Image Compression Using Neural Network

Sangeeta Mishra
TCET,Mumbai
Thakur Village, Kandivli(E)-101

Sudhir Savarkar
DMCE Airoli

ABSTRACT

Apart from the existing technology on image compression represented by series of JPEG, MPEG and H.26x standards, new technology such as neural networks and genetic algorithms are being developed to explore the future of image coding. Successful applications of neural networks to basic propagation algorithm have now become well established and other aspects of neural network involvement in this technology.

General Terms

Experimentation, Verification,Communication

Keywords

Artificial neural network, Image compression, Levenberg Marquardt algorithm

1. INTRODUCTION

Neural networks are inherent adaptive systems, they are suitable for handling nonstationaries in image data. Artificial neural network can be employed with success to image compression. The greatest potential of neural networks is the high speed processing that is provided through massively parallel VLSI implementations.

The most important part of a neuron is the multiplier, which performs high speed pipelined multiplication of synaptic signals with weights. As the neuron has only one multiplier the degree of parallelism is node parallelism. Each neuron has a local weight ROM (as it performs the feed-forward phase of the back propagation algorithm) that stores, as many values as there are connections to the previous layer.

The crucial problems of neural network hardware are fast multiplication, building a large number of connections between neurons, and fast memory access of weight storage or nonlinear function look up tables.

The aim is to design and implement image compression using Neural network to achieve better SNR and compression levels. The compression is first obtained by modeling the Neural Network in MATLAB. This is for obtaining offline training.

2. Neural network

2.1 Neural network outline

Neural network(NN) is a nonlinearity complex network system which consists of large quantities of processing unit analogying neural cells,has the capability of approaching the nonlinear function,very strong fault-tolerant ability and the quick study speed of local network.. Among them, feedforward network consists of three layers: the input layer, the hidden layer and the output layer. The hidden layer may have many layers,each layer neurons only accepts the previous layer neurons' output.And then ,this realizes the input and output nonlinear mapping relationship by adjusting the connected weighted value and the network structure[1].At present the neural network research method has formed many schools, the most wealthiest research work includes:the multi-layer network BP algorithm, Hopfield neural network model,adaptive resonance theory,self-organizing feature map theory and so on.This paper is based on researching BP neural network training quantification parameter [2],presents a network that is better than BP (Back Propagation) network in its property of optimal approximation, classify ability and the rapidity of study—Radial Basis Funtion(RBF).Many experimental results fully indicate, the RBF network have the stronger classification ability than other networks, the second place is the time series analysis problem.

2.2 Application neural network in video compression

This paper adopts prediction method based on neural network to realize forecast fast and effective for each encoding quantification parameter.This method trains neural networks through selecting video part continuous frame,uses the trained neural networks to obtain predictive value of the quantization parameter fastly, depends on the neural network unique faulttolerant ability to enhance the self-adaptivity of system.And then analogs signals' reconstruction model by the neural network, makes use of the image degradation model and reconstruction model to recover

3. Image compression using neural network

A two layer feed-forward neural network and the Levenberg Marquardt algorithm was considered. Image coding using a feed forward neural network consists of the following steps:

1.An image, F, is divided into rxc blocks of pixels. Each block is then scanned to form a input vector x (n) of size p=rxc

2. It is assumed that the hidden layer of the layer network consists of L neurons each with P synapses, and it is characterized by an appropriately selected weight matrix $\mathbf{W_h}$.

3. All N blocks of the original image is passed through the hidden layer to obtain the hidden signals, h(n), which represent encoded input image blocks, x(n) If L<P such coding delivers image compression.

4. It is assumed that the output layer consists of m=p=rxc neurons, each with L synapses. Let Wy be an appropriately selected output weight matrix. All N hidden vector h(n), representing an encoded image H, are passed through the output layer to obtain the output signal, y(n). The output signals are reassembled into p=rxc image blocks to obtain a reconstructed image, Fr.

5. There are two error matrices that are used to compare the various image compression techniques. They are Mean Square Error (MSE) and the Peak Signal-to-Noise Ratio (PSNR). The MSE is the cumulative squared error between the compressed and the original image whereas PSNR is the measure of the peak error.

$$MSE = {}^{I}\!/_{MN} \sum_{y=1}^{m} \sum_{x=1}^{n} [I(x,y) - I'(x,y)]^2 \quad .....1$$

6. The quality of image coding is typically assessed by the Peak signal-to-noise ratio (PSNR) defined as

PSNR = 20 $\log_{10}$ [255/sqrt(MSE)]…………………..2

7. Training is conducted for a representative class of images using the Levenberg Marquardt algorithm.

8. Once the weight matrices have been appropriately selected, any image can be quickly encoded using the W_h matrix, and then decoded (reconstructed) using the W_y matrix.

Levenberg Marquardt Algorithm

The Levenberg Marquardt algorithm is a variation of Newton's method that was designed for minimizing functions that are sums of squares of other nonlinear functions. This is very well suited to neural network training where the performance index is the mean squared error.

Basic Algorithm:

Consider the form of Newton's method where the performance index is sum of squares. The Newton's method for optimizing a performance index F(x) is

$X_{k+1} = X_k - A_k^{-1} g_k$, …………………………………………….3

Where $Ak = \nabla^2 F(x)$ and $g_k = \nabla F(x)$;

It is assume d that F (x) is a sum of squares function:

$$F(x) = \sum_{r=1}^{n} v_i^2(x) = V^T(x) v(x) \quad ……………………….4$$

Then the j^{th} element of the gradient will would be

$$[\nabla F(x)]_j = \delta F(x) / \delta S_j = 2 \sum_{i=1}^{n} V_i(x) \delta v_i(x) / \delta x_j$$

……………………..5

The gradient can be written in matrix form:

$\nabla F(x) = 2J^T(x)\, v(x)$,………………………………………...6

Where J(x) is the Jacobian matrix.

Next the Hessian matrix is considered. The k.j element of Hessian matrix would be

$$[\nabla^2 F(x)]_{kj} = \delta^2 F(x) / \delta x_k\, \delta x_j$$

The Hessian matrix can then be expressed in matrix form: $\nabla^2 F(x) = 2 J^T(x) J(x) + 2 S(x)$ Where this algorithm has the very useful feature that as μk is increased it approaches the steepest descent algorithm with small learning rate.

$$S(x) = \sum_{i=1}^{n} V_i(x) . \nabla^2 v_i(x)$$

Assuming that S(x) is small, the Hessian matrix is approximated as $\nabla^2 F(x) \equiv 2 J^T(x) J(x)$

Substituting the values of ∇2 F(x) & ∇ F(x), we obtain the Gauss-Newton method:

$$X_{k+1} = X_k - [J^T(X_k) J(X_k)]^{-1} J^T(X_k) V(X_k)$$

One problem with the Gauss-Newton over the standard Newton's method is that the matrix $H=J^TJ$ may not be invertible. This can be overcome by using the following modification to the approximate Hessian matrix:G = H + μ I.

This leads to Levenberg –Marquardt algorithm

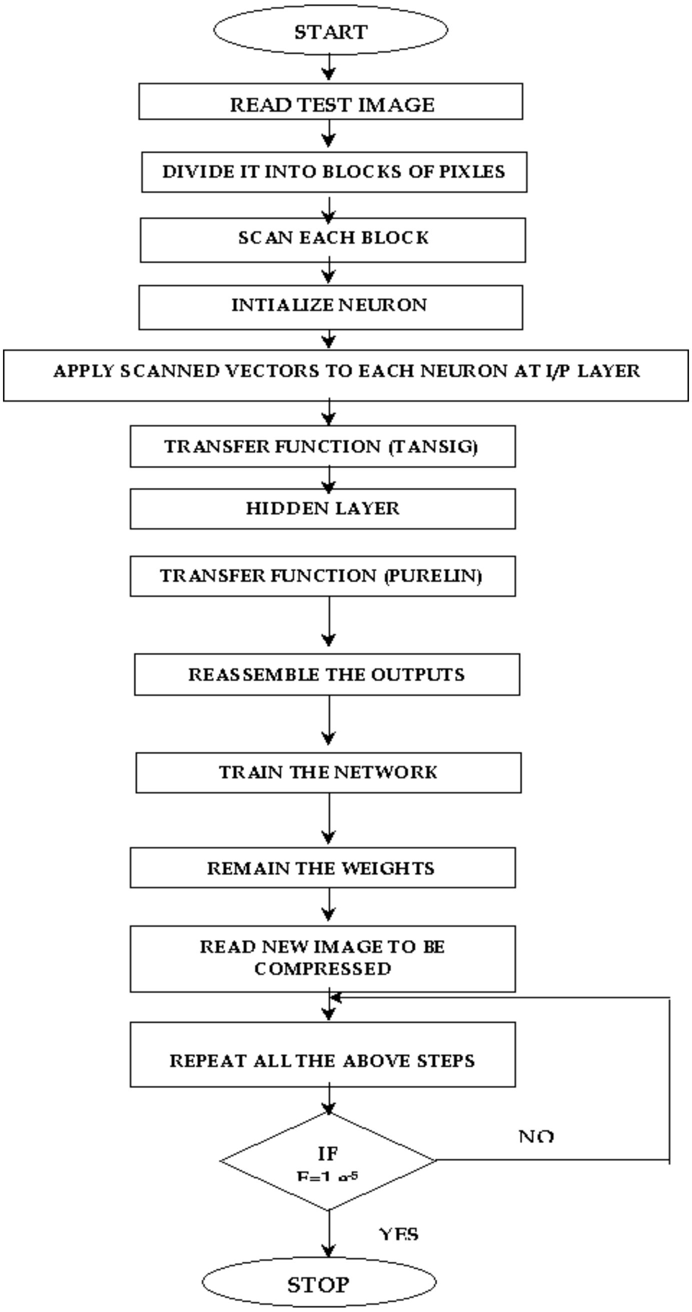
$$X_{k+1} = X_k - [J^T(X_k) J(X_k) + \mu_{kI}]^{-1} J^T(X_k) V(X_k)$$

Or $\Delta X_k = - [J^T(X_k) J(X_k) + \mu_{kI}]^{-1} J^T(X_k) V(X_k)$

The iterations of the Levenberg- Marquardt back propagation algorithm (LMBP) can be summarized as follows:

1. Present all inputs to the network and compute the corresponding network outputs and the errors $e_q = t_q - a^M{}_q$. Compute the sum of squared errors over all inputs. F(x).
2. $F(x) = \sum e_q{}^T e_q = \sum\sum (e_{j.q})^2 = \sum (v_i)^2$

Compute the Jacobian matrix. Calculate the sensitivities with the recurrence relation. Augment the individual matrices into the Margquardt sensitivities.



3. Obtain ΔX_k.

Recompute the sum of squared errors using xk + ΔX_k. If this new sum of squares is smaller than that computed in step 1 then divide μ by v, let $X_{k+1} = X_k + \Delta X_k$ and go back to

1. step 1. if the sum of squares is not reduced, then multiply μ by v and go back to step 3.

Training Procedure

During training procedure data from a representative image or a class of images is encoded into a structure of the hidden and output weight matrices.It is assumed that an image, F, used in training of size Rx C and consists of rxc blocks.

1. The first step is to convert a block matrix F into a matrix X of size P x N containing training vectors, x(n), formed from image blocks.
That is:

$$P = r.c \text{ and } p.N = R.C$$

2. The target data is made equal to the data, that is:

$$D=X$$

3. The network is then trained until the mean squared error, MSE, is sufficiently small.

The matrices W^h and W^y will be subsequently used in the image encoding and decoding steps.

Image Encoding

The hidden-half of the two-layer network is used to encode images. The Encoding procedure can be described as follows:

$$F \rightarrow X,\ H = (W^h.\ X)$$

Where X is an encoded image of F.

Image Decoding

The image is decoded (reconstructed) using the output-half the two-layer network. The decoding procedure is described as follows:

$$Y = (W^y.\ H),\ Y \rightarrow F$$

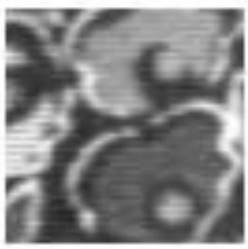
These steps were performed using MATLAB (Matrix laboratory). The compression so obtained was though offline learning. In the off-line learning methods, once the systems enters into the operation mode, its weights are fixed and do not change

- A skeleton mounted on a support structure to acquire the finger position of the patient who puts his hand in a glove vertically.
- Variable rotational resisters, sited in correspondence to the phalanx, that are able to retrieve finger movements.
- Surface active electrodes that send an electromyographic signal through a double-channel probe.

Some strain gauges are mounted on the structure in order to measure the force of the contact.

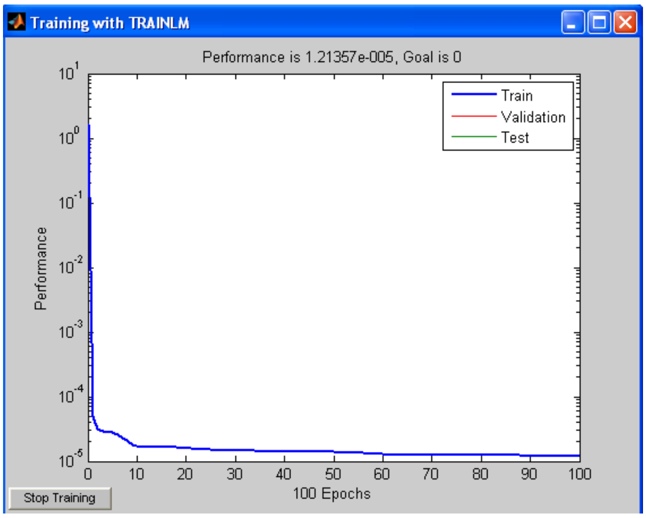
The skeleton is supported from a PC equipped with an ISA/EISA data acquisition card and system management software developed in C language using the lab windows library for the analysis, acquisition of the data, and the construction of the graphical interface.

4. Results



Original Image

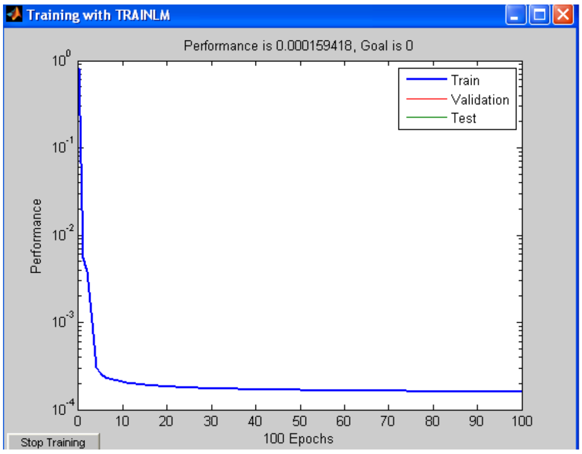
Image Into Blocks



Training is achieved

Original Image

First Block Reconstructed Image



Training is achieved

5. CONCLUSION

The computing world has a lot to gain from neural networks. Their ability to learn by example makes them very flexible and powerful. Furthermore there is no need to devise an algorithm in order to perform a specific task; i.e. there is no need to understand the internal mechanisms of that task. They are also very well suited for real time systems because of their fast response and computational times which are due to their parallel architecture. Neural networks also contribute to other areas of research such as neurology and psychology. They are regularly used to model parts of living organisms and to investigate the internal mechanisms of the brain.

6. FUTURE SCOPE:

Artificial Neural Networks is currently a hot research area in image processing and it is believed that they will receive extensive application to various fields in the next few years. In contrast with the other technologies, neural networks can be used in every field such as medicine, marketing, industrial process control etc. This makes our application flexible and can be extended to any field of interest. Integrated with the other fields like Artificial intelligence, fuzzy logic neural networks have a huge potential to perform. Neural networks have been applied in solving a wide variety of problems. It is an emerging and fast growing field and there is a huge scope for research and development.

7. REFERENCES

1. H.Demuth and M. Beale. Neural Network TOOLBOX User's Guide. For use with MATLAB. The Math Works lne.. (1998)
2. Henrieue Ossoinig. Erwn Reisinger., Reinhold Weiss Design and FPGA-Implementation of a Neural Network
3. Kiamal Z. Pekmestzi. Multiplexer-Based Array Multipliers. IEEE Transcation on computers, vol, 48 , JAN (1998)
4. Hennessy. J.L and Patterson. D.A. Compter Architecture: A quantitative Approach. Morgan Kaufmanns, (1990)
5. J. Jiang. Image compression with neural networks. Signal Processing: Image Communication 14 (1999) 737-760