Image Compression Based On Inverse Difference Pyramid with BPNN

Noha A. Hikal[1], Roumen K. Kountchev

Abstract - **In this paper a new developed algorithm for lossless compression of still grayscale images based on Back Propagation learning Neural Networks in correspondence with the method of Inverse Difference Pyramid (IDP) decomposition is presented. This algorithm is well suited to be used in Progressive Image Transmission (PIT). Advantage of the method is the adaptation of the neural network in accordance with the image contents, the minimization of the total number of pyramid levels and the increasing of the restored image quality.**

Keywords - **Inverse Difference Pyramidal Decomposition, Back Propagation Neural Networks, Image Coding**

I. INTRODUCTION

Image compression is an important tool to store and transmit visual information used for several applications. Compression of an image refers to a process in which the amount of data used to represent an image is reduced to meet a bit rate requirement (below or at most equal the maximum available bit rate), while the quality of the reconstructed image satisfies a requirement for a certain application and the complexity of the computation involved is affordable for the application. Progressive Image Transmission (PIT) [1] concept is of particular importance in browsing large image files. Progressive transmission of an image permits the initial reconstruction of an approximation followed by a gradual improvement of quality in image reconstruction. A coarse copy of the image is sent first to give the receiver an early impression of image contents then subsequent transmission provides image detail of progressively finer resolution. The observer may terminate transmission of an image as soon as its contents are recognized. In order to send image data progressively, the data should be organized hierarchically in the order of importance, from the global characteristics of an image to the local details. There are two types of data structures for progressive transmission depending upon the encoding method employed [2]:

- transform based image encoding,
- spatial encoding.

In a transform based encoding the image is first divided into a set of contiguous non-overlapping blocks, and then each block is transformed into a set of transform coefficients, (e.g, Discrete Cosine Transform (DCT) [3], the coefficients are then quantized before initiating its transmission. On the other hand, the spatial approach, like pyramidal encoding, generates a sequence of images with different resolution (corresponding with the pyramid levels), the image is successively reduced in spatial resolution and size by sub-sampling or averaging. The images are restored using the data from all pyramid levels, which is arranged, interpolated and summed.

II. PYRAMIDAL REPRESENTATION - STATE OF ART

The first pyramidal data structure is the Gaussian - Laplacian pyramid [4]. Gaussian Pyramid (GP) can be viewed as a set of low pass filtered copies of the original image. Laplacian Pyramid (LP) is a sequence of error images, each is a difference between two successive levels of Gaussian pyramid. Various pyramid data structures for progressive image transmission have been proposed like [5]: Mean Pyramid (MP), Reduced Sum Pyramid (RSP), and Reduced Difference Pyramid (RDP). For further refinement of the Laplacian pyramid was developed the Least Squared Laplacian Pyramid (LSLP) [6]. The corresponding LSLP is generated by adding an extra filter with auxiliary coefficients sequence after the down sampling process, this filter works on minimizing the energy of the Laplacian coefficients. Reduced Laplacian Pyramid (RLP) [7] is designed by discarding the reduction filter and adopting a halfband interpolator. In [8] a contrast pyramid coding technique, which differs in the way of computing the difference image was discussed. Instead of using the difference image to represent the information difference between two successive levels, a coding scheme consists of the generation of the contrast image with a simple nonlinear algorithm and a simple compandor model is used. The efficiency of contrast pyramid method comes from coding the contrast image. Centered pyramid [9] differ from other pyramids methods in that each coarser level node is suited exactly in the center of its finer level predecessors, which offers an accurate way of up projection and makes it helpful in contour, multi-scale detection and object recognition. Hierarchy Embedded Differential Image (HEDI) [10] is a technique similar to RDP but expanded and generalized to improve the speed for compressing and decompressing processes. Many pyramidal decomposition techniques showed improvements over Joint Photographers Experts Group (JPEG). The Inverse Pyramidal Decomposition with multiple Discrete Cosine Transform (IDP/DCT) was presented in [11]. The IDP decomposition differs in the way of obtaining pyramid levels, the word "inverse" refers to compute the pyramid levels from bottom (level zero) to the top. The novelty lies in the modeling performed at each pyramid level, which relies on the DCT of the input subimage. This new pyramid can be compared to subband DCT [12] to notice that the IPD/DCT offers better performances in terms of compression ratio with a fixed PSNR for each pyramid

[1]Noha A. Hikal is with Mansoura University, Communications Engineering Dept, Egypt,E-mail: eng_noha_hikal@yahoo.com

[2]Roumen K. Kountchev is with FCTT, Radiocom. Dept., Technical University of Sofia, Bul. Kl. Ohridsky 8, Sofia, Bulgaria, E-mail: rkountch@tu-sofia.bg

level. Coding and decoding processes are simple and flexible. Therefore, it is well adapted to the needs of users, and it is relatively simple for real time processing. Since the pyramid top now consists of the low frequency coefficients of the DCT, this makes it in correspondence with the requirement for PIT. As it is known, the DCT is image independent, which provides more simplicity than other methods (Fourier transform. KLT, etc). The number of coefficients necessary to ensure high quality of the restored image using DCT is relatively high. The NN's are interesting alternatives to this classical approach for image processing and compression techniques due to their quality image reproduction, lower matrices computational effort, and adaptation. In addition to the NN's structure features are such as their massively parallel structure, high degree of interconnections, capability of learning and self-organizing, which allow them to solve several problems in processing image data.

In this work, a new technique for building Inverse Difference Pyramidal (IDP) Decomposition for image compression based on merging IDP and Back Propagation learning rule NN's (BPNN) [13] is considered. This technique combines the advantages of both methods IDP and NN's, the adaptation and learning capabilities of NN's could improve the performance of IDP, decrease the number of pyramid levels, increase the quality of the reconstructed image, maximize PSNR and minimize MSE.

III. IDP-BPNN Image Compression Algorithm

A. IDP-BPNN Coding

Lossless coding of image data in accordance with IDP-BPNN procedure is performed in the following steps:

Step 1: The whole image B(i,j) of size $2^N x 2^N$ is divided in blocks each of them of size (*m* x *m)* pixels.

Step 2: For each block, a three layer NN (input layer, hidden layer, and output layer) is built. The large input layer consisting of (m^2) neurons feeds the small hidden layer consisting of (*m*) neurons, which then feeds the large output layer consisting of *(*m^2*)* neurons. This structure is referred to as a "bottleneck" type networks ($m^2/m/m^2$), shown in Fig. 1.

Step 3: Using row scanning, pixels of each block are arranged as a vector of length *(*m^2*)*; these (m^2) components are considered the input vector of the three-layer NN.

Step 4: The Back propagation learning algorithm is used as an adaptive approximation algorithm, providing the most suitable (*m)* hidden weights values that approximate the (m^2) input vector with minimum mean squared error. This form of number of pixels reduction for each block is considered a spatial reduction form, which serves the PIT. The training of the NN proceeds as follows: for example, a 256x256 pixels training image is used to learn the bottleneck type network to create the required identity map. Training input–output pairs are produced from the training image by using blocks of size 8x8 pixels of the image itself. Once training is complete, image compression is demonstrated in the recall phase. The total number of *(m)* hidden weights from all blocks forms the coefficients of the first pyramid level (p=0).

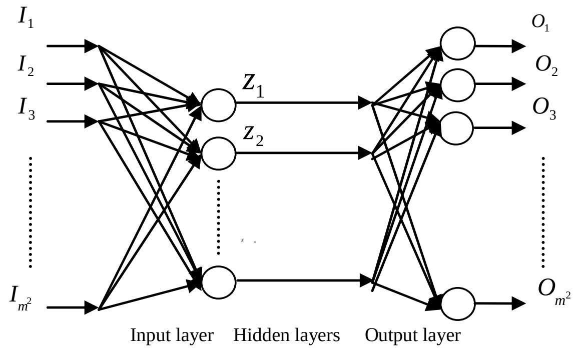


Fig. 1. Bottleneck type Neural Network

Step 5: The coefficients obtained are encoded using Run Length Encoding (RLE) [14] and transmitted.

Step 6: At the transmitter side, the reconstructed blocks will be recovered from these coefficients (after decoding process) in reverse arrangement using one layer BPNN *(*m/m^2) for each block, Fig.2. The difference is calculated pixel by pixel between the original image and the reconstructed blocks, which approximate the image results and in result is defined a difference image of the same size as the original one ($2^N x 2^N$).

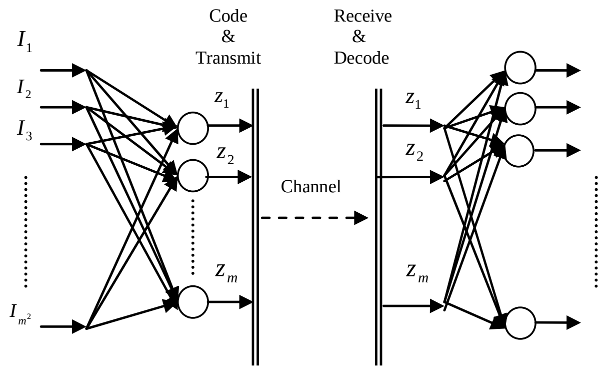


Fig.2. Compression/Decompression scheme for one block of size *m*x*m* using trained Bottleneck type Neural Network

Step 7: For the obtained differences image E_0(i,j), each block is divided again into four sub-blocks with size (*m/2*x*m/2)*, and again for each sub-block, a three layer NN is built of type ($(m/2)^2/m/2/(m/2)^2$).

Step 8: Following the same procedure, after training NN provides output sub-blocks of size (*m/2*x*m/2)*, approximating the input sub-block. The total number of *m/2* hidden weights from all sub-blocks forms the coefficients of the second pyramid level (p=1), which will be encoded and transmitted later.

Step 9: At the transmitter side, the coefficients of the second pyramid level are used to generate a reconstructed version of the difference image E_0(i,j). Pixel by pixel difference is obtained between E_0(i,j) and its reconstructed version E_0'(i,j). A new difference E_1(i,j) is obtained, which must be divided into 4 sub-blocks again, each of size (*m/4*x*m/4)*, and a new three layer NN is build in similar way as preceding ones. The total number of *m/4* hidden neurons for all sub-blocks form the coefficients of the third pyramid level (p=2), which will be encoded and transmitted later. The process continues with the same manner till having a high quality reconstructed image. True image restoration is ensured when the hidden layer weight values are encoded and added in the transmitted data as well. It can be noticed that the obtained compressed coefficients (the values of the hidden weights) for each pyramid level contain a number of similar values. Thus, scanning these coefficients and apply Run-Length Encoding (RLE) algorithm on the scanned data could achieve further compression.

B. IDP-BPNN Decoding

The decoding of the lossless compressed image data in accordance with the general BPNN-IDP procedure is performed in following the steps.

Step 1: For each level p, Run Length Decoding process will be done to get the hidden weights values.

Step 2: The reconstructed blocks of the image are calculated using the corresponding reconstruction BPNN's arrangement for each level as the one's used at the transmitter side.

Step 3: The elements $B'(i,j)$ of the restored image from all levels are calculated in accumulation way

$$B'(i,j) = \sum_{p=0}^{P-1} \tilde{B}(i,j) \quad (1)$$

for i, j = 0, 1, . ., 2^n-1 and $\tilde{B}(i,j)$ is the reconstructed image from each level p using BPNN's at the receiver side. Fig.3 illustrates the procedure of using IDP-BPNN at the transmitter and at the receiver. Fig.4 shows a general block diagram of IDP-BPNN decomposition and reconstruction.

C. Coding of color images

The coding of color images (written in format 4:4:4), based on the described algorithm, can be done by applying the algorithm on the matrix of every primary color component: R, G, B. In order to obtain higher compression ratio, the R, G, B components of every pixel (i,j) are transformed in Y, Cr, Cb [14] and the 4:4:4 format was converted into 4:2:0. Each one of the components Y, Cr, Cb is processed, applying the already described general BPNN-IDP coding algorithm to get a new matrices $\hat{Y}$, $\hat{C}r$, $\hat{C}b$ for each level, which approximate the original ones.

The decoding of the compressed color images is performed applying the BPNN-IDP decoding algorithm on the components $\hat{Y}$, $\hat{C}r$, $\hat{C}b$.

IV. Performance Evaluation of BPNN-IDP

To evaluate the performance of the proposed BPNN-IDP algorithm, the commonly known measures will be used which are:

- The Peak Signal to Noise Ratio (PSNR) obtained for the reconstructed image at each level p of the pyramid as:

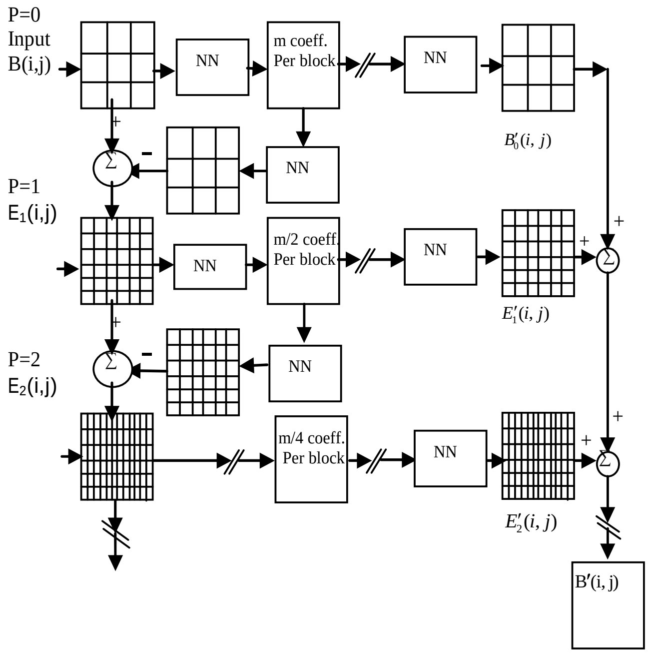


Fig.3. Block diagram of the 3-level IDP-BPNN decomposition and reconstruction

$$PSNR(p) = 10\log_{10}\frac{B_{max}^2}{\bar{\varepsilon}^2(p)}, dB \quad (2)$$

Where $\bar{\varepsilon}^2(p)$ is the mean squared error (MSE) at level *p* and it is computed as:

$$\bar{\varepsilon}^2(p) = 4^{-n}\sum_{i=0}^{2^n-1}\sum_{j=0}^{2^n-1}[B(i,j) - \hat{B}_p(i,j)]^2 \quad (3)$$

- The amount of image compression (measured in bits-per-pixel): it was computed as the ratio of the total number of bits transmitted to the total number of pixels in the original image.

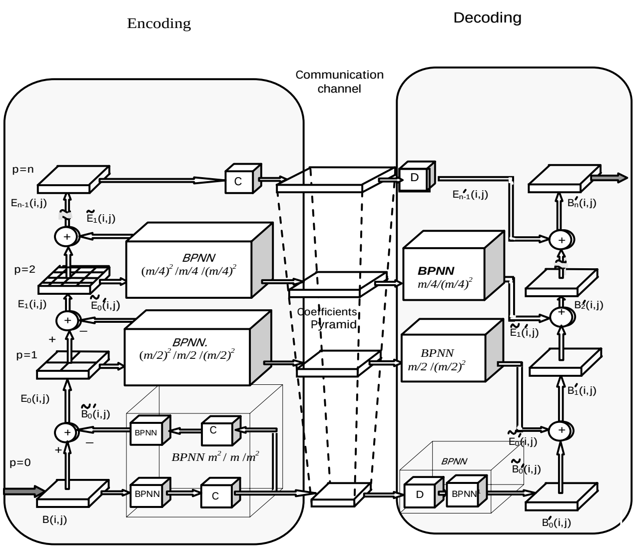
IV. Simulation Results

Simulations have been performed on the original bitmap images "pepper" with size 512x512 pixels and "bird" with size 256x256 pixels. The investigation was done only for lossless compression (i.e, without quantization). Results in terms of PSNR, compression ratio, and number of levels are shown in Table 1. Figs.5 and 6 illustrate the quality results obtained on "pepper" and "bird" images with successive pyramid levels. Results obtained with other test image are very similar to the examined ones.

V. Conclusion

The goal of this study was to develop a new algorithm based on the inverse difference pyramidal decomposition. The novelty lies in modeling each pyramid level using BPNN. This new algorithm can be compared to the most similar pyramids proposed in articles concerning pyramidal decompositions. It can be underlined that, compared to IDP-

DCT decomposition, IDP-BPNN using the gradient descent rule which aims to minimize the mean squared error between the original and the reconstructed image in iterative way reduces the number of levels, required to reconstruct the image at the receiver side. The coding and decoding of the hidden weights values are relatively simple. The training process is time consuming, but the compression ratio and the quality of the reconstructed image are considerable. The future method development will be aimed at the adopting of efficient algorithms for speeding up the NN learning and at the compression ratio increasing with quantization of the hidden-layer coefficients.



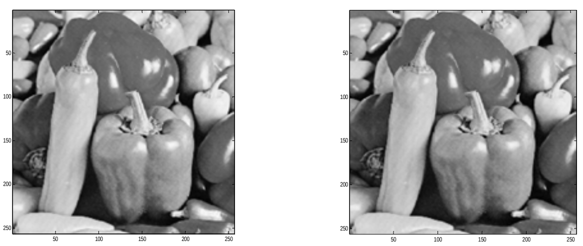
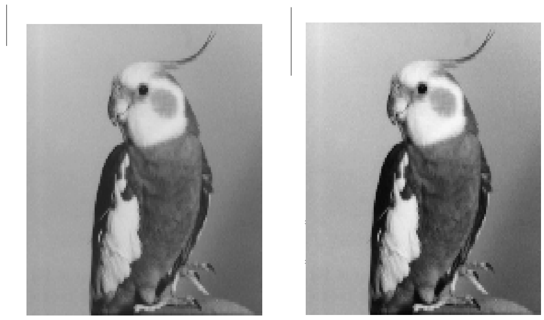


Fig.5. Quality results obtained for the original “bird” image with two-level pyramid

TABLE I. MODELING results for IDP-DCT [11] COMPARED WITH BPNN-IDP, ORIGINAL IMAGE "Pepper"

COMPARISON CRITERIA	IDP-DCT	IDP-BPNN
Level numbers	2	2
Transformation	DCT	BP learning
Subimage size p=0	8x8	16x16
Number of coefficients	4 per block	Square root [block Size] per block
PSNR at last level	31.06 dB	60.13 dB
bpp at last level	0.52	0.125



p=0; PSNR=20.8 dB; 0.016 bpp; p=1; PSNR=24.3 dB; 0.031 bpp

p=2; PSNR=26 dB; 0.125 bpp; p=3; PSNR=27.7 dB; 0.25 bpp

Fig.6. Quality results obtained with the original image “bird” for four-level pyramid

REFERENCES

[1] K. Tzou, "Progressive Image Transmission: A review and comparison of techniques" *Opt. Eng., Vol 26, pp. 581-589 July 1987*.

[2] A. Poularikas. "The Transforms and Applications Handbook". CRC Press 2000.

[3] K. Rao, P. Yip. " The Transform and Data Compression Handbook". CRC Press 2000.

[4] P. Burt. "The Laplacian Pyramid as a Compact Image Code". IEEE Trans. Commun., Vol 31, No. 4, April 1983.

[5] M. Goldberg. "Comparative Performance of Pyramid Data Structures for Progressive Image Transmission" IEEE Trans .Commun. Vol. 39, No.4, April 1991.

[6] M. M. Unser. "An Improved Least Square Laplacian Pyramid for Image Compression" IEEE Signal Processing, 27, pp. 187-203, 1992.

[7] B. Aiazzi, et al. "A Reduced Laplacian Pyramid for Lossless and Progressive Image Communication". IEEE Trans. Commun. Vol. 44, No.1 , 1996.

[8] Tian Hu Yu. "Novel Contrast Pyramid Coding of Images". Proceedings of the 1995, ICIP.

[9] P. Brigger et al. "Centered Pyramids" IEEE Trans - Image Processing, Vol-8 No-9, Sep. 1999.

[10] W. Kim, et al. "Hierarchy Embedded Differential Image for Progressive Transmission using Lossless Compression" IEEE Trans. Circints and Systems for Video Technology. Vol. 5, No.1, Feb. 1995.

[11] R. Kountchev, et al. "Inverse Pyramidal Decomposition with Multiple DCT". Signal Processing. Image Communication, 17 2002, pp. 201-218.

[12] Y. Wu, et al. "Low Bit Rate Subband DCT Image Compression" IEEE Trans. Consumer Election. 43 (2) May. 1997, pp. 134-140.

[13] Y. H. Hu, J. N. Hwang "Handbook of Neural Networks and Signal Processing”, CRC Press 2002.

[14] D. Salomon, "Data Compression: The complete reference" 3rd Ed., Springer-Verlag, New York, 2004.