

Self organizing map and wavelet based image compression

T. Kathirvalavakumar · E. Ponmalar

Received: 26 May 2011 / Accepted: 14 May 2012 / Published online: 2 June 2012
© Springer-Verlag 2012

Abstract The proposed work is hybridizing self organizing map (SOM) and wavelet transform for performing image compression. The novelty in this work is applying discrete wavelet transform (DWT) on the code vector obtained from SOM after vector quantization and storing only the approximation coefficients along with the index values of the SOM. The experimental results of the proposed method show better compression and better peak signal to noise ratio (PSNR) value of the decompressed image.

Keywords Neural network · SOM · Wavelet transform · Approximation coefficients · Decompression

1 Introduction

The explosive growth of data to be stored and/or transmitted in real world applications has been the driving force behind the need of developing algorithms and techniques for data compression in particular image compression [1]. Neural network is an area under research for years together. Neural networks are utilized in wide variety of applications including image compression. Similarly wavelet transform has also formed the basis of numerous applications and image compression in particular.

The use of SOM started from 1990. In 1990, an in depth study of SOM and its variants namely Learning Vector Quantisation 1 (LVQ1), Learning Vector Quantisation 2 (LVQ2) and their practical applications have been performed by Kohonen [2]. Jiang [3] conducted a survey on the effect and impact of various neural networks over image compression namely back propagation neural network based image compression, Hebbian learning based image compression, vector quantization based image compression, wavelet neural network based image compression, fractal neural network based image compression. A family of networks that learn from input namely Kohonen Self Organizing Feature Map (KSOFM), competitive learning, frequency sensitive competitive learning, fuzzy competitive learning and predictive neural networks are used for vector quantization. A review of vector quantization of images for code book design for image compression has been performed by Lu and Shin [4]. They have also proposed a compression technique by first classifying edge block and background block and then the design of separate code-books by KSOFM. Amerijckx et al. [5] have worked on developing a Self Organizing Kohonen Map based image compression over still-images maintaining visual quality. Park and Woo [6] have proposed an edge preserving image compression algorithm based on unsupervised competitive neural network called Weighted Centroid Neural Network. Annadurai and AnnaSaro [7] have proposed a method to reduce the convergence time in Self Organizing Feature Map (SOFM) based image compression. In this, cumulative distribution function is first estimated and used for mapping image pixels which act as input for SOFM. Sharma et al. [8] have proposed a six step SOM algorithm for image compression using Kohonen's Self Organizing Map which is a class of neural networks. Hierarchical SOM technique for efficient and effective code book design has been proposed

T. Kathirvalavakumar (✉)
Department of Computer Science, VHNSN College,
Virudhunagar 626 001, India
e-mail: kathirvalavakumar@yahoo.com

E. Ponmalar
Department of Computer Science, SFR College for women,
Sivakasi 626 123, India
e-mail: ponmalar2006@yahoo.co.in

by Tsai et al. [9] for image compression. Sarlin [10] has focused the use of SOM neural network for monitoring millennium development goals.

A framework for analysing various algorithms for image compression based on wavelet approximations has been presented by DeVore et al. [11]. Karayiannis et al. [12] have combined wavelet image decomposition, vector quantization using LBG (Linde, Buzo and Gray) and fuzzy algorithm to perform image compression. Cho and Pearlman [13] have used progressive resolution coding for fast and efficient image decomposition based on prediction of dynamic ranges of wavelet sub bands. Jain and Jain [14] have performed a detailed study on image compression based on wavelet transform along with evaluation and comparison of seven wavelet families by applying them over test images. Venkateswaran and Rao [15] have used different methodology to achieve image compression. In their work, instead of applying DWT over whole image, sub blocks of the image are subjected to wavelet decomposition and the wavelet coefficients are clustered using K-Means clustering. Designing a vector quantization code book using fuzzy Probabilistic C Means clustering algorithm over wavelet packet tree coefficients has been implemented by Nagendran and Arockia Jansi Rani [16]. Improvement on clustering based on feature weight learning has been proposed by Wang et al. [17] and Yeung and Wang [18]. Image compression should not remove the important features of the image but at the same time unwanted and redundant feature need not be considered for compression. New sensitivity measure has been proposed in a literature to remove the redundant feature of the network [19].

SOM and wavelets combined together have been used for different applications like image segmentation, code book generation and image compression. Nunez and Llacer [20] have implemented image segmentation of astronomical images using SOM and wavelets. While designing a scheme for transmission of fixed images for wireless communication, Chatellier et al. [21] have developed compression module by applying DWT over the image first and then SOM vector quantization is applied to generate code books. Pandian and Anitha [22] have proposed a technique where generic code book is implemented using SOFM, discrete cosine transform (DCT) and DWT. Similar work has been proposed by Dandawate and Londhe [23]. Image compression using set partitioning in hierarchical trees (SPIHT) and SOFM vector quantization has been implemented by Rawat and Meher [24]. Another work of Dandawate et al. [25], has introduced an idea of designing code book for vector quantization based on SOFM and DWT. Huang et al. [26] have made a survey on training approaches for neural networks and extreme learning machines. Upper integral network with extreme learning

mechanism has been proposed for classification system by Wang et al. [27].

The proposed work aims at reducing the size of the image file by hybridizing SOM and wavelet transform. DWT is applied on the code vector which is obtained from SOM neural network after vector quantization and storing only the approximation coefficients along with the index values obtained from SOM. Experimentally, the proposed method has been tested and found that better compression is achieved and retaining the visual quality of the image. The efficiency of the compression achieved by the proposed work is shown by applying over six images.

2 Design of compression and decompression techniques

SOM is a type of artificial neural network consisting of an input layer and Kohonen layer, that is trained using unsupervised learning to produce a two-dimensional, discretized representation of the input space of the training samples, called a map. SOMs operate in two modes namely training and mapping. Training builds the map using input examples and is called as vector quantization. After training, mapping automatically classifies a new input vector.

A self-organizing map consists of components called nodes or neurons. The neurons in the Kohonen layer are arranged in the form of grid. Associated with each node is a weight vector of the same dimension as the input data vectors and a position in the map space. The self-organizing map describes a mapping from a higher dimensional input space (input patterns) to a lower dimensional map space (weight vector or code vector). The input pattern is mapped to the neuron having weight vector closest to the input pattern. Once the closest neuron is located, the index of the neuron is assigned to the input pattern.

In the proposed work after a number of trials, it was decided to have 16×16 neurons in the Kohonen layer of the SOM network with 16 inputs as shown in Fig. 1.

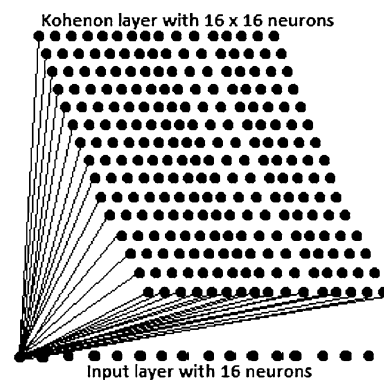


Fig. 1 Structure of SOM neural network

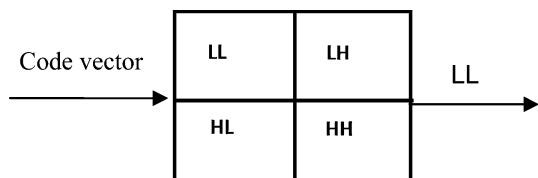


Fig. 2 Single level DWT

DWT when applied over a set of data in the form of a matrix, decomposes it to produce four components namely, LL, LH, HL, HH coefficients and it can be applied to many levels. Figure 2 shows decomposition of single level Haar DWT used in the proposed work.

In the proposed work, during compression, a gray scale image with 256×256 pixels is first divided into 4,096 blocks, each of size 4×4 . Every 4×4 block is converted into 16 element vector. There are 4,096 vectors corresponding to 4,096 blocks. They are given as input to SOM network. SOM network is trained using unsupervised batch weight/bias training with 4096 input patterns. After training the SOM network, the weight matrix of size 16×256 is obtained and 4,096 indexes corresponding to 4,096 input patterns are obtained. The 256 weight vectors called as code vectors and the 4,096 indexes obtained as output from the trained SOM network are processed separately for encoding. Single level Haar DWT is applied over the code vectors and 4,096 indexes corresponding to the input patterns of 4,096 blocks are encoded using arithmetic encoding.

Each code vector is converted into 4×4 block and when DWT is applied over it, LL (approximation coefficients), LH (horizontal coefficients), HL (vertical coefficients) and HH (diagonal coefficients) components are obtained. For each code vector, only LL coefficients of size 2×2 are stored because the detailed coefficients contain less important information and will not improve the visual quality of the image. The approximation coefficients in floating point are quantized and encoded into integers. Arithmetic encoded values of 4,096 indexes are stored along with the encoded approximation coefficients as binary file, which is the

compressed form of the 256×256 input image. Figure 3 shows the steps involved in compression.

During decompression, first the index part and LL part are read from the compressed binary file. Arithmetic decoding is performed over the indexes resulting in 4,096 indexes. LL components of the code vectors are dequantized. LH, HL and HH components are assigned zero as they have not been stored during compression. Inverse DWT is applied over LL, LH, HL and HH components of the 256 code vectors. Mapping the retrieved 4,096 indexes onto the code vectors results in 4,096 output vectors each with 16 elements. Each output vector is converted into 4×4 sub block which forms the decompressed image. The steps involved in the decompression process are shown in Fig. 4.

3 Proposed algorithm for compression and decompression

Algorithm for compression

- Step 1: Split the image into blocks of size 4×4 . Convert each block as vector.
- Step 2: Input the 4,096 vectors to SOM network and train the network. 16×256 weight matrix and 4,096 indexes are obtained as a result of training.
- Step 3: Apply DWT over each 4×4 sub block of the weight matrix (code vector).
- Step 4: Encode the 4,096 indexes using arithmetic encoding.
- Step 5: Store the encoded indexes.
- Step 6: Quantize only the LL components of code vector into integers.
- Step 7: Store the encoded LL components of code vectors.

Algorithm for decompression

- Step 1: Read the 4,096 encoded indexes from the compressed binary file.

Fig. 3 Block diagram showing compression

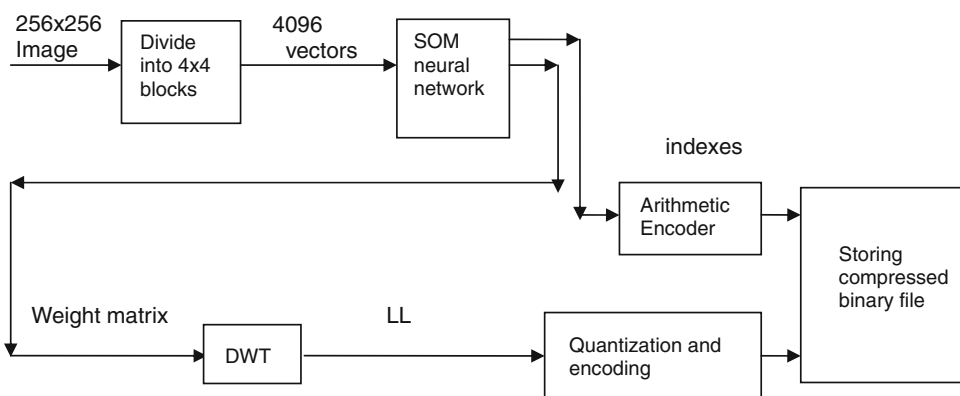
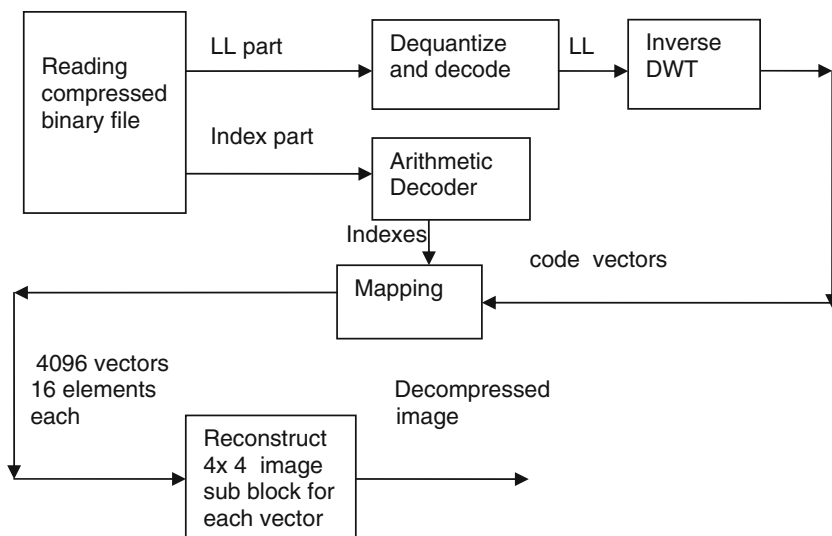


Fig. 4 Block diagram showing decompression



- Step 2: Apply arithmetic decoding over the 4,096 indexes.
- Step 3: Read the LL components of code vectors.
- Step 4: Dequantize the LL components of the code vectors.
- Step 5: Apply inverse DWT over the dequantized LL components by considering LH, HL, HH components as zeros and then reconstruct the 256 code vectors.
- Step 6: Map the 4,096 indexes with the 256 reconstructed code vectors resulting in 4,096 output vectors each with 16 elements.
- Step 7: Convert each 16 element output vector into 4 × 4 block to reconstruct the decomposed image.

4 Experimental results and discussion

The implementation of the proposed work was applied over six images namely lena, pepper, zelda, cameraman, girlface and goldhill. Haar wavelet has been chosen as it has the advantages such as best performance in terms of computation time, memory efficiency. Many strategies were tested such as change in the number of input neurons, change of number of neurons in the Kohenon layer, analysing and deciding the number of detailed coefficients to skip without storing them. It has been decided after many trials that the proposed method could be applied using SOM with 16 input neurons, 16 × 16 neurons in Kohenon layer. All the detailed coefficients are not stored as they contain less important information when DWT is applied over the code vector.

The input image was represented as a two dimensional matrix. After applying SOM and DWT, the 4,096 indexes and LL components of the 256 code vectors were stored as binary file. The size of image lena.bmp is 66,616 bytes.

After compression the size of the binary file becomes 4,869 bytes. For any input image of size 256 × 256, we are storing only the encoded 4,096 indexes and LL components of 256 code vectors. Hence the size of the compressed binary file for all 256 × 256 images will be at the maximum 5 KB. Table 1 shows the file size of the considered original images, equivalent jpeg images and their compressed images.

The size of the binary files obtained using the proposed method is less than the size of the file compressed using jpeg. RMSE and PSNR between the original images and the decompressed images are tabulated in Table 2.

Root mean square error is calculated as $RMSE = \sqrt{\frac{\sum_{i=1}^N (y_i - x_i)^2}{N}}$ where y_i = intensity of the ith pixel of the original image, x_i = intensity of the ith pixel of the decompressed image, N = number of pixels in the image.

Peak signal to noise ratio is computed as $PSNR = 20 \log_{10} \frac{255}{RMSE}$.

Compression ratio is calculated as $CR = \frac{\text{Number of bits in the original image}}{\text{Number of bits in the compressed image}}$.

Table 1 Original file size, jpeg file size and compressed file size

Original image	Original file size in bytes	Size of jpeg file in bytes	Size of compressed binary file in bytes
Lena256.bmp	66,616	9,595	4,869
Pepper256.bmp	66,616	9,966	4,941
Zelda256.bmp	66,616	9,552	5,041
Cameraman256.bmp	66,616	12,012	4,688
Girlface256.bmp	66,616	9,711	4,735
Goldhill256.bmp	66,616	11,764	5,065

Table 2 RMSE and PSNR of decompressed images of the proposed method

Original Image	Decompressed bmp file	RMSE	PSNR
Lena256.bmp	Lenasdrecons.bmp	4.8700	34.3802
Pepper256.bmp	Peppersdrecons.bmp	5.0036	34.1451
Zelda256.bmp	Zeldasdrecons.bmp	5.0036	34.2940
Camerman256.bmp	Cameramansdrecons.bmp	4.9186	32.3967
Girlface256.bmp	Girlfacesdrecons.bmp	5.0545	34.0573
Goldhill256.bmp	Goldhillsdrecons.bmp	6.1545	32.3467

A review over the researches made over 256×256 Lena image for image compression revealed the following results. For Lena image, according to Lu and Shin [4], PSNR value obtained using LBG is 29.40 dB but by the classified vector quantization technique and KSOFM is 30.03 dB. They classified the blocks into background, horizontal edge, vertical edge and diagonal edge and applied KSOFM. Applying pure Self Organizing Kohenon Map, Amerijckx et al. [5] have got PSNR of 24.74 dB at the compression rate 25.22 and at the compression rate 38, PSNR obtained is 24 dB. DCT was applied over image blocks followed by KSOM. According to Weighted Centroid Neural Network (WCNN) of Park and Woo [6], PSNR obtained is 31.04 dB. WCNN was proposed to reduce the edge degradations in reconstructed images. Tsai et al. [9] have achieved a PSNR of 34.016 dB using New Hierarchical SOM by splitting LBG to speed up convergence. Compared with the PSNR obtained by all the above four methods, proposed method gives better PSNR of 34.3802 dB. This is shown in Table 3.

DeVore et al. [11] have stored 20,236 DWT coefficients in 14,604 bytes and 12,068 DWT coefficients with 8,925 bytes but the proposed method requires 4,869 bytes for storing 1,024 coefficients and 4,096 indexes. With hierarchical partition priority wavelet image compression to retain maximum fidelity, Efstratiadis et al. [28] have achieved PSNR of 32 dB at 0.65 bpp (bits per pixel). At compression ratio of 4:1, using SOFM and applying cumulative distribution function (cdf), the PSNR of Lena image according to Annadurai and Saro [7] is 28.91 dB. Laha et al. [29] have achieved PSNR 28.47 dB at 0.218 bpp by applying Restricted Window Search with

L2-SOM (RWS). Applying Linked Significant Tree method (LST), Muzaffar and Choi [30] have made PSNR of 24.58 dB at 0.1 dB, 27.22 dB at 0.2 bpp, 28.96 dB at 0.3 bpp, 30.81 dB at 0.4 bpp, 31.87 dB at 0.5 bpp. They have used LST wavelet image coding algorithm based on SPIHT. Over the same lena image, Chopade and Ghatol [31] have used wavelet based SPIHT and have achieved 32.52 dB at 0.2 bpp. By SOM vector quantization on DWT coefficients with 256 Quadrature Amplitude Modulation (QAM), Chatellier et al. [21] have got PSNR of 31.34 dB. When considering the PSNR values of different techniques at various bpp, it has been observed that the proposed method outperformed them by achieving PSNR of 34.3802 dB at 0.5954 bpp as shown in Table 4.

Sanchez et al. [32]. have obtained PSNR of 18.3135 at 0.5 bpp using adaptive filter whereas proposed method results in 34.3802 dB at 0.5954 bpp. With SPIHT and Embedded Block Coding with Optimized Truncation (EBCOT), Sudhakar et al. [33]. have made it 26.81 dB at compression ratio of 13.03 and 31.28 dB at compression ratio of 6.57. Using curvelet, widgelet and ridgelet Transforms, Joshi et al. [34]. have achieved PSNR of 18.3, 17.57 and 27.1 dB respectively over lena image at compression factor of 7.49. The proposed method outperforms these methods by achieving PSNR of 34.3802 dB at compression ratio of 13.68:1 and 0.5954 bpp as shown in Table 5.

Sonja Grgic et al. [35]. have got PSNR of 32.52 dB at compression Ratio 10:1 whereas in the proposed work PSNR is 34.3802 dB. Pandian and Anitha [22] have used spatial quantization and achieved PSNR 36.16 dB at compression ratio 10.05:1. By designing a Neuro-wavelet based codebook with vector quantizer using SOM and

Table 3 Comparison over existing techniques based on PSNR

Method	Work	PSNR
LBG	Lu and Shin (LBG)	29.400
VQ	Lu and Shin (VQ II)	30.030
Pure SOM	Amerijckx	24.740
Weighted CNN	Park and Woo	31.040
New Hierarchical SOM	Tsai et al. [9]	34.016
SOM and DWT	Proposed method	34.3802

Table 4 Comparison over existing techniques based on PSNR and bpp

Method	Work	PSNR	bpp
Hierarchical partition	Efstratiadis	32.00	0.6500
SOFM and cdf	Annadurai and Saro	28.91	2.0000
RWS with L2-SOM	Laha	28.47	0.2186
Linked significant tree	Muzaffar and Choi	31.87	0.5000
SPIHT	Chopade and Ghatol	32.52	0.2000

neural network, Dandawate et al. [25] have got PSNR 35.89 dB and the compressed file size 15.2258 KB. But in the proposed method, the compressed file size is only 4.75 KB and PSNR achieved is 34.3802 dB at compression ratio 13.68:1. It is noted that better compression is achieved by the proposed work. Table 6 shows the bit rate achieved by the proposed work. The result reveals that the implementation of the proposed work gives good compression rate with acceptable loss in image quality. Table 7 shows the entropy of the decompressed images after decompression and Table 8 gives the compression ratio obtained.

Figure 5 shows the original Lena image and the decompressed Lena image of the proposed method. It has been observed from the decompressed image that the visual

quality of the decompressed image is acceptable because PSNR value is greater than 30 [36]. Similarly the work has been applied over the other five images and it has been observed that the proposed work outperformed well on each image.

5 Conclusions

Proposed a method for compressing images using a hybrid of SOM neural network and wavelet transform and implemented successfully over 256×256 gray scale images. Novelty of this hybrid work is applying DWT on the code vector obtained from SOM after vector

Table 5 Comparison over existing techniques based on PSNR and CR

Work	PSNR	CR
Sanchez	18.3135	16.00
Sudhakar I	26.8100	13.03
Sudhakar II	31.2800	6.57
Joshi	27.1000	7.49
Sonja Grgic	32.5200	10.00
Proposed method	34.3802	13.68

Table 6 Bit rate of compressed images

Original image	Compressed binary file	Bits per pixel
Lena256.bmp	Lenainterenco.bin	0.5954
Pepper256.bmp	Pepperinterenco.bin	0.6031
Zelda256.bmp	Zeldainterenco.bin	0.6154
Cameraman256.bmp	Cameramaninterenco.bin	0.5723
Girlface256.bmp	Girlfaceinterenco.bin	0.5780
Goldhill256.bmp	Goldhillinterenco.bin	0.6183

Table 7 Entropy of decompressed images

Original image	Decompressed file	Entropy
Lena256.bmp	Lenasdrecons.bmp	6.1240
Pepper256.bmp	Peppersdrecons.bmp	5.0036
Zelda256.bmp	Zeldasdrecons.bmp	6.1364
Cameraman256.bmp	Cameramansdrecons.bmp	5.5044
Girlface256.bmp	Girlfacesdrecons.bmp	6.2662
Goldhill256.bmp	Goldhillsdrecons.bmp	6.3287

Table 8 Compression ratio

Original image	Compressed binary file	CR
Lena256.bmp	Lenainterenco.bin	13.68
Pepper256.bmp	Pepperinterenco.bin	13.48
Zelda256.bmp	Zeldainterenco.bin	13.21
Cameraman256.bmp	Cameramaninterenco.bin	14.21
Girlface256.bmp	Girlfaceinterenco.bin	14.07
Goldhill256.bmp	Goldhillinterenco.bin	13.15

Fig. 5 Original image lena256.bmp and decompressed lena image lenasdrecons.bmp



ORIGINAL IMAGE lena256.bmp

DECOMPRESSED IMAGE lenasdrecons.bmp

quantization and storing only the approximation coefficients. The proposed method was tested over images of different sizes and observed that compression and decompression are performed well. Considerable reduction in the file size is achieved. Both Vector Quantization using SOM and wavelet transform are lossy compression techniques. The size of the image file is reduced to a considerable extent and the decompressed images are visually acceptable. Since the compressed binary file contains the indexes and LL components of the code vectors, the size of the file is reduced and reconstruction is faster when compared with compression. The experimental results obtained from this technique reveals that this hybrid method leads to improved performance measures namely Peak signal to noise ratio and bits per pixel over existing techniques. The PSNR values are greater than 30 showing that the decompressed images are acceptable [36].

References

- Khalid Sayood (2006) Introduction to data compression, 3rd edn. Elsevier, San Francisco
- Kohonen T (1990) The Self organizing map. In: Proceedings of the IEEE, vol 78(9), pp 1464–1480
- Jiang J (1999) Image compression with neural networks—a survey. *Signal Process Image Commun* 14:737–760
- Lu C, Yong Ho Shin (1992) A neural network based image compression system. *Contributed paper in IEEE*, pp 25–29
- Amerijckx C, Verleysen Michel, Thissen Philippe, Legat Jean-Didier (1998) Image compression by self organized kohonen map. *IEEE Trans Neural Networks* 9(3):503–507
- Park D, Woo Young-June (2001) Weighted centroid neural network for edge preserving image compression. *IEEE Trans neural networks* 12(5):1134–1146
- AnnaDurai S, AnnaSaro E (2006) An improved image compression approach with self organizing feature maps using cumulative distribution function. *ICGST GVIP J* 6(2):41–48
- Sharma DK, Loveleen Gaur, Daniel Okunbor (2007) Image compression and feature extraction with neural network. In: Allied academies international conference, proceedings of the academy of information and management sciences, vol 11(1), pp 33–37
- Tsai C, Chen-An Jhuang, Chih-Wei Liu (2008) Gray image compression using new hierarchical self-organizing map technique. In: 3rd international conference innovative computing information and control
- Sarlin P (2011) Visual tracking of the millennium development goals with a fuzzified self-organizing neural network. *Int J Mach Learn Cybern*. doi:10.1007/s13042-011-0057-S
- DeVore RA, Jawerth Bjorne, Lucier Bradley J (1992) Image compression through wavelet transform coding. *IEEE Trans Inform Theory* 8(2):719–746
- Karayiannis NB, Pin-I.Pai, Nicholas Zervos (1998) Image compression based on fuzzy algorithm for learning vector quantization and wavelet image decomposition. *IEEE Trans on image processing* vol 7(8), pp 1223–1230
- Cho Y, Pearlman William A (2007) Hierarchical dynamic range coding of wavelet subbands for fast and efficient image decomposition. *IEEE Trans on image processing* 16(8):2005–2015
- Jain YK, Jain Sanjeev (2006) Performance evaluation of wavelets for image compression. *Asian J Inf Technol* 5(10):1104–1112
- Venkateswaran N, Ramana Rao YV (2007) K-Means clustering based image compression in wavelet domain. *Inf Technol J* 6(1):148–153
- Nagendran R, Arockia Jansi Rani P (2010) A Novel FPCM based vector quantizer codebook design for image compression in the wavelet packet domain. *Int J Multiméd Appl* 2(1):1–7
- Wang Xizhao, Wang Yadong, Wang Lijuan (2004) Improving fuzzy c-means clustering based on feature-weight learning. *Pattern Recogn Lett* 25(10):1123–1132
- Yeung Daniel S, Wang Xizhao (2002) Improving performance of similarity-based clustering by feature weight learning. *IEEE Trans Pattern Anal Mach Intell* 24(4):556–561
- Wang Xi-Zhao, Li Chun-Guo (2008) A definition of partial derivative of random functions and its application to rbfn sensitivity analysis. *Neurocomputing* 71(7–9):1515–1526
- Nunez J, Jorge Llacer (2003) Astronomical image segmentation by self-organizing neural networks and wavelets. Elsevier, *Neural Networks 16: special Issue*
- Chatellier C, Boeglen H, Perrine C, Olivier C, Haerberle O (2007) A robust joint source channel coding scheme for image transmission over the ionospheric channel. *Signal Process Image Commun* 22:543–556
- Immanuel Pandian S, Alex Anitha J (2009) A neural network approach for color image compression in transform domain. *Int J Recent Trends Eng* 2(2):152–154
- Dandawate YH, Shreenivas N. Londhe (2009) Image compression using generic vector quantizer designed using transform coding: The quality analysis perspective. In: Second international conference on computer and electrical engineering, IEEE
- Rawat CD, Meher Sukadev (2009) A hybrid coding scheme combining SPIHT and SOFM based vector quantization for effectual image compression. *European J Scientific Res* 38(3): 425–440
- Dandawate YH, Jadhav TR, Chitra AV, Joshi MA (2009) Neuro-Wavelet based vector quantizer design for image compression. *Indian J Sci Technol* 2(10):56–61
- Guang-Bin Huang, Wang Dian Hui, Lan Yuan (2011) Extreme learning machines: a survey. *Int J Mach Learn Cybern* 2:107–122. doi:10.1007/s13042-011-0019-y
- Wang Xizhao, Chen Aixia, Feng Huimin (2011) Upper integral network with extreme learning mechanism. *Neurocomputing* 74(16):2520–2525
- Efstratiadis SN, Tzovaras Dimitrios, Strintzis Michael G (1996) Hierarchical partition priority wavelet image compression. *IEEE Trans Image Process* 5(7):1111–1123
- Laha A, Chanda Bhabatosh, Pal Nikhil R (2008) Fast codebook searching in a SOM based vector quantizer for image compression. *SIVIP* 2:39–49
- Muzaffar T, Choi TS (2008) Linked significant tree wavelet-based image compression. *Signal processing* 88:2554–2563
- Chopade NB, Ghatol AA (2009) Analysis of discrete wavelet based image compression—a review. *J Sci Ind Res* 68:915–919
- Sanchez LEU, Hector M. Perez Meana, Mariko Nakana Miyatake (2004) Wavelet image compression using universal coefficients matrix detail estimation. In: Proceedings 14th international conference on electronics, communications and computers, IEEE, pp 277–282
- Sudhakar R, Karthiga R, Jayaraman S (2007) Image compression using coding of wavelet coefficients—a survey. *GVIP Special Issue on Image Compression, ICGST* 7:1–13
- Joshi MS, Manthalkar RR, Joshi YV (2008) Image compression using curvelet, ridgelet and wavelet transform, a comparative study. *ICGST-GVIP, ISSN 1687-398X*, 8(III):25–34

35. Grgic Sonja, Grgic Mislav, Zovko-Cihlar Branka (2001) Performance analysis of image compression using wavelets. *IEEE Trans Ind Electron* 48(3):682–695
36. Vilovic I (2006) An experience in image compression using neural networks. In: the 48th international symposium ELMAR, Zadar, Croatia, pp 95–98