

## Image Compression Based on Clustering Fuzzy Neural Network

Shahba I. Khaleel

Jamal S. Majeed

Bayda I. Khaleel

College of Computer Sciences and Mathematics

University of Mosul, Iraq

Received on: 27/08/2007

Accepted on: 26/09/2007

### ABSTRACT

The problem inherent to any digital image is the large amount of bandwidth required for transmission or storage. This has driven the research area of image compression to develop algorithm that compress images to lower data rates with better quality.

This research present, a new approach to image compression based on clustering. This new approach includes new objective function, and its minimization by energy function based on unsupervised two dimensional fuzzy Hopfield neural network. New objective function consists of a combination of classification entropy function and average distance between image pixels and cluster centers. After applying new method on gray scale sample images at different number of clusters, better compression ratio and signal to noise ratio was observed. The new method is also a new clustering analysis method, and it provides more compact and separate clustering.

**Keywords:** image compression, clustering, unsupervised fuzzy Hopfield neural network, classification entropy function.

كبس الصورة بالاعتماد على الشبكة العصبية المضطبة المعنقدة

بيداء إبراهيم خليل

جمال سيد مجيد

شهبا إبراهيم خليل

كلية علوم الحاسوب والرياضيات، جامعة الموصل

تاريخ القبول: 2007/09/26

تاريخ الاستلام: 2007/08/27

### المخلص

إن المشاكل والمعوقات التي ترافق أية صورة رقمية تتطلب عرض حزمة كبيرة لنقلها من مكان إلى آخر وكذلك تحتاج إلى مساحة خزنية كبيرة. هذه المعوقات قادت إلى البحث عن تحسينات لخوارزميات الكبس لتقليل نسبة البيانات المبعوثة وبأفضل نوعية أي من دون تأثير في البيانات الحقيقية للصورة.

في هذا البحث تم تقديم طريقة جديدة لكبس الصورة بالاعتماد على العنقدة. طريقة الكبس الجديدة تتضمن دالة هدف جديدة التي تقل قيمتها بواسطة دالة الطاقة المعتمدة على شبكة الهوبفيلد العصبية الاصطناعية المضطبة ثنائية الأبعاد ذات التدريب دون إشراف. تتكون دالة الهدف الجديدة من ربط دالة الانتروبي التصنيفية ومعدل المسافة بين نقاط الصورة ومراكز العنقدة. تم تطبيق

الطريقة الجديدة على نماذج صور ذات تدرج رمادي وبإعداد متنوعة من مراكز العنقدة وتم الحصول على أفضل نسبة كبس. وهذه الطريقة الجديدة تعتبر أيضا طريقة عنقدة جديدة قوية لنقاط الصورة. **الكلمات المفتاحية:** كبس الصورة، العنقدة، شبكة الهوبفيلد العصبية الاصطناعية المضبية، دالة الانترنتي التصنيفية.

## **1. Introduction**

Image compression has been pushed to the forefront of the image processing field. This is largely a result of the rapid growth in computer power, the corresponding growth in the multimedia market, and the advent of the World Wide Web, which makes the Internet easily accessible for everyone. Additionally, the advances in video technology, including high-definition television, are creating a demand for new, better, and faster image compression algorithms. The storage and transmission of such data require large capacity and bandwidth which could be very expensive. Image data compression techniques are concerned with reduction of the redundancies in data representation in order to decrease data storage requirements and hence communication costs. Reducing the storage requirements is equivalent to increasing the capacity of the storage medium and hence communication bandwidth. Thus the development of efficient compression techniques will continue to be a design challenge for future communication systems and advanced multimedia applications[1,2].

Clustering is a useful approach in several exploratory pattern-analysis, grouping, and machine-learning situations, including data mining, document retrieval, image segmentation, and pattern classification[3,4].

In image segmentation coding techniques, image is segmented to different regions separated with contours, and coded with different coding techniques. Region growing, k-means, c-means, and split and merge methods are used generally for image segmentation. Beside of this, crisp classical segmentation methods, the fuzzy logic segmentation methods were also seen very effective for coding [5,6].

The Hopfield neural network is well-known technique used for solving optimization problems based on energy function[7]. In this study, a new image clustering and compression method based on fuzzy Hopfield neural network was introduced for gray scale images. This new approach includes new objective function, and its minimization by energy function based on unsupervised two dimensional fuzzy Hopfield neural network. After applying new method on gray scale sample images at different number of clusters, better compression ratio was observed.

## **2. Classification of compression algorithms**

In an abstract sense, we can describe data compression as a method that takes an input data  $D$  and generates a shorter representation of the data

$c(D)$  with a fewer number of bits compared to that of  $D$ . the reverse process is called decompression, which takes the compressed data  $c(D)$  and generates or reconstructs the data  $D'$  as shown in the figure 1. Sometimes the compression (coding) and decompression (decoding) systems together are called a CODEC.

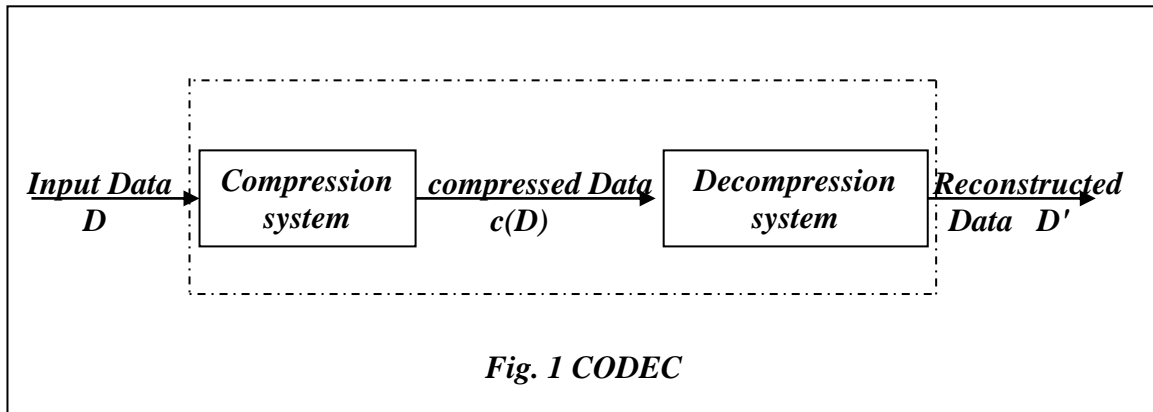


Fig. 1 CODEC

The reconstructed data  $D'$  could be identical to the original data  $D$  or it could be an approximation of the original data  $D$ , depending on the reconstruction requirements. If the reconstructed data  $D'$  is an exact replica of the original data  $D$ , we call the algorithm applied to compress  $D$  and decompress  $c(D)$  to be lossless. On the other hand, we say the algorithms are lossy when  $D'$  is not an exact replica of  $D$ . hence as far as the reversibility of the original data is concerned, the data compression algorithms can be broadly classified into two categories, lossless and lossy, we will focus our discussions on lossless coding[2,8].

### 2.1 Coding (Compression) Method

The neighboring pixels in a typical image are highly correlated to each other. Often it is observed that the consecutive pixels in a smooth region of an image are identical or the variation among the neighboring pixels is very small.

Run length coding(RLC) is an image compression method that works by counting the number of adjacent pixels with the same gray level value. This count, called the run length, is then coded and stored.

Run length coding is a simple approach to source coding when there exists a long run of the same data, in a consecutive manner, in a data set. As an example, the data set that represents by the matrix  $d = [ 5 5 5 5 5 5 5 19 19 19 19 19 19 19 19 19 19 19 19 0 0 0 0 0 0 0 8 23 23 23 23 23 23 ]$  contains long runs of 5's, 19's, 0's, 23's,etc. rather than coding each sample in the run individually, the data can be represented compactly by simply

indicating the value of the sample and the length of its run when it appears. In this manner, the data  $d$  can be run length encoded as (5 7) (19 12) (0 8) (8 1) (23 6). For ease of understanding, we have shown a pair in each parentheses. Here the first value represents the pixel, while the second indicates the length of its run[1,2].

In some cases, the appearance of runs of symbols may not be very apparent. But the data can possibly be processed in order to aid run length coding. Here we apply classical clustering algorithm, fuzzy clustering algorithm and using fuzzy with neural network on the gray level image. Then, after obtaining the cluster centroids the clustering image is created and it is coded by run length coding algorithm in one and two dimensions. When applying two dimensions of run length coding, we use zig-zag ordering of coefficients of the clustered image as shown in the following figure.

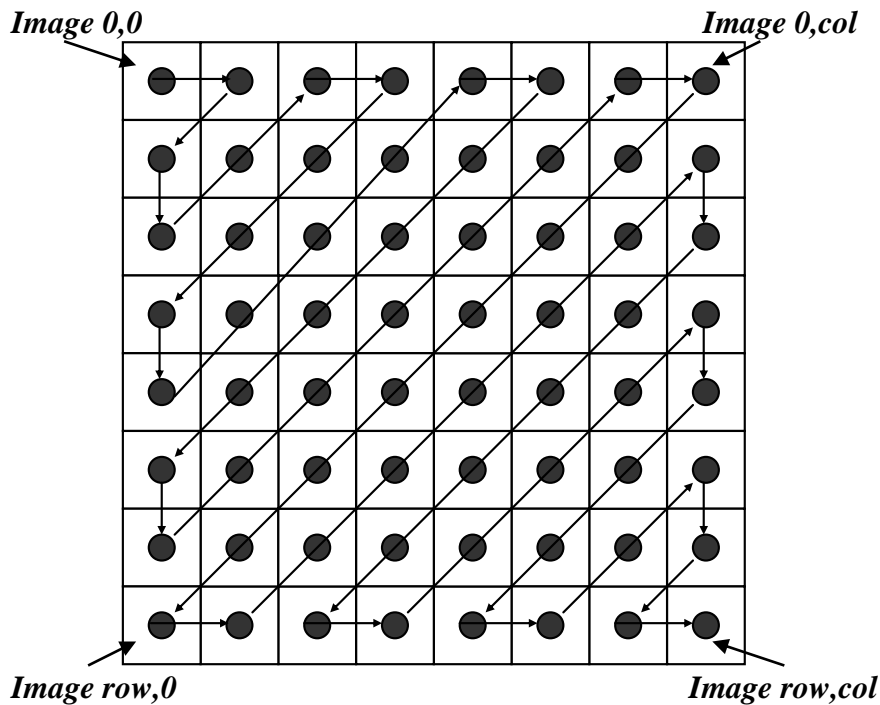


Fig. 2 Zig-Zag Ordering of Pixels of Image

## 2.2 Fidelity Criteria

In some image transmission systems some errors in the reconstructed image can be tolerated. In this case a fidelity criterion can be used as a measure of system quality[16]. After completing the decoding process, the root mean square error  $e_{RMS}$ , the root mean square signal to noise ratio  $SNR_{RMS}$ , and the peak signal to noise ratio  $SNR_{PEAK}$ , should be calculated

between the reconstructed image and the original image to verify the quality of the decode image with respect to the original one.

We can define the error between an original, uncompressed pixel value and the reconstructed (decompressed) pixel value by the following equation :

$$error(r,c) = \hat{I}(r,c) - I(r,c)$$

where:  $I(r,c)$  = the original image

$$\hat{I}(r,c) = \text{the decompressed image}$$

next we can define the total error in an  $N * N$  decompressed image as:

$$Total\ error = \sum_{r=0}^{N-1} \sum_{c=0}^{N-1} [\hat{I}(r,c) - I(r,c)]$$

The root mean square error is found by taking the square root of the error squared divided by the total number of pixels in the image.

$$e_{RMS} = \sqrt{\frac{1}{N^2} \sum_{r=0}^{N-1} \sum_{c=0}^{N-1} [\hat{I}(r,c) - I(r,c)]^2}$$

The smaller the value of the error metrics, the better the compressed image represents the original image. Alternately, with the signal to noise (SNR) metrics, a larger number implies a better image. The SNR metrics consider the decompressed image ( $\hat{I}(r,c)$ ) to be the 'signal' and the error to be 'noise'. We can define the root mean square signal to noise ratio as :

$$SNR_{RMS} = \sqrt{\frac{\sum_{r=0}^{N-1} \sum_{c=0}^{N-1} [\hat{I}(r,c)]^2}{\sum_{r=0}^{N-1} \sum_{c=0}^{N-1} [\hat{I}(r,c) - I(r,c)]^2}}$$

another related metric, the peak signal to noise ratio, is defined as:

$$SNR_{PEAK} = 10 \log_{10} \frac{(L-1)^2}{\frac{1}{N^2} \sum_{r=0}^{N-1} \sum_{c=0}^{N-1} [\hat{I}(r,c) - I(r,c)]^2}$$

where:  $L$  = the number of gray levels  
(e.g. , for 8 bits  $L=256$ )

To check the compression performance, the values (CR) compression ratio and Bpp rate (bit per pixel rate ) are calculated. The compression ratio is the amount of compression, while the Bpp rate is the number of bits required to represent each pixel value of the compressed image. The compression ratio is defined by:

$$compressionratio(CR) = \frac{UncompressedFileSize}{CompressedFileSize}$$

The bits per pixel for N\*N image is :

$$\text{BitsPerPixel} = \frac{\text{NumberofBits}}{\text{NumberofPixels}} = \frac{(8)(\text{Numberofbytes})}{N * N}$$

### 3. K-means Clustering Algorithm

The Standard K-means clustering algorithm is well known and understood algorithm. Its computational complexity is O(n), where n is the number of data points (feature vectors) to be clustered. The k-means is one of a group of algorithms which aim to minimize an objective function[2,9].

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers. The k-means algorithm can be run multiple times to lessen this affect[10,11].

The k-means is a simple algorithm that has been adapted to many problem domains. It is a good candidate for extension to work with fuzzy feature vectors.

### 4. Fuzzy Approach to Pixel Classification

The basic postulates of fuzzy clustering is that a member may have partial memberships grades in several fuzzy clusters. A membership value in the interval[0,1] is assigned to each sample in every cluster, based on certain measurements[2,10].

In fuzzy clustering a pattern is assigned with a degree of belongingness to each cluster in a partition. Here we will present the most popular fuzzy clustering algorithm, known as fuzzy c-means algorithm[12,13].

#### 4.1 Fuzzy C-Means

Fuzzy c-means is a commonly used clustering approach. It is a natural generalization of the k-means algorithm allowing for soft segmentation based on fuzzy set theory. As in hard k-means algorithm, fuzzy c-means algorithm is based on the minimization of a criterion function.

The following criterion function may be chosen. Which differs from the k-means objective function by the addition of the membership values  $u_{ik}$  and the fuzzifier m that shown in the following equation.

$$J(u, v) = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^m |x_k - v_i|^2$$

where:

- $x_1, x_2, \dots, x_n$  are the  $n$  sample points.
- $V = \{v_1, v_2, \dots, v_c\}$  are the cluster centers.
- $U = \{u_{ik}\}$  is  $c \times n$  matrix, where  $u_{ik}$  is the membership value of the  $k^{\text{th}}$  input sample  $x_k$  in the  $i^{\text{th}}$  cluster. The membership values satisfy the following conditions:

$$\left. \begin{aligned} 0 \leq u_{ik} \leq 1 \\ \sum_{i=1}^c u_{ik} = 1 \\ 0 < \sum_{k=1}^n x_k < n \end{aligned} \right\}$$

- $m \in [1, 0]$  is an exponent weight factor . There is no fixed rule for choosing the exponent weight factor. However, in many applications  $m=2$  is a common choice.

The above three conditions imply the followings:

- The membership values of each sample  $x_k$  to a particular cluster should lie between 0 and 1.
- Each sample  $x_k$  must belong to at least one cluster.
- Each class must have at least one sample and all the sample cannot belong to a particular class.

The objective function in this case is the sum of the squared Euclidean distances between each input sample and its corresponding cluster center, weighted by the fuzzy membership values.

The algorithm iteratively updates the cluster centers using the expression:

$$v_i = \frac{\sum_{k=1}^n u_{ik}^m x_k}{\sum_{k=1}^n u_{ik}^m}, \quad i = 1, 2, 3, \dots, c \quad (1)$$

The fuzzy membership function of the  $k^{\text{th}}$  sample  $x_k$  to the  $i^{\text{th}}$  cluster is given by the following:

$$u_{ik} = \frac{\left[ \frac{1}{(x_k - v_i)^2} \right]^{\frac{1}{m-1}}}{\sum_{j=1}^c \left[ \frac{1}{(x_k - v_j)^2} \right]^{\frac{1}{m-1}}}, \quad i = 1, 2, \dots, c; k = 1, 2, \dots, n \quad (2)$$

It can be noted that the weight factor  $m$ , reduces the influence of small membership values.

The fuzzy  $c$ -means algorithm is thus summarized as follows:

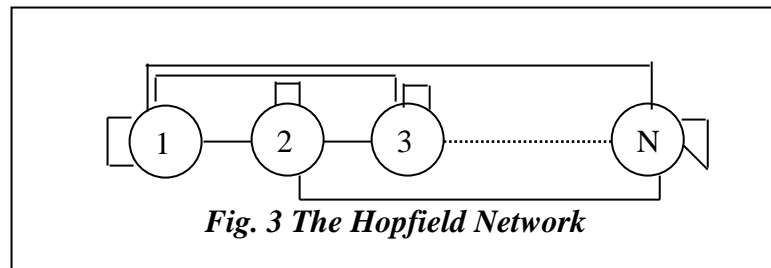
- Step1:** Initialize  $U^{(0)}$  randomly or based on some approximation. Initialize  $V^{(0)}$  and calculate  $U^{(0)}$ . Set the iteration counter  $t=1$ . Select the number of cluster centers  $c$  and choose the value of  $m$ .
- Step2:** Compute the cluster centers. Given  $U^{(t)}$ , calculate  $V^{(t)}$  according to the Eq. 1.
- Step3:** Given  $V^{(t)}$  update the membership values to  $U^{(t+1)}$  according to Eq. 2.
- Step4:** Stop iteration if  $|u_{ik}^{(t+1)} - u_{ik}^{(t)}| \leq \varepsilon$ , where  $\varepsilon$  is a small positive number.
- Step5:** Increment iteration counter to  $t=t+1$  Go to step2.

It may be noted that, we are applying the fuzzy c-means to image the data points or the sample points  $x_1, x_2, \dots, x_n$  are the pixel gray values. Thus  $n$  represents the total number of pixels in the image [2,10,12].

### 5. Hopfield Neural Network

Artificial neural networks are mimicking the neurophysiology of the human brain. They have an ability to learn from examples in order to find patterns in data or classify data. Once trained on training data they have predicting ability on new data. They perform global search on data, however their shortcoming is that they represent the kind of black box, where a user can hardly understand the underlying principles that are used to classify the data. On the other hand, it could perform well on recognizing images and similar tasks. The most common method used for learning of neural network is Hopfield[14].

The Hopfield neural network can be used as a content addressable memory. Knowledge and information can be stored in single layered interconnected neurons (nodes) and weighted synapses(links) (as shown in the following figure), and can be retrieved based on the network's parallel relaxation method, nodes are activated in parallel and are traversed until the network reaches a stable state (convergence). It had been used for various classification tasks and global optimization[15]. In this study, a new image clustering and compression method based on fuzzy Hopfield neural network was introduced for gray scale images.



**Fig. 3 The Hopfield Network**



## 6. The Proposed Method

The Hopfield neural network is a well-known technique used for solving optimization problems based on the energy function. In this method, two dimensional Hopfield neural network consists of  $N \times c$  neurons which are fully interconnected neurons. The total weighted input for neuron  $(x,i)$  is given as:

$$Net_{x,i} = \sum_{y=1}^N \sum_{j=1}^c W_{x,i;y,j} V_{y,j} + I_{x,i}$$

where  $N$  is the number of data points,  $c$  is number of cluster,  $V_{y,j}$  denotes the binary state of neuron  $(y,j)$ ,  $W_{x,i;y,j}$  is interconnection weight between neuron  $(x,i)$  and neuron  $(y,j)$ ,  $I_{x,i}$  is external bias vector for neuron  $(x,i)$ . energy function of two dimensional Hopfield neural network is also given as:

$$E = -\frac{1}{2} \sum_{x=1}^N \sum_{y=1}^N \sum_{i=1}^c \sum_{j=1}^c V_{x,i} W_{x,i;y,j} V_{y,j} - \sum_{x=1}^N \sum_{i=1}^c I_{x,i} V_{x,i}$$

The neural network reaches a stable state, when the energy function is minimized. The optimization problem can be mapped into a two dimensional fully interconnected Hopfield neural network with the fuzzy reasoning strategy. Instead of using the competitive learning strategy, the fuzzy Hopfield neural network uses the fuzzy reasoning algorithm to eliminate the need for finding weighting factors in the energy function.

Iterative minimization of new objective function consists of the following steps:

**Step1:** Choose number of cluster  $c$ , iteration criteria  $\varepsilon$ , fuzzification parameter  $m$  chosen to be 2, and primary centroids  $v^0$ .

**Step2:** Compute initial membership values:

$$u_{x,i} = \frac{\left[ \frac{1}{\|z_x - v_i^0\|^2} \right]^{\frac{1}{m-1}}}{\sum_{k=1}^c \left[ \frac{1}{\|z_x - v_k^0\|^2} \right]^{\frac{1}{m-1}}}, \quad x = 1, 2, \dots, N; \quad i = 1, 2, \dots, c$$

Where  $m$  is the fuzzification parameter and membership value  $u_{x,i}$  is the output state at neuron  $(x,i)$ ,  $z_x$  is  $x$  pixel value of image. A neuron  $(x,i)$  in a maximum membership state indicates that  $z_x$  pixel belongs to class  $i$ . The summation of membership values of each pixel in different classes

equals 1 and total membership values for N image pixel equal N. v represents the cluster center.

**Step3:** Compute new membership values according to the following equation:

$$u_{x,i} = \left[ \sum_{j=1}^c \left( \frac{Net_{x,i}}{Net_{x,j}} \right)^{\frac{1}{m-1}} \right]^{-1} \quad \text{for all x and i}$$

this membership function is effective to minimize new objective function in iteration.

When the input to neuron (x,i) can be expressed as:

$$Net_{x,i} = \left| z_x - \sum_{y=1}^N \frac{1}{\sum_{h=1}^N u_{h,i}^m} z_y u_{y,i}^m \right|^2$$

**Step4:** If  $u_{x,i}$  membership values is out of  $0 \leq u_{x,i} \leq 1$  range, new  $u_{x,i}$  values are given as:

$$u_{x,i} = \frac{|u_{x,i}|}{\max_{x,i} |u_{x,i}|} \quad \text{for all x and i}$$

**step5:** Compute new cluster centroids:

$$V_i = \frac{\sum_{x=1}^N u_{x,i}^m z_x}{\sum_{x=1}^N u_{x,i}^m}$$

**Step6:** Compute new objective function  $J^k$ :

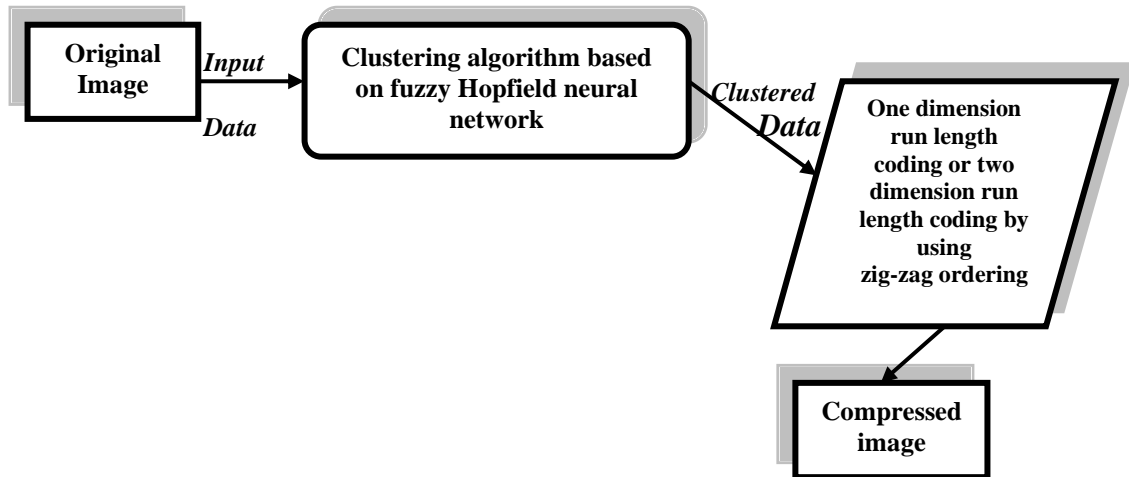
$$J^k = \frac{1}{N} \sum_{x=1}^N \sum_{i=1}^c u_{x,i}^m d(z_x, v_i) - \frac{1}{N} \sum_{x=1}^N \sum_{i=1}^c u_{x,i} \ln u_{x,i}$$

New objective function consists of equal weighted combination of classification entropy function and average distance between image pixels and cluster centroids for separate and compact clustering.

**Step7:** If  $|J^{k+1} - J^k| \leq \varepsilon$  stop, otherwise go to step3.

**Step8:** Segmented or clustering image is created when obtaining the membership values and the cluster centroids, after this it is coded by run length coding in one or two dimensions.

The following block diagram explain the flow of the new image clustering and compression method based on fuzzy Hopfield neural network.



## 7. Results and Conclusions

This new method fuzzy hopfield neural network was applied to 256\*256 dimensional four sample grayscale images and compared with results of k-means and fuzzy c-means algorithms and also compared with original run length coding with one and two dimensions. Comparing parameters is a signal to noise ratio, compression ratio and bits per pixel. Comparison results are given at table 1. Original images and reconstructed image by k-means, fuzzy c-means and fuzzy hopfield neural network method were also given in figure 5,6,7 and 8 corresponding to different sample of images fuzzy hopfield neural network method provides better image compression than other methods according to results.

Importance of image clustering and compression methods increases nowadays. A new image clustering and compression method based on fuzzy hopfield neural network provides better compression ratio. This method can be used for pattern recognition additionally, because it provides good validity measure. There isn't possibility to reach incorrect results. But some of methods as k-means has high possibility to go to a local minimum according to selection of initial values and may not give correct results. The fuzzy hopfield neural network can also provide a more efficient mechanism. This new method is a good alternative for image clustering and compression.

**Table (1) Comparison Results**

<i>Image1</i>	<i>K_means</i>	<i>Fuzzy C_means</i>	<i>Fuzzy Hopfield Neural Network</i>
<i>SNR_RMS</i>	5.476125	6.567393	20.325558
<i>SNR_PEAK</i>	19.324581	21.343241	31.228843
<i>Compression ratio</i>	1.739	4.72	14.56
<i>Bpp</i>	4.5999	1.6924	0.5493

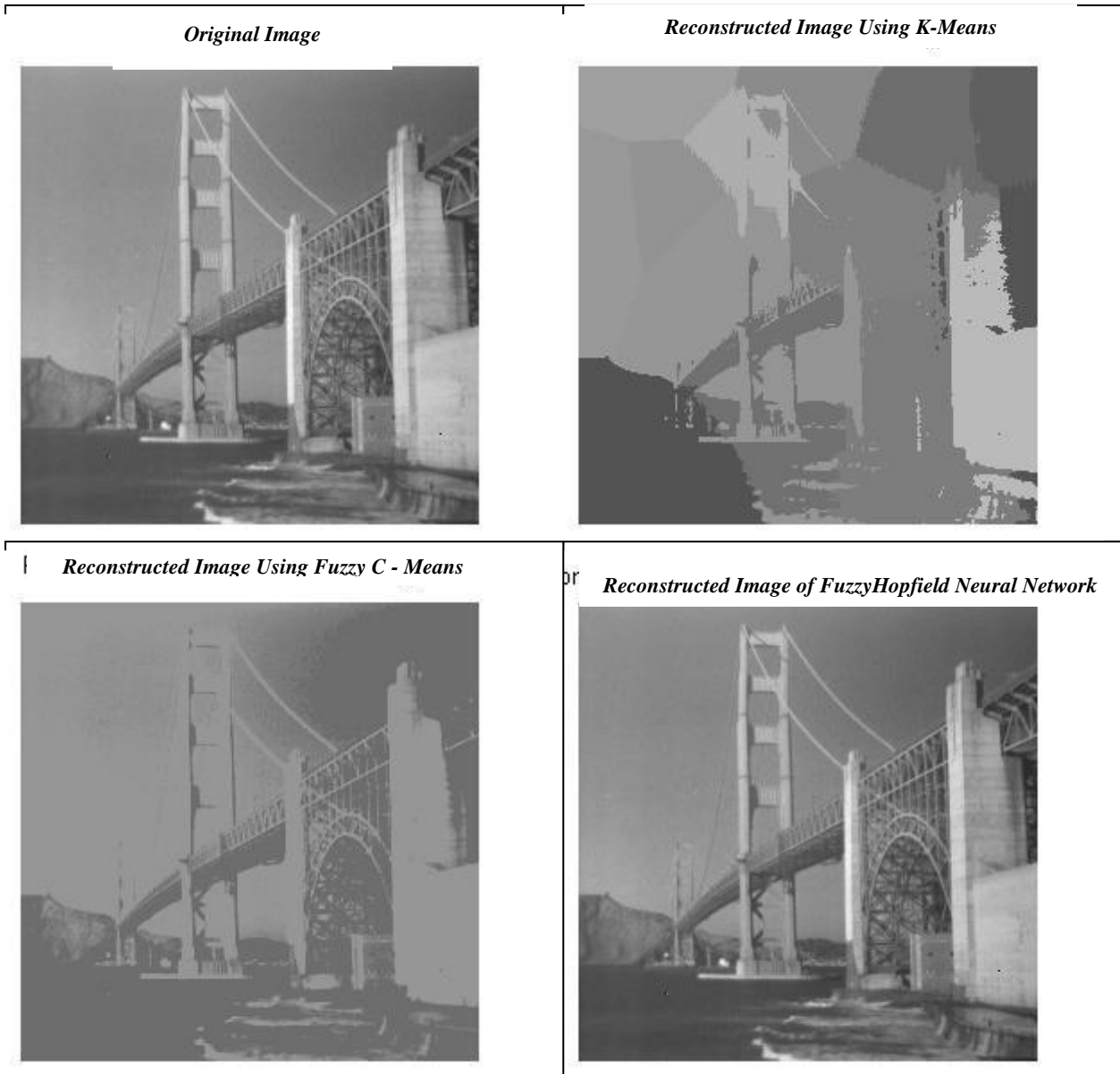
<i>Image2</i>	<i>K_means</i>	<i>Fuzzy C_means</i>	<i>Fuzzy Hopfield Neural Network</i>
<i>SNR_RMS</i>	6.448968	7.135414	18.402799
<i>SNR_PEAK</i>	21.632525	22.764975	30.87243
<i>Compression ratio</i>	1.59656	8.81808	20.58
<i>Bpp</i>	5.0107	0.9072	0.390625

<i>Image3</i>	<i>K_means</i>	<i>Fuzzy C_means</i>	<i>Fuzzy Hopfield Neural Network</i>
<i>SNR_RMS</i>	4.407593	6.739408	18.277225
<i>SNR_PEAK</i>	18.926881	22.383103	30.428342
<i>Compression ratio</i>	2.162048	8.187727	17.712432
<i>Bpp</i>	3.7002	0.9778	0.4516601

<i>Image4</i>	<i>K_means</i>	<i>Fuzzy C_means</i>	<i>Fuzzy Hopfield Neural Network</i>
<i>SNR_RMS</i>	2.400564	3.702035	13.722287
<i>SNR_PEAK</i>	17.544732	19.943472	27.32581
<i>Compression ratio</i>	4.743485	16.29438	32.768
<i>Bpp</i>	1.6865	0.4910	0.24414



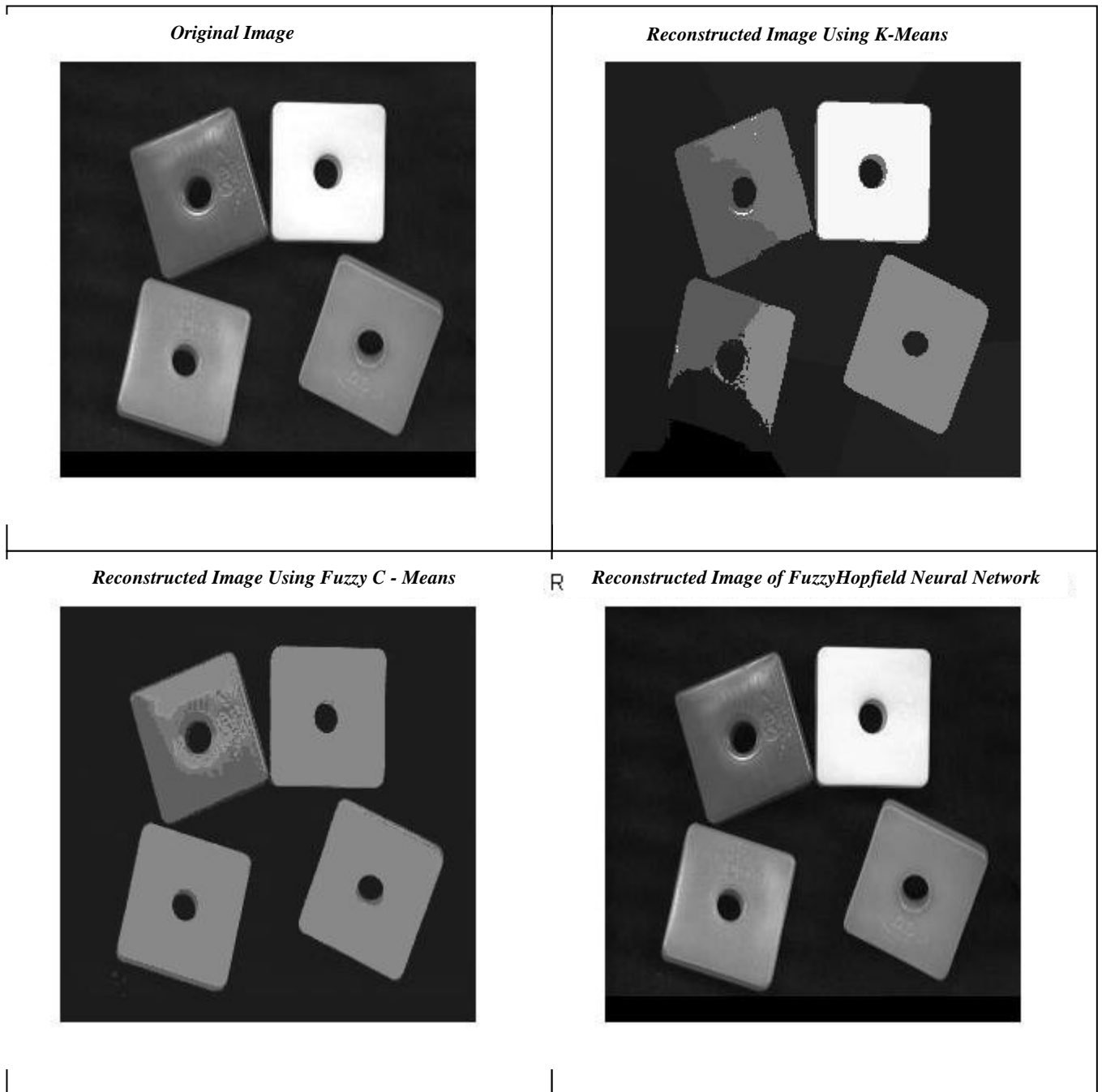
***Fig.5 : Original image , Reconstructed image using K-means, Reconstructed image using fuzzy c-means, Reconstructed image using fuzzy Hopfield neural network***



***Fig. 6 : Original image , Reconstructed image using K-means, Reconstructed image using fuzzy c-means, Reconstructed image using fuzzy Hopfield neural network***



***Fig. 7 : Original image , Reconstructed image using K-means, Reconstructed image using fuzzy c-means, Reconstructed image using fuzzy Hopfield neural network***



*Fig. 8 : Original image , Reconstructed image using K-means, Reconstructed image using fuzzy c-means, Reconstructed image using fuzzy Hopfield neural network*



**REFERENCES**

- [1] Umbaugh, S.E. (2006) **Computer Vision and Image Processing**, Prentice Hall PTR,USA.
- [2] Acharya, T. and Ray A.K. (2005) **Image Processing Principles and Applications**, Wiley & Sons Inc., Hoboken, New Jersey.
- [3] Palus, H. (2003) **Clustering Techniques in Colour Image Segmentation**, Artificial Intelligence methods, pp.223-226.
- [4] Tenas, A. (2001) **Image Segmentation by Unsupervised Adaptive Clustering in the Distribution space for AUV guidance a long sea-bed boundaries using vision**, OCEANS,USA.
- [5] Eschrich, S. (2002) **Fast Accurate Fuzzy Clustering through Data Reduction**, IEEE, Transaction on pattern analysis and machine intelligence, PP. 1-18.
- [6] Kanungo, T. (2002) “An Efficient K-means Clustering Algorithm : Analysis and Implementation”, **IEEE, Transaction on pattern analysis and machine intelligence**, Vol. 24, No. 7, PP. 881-892.
- [7] Valluru, B. Rao and Hayagriva V. (1993) **C++ Neural Network and Fuzzy Logic**, Henry Holt and company, Inc., New York.
- [8] Ciavarella, M. (2004) **Lossless Image Compression using Pixel Reordering**, Australian Computer Society, Inc., Vol. 26.
- [9] Tou, J. and Gonzalez R. (1974) **Pattern Recognition Principles**, Wesley Publishing Company, Inc., Canada.
- [10] Rashid, T. (2003) “Clustering of Fuzzy Image Features”, MSc. Thesis, **University of Bristol**.
- [11] Wu ian-Kang (1997) **Neural Networks and Simulation Methods**, Marcel Dekker, Inc., New York.
- [12] Zhu, C. (2003) “Multi context Fuzzy Clustering for Separation of brain tissues in magnetic resonance images”, **Institute Automation Chinese Academy of Sciences**, Vol. 18, PP. 685-696.
- [13] Kosko, B. (1992) **Neural Networks and Fuzzy System**, A Dynamical system approach to machine intelligence, Prentice-Hall, Inc., USA.
- [14] Takac, A. (2003) “Genetic Programming in Data Mining :Cellular Approach”, MSc. Thesis, **Comenius University**, Bratislava, Slovakia.

- [15] Chen, H. (1995)“Machine Learning for information Retrieval: Neural Networks, Symbolic Learning, and Genetic Algorithms”, **The America Society for information science**, Vol.46, No.3, PP.194-216.
- [16] Gonzales, R. and Wintz P. (1987) **Digital Image Processing**, Addison Wesley Publishing Company.