined outcome as compared to the MLP network in some of the other challenging applications. This is due to the lack of a universal approximation characteristic of any neural network based filter. Hence, the intent of this research article is to introduce a network named as FLMLP that can overcome the drawback of both MLP and FLANN, by exploiting their merits. Instead of the classical derivative approach to train this network, a recently developed parameter less heuristic-technique, i.e. Jaya, has been applied as a learning technique. This heuristic technique will avoid trapping of solutions at local minima, with the intention that the cost function (estimated restoration error) can be minimized. Network sluggishness is also circumvented by the proposed algorithm which is usually realized during gradient based learning.

The architecture of this article is as follows: Section 2 reveals the previous research in the field image denoising using ANN based neural filters. Section 3 describes the way in which the Jaya-FLMLP filter is formulated and applied for the X-ray's image denoising process. In

Section 4, the qualitative and qualitative results of the employed filtering scheme are highlighted. Sections 5 and 6 present the discussion and conclusion of this paper.

## 2 Related work

The role and development of the neural network in the field of the image preprocessing task such as denoising, enhancement, restoration, de-blurring have been reported by many researchers. The neural network can be classified depending on the learning model and the optimization algorithm applied to update the weights. How nature-inspired optimization technique based neural filter is utilized to remove noise from different images is discussed in this section which emphasizes the significance of the proposed methodology.

### 2.1 Derivative learning based neural network filter

Zhou *et al.* [34] have presented the application of a nonlinear multilayer neural network based filter in the area of image restoration. Similarly, Siva *et al.* [25] have discussed the utility of a neural network filter and also shows the limitation of different linear restoration filters such as Wiener, Kalman, inverse and pseudo inverse filter. The limitation includes the assumption of any system as a wide sense stationary process and the condition of high signal to noise ratio. Also, the computational complexity is very high for these types of filters. Similarly, many researchers have applied nonlinear spatial filters such as Anisotropic Diffusion (AD), Non-Local Means (NLM), Total Variance (TV), Partial Differential Equation (PDE) etc. to overcome these bottom-holes [1, 8, 10, 30]. However, these filters have many arbitrary parameters which regulate their performance. Furthermore, all of them are computationally expensive and effective for a particular noise. Suzuki *et al.* [1, 26] have shown the different multilayer networks based adaptive filter for the reduction of noise from radiographic X-ray and other images while preserving the edges. Zhao et al. [33] have discussed the importance of a single layer FLANN filter over other feedforward multilayer networks due to the reduced computational burden because of the presence of single layer. It expands the dimensionality of the input image pixel data using nonlinear independent standard functions, such as Chebyshev, trigonometric, power, exponential, etc., and it can be utilized in the problem of image segmentation, restoration, and identification [9, 15, 18]. At the same time, Radial Basis Function (RBF) has limited hidden layers and attracted the attention of researchers as an alternative to MLP. RBF performs very well with less computational resources. However, performances wise RBF is highly dependent upon the truncated Volterra series [32]. Hence, FLANN is recommended by many researchers to solve nonlinearity and any optimization problem. Usually, training of neural network filters is executed through gradient descent (derivative) learning techniques, and these techniques are applied for obtaining the optimal set of weight. Here, the error is considered to be a cost function that needs to minimize. However, these techniques are not recommended for flat or irregular functions. Apart from the weight selection, various other parameters are associated and need to fix carefully for the seamless working of neural network filter. The simple gradient descent learning based neural network filters have all these restrictions and hence not endorsed in the highly nonlinear environment. To eliminate above intricacy, heuristic (nature-inspired/evolutionary) learning based neural network filters are introduced where

heuristic technique has been applied for adjusting weights of neural network filtering model by overcoming all the bottleneck of derivative based neural network filter.

### 2.2 Heuristic learning based neural network filter

Any neural network has the ability to learn from the optimization of a suitable error function. Hence, optimization methods such as derivative or evolutionary based approaches could be used in learning. Though the gradient based techniques are very efficient, they may get stuck at the local and global optimal point of the solution and thus are not preferable for irregular function [7, 16, 31]. In fact, state-of-the-art evolutionary techniques are utilized for training the neural network in place of gradient approaches such as Least Mean Squares (LMS), Leaky LMS (L-LMS), Normalize LMS (N-LMS), Recursive Least Squares (RLS) and the Backpropagation (BP) algorithm. The optimization technique could be a Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), Cat Swarm Optimization (CSO), Ant Colony Optimization (ACO), etc. Similarly, authors in [23] have discussed other benefits of ANN's (a) in hardware implementation such as FPGA and VLSI, (b) providing parallel processing and (c) performing an adaptive and nonlinear task like the de-blur of an image. They have also discussed the efficacy of the metaheuristic ABC algorithm for the training of neural network filter and have successfully applied to de-blur the various medical radiological images. Kumar et al. have provided a brief idea of the FLANN filter, and also discussed the application of Particle Swarm Optimization (PSO), and Cat Swarm Optimization (CSO) in selecting the best possible weights of in such a way that the estimated error function get minimized in [11, 13]. They proposed the FLANN network for removing noise from CT and X-ray images. Similarly, authors in [3, 6] cast off nature inspired techniques such as the GA and pigeon inspired based ANN filters to denoise several digital images. Wang et al. have proposed a novel single layer neural filter and applied the Predator-Prey PSO for updating weights [28].

Parameter dependent techniques optimize mostly all the above mentioned heuristic neural network filters, and the efficiency of these filters depends on the controlling parameters value. These parameters could be a learning rate, local, social or inertial parameters, crossover or mutation etc. Hence, for this proposed work, the Jaya optimization technique is applied for the training of the FLMLP filter, and collectively it is named as Jaya-FLMLP. The performance of the recently introduced Jaya algorithm does not get affected due to any controlling parameter.

## 3 Proposed Jaya-functional link multilayer perceptron (Jaya-FLMLP) filter

### 3.1 Multilayer perceptron (MLP) filter

The structure of an MLP filter consists of one or more hidden layer between the input and output layers. The Fig. 1 shows the structure of the Multilayer Neural Network filter for image denoising. Initially, all the nodes of any layers and its adjacent layer are interconnected with a pre-assigned set of random weights. In the input layer, '$X_i(n)$' are the noisy pixel intensities which are supplied to the inputs nodes of MLP network. These pixel intensities are obtained from the dynamic window of the noisy image.

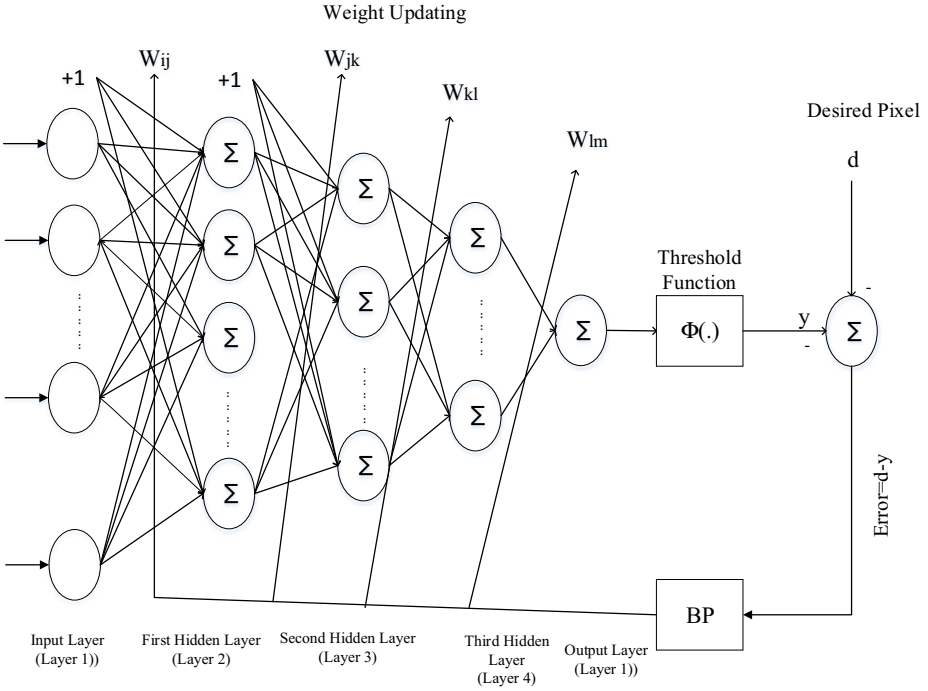



**Fig. 1** Structure of Multilayer Neural Network filter

Usually, the size of the window is considered as 3 × 3 hence the number of inputs become 9. All the nodes of hidden and output layers comprise of nonlinear threshold (activation) function known as sigmoid function. The weighted sum of inputs of each layer, and activation function of both hidden and output layers are utilized to evaluate the final output i.e. '*y*'. The final output '*y*' and center most pixel '*d*' of the corresponding window of the target (noise-free) image are to be compared to evaluate error '*e*' which is subsequently used as an objective function. The Backpropagation (BP) algorithm attempts to minimize this objective function by updating all the weights of the network. Similarly, the above-discussed steps are going to repeat for the other windows starting from top left corner to bottom right corner of the image. The iteration will terminate if it encounters the stopping criteria i.e. the maximum number of iteration, or mean square error below the certain threshold. After the completion of iteration, the MLP network gets trained and obtained optimal weights is utilized for filter out any other noisy pixel.

### 3.2 Functional link artificial neural network (FLANN) filter

FLANN is a variant of neural network based filter, and unlike MLP network, it has no hidden layer. The BP learning algorithm is applied to update the weights of MLP which is complex and computationally expensive. As FLANN network is a single layer network, it avoids this cumbersome algorithm and improves the training time significantly by applying simple algorithms like LMS, L-LMS, N-LMS etc.

The noisy input data of the FLANN filter are expanded nonlinearly by applying functions such as power, trigonometric, exponential, Chebyshev, *etc*. Fig. 2 shows the architecture of the FLANN filter, where, 'X1, X2 … … X9' are the nine noisy input pixels obtained from the dynamic window

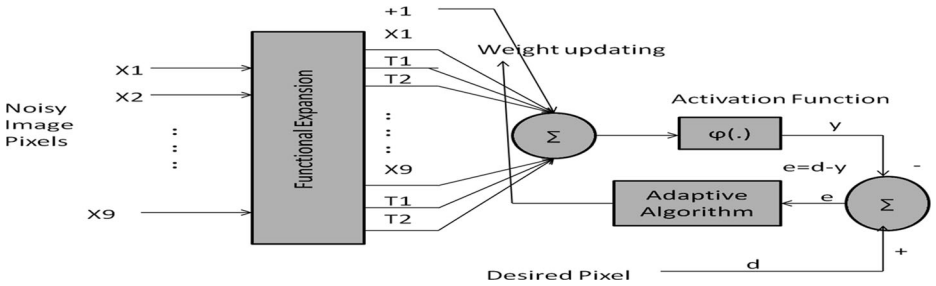



**Fig. 2** The structure of FLANN filter

of size 3 × 3. Then, 'X1' is expanded by any of above-mentioned nonlinear function and stored into a single dimensional array as [X1, T1, T2… T4]. For example, if this expansion is exponential then, the X1 is further extended as $T1 = e^{X1}$, $T2 = e^{2X1}$, etc. Similarly, other eight pixels will be expanded and appended next to the same previous array. By considering pre-initialized weights and activation function 'φ', the FLANN output '$y$' is computed. The obtained output 'y' is compared with desired (noise-free) pixel 'd' to evaluate error 'e' which is considered as the cost function. By utilizing this error function, the weights of FLANN filter is updated by applying any of the adaptive algorithms. These algorithms may be either derivative based searches LMS, RLS, N-LMS, L-LMS, etc. or may be of derivative-free algorithms such as GA, PSO, CSO, etc. In this research article, most recently developed derivative-free Jaya algorithm has been applied.

### 3.3 Jaya

The parameterless heuristic optimization technique Jaya has recently been introduced by Rao et al. in [12, 20–22]. They have deciphered several constrained and unconstrained benchmark problems applying this technique. They compared Jaya with other state-of-the-art techniques such as PSO, Teaching Learning Based Optimization (TLBO), Differential Equation (DE), Artificial Bee Colony (ABC), etc. and show its superiority. The word Jaya means 'Victory,' and it originates from the ancient Sanskrit language. Jaya is an algorithm-specific and parameterless algorithm, which makes it different from any other evolutionary based or swarm intelligence based techniques. The notion of this algorithm is that it keeps information of best and worst solution with successive iteration and updates itself according to the best one. Algorithm 1 shows the basic steps of the Jaya algorithm. 'X_(j,k, i)' is the value of the $j^{th}$ variable for the $k^{th}$ contender at the $i^{th}$ iteration and it can be updated as per the Eq. (1), where r1 and r2 are the uniformly distributed random numbers in the range [0–1]. The term "r1(X_(j,best,i)-|X_(j,k,i)|)" indicates the tendency to move closer to the optimum solution and "-r2(X_(j,worst,i)-|X_(j,k,i)|)" indicates the solution to avoid the worse one.

$$X'_{j,k,i} = X_{j,k,i} + r1\left(X_{j,best,i} - \left|X_{j,k,i}\right|\right) - r2\left(X_{j,worst,i} - \left|X_{j,k,i}\right|\right) \tag{1}$$

Algorithm. 1 Procedure of Jaya optimization techniques

| Steps | Process |
|---|---|
| Step.1. | Identify the fitness function to be optimized. |
| Step.2. | Initialize the population size, the number of iteration and variables. |
| Step.3. | Evaluate the fitness function according to the variables values *i.e.* '$X_{j,\,k,\,i}$' and store the best and the worse solutions |

| Steps | Process |
|---|---|
| Step.4. | Modify the variable values using Eq. (1). |
| Step.5. | Evaluate the fitness function according to modified '$X'_{j,k,i}$'. |
| Step.6. | Compare the fitness function with a recent and previous solution and accept the best one only. |
| Step.7. | Repeat the process from Step.3. for '$i^{th}$' iteration time. |
| Step.8. | Save the best optimum solution. |

### 3.4 Proposed Jaya-FLMLP filter

The basic architecture FLMLP filter structure is designed by combining MLP and FLANN network together and has advantages of both the networks. The FLANN network has no hidden layer and MLP may have one or more hidden layer. In the FLMLP, the input layer is expanded by any nonlinear function similar to FLANN filter and having one hidden layer like MLP. The nonlinearity improves due to the presence of expanded inputs and hidden layer as compared to both MLP as well as FLANN. In fact, due to the limited layer, programming complexity decreases as compared to MLP. Here, the Jaya optimization technique is proposed to update the weights of this FLMLP network. Figure 3 shows the corrupted image window of size 3X3 and exponentially expanded input pixel T1. The Fig. 4 shows the basic structure of Jaya-FLMLP filter. It is clear that the inputs to this network are received from the noisy X-ray image and desired pixel is obtained from the window of reference (target) image respectively. For the training of the proposed Jaya-FLMLP filter, the noisy image '$x$' is obtained from the original image '$y_r$' which is synthetically contaminated by Poisson noise '$\eta$'. The elementary model which takes noise into account is mentioned in Eq. (2).

$$x = y_r + \eta \times y_r \tag{2}$$

where, $\{X1, X2, \ldots\ldots, X9\}$ are the inputs of the Jaya-FLMLP network and are obtained from$x$. These pixels are selected by overlapping shifted window blocks of $3 \times 3$ size starting from the first to the third row and the first column to the third column. Subsequently, these pixels intensities are functionally expanded exponentially using

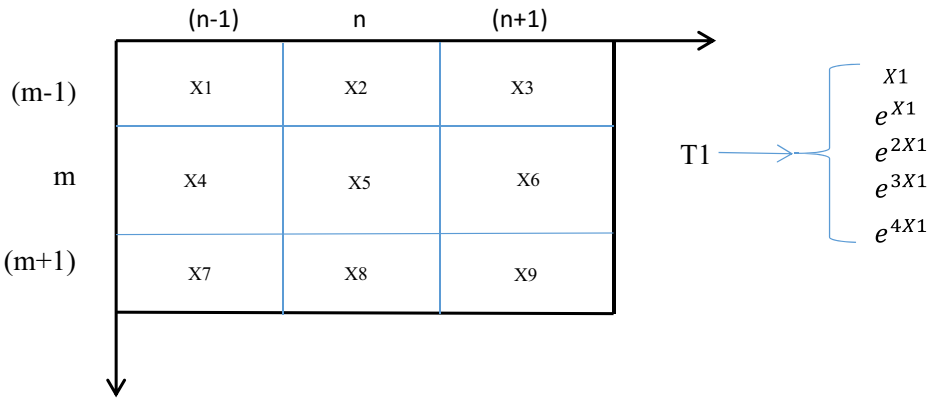



**Fig. 3** Corrupted image window of size 3X3 and exponentially expanded input pixel T1

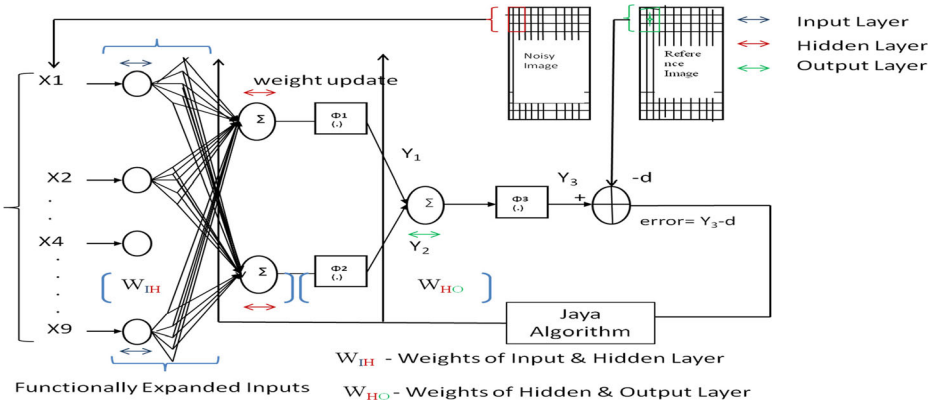



**Fig. 4** Structure of proposed Jaya-FLMLP

Eq. (3).

$$T1 = \begin{Bmatrix} x_1 \\ e^{1x_1} \\ e^{2x_1} \\ e^{3x_1} \\ e^{4x_1} \end{Bmatrix}; \text{similarly, } T2 = \begin{Bmatrix} x_2 \\ e^{1x_2} \\ e^{2x_2} \\ e^{3x_2} \\ e^{4x_2} \end{Bmatrix}; T3 = \begin{Bmatrix} x_3 \\ e^{1x_3} \\ e^{2x_3} \\ e^{3x_3} \\ e^{4x_3} \end{Bmatrix}; \text{and so on.} \tag{3}$$

The equivalent exponentially expanded nine noisy pixels intensities from one window in an array are:

$$T = [T1, T2, T3, \cdots\cdots T8, T9] \tag{4}$$

At the same time, two numbers of 10 sets of random weights $W_1(j)$and $W_2(j)$ are initialized between values 0–1, and multiplied with nonlinearly expanded intensities. The corresponding set of weights at the n$^{th}$ iteration is

$$W_1(n) = \{W1_{IH}, W1_{HO}\} \tag{5}$$

$$W_2(n) = \{W2_{IH}, W2_{HO}\} \tag{6}$$

where, $W1_{IH}$and $W2_{IH}$, denote the random weights connectivity of input and hidden layer and are arranged in the row matrix of size 1X 45$w1$ i.e. $W1_{IH}$= { $w1$,$w2$,... ...,$w45$}and $W2_{IH}$= {,$w2$,... ...,$w45$}. Similarly, $W1_{HO}$ and $W2_{HO}$ are the weight connectivity between the hidden and output layer having size of 1X1 i.e. $W1_{HO}$= {$w46$}. Then, weighted inputs are passed through activation functions $\phi$1(.) and $\phi$2(.) in the hidden layers to evaluate the outputs '$y1$' and '$y2$' respectively. Similarly, the activation function $\phi$3(.) is employed in the output layer to get the final output '$y3$'. Again, y3 is compared with the desired pixel $d$ for evaluating error '$e$'. Where, $d$ is the center most pixel from the moving contextual window of the reference image.

$$y1(n) = \phi 1^* \left[ T(n) * W1_{IH}(n)^T \right] \tag{7}$$

$$y2(n) = \phi 1^{*}\left[T(n)*W2_{IH}(n)^{T}\right] \tag{8}$$

$$y3(n) = \phi 2\left[y1(n)*W1_{HO}(n)^{T} + y2(n)*W2_{HO}(n)^{T}\right] \tag{9}$$

$$\text{error} : e(n) = d(n) - y3(n) \tag{10}$$

For each set of weights the objective function *i.e. 'error(.)'* will be quantified and according to the minimum and maximum errors value the best and worst set weights are selected which is further utilized by the Jaya algorithm to generate a new 10 sets weight position. Again, the cost function is evaluated with new weights. In each iteration, the previous and current errors are to be compared and on the basis of the minimum error, the best set of weight will be stored during the training of the Jaya-FLMLP network. By

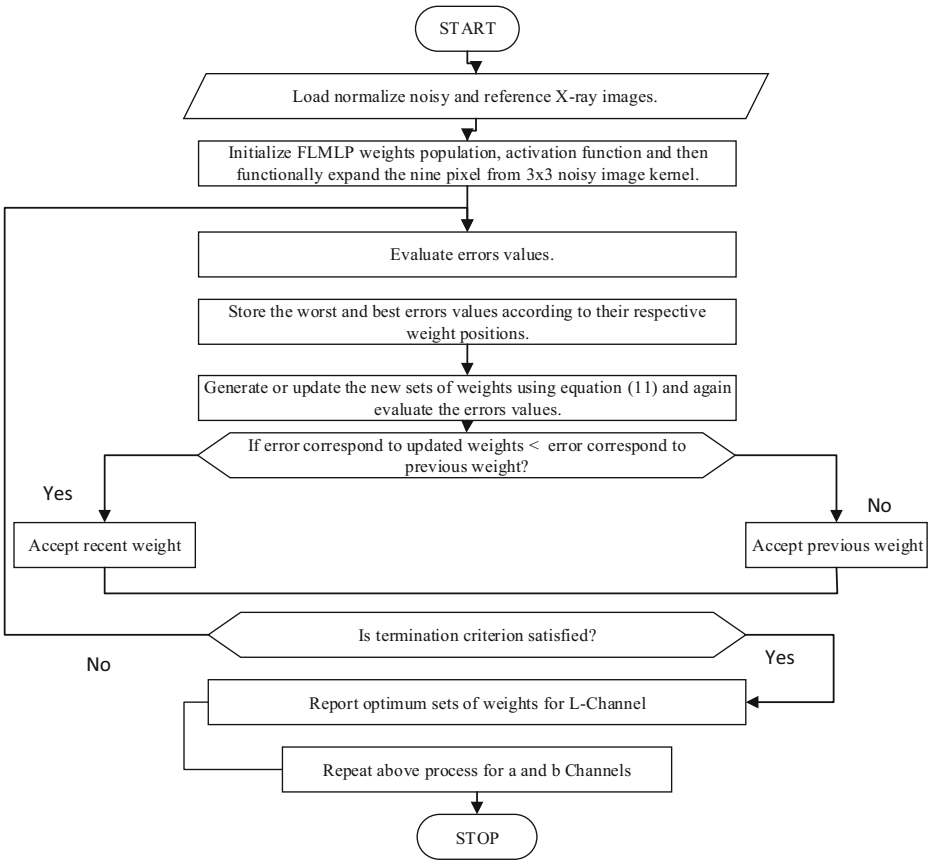



**Fig. 5** Flowchart of Jaya-FLMLP filter

utilizing the optimal set of weights, any image corrupted by Poisson noise can be tested for denoising with the proposed Jaya-FLMLP filter.

$$w(n+1) = w(n) + r1(w_{best}(n) - |w(n)|) - r2(w_{worst}(n) - |w(n)|) \tag{11}$$

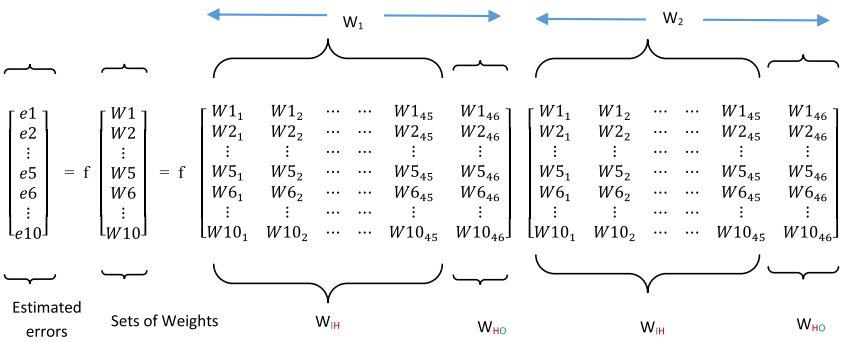
### 3.5 Jaya-FLMLP filter encoding strategy

In this section, a vector representation of the Jaya-FLMLP algorithm has been presented. It provides a brief idea about the matrix size of the weight vector. It presents the procedure of finding the optimal sets of the weight of Jaya-FLMLP which is based on minimum estimated errors. Here, the weights of hidden and output layers simultaneously updated in successive iterations, unlike backpropagation algorithm where weight updating is from output to input layer. Equation (12) demonstrate the functional dependency of the estimated errors i.e. '$\hat{e}(m,n)$' upon the initialized weight of the Jaya-FLMLP filter. Jaya optimization technique helps to search best possible set of weights, where, $W_1$ and $W_2$ represent the weight connectivity between node 1 and node 2 of the hidden and output layers. Equation (13) is a simplified matrix representation of the proposed filter that displays the associated random weights connectivity of the input and hidden layer i.e. *'$W_{IH}$ = {w1, w2... w45}'* as well as the hidden and the output layer i.e. *'$W_{HO}$ = {w46}'*. Similarly, '*$Y_1$', and '$Y_2$'* are the outputs of input-hidden layer, and likewise *'$Y_3$'* is the output of the hidden-output layers. The flowchart of the proposed Jaya-FLMLP filter is depicted in Fig. 5.

$$\text{Estimated error, } \hat{e}(m,n) = f(X) \tag{12}$$

where, $X = \{W_{IH}, W_{HO}\}$

$$\begin{bmatrix} e1 \\ e2 \\ \vdots \\ e5 \\ e6 \\ \vdots \\ e10 \end{bmatrix} = f \begin{bmatrix} W1 \\ W2 \\ \vdots \\ W5 \\ W6 \\ \vdots \\ W10 \end{bmatrix} = f \begin{bmatrix} W1_1 & W1_2 & \cdots & \cdots & W1_{45} & W1_{46} \\ W2_1 & W2_2 & \cdots & \cdots & W2_{45} & W2_{46} \\ \vdots & \vdots & \cdots & \cdots & \vdots & \vdots \\ W5_1 & W5_2 & \cdots & \cdots & W5_{45} & W5_{46} \\ W6_1 & W6_2 & \cdots & \cdots & W6_{45} & W6_{46} \\ \vdots & \vdots & \cdots & \cdots & \vdots & \vdots \\ W10_1 & W10_2 & \cdots & \cdots & W10_{45} & W10_{46} \end{bmatrix} \begin{bmatrix} W1_1 & W1_2 & \cdots & \cdots & W1_{45} & W1_{46} \\ W2_1 & W2_2 & \cdots & \cdots & W2_{45} & W2_{46} \\ \vdots & \vdots & \cdots & \cdots & \vdots & \vdots \\ W5_1 & W5_2 & \cdots & \cdots & W5_{45} & W5_{46} \\ W6_1 & W6_2 & \cdots & \cdots & W6_{45} & W6_{46} \\ \vdots & \vdots & \cdots & \cdots & \vdots & \vdots \\ W10_1 & W10_2 & \cdots & \cdots & W10_{45} & W10_{46} \end{bmatrix} \tag{13}$$




## 4 Results

In this article, five other filtering techniques are also considered for comparative study so that the efficacy of the proposed filter can be examined. Hence, all the competitive filters are applied to remove Poisson noise from the different benchmark noisy images. For experimentation purpose, four grayscale images are taken for testing. Three of them are radiological X-ray images named as 'Chest', 'Severe pneumonia' and 'Pelvis' and the last one is a panoramic 'Hand1'. The Poisson noise has been introduced synthetically and generated from input image data itself. In fact, the density or scaling factor of noise depends upon the precision (single/double) and mean of Poisson distribution [19]. The image size of the 'Hands1' is 240X320 and

the X-rays images are 616X447. All the four images are in the JPEG format. The simulation tasks are performed in MATLAB® programming environment using a personal computer with the specification of Intel Core i3 at 1.40 GHz, 4 Gb of RAM, 64-bit bus and Windows 10 operating system. The obtained results are examined visually along with various quality metrics and nonparametric statistical tests.

The final fine-tuned parameters obtained analytically for different denoising algorithm are presented in Table 1. The size of BP-MLP network is of 9–3–2-1 structure i.e. having two hidden layers. The number of nodes in input, two hidden and output layer is 9,3,2 and 1 respectively. The number of weights between input and first hidden layer, first and second hidden layer, hidden and output layer are 27, 6 and 2 respectively. The learning rate of back propagation algorithm is taken as 0.02. The sigmoid functions used in two hidden layers and output layers are tanh(.) and purelin(.) respectively. In LMS-FLANN, PSO-FLANN, CSO-FLANN and JAYA-FLMLP filter, the input values are the 9 noisy pixels which are obtained from the corrupted X-ray image. For the generality of comparison, exponential expansion is applied and each pixel expanded for five times in all these approaches. Thus, LMS-FLANN attains a total of 45 weights and for 10 sets of weights PSO-FLANN, CSO-FLANN have 450 {10×45} weights. Similarly, for the Jaya-FLMLP filter, the number of weights of input to hidden and hidden to output layer are also depicted in this table. The learning rate is considered

**Table 1** Parameters employed in different ANNs filters

| Sl. No. | ANN based denoising methodologies | Parameters |
|---|---|---|
| 1 | LMS-FLANN | Number of Weights = {9X5} = 45<br>Learning Rate = 0.02<br>Functional Expansion = exponential(.)<br>Activation Function = logsig(.) |
| 2 | BP-MLP | Number of Weights:<br>Number of weights in Hidden layer1 = {9X3} = 27<br>Number of weights in Hidden layer2 = {3X2} = 6<br>Number of weights in Output layer = {2X1} = 2<br>Activation Functions:<br>Hidden Layers 1,2 = tanh(.)<br>Output Layer = purelin(.)<br>Learning Rate = 0.02 |
| 3 | PSO-FLANN [11] and CSO-FLANN [13] | Number of Weights = {10X9X5} = 450<br>Functional Expansion = exponential(.)<br>Activation Function = logsig(.)<br>Inertia = mean((2/ Number of Iteration)^0.3)<br>PSO Acceleration Constants C1, C2 = 2<br>CSO Acceleration Constant = 2 |
| 4 | JAYA-FLMLP (Proposed) | Number of Weights:<br>Weights for Input node to Hidden node1 = {10X45} = 450<br>Weights for Input node to Hidden node 2 = {10X45} = 450<br>Weights for Hidden node1 to Output node = {10X1} = 10<br>Weights for Hidden node 2 to Output node = {10X1} = 10<br>Functional Expansion = exponential(.)<br>Activation Function:<br>Hidden Layer activation function = tanh(.)<br>Output Layer activation function = purelin(.) |

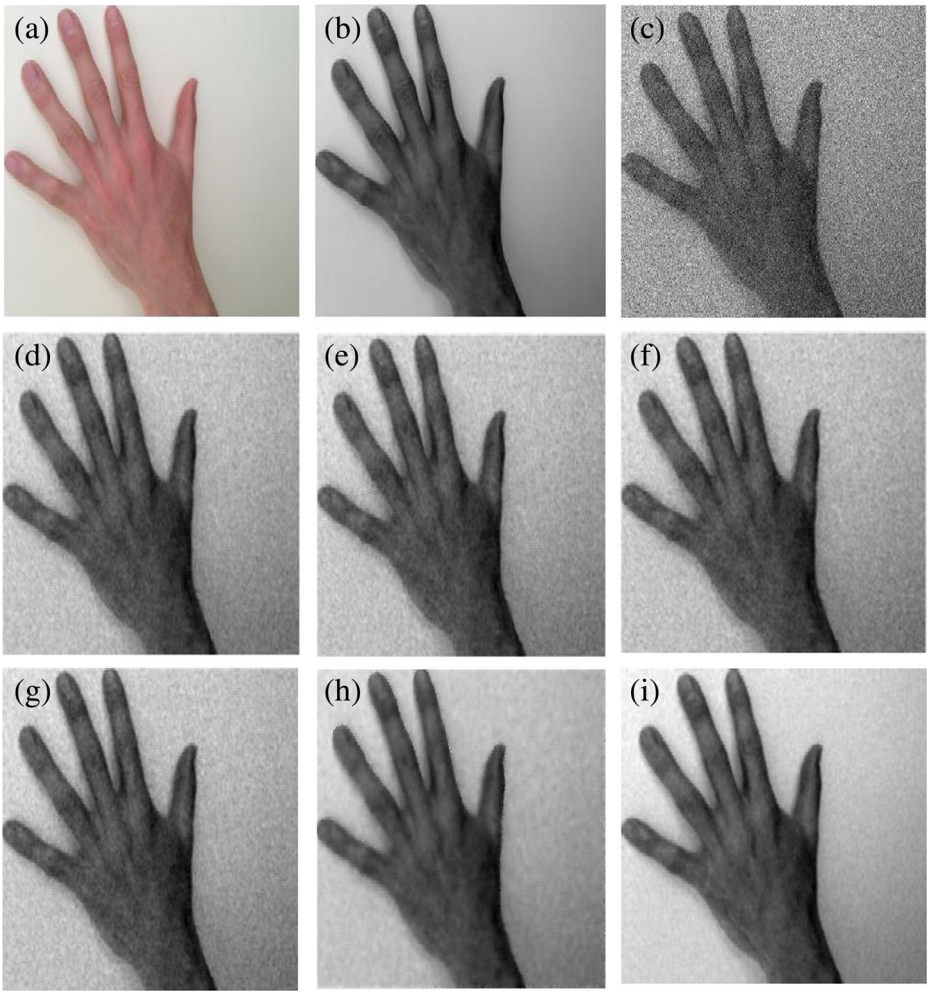



**Fig. 6** **a** Images of Hands1, (a) Original image (b) Grayscale image (c) Corrupted image by Poisson noise (d) Filtered image using LMS-FLANN (e) Filtered image using BP-MLP (f) Filtered image using PSO-FLANN (h) Filtered image using CSO-FLANN (i) Filtered image using Jaya-FLMLP. (Image Courtesy: Matlab® 2015a Image data). **b** Images of Chest X-rays, (a) Original image (b) Grayscale image (c) Corrupted image by Poisson noise (d) Filtered image using LMS-FLANN (e) Filtered image using BP-MLP (f) Filtered image using PSO-FLANN (h) Filtered image using CSO-FLANN (i) Filtered image using Jaya-FLMLP (j) Corrupted image by Gaussian noise ($\mu = 0.04$, $\sigma^2 = 0$) (k) Removal of Gaussian noise image using Jaya-FLMLP (l) Corrupted image by impulse noise (a==0.04) (m) Removal of impulse noise image using Jaya-FLMLP. (Image Courtesy: radiology images database - http://cdn.lifeinthefastlane.com)

to be 0.02 for LMS-FLANN network. In PSO-FLANN filter acceleration constant i.e. C1 and C2 which are also known as cognitive and social parameters respectively which is considered to be 2 in our algorithm. Similarly, CSO-FLANN depends upon social parameter i.e. C1 and it is also fixed at 2. The inertia is another common parameter for both PSO-FLANN and CSO-FLANN filters. We have considered adaptive inertia parameter for both networks whose mathematical expression is shown in Table 1. Logsig(.) activation function is used in both of these filters. Unlike PSO-FLANN and CSO-FLANN filters, Jaya-FLMLP does not require

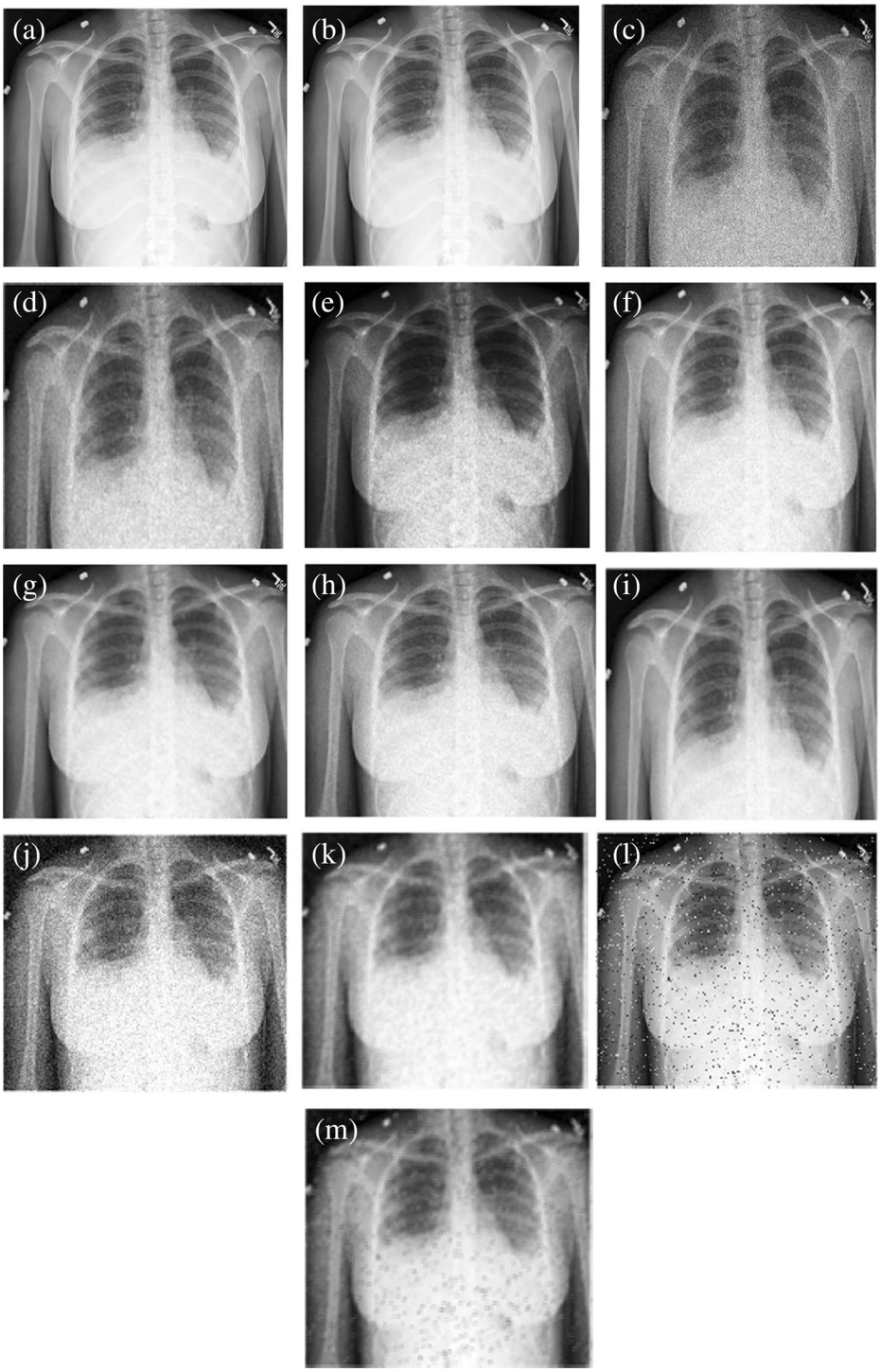



**Fig. 6** (continued)

**Table 2** Image quality metrics values obtain from Experiment I

| Image | | Filters | Image Quality Metrics | | | |
|---|---|---|---|---|---|---|
| | | | SSIM | PSNR | NRDB | NMSE |
| Hands1 | SSIM | WIENER | 0.9089 | 31.88 | 20.01 | −25.99 |
| | 0.8046 | BP-MLP | 0.8534 | 31.59 | 20.11 | −26.79 |
| | PSNR | LMS-FLANN | 0.9019 | 31.75 | 20.13 | −27.96 |
| | 28.87 | PSO-FLANN | 0.9153 | 31.92 | 21.87 | −28.04 |
| | Noisy | CSO-FLANN | 0.9462 | 32.03 | 22.10 | −28.08 |
| | | Jaya-FLMLP | 0.9473 | 33.97 | 23.49 | −28.37 |

parameters like inertia and acceleration constant. The activation functions such as tanh(.) and purelin(.) are considered for hidden and output layers of Jaya-FLMLP network. The total number of iteration is set to be 3000 for all the competitive ANN's networks. The first two rows and columns pixels are padded to the noisy input image to preserve the size of the filtered image with that of the input image during processing. This step will help in obtaining the image quality metrics (PSNR, SSIM, NRDB, NMSE) correctly without cropping the input image. The original distortion-free Chest X-rays image is taken as a reference image. The other two X-rays images i.e. 'Pelvis' and 'Pneumonia' as given in the Appendix Fig. 11 of this article are tested by applying all the proposed networks.

### 4.1 Subjective evaluation

The performance of the applied adaptive filters can be judged by observing and comparing the original and the filtered image. The grayscale images have been taken to carry out all the experiment. The following Figs. 6a and b show the filtered images of the 'Hand1' and 'Chest X-ray' image.

### 4.2 Adopted evaluation metrics

Performance metrics such as PSNR, NRDB, and NMSE are chosen to measure the quantitative aspect of the adopted filters. The mathematical forms of the respective quality metrics are mentioned in Eqs. (14)–(18). Studies such as the SSIM are also included in this paper for analyzing the qualitative and quantitative perspectives of the denoised image. The SSIM evaluates the structural similarity by comparing local patterns of the pixel intensities that have been normalized for contrast and luminance. It is based on the principle of a human visual

**Table 3** Image quality metrics obtain from Experiment II

| Image | | Filters | Image Quality Metrics | | | |
|---|---|---|---|---|---|---|
| | | | SSIM | PSNR | NRDB | NMSE |
| Chest X-ray | SSIM | WIENER | 0.5627 | 33.86 | 20.14 | −25.99 |
| | 0.4290 | BP-MLP | 0.6086 | 33.97 | 20.68 | −26.54 |
| | PSNR | LMS-FLANN | 0.7029 | 34.21 | 20.90 | −26.67 |
| | 30.71 | PSO-FLANN | 0.7821 | 34.34 | 21.54 | −27.47 |
| | Noisy | CSO-FLANN | 0.7932 | 35.14 | 22.10 | −27.77 |
| | | Jaya-FLMLP | 0.7981 | 35.26 | 25.36 | −27.95 |

**Table 4** Image quality metrics obtain from Experiment III

| Image | | Filters | Image Quality Metrics | | | |
|---|---|---|---|---|---|---|
| | | | SSIM | PSNR | NRDB | NMSE |
| Severe pneumonia | SSIM | WIENER | 0.6277 | 32.40 | 21.16 | −25.99 |
| | 0.6076 | BP-MLP | 0.6365 | 32.70 | 21.19 | −26.21 |
| | PSNR | LMS-FLANN | 0.6388 | 33.31 | 21.54 | −26.21 |
| | 29.26 | PSO-FLANN | 0.6599 | 33.61 | 21.89 | −26.89 |
| | Noisy | CSO-FLANN | 0.6798 | 34.22 | 22.10 | −27.51 |
| | | Jaya-FLMLP | 0.6801 | 34.10 | 24.16 | −27.80 |

system for extracting information from any structure. The SSIM can be expressed mathematically by combining three attributes i.e. luminance '*l*', contrast '*c*' and structure '*s*' as shown in Eq. (14), where, SSIM depends upon the local mean, standard deviation and cross-covariance of an image. The default value exponents '$\alpha$', '$\beta$' and '$\Upsilon$ are considered as 1 [27].

$$SSIM(i,j) = [l(i,j)]^{\alpha} \times [c(i,j)]^{\beta} \times [s(i,j)]^{\gamma} \tag{14}$$

PSNR is the ratio between the maximum possible power of an image signal and the power of the corrupting noise. NMSE is an estimator of the overall deviations between the original and restored images. Similarly, NRDB is applied to measure noise removal capability of any filtering technique. These performance indices are mathematically expressed as:

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left[ x(i,j) - \hat{x}\left(i,j\right) \right]^2 \tag{15}$$

$$PSNR = 10 \times \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{16}$$

$$NRDB = 10 \times \log_{10}\left(\frac{MSE_{IN}}{MSE_{OUT}}\right) \tag{17}$$

$$NMSE = \frac{\sum_{i=1}^{r} \sum_{j=1}^{c} \left(x(i,j) - \hat{x}(i,j)\right)^2}{\sum_{i=1}^{r} \sum_{j=1}^{c} \left(x(i,j)\right)^2} \tag{18}$$

**Table 5** Image quality metrics obtain from Experiment IV

| Image | | Filters | Image Quality Metrics | | | |
|---|---|---|---|---|---|---|
| | | | SSIM | PSNR | NRDB | NMSE |
| Pelvis | SSIM | WIENER | 0.7767 | 34.55 | 20.21 | −25.89 |
| | 0.7247 | BP-MLP | 0.7791 | 34.68 | 20.26 | −26.12 |
| | PSNR | LMS-FLANN | 0.7804 | 34.96 | 20.54 | −26.67 |
| | 27.98 | PSO-FLANN | 0.7891 | 35.84 | 21.43 | −27.54 |
| | Noisy | CSO-FLANN | 0.8231 | 36.39 | 20.97 | −27.83 |
| | | Jaya-FLMLP | 0.8245 | 36.62 | 22.91 | −26.96 |

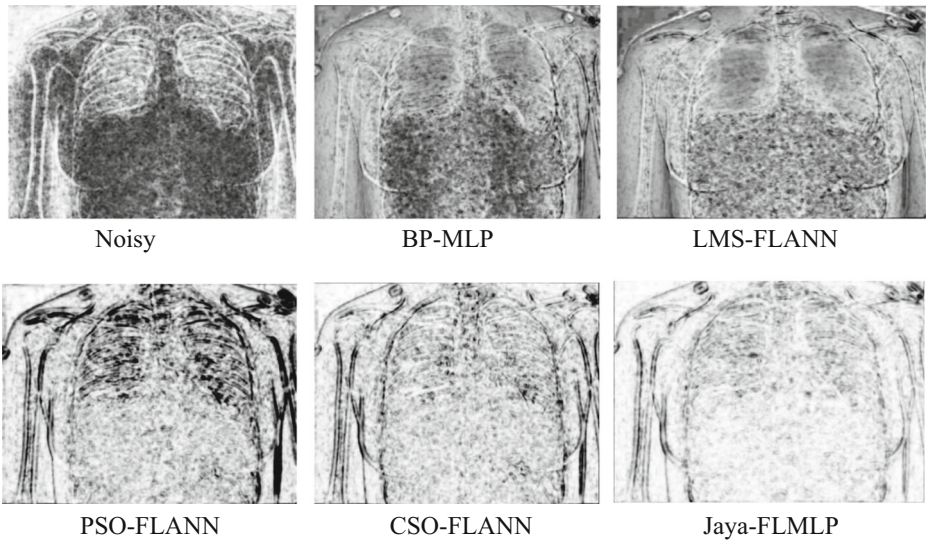



**Fig. 7** Images obtained from difference SSIM Index Map and Mean SSIM values

where, '$MSE_{IN}$' is the mean square error between the original and noisy images and '$MSE_{OUT}$' is the mean square error between the original and filtered images. Similarly, '$x(i,j)$' and '$\hat{x}(i,j)$' are the original and restored image respectively. The other indices such as convergence characteristics rate and computational time are also examined for the performance evaluation of the different filters. Quantitative measures obtained during the simulation studies are depicted in Tables 2, 3, 4, and 5 and are utilized to compare the proposed Jaya-FLMLP filter with the linear adaptive (Wiener) and nonlinear adaptive neural (BP-MLP, LMS-FLANN, PSO-FLANN and CSO-FLANN) filters. In the experiment, the PSNR, SSIM values of the noisy Hands1 image are 28.87 dB and 0.8046 respectively and are represented vertically in Table 2. The values of the performance indices of the filtered images are presented horizontally in this table. Accordingly, Tables 3, 4, and 5 specify the experimental output obtained from the radiographic X-ray images by applying these filters.

The Figs. 8 and 9 are showing the convergence characteristics and noise reduction capability of all competitive image filters. The convergence graph is acquired from the filtration of Chest X-ray image. The average time taken during training and testing of all experiment is depicted in Fig. 10. The time consumed by LMS-FLANN is as 1684.09 s. However, training time of proposed Jaya-FLMLP was as 3466.80 s.

## 5 Discussion

Table 1 demonstrates the parameter values obtained for the different neural filter algorithms, and it clearly shows the significance of how less number of controlling parameters are required

**Table 6** Average ranking of filtering algorithms based on the Friedman's test using PSNR

| Algorithms | WIENER | BP-MLP | LMS-FLANN | PSO-FLANN | CSO-FLANN | Jaya-FLMLP |
|---|---|---|---|---|---|---|
| Ranking | 5.80 | 5.06 | 4 | 3.06 | 1.96 | 1.1 |

**Table 7** Friedman Statistical test parameters using PSNR

| Source | Sum of square (SS) | Degree of freedom (df) | Mean square (MS) | Chi-square | Critical value (p) |
|---|---|---|---|---|---|
| Columns | 488.8 | 5 | 97.6 | 139.66 | $2.11e^{-28}$ |
| Error | 36.2 | 145 | 0.2497 | | |
| Total | 525 | 179 | | | |

for the proposed Jaya-FLMLP filter than the others adaptive filters. In Figs. 6a and b, the denoised images are obtained from applying the various neural filter, and these filters can remove Poisson noise from the ordinary and the X-ray images. It is evident from Fig 6b (j) - (m), that the trained Jaya-FLMLP filter can remove common noise like Gaussian and impulse noises. For the experimental task, the mean 'μ' and variance '$\sigma^2 = 0$' of Gaussian noise were 0.04 and 0 respectively. Similarly, density 'a' of impulse noise was considered as 0.04. In the same manner Fig.7 shows the qualitative view of the filtration chest X-ray image in terms of SSIM. In fact, the above images represent the gradual improvement in clearing noise using the state-of-the-art filters like PSO-FLANN, CSO-FLANN, and Jaya-FLMLP. After observing the images of Figs. 6 and 7 it is evident that the proposed Jaya-FLMLP filter provides comparatively better result than the other evolutionary neural network filters. If we consider the quantitative aspect of Experiment I, the PSNR, SSIM, NRDB, NMSE values of the filtered image using the Jaya-FLMLP network are 33.97, 0.7981, 23.49 and −28.37 respectively which are the highest among the other networks. However, In Table 2, the SSIM and PSNR values of

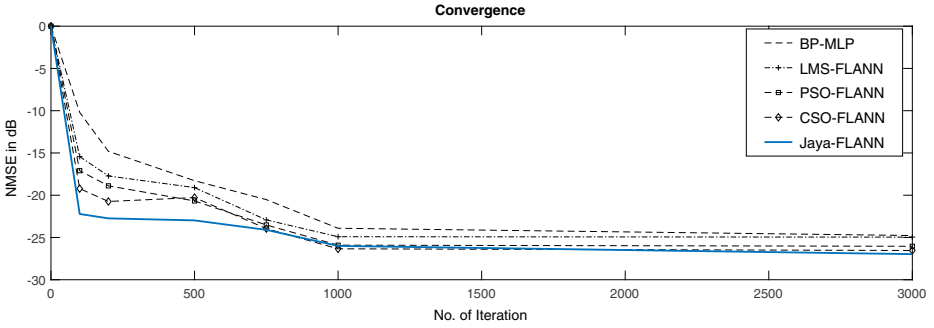



**Fig. 8** Convergence characteristics of ANNs filters

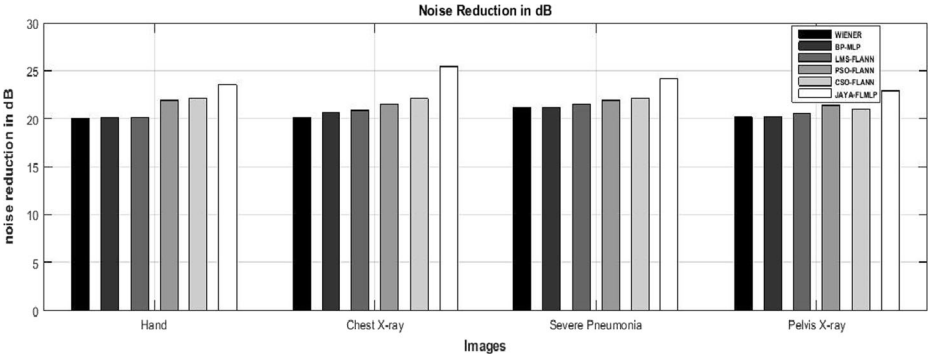



**Fig. 9** Noise reduction bar-graph capabilities of various image filters

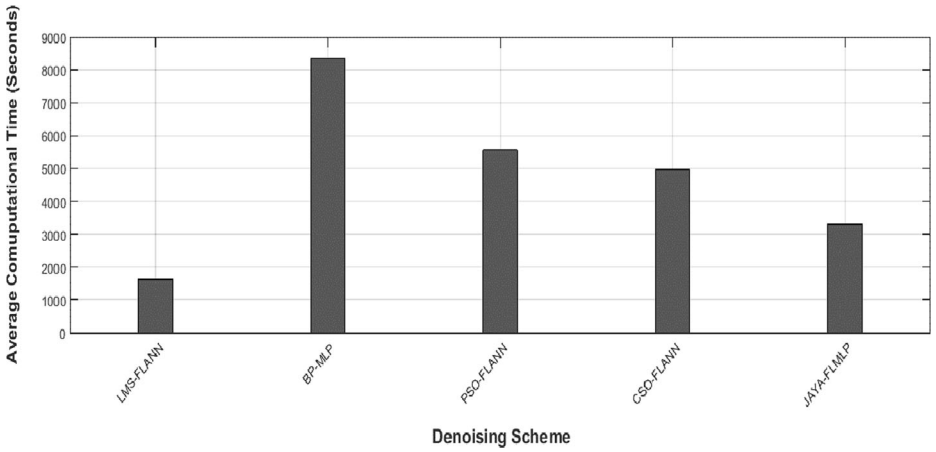



**Fig. 10** Average computational time of various image filters

the Wiener filter are more than those of the BP-MLP and LMS-FLANN filter. Hence, the linear adaptive filter may also provide better results than the nonlinear neural filter. Hence, to avoid vagueness, the non-parametric statistical test, i.e. Friedman's test is conducted for comparing all the implemented methodologies. Tables 6 and 7 present the results obtained from the Friedman's test, and the lower ranking and critical values suggest the superiority of the proposed Jaya-FLMLP filter. The PSNR data of the 30 images are considered for analyzing the nonparametric test.

Figure 8 shows the convergence characteristics graph of ANN filters in terms of the NMSE and the faster convergence rate of the Jaya-FLMLP network. Figure 9 shows the noise clearing ability of all the employed filters, and it indicates that the efficiency of the proposed filter is the highest among all. Similarly, Fig. 10 demonstrates the average computational time for the training all the networks. The Jaya-FLMLP takes less time for learning than any other evolutionary network-based filters such as PSO-FLANN and CSO-FLANN. However, the LMS-FLANN and BP-MLP adaptive filters have taken the minimum and maximum time for training the network. The reason is, it mostly depends on the network architecture and the size of the image because that much of execution loop will increase. In the case of evolutionary ANNs filtering techniques, it depends upon the population size and controlling parameters of the filtering algorithms.

## 6 Conclusion

The above result of the simulation studies shows that the proposed Jaya-FLMLP filter can be successfully applied to filter 'Poisson' noise from the X-ray images and it exhibits satisfactory performance under different noise conditions. The qualitative and quantitative inspections establish the excellence of the Jaya-FLMLP over the other five filtering techniques i.e. Wiener, LMS-FLANN, BP-MLP, PSO-FLANN, and CSO-FLANN. The comparison of performance comprises of computational time, performance indices such as PSNR, SSIM, NRDB, NMSE and convergence rate. The simulation results show that the parameterless Jaya has enormous potential to solve any optimization problem in lesser time with limited hidden nodes and it can be used as a learning algorithm for any ANNs. The Friedman's test is also considered to assess

the dominance of the proposed Jaya-FLMLP filter over others. Our future line of research includes testing the effectiveness and adaptability of the Jaya-FLMLP filter, irrespective of the noise and medical image type. It can be hybridized with other heuristic techniques for achieving a more accurate level of the FLMLP.

## Appendix

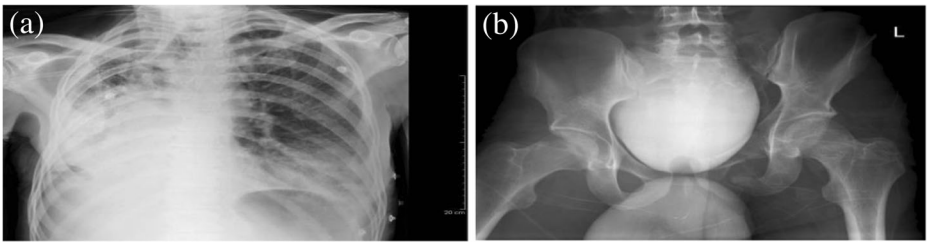



**Fig. 11** X-rays Images (**a**) Severe pneumonia (**b**) Pelvis. (Image Courtesy: radiology images database - http://cdn.lifeinthefastlane.com)

## References

1. Chen S, Suzuki K (2014) Bone suppression in chest radiographs by means of anatomically specific multiple massive-training ANNs combined with total variation minimization smoothing and consistency processing. Comput Intell Biomed Imaging 33:211–235
2. Chen F, Wu Y (2015) Improving image recognition by hierarchical model and denoising. 11th International Conference on Natural Computation, 2015
3. Dash PP, Patra D (2011) Evolutionary neural network for noise cancellation in image data. Int J Comput Vis Robot 2:206–217
4. Dehuri S, Cho SB (2010) Evolutionarily optimized features in functional link neural network for classification. Expert Syst Appl 37:4379–4391
5. Dokur Z, Olmez T (2002) Segmentation of ultrasound images by using a hybrid neural network. Pattern Recogn Lett 23:1825–1836
6. Duan H, Wang X (2016) Echo state networks with orthogonal pigeon-inspired optimization for image restoration. IEEE Trans Neural Networks Learn Syst 27:1–13
7. García DM, Gutiérrez JG, Santos JCR (2012) On the evolutionary optimization of k-NN by label-dependent feature weighting. Pattern Recogn Lett 33:2232–2238
8. Giakoumis I, Nikolaidis N, Pitas I (2006) Digital image processing techniques for the detection and removal of cracks in digitized paintings. IEEE Trans Image Process 15:178–188
9. Ginneken BV, Stegmann MB, Loog M (2006) Segmentation of anatomical structures in chest radiographs using supervised methods: A comparative study on a public database. Med Image Anal 10:19–40
10. He L, Land R, Greenshields I (2009) A nonlocal maximum likelihood estimation method for rician noise reduction in mr images. IEEE Trans Med Imaging 28:165–172
11. Kumar M, Mishra SK (2015) Kumar M, Mishra SK (2015) Particle swarm optimization-based functional link artificial neural network for medical image denoising. In: Computational Vision and Robotics. 1st edn, Springer, New Delhi, pp 105–111. https://doi.org/10.1007/978-81-322-2196-8_13
12. Kumar M, Mishra SK (2017) Jaya-FLANN based adaptive filter for mixed noise suppression from ultrasound images. Biomed Res 28:4159–4164
13. Kumar M, Mishra SK, Sahu SS (2016) Cat swarm optimization based functional link artificial neural network filter for gaussian noise removal from computed tomography images. Appl Comput Intell Soft Comput 2016:1–6
14. Li Y, Lu J, Wang L, Fan Y, Li S, Yahagi T (2007) Removing noise from medical CR image using multineural network filter based on noise intensity distribution. Proc. - Third Int. Conf. Nat. Comput, 2007

15. Majhi B, Panda G (2011) Robust identification of nonlinear complex systems using low complexity ANN and particle swarm optimization technique. Expert Syst Appl 38(1):321–333
16. Mirjalili S, Hashim SZM, Sardroudi HM (2012) Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Appl Math Comput 218:11125–11137
17. Pao Y (1989) Adaptive pattern recognition and neural networks. Addison Wesley, Boston
18. Perry SW, Guan L (2000) Weight assignment for adaptive image restoration by neural networks. IEEE Trans Neural Netw 11:156–170
19. Poisson noise (2016) (Online). Available: https://in.mathworks.com/help/images/ref/imnoise.html. Accessed 25 Jun 2016
20. Rao RV (2016) Jaya : A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int J Ind Eng Comput 719–34:2016
21. Rao RV, Waghmare GG (2016) A new optimization algorithm for solving complex constrained design optimization problems. Eng Optim 49:1–24
22. Rao RV, More KC, Taler J, Ocłoń J (2016) Dimensional optimization of a micro-channel heat sink using Jaya algorithm. Appl Therm Eng 103:572–582
23. Saadi S, Guessoum A, Bettayeb M (2013) ABC optimized neural network model for image deblurring with its FPGA implementation. Microprocess Microsyst 37:52–64
24. Sicuranza GL, Carini A (2011) A generalized FLANN filter for nonlinear active noise control. IEEE Trans Audio Speech Lang Process 19:2412–2417
25. Sivakumar K (1993) Image Restoration Using a Multilayer Perceptron with a Multilevel Sigmoidal Function. IEEE Trans of Signal Process 41:2018–2022
26. Suzuki K, Horiba I, Sugie N (2003) Neural edge enhancer for supervised edge enhancement from noisy images. IEEE Trans Pattern Anal Mach Intell 25:1582–1596
27. Wang A, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: From error visibility to structural similarity. IEEE Trans Image Process 13:600–612
28. Wang H, Lv Y, Chen H, Li Y, Zhang Y, Lu Z (2016) Smart pathological brain detection system by predator-prey particle swarm optimization and single-hidden layer neural-network. Multimed Tools Appl. https://doi.org/10.1007/s11042-016-4242-0
29. Wen H, Wen J (2013) Image Denoising and Restoration Using Pulse Coupled Neural Networks. 6th Int. Congr. Image Signal Process. 2013
30. Zeng W, Lu X, Tan X (2013) A local structural adaptive partial differential equation for image denoising. Multimed Tools Appl 74:743–757
31. Zhang D, Mabu S, Hirasawa K (2010) Noise reduction using genetic algorithm based pcnn method. IEEE conf. Systems Man and Cybernetics (SMC). 2010
32. Zhao H, Zhang J (2008) Functional link neural network cascaded with Chebyshev orthogonal polynomial for nonlinear channel equalization. Signal Process 88:1946–1957
33. Zhao H, Zhang J (2010) Pipelined Chebyshev functional link artificial recurrent neural network for nonlinear adaptive filter. IEEE Trans Syst Man, Cybern Part B Cybern 40:162–172
34. Zhou YT, Chellappa R, Vaid A, Jenkins BK (1988) Image restoration using a neural network. IEEE Trans Acoust 36:1141–1151

**Manish Kumar** received the B.Tech. Degree in Applied Electronics and Instrumentation Engineering from Biju Patnaik University, Rourkela, India, in 2010 and the M.Tech. Degree in Biomedical Engineering from the Manipal Univesity, Udupi, India, in 2013. He is currently pursuing Ph.D. degree in Electrical and Electronics engineering. His area of interests are medical image processing, neural network and evolutionary techniques.

**Sudhansu Kumar Mishra** received the M.E. and Ph.D. degree in Electrical and Electronics Engineering from the National Institute of Technology, Rourkela, India. He is currently an Assistant Professor with the Department of EEE, Birla Institute of Technology. His area of interest are image processing, system identification and computational intelligence.