International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

Medical Image Compression using Neural Network with HGAPSO Optimization

Ravikiran H.K, Jayanth J

Abstract*: Lossy medical image compression has become increasingly attractive due to a drastic increase in the number of images used for diagnosis and treatment. The work focused on developing a feed-forward neural network for compression of medical images with optimization of weights using hybrid genetic and particle swarm (HGAPSO) optimization technique. The neural network can achieve a better compressed & decompressed image only with the proper training and optimized weights. Training algorithms such as back-propagation algorithm (BPA) traps to local minima rather than the global one which degrades the quality of a reconstructed image. In this work, HGAPSO optimization is adopted to overcome the drawback of BPA. HGAPSO parameters are carefully chosen to have better exploitation & exploration in the search area, which avoids the algorithm from trapping to the local minima. High-quality results of Genetic Algorithm (GA) obtained using selection, crossover and mutation can provide quality guidance for PSO which improves the results of the proposed system. The performance of the proposed work is evaluated on a raw medical image database based on PSNR, MSE, and CR. The experiment is simulated for 16-4-16 neural network architecture and a compression ratio of 75 % is achieved. The results obtained indicated that with proper training PSNR could be improved by 1.98 %.

***Keywords* : Artificial neural network, genetic algorithm and particle swarm optimization, image compression.**

I. INTRODUCTION

Even if numerous studies have been developed in the field of interest, biomedical image compression remains an important issue. Due to innovations in medical engineering, digital acquisition in medical images tends to improve the resolutions of the medical images, which intensifies the mass of data to archive [2]. Hence a large storage space is needed and requires more bandwidth during transmission through a communication link. Medical images used for diagnostics must be kept in the same state for several periods for the judicial perspectives. To save a large number of image databases, an efficient compression system is required. The correlation between the data within the image enables a reduction in the image data contents without much degradation in the quality of the image [3].

Revised Manuscript Received on December 30, 2019.
***** Correspondence Author
Ravikiran H.K.*, Department of Electronics and Communication, GSSS Institute of Engineering and Technology for Women, Affiliated to Visvesvaraya Technological University, Mysuru, India.
Dr. Jayanth J, Department of Electronics and Communication, GSSS Institute of Engineering and Technology for Women, Affiliated to Visvesvaraya Technological University, Mysuru, India.

There are several articles published which use Artificial neural networks (ANN) to compress image with BPA as a training algorithm [1] [2]. The BPA is the simplest training algorithm of ANN used for image compression. ANN adapts its weights and bias based on the difference between the actual and desired output. One of the major drawbacks of BPA is very slow convergence and trapping into local minima. The above said problem needs to be addressed in an efficient way which has made the neural network less efficient for image compression. With the motivation of biological systems, several computational intelligence (CI) techniques have emerged which provide well enough optimized weights [4][17].

Hybridization of backpropagation with soft computing techniques can eliminate such problems and can give a better solution [14]. Some of the optimization techniques available in literature for weights optimization are the GA [6], Particle Swarm Optimization (PSO) [7], Hybrid GA and PSO [11], Cuckoo Search [9], Grey Wolf Optimizer (GWO) [8] and whale optimization [10].

The main contribution of the paper is to integrate GA and PSO optimization strategy to train the ANN with BPA to achieve global minima, and then these global optimum weights and thresholds connections are saved for the operational phase to compress and decompress the image.

The theoretical backgrounds are explained in Section II. Implementation details are explained in Section III. Database details are mentioned in section IV. The experimental results and validation of the proposed work are been explained in section V. The final remarks are discussed in section VI.

II. BACKGROUNDS

A. *Artificial Neural network for Image Compression*

A feed-forward ANN comprises of input, hidden and an output layer, and operates in two phases such as a training phase and an operation phase [4]. The knowledge acquired by ANN is stored as weights. The main aim of the training phase is to calculate the appropriate weights between the layers, which will allow the network to operate correctly during the operation phase. The most common training algorithm to update the weights and bias of an ANN is the backpropagation algorithm [5]. The algorithm starts with assigning a set of weights and bias values. The initial weights and bias are set to random values except zero because it will result in having identical input and output vectors. During the training phase, the input image is applied to the input layer as a vector where N denotes the block size. The output layer presents the desired output. An error is calculated on comparing the obtained output and the actual output during every forward pass.

The error is used to determine the weights and bias according to the learning rule [7]. As the neural network has to be trained with various paradigms the training phase plays a key step. Hence the weights and bias play a significant part in image compression & decompression. The structure of typical feed- forward ANN is as shown in Fig.1.

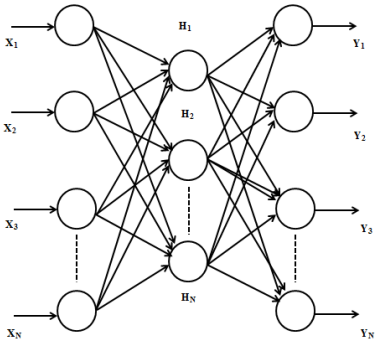

Fig.1. Typical Feed Forward ANN

Algorithm to train the artificial neural network (ANN) can be summarized as follows:

Algorithm 1:

Step 1: Define input parameters in

$$[image, blocks, patterns, error\ tolerence] = read\_param$$

Step 2: Initialization of network weights, bias to a random value. Set iterations to zero.
Step 3: Select the folder which holds the training image data.
Step 4: Normalize the image data.
Step 5: Train the NN =

$$TrainImagePattern(InImage, blocksize, Image\ Number)$$

Step 6: Adjust weights and bias using BPA.
Step 7: Halt if any stopping criteria are satisfied.
Step 8: Save the optimized weights and bias.

The algorithm for the ANN operational phase can is summarized as follows.

Algorithm 2:

Step 1: Define input the image to be compressed.
Step 2: Read trained weights from the files.

$$\begin{bmatrix} I2H, H2O, I2H_{bias}, H2O_{bias} \end{bmatrix} = Read\_Weight(WeightFiles)$$

Step 3: Prepare the data for input- hidden layer as which represents the compressed image.

$$H_{ih} = Tansig((Pattern * I2H) + I2H_{bias})$$

Step 4: Apply image reconstruction between hidden – output layer as

$$\begin{bmatrix} H_{ho} \end{bmatrix} = Purelin((Pattern * H2O) + H2O_{bias})$$

B. *Genetic algorithm*

Genetic algorithms (GA) are a kind of stochastic search procedure that has been developed based on the adaptive processes of natural systems [15]. The mechanism of GA involves natural selection, genetics, and evolution of genes. In GA, a prospective way out to a particular problem is called an individual or chromosomes. Each chromosome represents a point in the search space. A search space will consist of a population of a finite numeral of chromosomes. These individuals are evaluated based on the fitness function. A range of new offspring is generated by applying crossover and mutation operator on the individuals which randomly modifies the individual contents. Any individuals with less performance are replaced systematically by the offspring's [12].The performances of these individuals are continuously evaluated until any stopping criterion is satisfied.

We can summarize the genetic algorithm as follows.

Algorithm 3:

Step 1: Set initial population P^0 according to the problem.
Step 2: Determine the parameters.
Step 3: Compute the fitness function of all individuals in P^0 .
Step 4: Repeat
Step 5: Evaluate the stop condition.
Step 6: Apply selection operator.
Step 7: Apply the crossover operator to all selected pairs of individuals and generate a new population.
Step 8: Update P^t and Go to Step 3.
Step 9: If no improvement by step 6. Perform mutation operation on the new population
Step 10: Update P^t and Go to Step 3.
Step 11: End
Step 12: Continue till a termination condition is fulfilled.

C. *Particle swarm optimization algorithm*

Particle swarm optimization (PSO) is a popular optimization technique developed as an inspiration from the behaviour of swarm [12]. PSO search for a best optimal result by analyzing the activity and clustering of birds [13].Each individual in PSO is called as particles and the entire population is called a swarm. At the beginning, the particles are randomly initialized in the search space with a certain velocity. In the search space, each particle exchanges the information between it and other neighbours and updates there velocity and direction. In each iteration global best particle position and particle's best position is saved and exchanged to calculate the next direction of move towards global best. The particle position is saved as a local best position, while the overall best particle positions are protected as the best global position, and are exchanged with all particles in the swarm until the best solution is obtained.

The algorithm for PSO algorithm is briefed as follows.

Algorithm 4:

Step 1: Randomly initialize the population and velocity for each particle with acceleration constants c1,c2. Each particle is characterized as a D dimensional vectors, $x_i = (x_{i1}, x_{i2}, ..., x_{iD}) \in S$. with an initial velocity $v_i = (v_{i1}, v_{i2}, ..., v_{iD})$.

Step 2: Repeat step 2 until the stopping condition is fulfilled.

Step2.1: At each iteration t, the position of each particle x_i^t is calculated

using

Equation 1 and the velocity v_i^t using Equation 2.

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}, \qquad i=1,\ldots,P \tag{1}$$

$$v_i^{(t+1)} = Wv_i^{(t)} + c_1 r_{i1} \times \left(P_{best\,i}^{(t)} - x_i^{(t)}\right) + c_2 r_{i2} \times \left(G_{best} - x_i^{(t)}\right) \tag{2}$$

Step 2.2: Compute the fitness function for each particle in the population and the new personal best solution P_{best} and global best solution G_{best} are assigned.

Step 2.3: Reduce the inertia weight by 5% in each iteration.

Step 3: Save the best solution obtained.

III. METHODOLOGY

A feed-forward neural network with 16 input nodes, 4 hidden nodes and 16 output nodes with the BPA as a training algorithm is considered in this work [1] [2]. However, BPA has some integral problems such as; the convergence of the algorithm depends on the initial values of the network parameters which are usually initialized as a random value. Also, BPA gets stuck into local optima, which is a major drawback. However, such a problem could be compensated using evolutionary algorithms such as GA and PSO, which are well identified for their strong global optimization. In this work, we have proposed a new algorithm that integrates the conventional back propagation algorithm with hybrid GA and PSO to do global and local search optimum [3] [4] [5]. Both GA & PSO are iterative optimization procedure. In this hybrid algorithm, GA and PSO represent the cascade structure. Both GA and PSO work with the same population, where a genetic algorithm is used to generate a perfect model for PSO from which the PSO particles are guided to evolve the model. The operations of GA along with well-diversified information of PSO particles can increase the search efficiency of both PSO and GA and could avoid the premature convergence of the algorithm. In this structure, the total numbers of iterations are equally shared by PSO and GA. Initially, the weights and bias of the ANN are chosen randomly and executes BPA until it reduces the MSE value or any one of the stopping conditions is satisfied. Then the weights and bias are represented as genes of the chromosomes for GA. The GA works on three operators, namely, the tournament parent selection, arithmetic crossover, and mutation. In each generation, GA evaluates fitness function (MSE) of all the individuals in the population and replaces the least performing individuals by performing crossover of the best parents. Further, if there is no progress in the fitness function genetic algorithm performs the mutation operation on the chromosomes. The PSO conducts search using a population of particles that corresponds to individuals in GA [16]. In each iteration, the solutions provided by the genetic algorithm forms exemplar for particles of PSO and PSO are executed to find P_{best} and G_{best}. The GA and PSO scheme stops if there is no further reduction in mean square error or the total number of iteration is reached. The best-obtained solution is stored for the operation phase of the neural network for further compression and decompression.

The structure of the methodology with HGAPSO algorithm can be briefed as follows.

Algorithm-5:

Step 1: Initialize and train the neural network as in algorithm-1.

Step 2: Save the weights and bias obtained in the neural network.

Step 3: From the obtained ANN solution form the population for both GA and PSO.

***/ Procedure for HGAPSO*/**

Step 3.1: Use GA (algorithm 3) to generate a model for PSO using selections, crossover and mutation operator.

Step 3.2: Use the solution of GA as input to PSO (algorithm4) and evaluate.

Step 3.3: Repeat 3.1.

Step 3.4: If no more improvement achieved with the usage of GA and PSO or any stopping condition is reached. Stop the algorithm.

Step 4: Save the optimized weights and bias for the operational phase.

IV. DATABASE DESCRIPTION

The performance of the proposed method is assessed on medical images. The images are classified as training and test images, 50 images are used to train the neural network and 20 images are used for testing the neural network. Some sample images are shown in Fig.2.

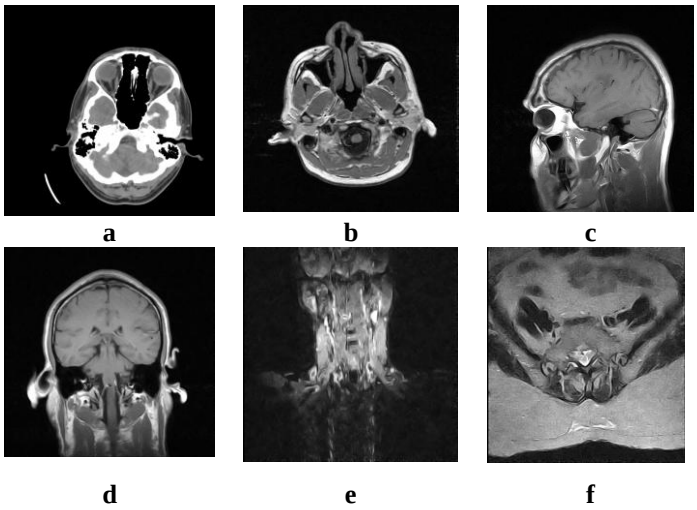

Fig.2.Sample images from the dataset.

V. RESULTS

This section presents experimental studies of medical image compression using HGAPSO optimized neural network. The study is conducted using MATLAB simulation tool in a windows platform with an i3 processor. Table I. shows the parameter values for the neural network and HGAPSO used for image compression.

The performance is measured using the following parameters:

1. PSNR
2. MSE
3. CR

- Peak Signal Noise Ratio (PSNR) is the ratio between the maximum possible power of the signal and the power of the corrupting noise. It is measured as

$$PSNR = 20 * log10(\frac{255}{\sqrt{MSE}}) \quad (3)$$

- Mean Squared Error (MSE) is the square of difference between the actual and desired output. It is measured as

$$MSE = \frac{1}{MN} \sum_{x=1}^{M} \sum_{y=1}^{N} (I(x,y) - I'(x,y))^2 \quad (4)$$

- Compression Ratio (CR) is measured in terms of saving realized using compression

$$CR = \frac{Compressed\ image\ size}{Uncompressed\ image\ size} \quad (5)$$

Table I: Parameter settings for experiment

Parameters	Value
Neural network structure	16-4-16
Block size for training	4*4
Maximum no of iteration for neural network	500
Minimum error for GA , PSO and Neural network	0.001
Population size of GA and PSO	50
Length of Chromosomes	148
Inertia Factor of PSO	1
No. of Iterations	100
Crossover Function	0.8
Mutation Function	0.1
Best Acceleration factor of PSO C1 & C2	1.5 & 2.0
Lower and upper bound variable	[-5 5]

Table II: Comparison of results

Compression ratio 75%	Neural Network		Neural Network Optimized with HGAPSO	
	PSNR	MSE	PSNR	MSE
Image_1	30.45	58.50	31.49	46.13
Image_2	32.98	32.72	33.60	28.35
Image_3	30.61	56.38	31.06	50.83
Image_4	36.13	15.82	36.80	13.57
Image_5	37.32	12.04	38.02	10.23
Image_6	31.08	50.61	31.59	45.01
Mean	33.09		33.76	

VI. CONCLUSION

In this paper, ANN is trained using an HGAPSO for medical image compression. The ANN training algorithm used combines traditional BPA with the Hybrid GA and PSO optimization technique. This hybrid algorithm combines the local search ability of PSO with the global search ability of GA. Thus the hybrid training algorithm improves the results by optimizing the weights and bias of ANN for achieving a good quality medical image after decompression than the traditional method. Future work could be concentrated on hybridizing other evolutionary algorithms for weights and bias optimization.

ACKNOWLEDGMENT

The authors would like to thank Doctoral committee members Dr.Raviraj P, Professor, GSSSITW, Mysuru and Dr. Shreekanth T, Project Manager, L&T Technology Services, Mysuru and also Dr. Shashikiran.B.R, Consultant radiologist, Sri Sai Ultrasound Centre, Hassan for their comments and suggestions in order to make this research work better.

REFERENCES

1. Verma, B., M. Blumenstein, and S. Kulkarni. "A new compression technique using an artificial neural network." Journal of Intelligent Systems 9, no. 1 (1999): 39-54.
2. Omaima NA, Al-Allaf, "Improving the performance of backpropagation neural network algorithm for image compression/decompression system." Journal of Computer Science 6, no. 11 (2010): 1347-1354.
3. P. VaRao, SbMadhusudana, S. SbNachiketh, KcKeerthi, "Image compression using artificial neural networks", ICMLC 2010 - The 2nd International Conference on Machine Learning and Computing, pp. 121-124, 2010.
4. Yoon, Byungjoo, Dawn J. Holmes, Gideon Langholz, and Abraham Kandel. "Efficient genetic algorithms for training layered feedforward neural networks." Information Sciences 76, no. 1-2 (1994): 67-85.
5. P.J.Werbos, Beyond regression:New tools for predictions and analysis in the behavioralscience,Ph.D.Thesis,Harvard University,1974
6. Chiroma, Haruna, SameemAbdulkareem, AdamuAbubakar, and TututHerawan. "Neural networks optimization through genetic algorithm searches: a review." Appl. Math 11, no. 6 (2017): 1543-1564.
7. Zajmi, Leke, Falah YH Ahmed, and Adam AmrilJaharadak. "Concepts, Methods, and Performances of Particle Swarm Optimization, Backpropagation,andNeural Networks." Applied Computational Intelligence and Soft Computing 2018 (2018).
8. Hassanin, Mohamed F., Abdullah M. Shoeb, and Aboul Ella Hassanien. "Grey wolf optimizer-based back-propagation neural network algorithm." In 2016 12th International Computer Engineering Conference (ICENCO), pp. 213-218. IEEE, 2016.
9. Nawi, NazriMohd, Abdullah Khan, and Mohammad ZubairRehman. "A new back-propagation neural network optimized with cuckoo search algorithm." In International Conference on Computational Science and Its Applications, pp. 413-426. Springer, Berlin, Heidelberg, 2013.
10. Aljarah, Ibrahim, HossamFaris, and SeyedaliMirjalili. "Optimizing connection weights in neural networks using the whale optimization algorithm." Soft Computing 22, no. 1 (2018): 1-15.
11. Gong, Y.J., Li, J.J., Zhou, Y., Li, Y., Chung, H.S.H., Shi, Y.H. and Zhang, J., 2015. Genetic learning particle swarm optimization. IEEE transactions on cybernetics, 46(10), pp.2277-2290.
12. Ali, Ahmed F., and Mohamed A. Tawhid. "A hybrid particle swarm optimization and genetic algorithm with population partitioning for large scale optimization problems." Ain Shams Engineering Journal 8, no. 2 (2017): 191-206.
13. Premalatha, K., and A. M. Natarajan. "Hybrid PSO and GA for global maximization." Int. J. Open Problems Compt. Math 2, no. 4 (2009): 597-608.
14. Nur, Abdirashid Salad, Nor HaizanMohdRadzi, and Ashraf Osman Ibrahim. "Artificial Neural Network Weight Optimization: A Review." TELKOMNIKA Indonesian Journal of Electrical Engineering 12, no. 9 (2014): 6897-6902.
15. Holland, John. "Holland, Adaptation in natural and artificial systems", MIT Press, Cambridge, MA (1992).
16. Nootyaskool, Supakit. "The hybrid implementation genetic algorithm with particle swarm optimization to solve the unconstrained optimization problems." In Knowledge and Smart Technology (KST), pp. 57-61. IEEE, 2012.
17. Han Feng, Man Tang, Jie Qi. "A backpropagation neural network based on a hybrid genetic algorithm and particle swarm optimization for image compression", 2011 4th International Congress on Image and Signal Processing, 2011.

AUTHORS PROFILE

Ravikiran H.K received the B.E. degree in Electronics and Communication Engineering from S.J.C. Institute of Technology, Chikballapur, Karnataka, India in the year 2008, and M.Tech. in Digital Electronics and Communication Systems from Malnad College of Engineering, Hassan, Karnataka, India, in 2010. He has a Teaching experience of 9 years. Currently he is a research scholar in GSSS Institute of Engineering and Technology for Women, affiliated to Visvesvaraya Technological University, Mysuru He has published 7 papers till date, of which 5 papers are from International Journals, rest are from conference. His subjects of interest include Image Compression, Neural Network, Soft Computing Techniques, etc.

.

Dr. Jayanth J received the B.E. degree in Electronics and Communication Engineering from Vidya Vikas Institute of Engineering and Technology, Mysore, Karnataka, India in the year 2008, and M.Tech. in Digital Electronics and Communication Systems from Malnad College of Engineering, Hassan, Karnataka, India, in 2010 and Ph.D from VTU, Karnataka, India, in 2017. He has a Teaching experience of 9 years. He has published 35 papers till date, of which 15 papers are from International Journals, rest are from conference. He has authored 3 Book chapter and written 3 books on Image Fusion and Image classification using fuzzy logic for remote sensed data classification. He is working as a PI for the DST NRDMS project research grant. He has completed 3 consultancy projects for Infy-cloud Bangalore. His subjects of interest include Image Processing, Swarm Intelligence and Remote Sensing.