Image Compression Using MLP Neural Network

Praveen Singh Tomar[1], Pankaj Kawadkar[2], Parmalik Kumar[3]

Dept. of Computer Science & Engg.,

Patel college of Science & Technology, Bhopal

Abstract: Image compression techniques are used to reduce the size of image, which helps to reduce the storage space and transmission cost. In the present research Multi Layer perceptron Algorithm has been used. MLP algorithm helps to decrease the time required for transmission and storage of image . The proposed scheme has been demonstrated through several experiments and very promising results in compression as well as in reconstructed image over convectional neural network based technique.

Index Terms: Neural Networks, Image Compression.MLP.

1. INTRODUCTION

Digital image compression is a key technology in the field of communications and multimedia applications. Image compression is often referred to as coding, where coding is very general term encompassing any special representation of data which satisfies a given need. As with any communication, compressed data communication only works when both the sender and receiver of the information understand the encoding scheme. For example, this text makes sense only if the receiver understands that it is intended to be interpreted as characters representing the English language. Similarly, compressed data can only be understood if the decoding method is known by the receiver. It is useful because it helps to reduce consumption of expensive resources such as hard disk space or transmission bandwidth i.e. a data file that suppose to takes up 50 kilobytes (KB) could be downsized to 25 kilobytes (KB), by using data compression software. A simple characterization of data compression is that it Involves transforming a string of characters in some representation (such as ASCII) into a new string (of bits, for example) which contains the same information but whose length is as small as possible. Data compression has important application in the areas of data transmission and data storage. Compressed data required smaller storage size and reduce the amount of data that need to be transmitted. Hence, it increases the capacity of the communication channel.

There are "lossless" and "lossy" forms of data compression. Lossless data compression is used when the data has to be uncompressed exactly as it was before compression. Text files are stored using lossless techniques, since losing a single character can in the worst case make the text dangerously misleading. Archival storage of master sources for images, video data, and audio data generally needs to be lossless as well. However, there are strict limits to the amount of compression that can be obtained with lossless compression. Lossless compression ratios are generally in the range of 2:1 to 8:1.

Lossy compression, in contrast, works on the assumption that the data doesn't have to be stored perfectly. Much information can be simply thrown away from images, video data, and audio data, and when uncompressed such data will still be of acceptable quality.

Compression ratios can be an order of magnitude greater than those available from lossless methods. Several techniques have been used for data compression. Each technique has their advantages and disadvantages A lot of image compression methods based on neural network have been presented for this purpose, They can be classified as lossless or lossy image compression technique. Rutherhart in his study explore the power of neural network to encode /decode data, which is later used by many experts for image compression using the back propagation & training algorithm. In these algorithms, an image is partioned into many non overlapping groups of pixel, and feded to network training. Image compression is obtained by encoding the pixels into trained set of weight, and it is send to side where the image is reconstructed. This method has certain advantages as compare to vector quantization because in this method no code books are required and encoding/decoding time are much less. But in these cases amount of compression obtained is limited because it oppressed only the correlation between pixel within each of the training patterns. To overcome this Adaptive one hidden layer feed forward neural network was invented to get better image compression. It minimizes the required network size and the time required for computation as well based on the image size. The basic goal is to design an edge preserving image compression technique using one hidden layer feed forward neural network of which the neurons are find adaptively on the basis of the images to be compressed. Edge detection encodes information on the basis of structure of the image so it is a important step of reduction. Using edge detection very important data of the image is preserved whereas keeping aside smaller data that effectively reduces dynamic vary of the image and components pixel redundancy. As a next step the image is thresholded to observe the pixel having less influence on the image and so removed. A function has been designed mistreatment grey level data of the edge detected image and applied to reduce the size additional. Finally thining operation has been applied supported the interpolation technique to cut back thickness of the image

II. IMPLEMENTATION OF MLP ALGORITHM

The algorithm for Perceptron Learning is based on the back-propagation rule discussed previously. This algorithm can be coded in any programming language, and in the case of this tutorial, Java for the applets. In this case we are assuming the use of the sigmoid function *f(net)* described earlier in the tutorial. This is because it has a simple derivative.

x = input training vector

t = Output target vector.

δk = portion of error correction weight for wjk that is due

to an error at output unit Yk; also the information about

the error at unit Yk that is propagated back to the hidden

units that feed into unit Yk

δj = portion of error correction weight for vjk that is due to the backpropagation of error information from the output layer to the hidden unit Zj

α = learning rate.

voj = bias on hidden unit j

wok = bias on output unit k

Algorithm:

Step 1.Initialize weights.

(set to small random values).

While stopping condition is false, do steps 2-9.

Step 2. For each training pair, do steps 3-8.

Feed forward:

Step 3. Each input unit(Xi,, i=1………..n)receives

input signal Xi and broadcasts this signal to all units in the layer (this hidden units).

Step 4. Each hidden units (Zj, j=1…………p) sums its weighted input signals.

applies its activation function to compute Its output signal

$$Zj = f'(z\_inj),$$

and sends this signal to all units in the layer above (output units).

Step5.Each output unit(yk,k=1…..m) sums its weighted input signals.

and applies its activation function to compute its output signal,

Error:

Step6. Each output unit(yk,k=1…………..…m) receives target pattern corresponding to the input training pattern, computes its error information term.

$$\delta k = (tk - yk) f'(y\_ink)$$

Calculate its weight correction term (used to update later).

Calculate its bais correction term (used to update later),

and send to units in the layer below.

Step 7. Each hidden units (zj,j=1……p) delta inputs

$$\delta\_inj = \sum_{k=1}^{m} \delta k wjk$$

Multiplies by derivation of its activation function to calculate its error information term

$$\delta j = \delta\_inj f'(z\_inj)$$

Calculates its weight correction term (used to update later)

$$\Delta vij = \alpha \delta j xi$$

and calculates its bais correction term (used to update later).

$$\Delta voj = \alpha \delta j$$

Update weights and biases:

Step8. Each output(yk,k=1……m) updates its bias and weight(j=0……p)

$$wjk(new) = wjk(old) + \Delta wjk$$

Each hidden units (zj,j=1…….p)updates its bais and weights(i=o……..n);

$$vij(new) = vij(old) + \Delta vi_j$$

Step 9. Test Stopping Condition.

III. ACTIVATION FUNCTION OF MLP ALGORITHM

If a multilayer perceptron consists of a linear activation function in all neurons, that is, a simple on-off mechanism to determine whether or not a neuron fires, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two-layer input-output model. What makes a multilayer perceptron different is that each neuron uses a nonlinear activation function which was developed to model the frequency of action potentials, or firing, of biological neurons in the brain. This function is modeled in several ways, but must always be normalizable and differentiable.

The two main activation functions used in current applications are both sigmoids, and are described by hyperbolic tangent which ranges from -1 to 1, and the latter is equivalent in shape but ranges from 0 to 1. Here yi is the output of the ith node (neuron) and vi is the weighted sum of the input synapses. More specialized activation functions include radial basis functions which are used in another class of supervised neural network models.

Most common activation functions are the logistic and hyperbolic tangent sigmoid functions.

The project uses hyperbolic tangent function:

$f(x)= \{2/(1+e-\lambda x)\}-1$

and derivative:

$f'(x)=f(x)(1-f(x))$

6.2.3 Weight AdjustmentThe networks weights need to be adjusted in order to minimize the difference or the error between the output and the expected output. This is explained in the equations below.The error signal at the output layer of the ith neuron at iteration n is given by

$ei\ (n) = X\ i\ (n) - X\ 'i\ (n)$

Where Xi represent the desired out put and X'i represent the actual out put. The error function over all neurons in output layer is given by Eq.

$$El\ (n) = \sum ei\ 2(n)$$

The error function, over all input vectors in the training image, is

$$E = \sum El\ ,\ El = (\ X\ '\ ,\ w)$$

where l indexes the image blocks (inputs vector), X' is the vector of outputs, and w is the vector of all weights. In order to minimize the error function with respect to weight vector (w) it is necessary to find an optimal solution (w*) that satisfy the condition

$$E\ (\ w^*) \leq E\ (\ w)$$

The necessary condition for the optimality is

$$\Delta E\ (\ w) = 0$$

where Δ is gradient operator

$\Delta = [\partial\ /\ \partial w]$ and $\Delta E\ (w)$ is gradient vector (g) of error function is defined as follows

$$\Delta E\ (\ w) = \partial E\ /\ \partial w$$

The solution can be obtained using a class of unconstrained optimization methods based on the idea of local iterative descent. Starting with initial guess denoted w(0), generate a sequence of eight vectors w(1),w(2)… such that the error function is reduced for each iteration

$$E\ (\ wn +1\ ) \leq E\ (\ wn\ )$$

IV. RESULT AND DISCUSSION

For the simulation of the whole compression process, we used 4x4 points image blocks (as a compromise between the compression rate and the correlation between successive blocks. The Kohonen MLP ALGORITHM algorithm was trained with an exponentially decreasing function $\alpha(t)$. While the simulations have been carried out on different images with similar results, the standard Lena image is used in this article for illustration purposes.

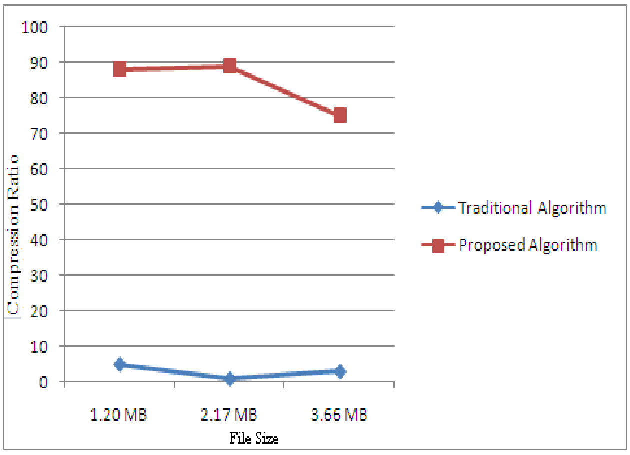
First, we have to show the consequences of the low-pass filtering: by removing a part of the high frequencies, we delete a part of the information contained in the image.An immediate consequence of this is a reduction of the compression ratio, even though the image visual quality remains more or less unchanged. In other words, before beginning the compression, the image quality will be degraded by the filtering. We calculate the compression ratio of traditional algorithm. Following table show the compression ratio of traditional algorithm

Result Compare Between Tradition & New Approach							
			Tradition approach				Proposed Approach
S.No	File Type	File Size	Huffman coding	Shano-fano	LZW	Ratio	Ratio
1	BMP	6.77 MB	5.88 MB	6.06 MB	6.07 MB	30%	77%
2	JPG	377 KB	378 KB	386 KB	528 KB	21%	65%
3	TIFF	2.17 MB	2.15 MB	2.20 MB	3.02 MB	18%	89%

Compression Ratio of Tradition Algorithm and proposed algorithm

The above table shows the time taken by the major operation in each algorithm. In, image compression using MLP Technique the major operation is finding Best Matching unit that is, finding square root of vectors where as in Image file. Experiments were also conducted to obtain the compression ratios obtained from the two algorithms i.e. MLP Technique and Traditional algorithm for different image resolutions and a graph is plotted as shown in figure 1. From the graph it is seen as the resolution of the images increases MLP Technique gives better compression ratio when compared to JPEG. It has been seen MLP ALGORITHM takes less amount of time to compress the blocks when compared to Image block compression.

Compression ratio of proposed approach



MLP ALGORITHM has the ability to enhance any noisy compressed image that had been corrupted during compressed image transmission through a noisy digital or analog channel. Practically, the MLP ALGORITHM has the ability to compress untrained images but not in the same performance of the trained images. This can be done especially when using small number of image block dimension (P).

The MLP ALGORITHM Algorithm offers us a way of encoding knowledge as a set of training examples rather than by a set of rules and is an effective technique for problem domains where there are many rules or the rules cannot be easily devised. This research also shows that conventional computing and artificial neural networks are not in conflict with each other but each can be exploited for the advantages they offer.

V. CONCLUSION

The “Image Compression using MLP Algorithm” has been successfully programmed and tested he 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 time taken for computations 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. Perhaps the most exciting aspect of neural networks is the possibility that some day 'conscious' networks might be produced. There is a number of scientists arguing that consciousness is a 'Mechanical' property and that 'conscious' neural networks are a realistic possibility. Even though neural networks have a huge potential we will only get the best of them when they are integrated with computing, AI, fuzzy logic and related subjects. Neural networks are performing successfully where other methods do not, recognizing and matching complicated, vague, or incomplete patterns.

VI. FUTURE SCOPE

Artificial Neural Networks is currently a recent research area in image processing and it is supposed that they will receive wide application to diverse 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. Artificial Neural Networks is currently a hot research area in Data compression. MLP ALGORITHM algorithm for data compression being a wide field which is rapidly finding use in many applied fields and technologies MLP ALGORITHM has some limitation. They can not compress higher size of audio and video file. So To Improve the Compression Ratio of higher size of audio and video file in future enhancement.

REFERENCES

[1] Jiang, J. A Neural Network Design for Image Compression and Indexing. International Conference on Artificial Intelligent Expert Systems and Neural Networks, Hawaii, USA, 1996, 296-299.

[2] Joutsensalo, J. Non Linear Data Compression and Representation by Combining Self Organising Map and Subspace Rule, Vol-2 IEEE International Conference on Neural Networks, 1994, 637-642.

[3] Blumenstein, M The Recognition of Printed and Handwritten Postal Address using Artificial Neural Networks. Dissertation, Griffith University, Australia, 1996

[4] Nelson M, The Data Compression Book M & T Publishing Inc. 1991

[5] Gelene R., and Sungar, M. Random Network Learning and Image Compression using Artificial Neural Networks, Vol-6 IEEE 1994, 3996-4001.

[6] Sicuranza, G.L, Ramponi G, and Marsi S. Artificial Neural Network for image compression, Electronic letters, 1990, 477-479.

[7] Ash, T. (1989) Dynamic Node Creation in Backpropagation Networks, 1989, 365-375.

[8] Kenue S. K. Modified Back-Propagation Neural Network with Applications to Image Compression, SPIE Vol.1709, Applications of Artificial Neural Networks, 1992, 394-401.

[9] Carbonara, M., Fowler J., and Ahalt, S, Compression of Digital Video Data using Artificial Neural Network Differential Vector Quantization, SPIE Vol. 1709, Applications of Artificial Neural Networks, 1992, 422-433.

[10] Min, K., and Min, H. Neural Network Based Image Compression using AMT DAP 610, SPIE Vol. 1709, Application of Artificial Neural Networks, 1992,386-393.

[11] Panchanathan, S., Yeap, T., and Pilache, B. A Neural Network for Image Compression, SPIE Vol. 1709, Application of Artificial Neural Networks, 1992 376-385.

[12] R. C. Gonzales, R. E. Woods, Digital Image Processing, Second Edition, Prentice-Hall, 2002.

[13] Veisi H., Jamzad M., Image Compression Using Neural Networks, Image Processing and Machine Vision Conference (MVIP), Tehran, Iran, 2005.