Compression of Medical Images using Improved Kohonen Algorithm

Mohamed Ettaouil
Modeling and Scientific Computing Laboratory,
Faculty of Science and Technology,
University Sidi Mohammed Ben Abdellah, Fez, Morocco

Mohamed Lazaar
Modeling and Scientific Computing Laboratory,
Faculty of Science and Technology,
University Sidi Mohammed Ben Abdellah, Fez, Morocco

ABSTRACT
Nowadays, neural networks are largely used in signal processing and images. In particular, Kohonen networks or Self Organizing Maps are unsupervised learning models. This method performs a vector quantization (VQ) on the values obtained after processing. The vector quantization has a potential to give more data compression maintaining the same quality. In this paper we propose new scheme to image compression using Kohonen networks. The main innovation is to use the optimal Kohonen topological map to determine the optimal codebook, which can reduce the storage space, simplify data transfer and accelerate the process of data compression, unlike in classical Kohonen approach. To test our approach, we use the medical images. The results demonstrated the effectiveness of the proposed approach.

Keywords
Kohonen Networks, Vector Quantization, Image Compression, Codebook.

1. INTRODUCTION
In the context of image processing, compression schemes are aimed to reduce the transmission rate for images, while maintaining a good level of visual quality [20]. Thus, image compression is a very important factor for better utilization of network bandwidth and computer storage [18]. The compression process is usually lossy and is based on redundancy and irrelevancy reduction, which are inherent in the image domain [16]. The objective of image compression may be summarized as that of finding good approximations to the original image that can be compactly represented [3].

The compression of images has been performed by several techniques among the best known: the JPEG is a loss method standardized by ISO in August 1990; these methods perform compression by performing a scalar quantization (SQ) on the values obtained after processing. The disadvantage of the scalar quantization is that it does not exploit the spatial correlation between different pixels of the image. Another more interesting way to achieve compression coding is not the values individually one after the other, but to encode a set of values simultaneously. This procedure is called vector quantization (VQ) [8]; the VQ has been successfully used for encoding the voice signal and for compressing still images [7].

The Artificial Neural Networks (ANN) are a very powerful tool to deal with many applications [14][17]. Approaches using artificial neural networks for intelligent processing of data seem to be very promising [1] [2], this is mainly due to their structures, providing opportunities for parallel computation and the use of the learning process allows the network to adapt the data to be processed [5]. New techniques based on neural networks as compression tools have been proposed by Jiang [9] and Robert [10]. The Kohonen network is a particular neural network; it can be used as a vector quantizer for images. The results obtained by the Kohonen networks are dependent on their parameters [6]. Among these parameters, we find the size of Kohonen topological map, which has a great impact on the results. In this work, we discusses a new method of images compression using a new model for optimizing the architectures of the Kohonen networks, this model permits to determining the optimal codebook and reduce the coding time.

This paper is organized as follows: In section 2 the vector quantization by Kohonen networks is described. In section 3, the proposed approach to determine the optimal codebook is presented. Experimental results are given in the section 4. The last section concludes our work.

2. VECTOR QUANTIZATION BY KOHONEN NETWORKS
Vector Quantization (VQ) has been observed as an efficient technique for data compression [8]. VQ compression system contains two components: VQ encoder and decoder. The principle of the VQ techniques is simple. In vector quantization we group the source image into blocks or vectors is:

$$X = \{x^1, x^2, \dots, x^n\}$$

This vector of source image forms the input to the vector quantizer. At both the encoder and decoder of the vector quantizer, we have a set of d-dimensional vectors called the codebook of the vector quantizer. The vectors in this codebook, known as code-word, are selected to be representative of the vectors we generate from the source output. Each code-vector is assigned a binary index. At the encoder, the input vector is compared to each code-word in order to find the code-word closest to the input vector. The elements of this code-word are the quantized values of the source image. In order to inform the decoder about which code-word was found to be the closest to the input vector, we transmit or store the binary index of the code-vector. Because the decoder has exactly the same codebook, it can retrieve the code-word given its binary index.

Many authors used the Kohonen's algorithm or Self-Organized feature Map to achieve the vector quantization process of image compression [19]. Kohonen's algorithm is a reliable and efficient way to achieve VQ [6], and has shown to be usually faster than other algorithm and to avoid the problem of "dead units" that can arise for example with the LBG algorithm [12].

Topological maps or Self-Organizing Maps (SOM) were first introduced by T. Kohonen [10]. The Self-Organizing Map is a very popular artificial neural network (ANN) algorithm based on unsupervised learning. The first model is designed for processing high-dimensional data. Since it was originally proposed, the SOM has many applications. The applications initially focused on engineering, including image processing, speech recognition, and flaw detection in machinery. Recently, applications to other fields have emerged including business and management, such as information retrieval, medical diagnosis, time-series prediction, optimization [11]. The SOM is able to project high-dimensional data in a lower dimension, typically 2D. This nonlinear projection produces a 2D pattern map that can be useful in analyzing and discovering patterns in the input space. Consequently the SOM can be used to identify clusters for similar inputs. New inputs can then be analyzed and projected in the trained map, and assigned to an existing cluster. The artificial model of the neurons and network is presented in Fig. 1. The network consists in a regular 2D grid of neurons, where the position of each neuron is fixed. Also, each neuron is identified by its location and weight vector. The neural connection strengths found in biological systems are represented by these weights, and each neuron is laterally connected to a subset of its neighbours.

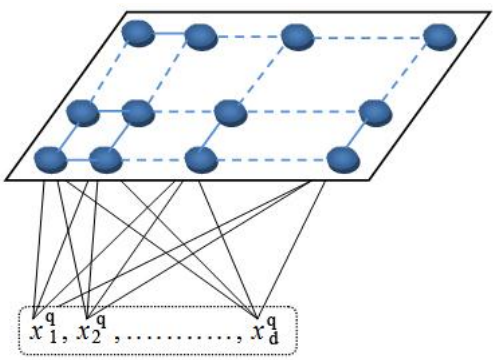


Fig. 1: Kohonen topological map

The SOM algorithm is based in the competitive learning concept where the neurons become gradually sensitive to different inputs, in a n-dimension input space. When an input pattern, from the training set, is presented to the network, a metric distance is computed for all weight vectors. The neuron with the most similar weight vector to the input pattern is called the Best Matching Unit (BMU). The weights of the BMU and its neighbors are then adjusted towards the input pattern. The magnitude of the changes decreases with time and is smaller for neurons far away from the BMU. Maps produced by the SOM algorithm are characterized by the fact that weight vectors which are neighbors in the input space are mapped onto neighboring neurons. If the dimensionality of the input space and the network differ, it is impossible to preserve all similarity relationships among weight vectors in the input space; only the most important similarity relationships are preserved and mapped onto neighborhood relationships on the network of neurons, while the less important similarity relationships are not retained in the mapping. If the input space and network are of the same dimensionality, the SOM algorithm can preserve all the similarity relationships and generates a distorted, but topographic map, of the input space, where more important regions of the input space are represented with higher resolution [19].

Kohonen’s algorithm has however another important property besides vector quantization: it realizes a mapping between an input and an output space that preserves topology; in other words, if vectors are near from each other in the input space, their projection in the output space will be close too. In the proposed compression scheme, we will use a two dimensional Kohonen map corresponding to a map of code-word, as the projection of an initial space including all vectors coming from blocks of the initial images.

SOM training

The SOM consists of a regular, usually two-dimensional (2-D), grid of map units. Each unit is represented by a prototype vector $w^j = \left(w_1^j, w_2^j, \ldots, w_d^j\right)$, where d is input vector dimension. The units are connected to adjacent ones by neighborhood relation. The number of map units, which typically varies from a few dozen up to several thousand, determines the accuracy and generalization capability of the SOM. During training, the SOM forms elastic net that folds onto the "cloud" formed by the input data. Data points lying near each other in the input space are mapped onto nearby map units. Thus, the SOM can be interpreted as a topology preserving mapping from input space onto the map units.

The SOM is trained iteratively. At each training step, a sample vector x is randomly chosen from the input data set, is compared with all w^i to find the reference vector w^g that satisfies a minimum distance or maximum similarity criterion. Though a number of measures are possible, the Euclidean distance is by far the most common:

$$g(t) = \arg \min_{i}^{N} \left\| x(t) - w^i(t) \right\| \quad (1)$$

N is the neurons number in the map. The best-matching unit and neurons within its neighborhood are then activated and modified:

$$w^i(t+1) = w^i(t) + \beta_{g,i}(t) \left\| x - w^i \right\| \quad (2)$$

One of the main parameters influencing the training process is the neighbourhood function $\beta_{g,i}(t)$ between the winner neuron g and neighbour neuron i. This function is positive and symmetric defines a distance-weighted model for adjusting neuron vectors. It is defined by the following relations:

$$\beta_{g,i}(t) = \exp\left(\frac{\left\| r_g - r_i \right\|}{2\sigma^2(t)}\right)$$

where $\left\| r_g - r_i \right\| \cong \left\| w^g - w^i \right\|$, r_g and r_i are positions of neurons g and i on the Kohonen topological map. The $\sigma(t)$ decreases monotonically with time. This function can introduces zones of influence around each winner neuron, the weightings of each neuron are changed, but the degree of change decreases with the distance on the map between the positions of neuron to neuron winner and to make updated. There is also a batch version of the algorithm where the adaptation coefficient is not used [13].

In the case of a discrete data set and fixed neighborhood kernel, the absolute error function E which is a quantitative criterion of the quality of our data modeling. These distances function between each stimulus and the nearest neuron. It is defined as follows:

$$E = \sum_{q=1}^{n} \left\{ \min_{i \in \{1,\ldots,N\}} \left\| x^q - w^i \right\| \right\}$$

where n is the stimuli number, N the number of neurons, the neuron index i and the stimulus q.

3. IMPROVED KOHONEN ALGORITHM

The performance of the Vector Quantization techniques depends largely on the quality of the codebook used in coding the data. If the codebook contains blocks that represent images, the compression quality will be good. For cons, the code words which are not similar to the blocks to be coded will not produce good quality of reconstructed images.

Many algorithms exist to generate the codebook [13], [15]. The Kohonen map is directly applicable to create the codebook. The training sequence contains the blocks of training images, and after training, the weight vectors represents the final code words. In general, this method use a learning database, this base is a set of blocks that are extracted from available images 'of the same type' as the images to be compressed. The learning base therefore contains blocks that are representative images coding. The objective is to create an optimal number N of vectors (code words) that the best represent the blocks of this learning base.

After the learning phase by Kohonen algorithm, we get the map presents in general two types of classes. The first class that does represent any observation (empty class), the second class represents the information. The first type of neurons is of useless part in the Kohonen map; these neurons will be called “useless neurons”.

The mean propose of this work is to delete the useless neurons. It should be noted that a good codebook permits to improve the performance of vector quantization by the Kohonen algorithm. To determine the optimal number of code words, we associated of each neuron, in the last iteration, one binary variable u_i, it is the binary variable for $i = 1, \ldots, N_{max}$ where N_{max} presents the maximal number of neurons (initial size the Kohonen map).

$u_i = 1$ if the i^{th} neuron is used;

$u_i = 0$ if the i^{th} neuron isn’t used;

The variables vector U is defined as follows:

$U = (u_i),\ i = 1, \ldots, N_{max}$

After the learning phase, the weights are fixed and we marked all neurons used in the last iteration. We introduce this phase in the Kohonen algorithm.

The classical method builds the codebook from all weight vectors the Kohonen network. But our method constructs the codebook only from vectors used; except the weight vector of each neuron i that their variable $u_i = 1$ where $i = 1, \ldots, N_{max}$.

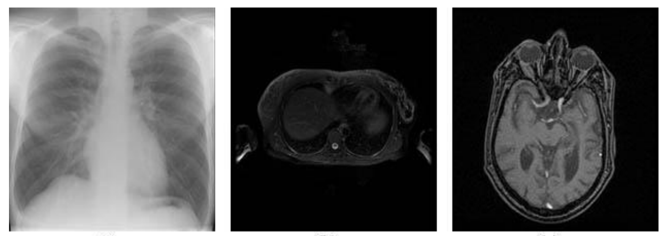
Finally, the codebook contains only the necessary code words, it is optimal his size N is determined from the used neurons. The number optimal of code words is defined as follows:

$$N = \sum_{i=1}^{N_{max}} u_i$$

N presents the optimal size of codebook.

4. EXPERIMENTAL RESULTS

We have tested the both approaches, the improved and the original SOM compression techniques on medical images and Lena image. These images of 256x256 pixels in size are used, each pixel are coded on 8 bits; the original medical images are showed in Fig. 2.



(a) (b) (c)

Fig. 2: Original medical images

Medical imaging was initiated and developed due to the diversity of physical phenomena being used (X-rays, ultrasound waves, magnetic nuclear resonance). Medical imaging was further developed with the increased use of computers in the acquisition process (real-time treatment of a large amount of information) as well as for image reconstruction (tomography).

The image is subdivided into blocks of a reduced size in order to take into account the local properties. The blocks present the input vectors of Kohonen map. After normalization, the blocks are used for generating the codebook by Kohonen algorithm and according to the rules of the Kohonen algorithm. In order to reconstruct the decoded image, a reverse scheme is applied.

Performances of the above algorithms are evaluated in terms of Peak Signal-to-Noise Ratio (PSNR) is given by:

$$\mathrm{PSNR} = 10 \log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right)$$

where Mean Squared Error (MSE) is defined as follows:

$$\mathrm{MSE} = \frac{1}{T}\sum_{i=1}^{T}(\hat{x}_i - x_i)^2$$

where $\hat{x}_i$ and x_i denote, respectively, the original and the encoded pixel values and T is the total number of pixels in an image.

Topological map square RxR (32x16 and 16x16) neurons were used. To compress on image, we dived it into blocks 4x4 pixels.

Numerical results obtained by applying the proposed method and the classical one (SOM) to medical images are presented in the Table 1. This table list, respectively, the test medical images, initial number of neurons (N_{max}), optimal number of neurons (N), Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), coding time reduced (CPU) by the improved method (Red CPU by millisecond) associated with two maps of different size (256 and 512 neurons). We remark from this table, a reduction in number of neurons. For example, a map of 256 neurons, the reduction is two neurons, but for a map of 512 neurons, the average of neurons reduces about 17 neurons.

Table 1: Numerical results for the medical images with size of block 4x4

Images	N_{max}=256 N=254			N_{max}=512 N=495		
	MSE	PSNR	Red CPU	MSE	PSNR	Red CPU
Image a	10.50	37.55	4	7.86	39.17	25
Image b	20.39	35.04	4	12.87	37.03	42
Image c	61.59	30.23	10	48.46	31.38	52

The reduction the neurons number, influence on the compression time for each image. For a map of 512 neurons, the time is reduced (Red CPU) to 52 milliseconds.

Recall that the proposed method contained in additional phase; this phase consists to determine the necessary neurons in order to remove the unnecessary neurons from the initial map.

Fig. 3 shows the three reconstructed images (a', b', c') obtained by the proposed approach. This later permit to construct the codebook of, approximatively, 495 (resp. 254) neurons from initial maps of 512 neurons (resp. 256 neurons) neurons. In this case, our approach is determinates the optimal codebook, because it is constructed only the necessary neurons.

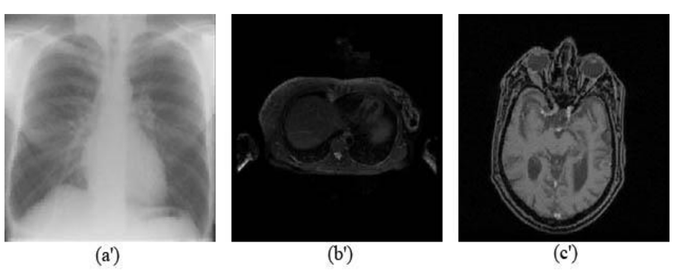


Fig. 3: Reconstructed Medical images

We note that the proposed approach maintains the same visual quality of reconstructed medical images (same PSNR) but decreases the coding time.

A simple comparison between the improved approach and the original method, we remark that the proposed method it is enables to reconstruct images of good visual quality with less compression time.

From a numerical point of view, our method permits to determine the optimal codebook from the optimal size of map. This optimal codebook gives same visual quality than the classical method, but with less compression time and reduces the memory space to store this codebook. Our method can also accelerate the phase of compression and decompression of an image.

5. CONCLUSION

In this paper, we have presented a new approach to determine the optimal codebook by the Self Organizing Maps (SOM) for image compression. The main innovation is to use the optimal map of Kohonen network to generate the optimal codebook. From a numerical point of view, the improved method gives better image quality and low compression time of the images, so reducing the size of the codebook and decrease the memory size to store the codebook.

The presented method considers the compression techniques on 2-D images. Our concept can be extended to 3-D images and Speech data. The research issues such as efficiency of the compression rate in 3-D watermarking will be studied in the future.

REFERENCES

[1] O. T.-C. Chen, B. J. Sheu, W.-C. Fang, 1994, “Image compression using self-organization networks”, IEEE Trans. Circuits Syst. Video Technol., vol. 4, pp. 480-489.

[2] J. Elhachmi, Z. Guennoun. 2011, “Evolutionary Neural Networks Algorithm For The Dynamic Frequency Assignment Problem”, International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 3.

[3] A. Eman, M. Asif, 2010, “Performance Analysis Of Multimedia Compression Algorithms”, International journal of computer science & information Technology (IJCSIT) Vol.2, No.5.

[4] E. Erwin, K. Obermayer, K. Schulten, 1992,“Self-organizing maps: Ordering, convergence properties and energy functions”. Biol. Cyb. 67(1), pp. 47-55.

[5] M. Ettaouil, Y. Ghanou, 2009, “Neural architectures optimization and Genetic algorithms”, Wseas Transactions On Computer, Issue 3, Volume 8, pp. 526-537.

[6] M. Ettaouil, Y. Ghanou, K. Elmoutaouakil, M. Lazaar, 2011, “A New Architecture Optimization Model for the Kohonen Networks and Clustering”, Journal of Advanced

Research in Computer Science (JARCS), Volume 3, Issue 1, pp. 14 - 32.

[7] M. Goldberg, P. R. Boucher, S. Shlien, 1986, “Image Compression using adaptive vector quantization”, IEEE Transactions on Communication, Vol. 34, No. 2, pp. 180-187.

[8] R. M. Gray, “Vector quantization”, IEEE Acoustics, speech and Signal Processing Magazine, 1984, pp. 4-29.

[9] J. Jiang, 1999, “Image compression with neural networks – A survey”, Signal processing :image communication n°14, pp 737-760.

[10] T. Kohonen, 1995, “Self-Organizing Maps”. Berlin/Heidelberg, Germany: Springer, vol. 30.

[11] T.Kohonen, Self-organizing maps", Springer, 1997.

[12] Y. Linde, A. Buzo, R.M. Gray, 1980,“An algorithm for vector quantizer design”, IEEE Transactions on Communication, Vol. 28, No. 1, pp. 84 - 95.

[13] N. Nasrabadi, Y. Feng, 1988,“Vector quantization of images based upon the Kohonen self-organizing feature maps”, in IEEE Int. Conf. Neural Networks, San Diego, CA, vol. 1, pp. 101-108.

[14] J. Pradeep, E. Srinivasan, S. Himavathi, 2011, “Diagonal Based Feature Extraction For Handwritten Alphabets Recognition System Using Neural Network”, International Journal of Computer Science & Information Technology (IJCSIT), Vol 3, No 1.

[15] Robert , Haykin, 1995, “Neural network approaches to images compression”, in .Proceedings of the IEEE, vol 82, n° 2.

[16] K. Sayood, “Introduction to Data Compression”, Elsever, 2006.

[17] J. Skowronski, I. Dologlou, 1997, "Image compression using permutative vector quantization", Signal Processing: Image Communications, Vol. 11, pp. 39-47.

[18] C. Stanley and al. , “ Competitive learning algorithms for vector quantization”, in Neural networks, vol 3, 1990, pp. 277-290.

[19] J. Ziv, A. Lempel, 1977, "A Universal Algorithm for Sequential Data Compression", IEEE Trans. on Inform. Theory, Vol. 23, May 1977, pp. 337-343.