Visushrink Pretreatment for Image Compression

Imen Chaabouni
Research Unit: Sciences and Technologies of Image and Telecommunncations
Higher Institute of Biotechnology, Sfax
Road of Soukra km 3, 3038 Sfax Tunisia

Wiem Fourati
Research Unit: Sciences and Technologies of Image and Telecommunncations
Higher Institute of Biotechnology, Sfax
Road of Soukra km 3, 3038 Sfax Tunisia

M. Salim Bouhlel
Research Unit: Sciences and Technologies of Image and Telecommunncations
Higher Institute of Biotechnology, Sfax
Road of Soukra km 3, 3038 Sfax Tunisia

ABSTRACT
In this paper we present a new method based on wavelet thresholding denoise for 2D image compression. The image blocks to be compressed are classified before presenting them to Fuzzy C Means clustering and Kohonen's network. We will compare performances of two methods, one compressed with FCM clustering methods [24] and another compressed with incremental self organizing map (ISOM) [27]. The results show that the wavelet thresholding denoise approach succeeded to improve high performances in terms of compression ratio and reconstruction quality. We noted that, when we apply an image pre-treatment for the original image to be compressed by FCM and by ISOM, the reconstructed image is better.

General Terms
Algorithms, Performance, Design.

Keywords
Wavelet thresholding denoise, image compression, fuzzy c mean, incremental self organizing maps.

1. INTRODUCTION
An image is often corrupted by noise in its acquisition or transmission. The goal of denoising is to remove the noise while retaining as much as possible the important signal features. The wavelet transform has good local performances in both spatial and frequency domains [3]. It improves the time resolution and frequency resolution compared with short – time Fourier transform. Recently, the wavelet transform has found many applications in the image processing and other fields [1], [14]. In image denoising problem [14], S.Mallat and W.L. Hwang [3] first showed that effective noise suppression may be achieved by transforming the noisy image into the wavelet domain, and preserving only the local maxima of the transform. A wavelet reconstruction using only large magnitude coefficients has also been shown to approximate well the uncorrupted image; in other words, noise suppression is achieved by thresholding the wavelet transform of the contaminated image. To choose the appropriate threshold, Donoho and Johnstone [6], have introduced the minimax and universal thresholding schemes, and discussed both hard and soft thresholds in a general context that included ideal denoising in both the wavelet and Fourier domains.
On a seemingly unrelated front, thresholding denoise has been proposed for lossy compression in several works [2], [7]. Concerns regarding the compression rate were explicitly addressed. This is important because any practical coder must assume a limited resource (such as bits) at its disposal for representing the data. Other works [14] also addressed the connection between denoising and compression, especially with nonlinear algorithms such as wavelet thresholding in a mathematical framework. Wavelet transform, because of its excellent localization property, has rapidly become an powerful signal and image processing tool for denoising [15], [8], [16]. Wavelet thresholding, first proposed by Donoho [4], [7], [8], is a signal estimation technique that exploits the capabilities of wavelet transform for signal denoising and has recently received extensive research attentions [17], [20], [21]. In this paper, we investigate wavelet-based techniques for denoising, focusing on shrinkage methods. The basic idea behind these techniques is to use wavelets to transform the data into a different basis. In details, the large coefficients are mainly the signal, and the smaller ones represent the noise. The wavelet coefficients are suitably modified and the denoised data is obtained by an inverse wavelet transform of the modified coefficients.
The paper is organizing as follows: the Fuzzy C Means (FCM) clustering methods is described in section 2. Section 3 presents the Incremental Self Organizing Map compression method. Section 4 details the pre-treatment of image with wavelet thresholding denoise using soft shrinkage. Section 5 provides the experimental results and the comparison between the two methods of compression and the effects of wavelet thresholding denoise on image. Finally we conclude this work.

2. Fuzzy C Means (FCM)
Lossy image compression is usually performed by FCM techniques. The implementation of FCM in image compression requires the decomposition of the image into a number of rectangular blocks. Each of these blocks forms a vector, also called training vector. Then, the objective of vector quantization is to classify all the training vectors into a number of clusters. This is accomplished by minimizing a distortion measure. The centers of the resulted clusters provide the codebook vectors, which constitute the codebook. The image is reconstructed by replacing each training vector by its closest codebook vector [24].

3. IMAGE COMPRESSION WITH INCREMENTAL SELF ORGANIZING MAPS (ISOM)

3.1. Description
There are many types of self-organizing networks applicable to a wide area of problems. One of the most basic schemes is competitive learning. The Kohonen network can be seen as an extension to the competitive learning network [22]. Self Organizing maps are a kind of artificial neural networks which inspire from the learning neural networks. This kind of neural network allows projecting an entry space on a one or two dimensional map called topological map. It's composed

of two layers, an entry vector, and a map where all elements are of the same dimension as for the entry.

3.2. Incremental Self Organizing Map

ISOM is an incremental network having an unsupervised learning scheme. ISOM is a two layer network, as shown in figure1. In the figure1, k represents the dimension of the input vector. The nodes in the first layer represent the coefficients vectors (code words). The number of nodes in the first layer is determined after the decomposition of the picture. The winner-takes-all guarantees that there will be only one node activated. While first layer represents input vector, index layer represents the index values of the input vector. The training of Kohonen networks [6],[9],[23], is competitive : when an entry vector is presented to neural network, all neurons get in completion to determine the winner neuron which is the one with the weight vector i closest to the entry vector according to a distance measure (generally the Euclidean distance). Thus the winner neuron get closer to the vector entry by adjusting his weight vectors according to the distance between him and the winner following this rule,

$$w_i(k)=w_i(k-1)+\eta(x(k-1)-w_i(k-1)) \qquad (1)$$

Where k is the iteration index, and η is the gain constant ($0<\eta\leq 1$).This process is to be iterated until the convergence of the map (when the distortion of the map between two consecutive iterations is smaller). The number of nodes in the first layer and the indexes of the output nodes are automatically determined during the learning. By the end of the training, the map is ready to be used. Thus when a new entry vector is presented to the map; the distance between this vector and each neuron is calculated to determine the winner which is the closest to the new vector, and hence it will be affected to the class which corresponds to the winner [26].

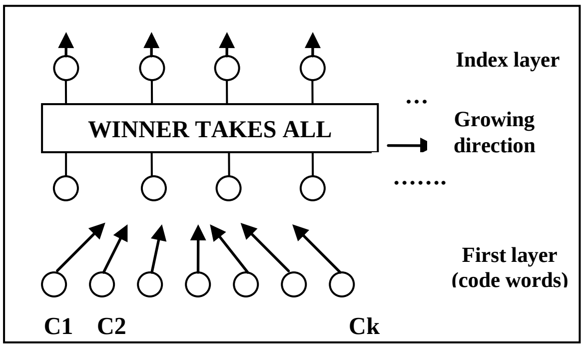


Fig 1: Structure of ISOM

3.3. Data Compression Using SOM

One important feature of SOM is the possibility of achieving high compression ratios with relatively small block size. Another important advantage of SOM image compression is its fast decompression by table lookup technique. SOM is basically a clustering method, grouping similar vectors (blocks) into one class [18].

Our basic approach to image compression consists of several key steps.

Step 1: Extract a square block from the image in an order.

Step 2: Compute the clustering to the block of the image by the methodology of incremental self organizing map.

Step 3: Compute the Euclidean distances between the input vector and the node in the first layer, and find the minimum distance.

Step 4: The competition between the neurons of the card of Kohonen is started. This competition is based on the strategy of "Winner Takes all". If the minimum distance exceeds the threshold ε fixed by the user, the weights will not be modified and increment the index counter by one, otherwise, the weights of the node of the card are updated nearest to the input vector according to equation 1 and the number of classes is the same one.

Step 5: Last step provides us a matrix 'classes' which contains the indices of classes of each block and a matrix 'weight' which represents the weights of the found classes respectively. This phase constitutes the compression phase of the image.

Step 6: We can improve this compression on applying the Huffman code on the matrix.

Step 7: Compute decompression phase by rebuilding the compressed image.

The self organizing maps are used just before the matching of the blocks and the range of it. The compression will be performed through these steps:

1. Compute Euclidean distance between the neurons of the card of Kohonen and the input vector.
2. If the minimum distance exceeds the threshold ε, the weights of neurons will not be modified and increment the number of classes by one, otherwise, the weights of the node of the card are updated nearest to the input vector according to equation 1 and the number of classes is the same one.
3. Segmented image is also compressed by Huffman code by applying this code on the indices of classes of the image.

The use of incremental self organizing map will provide us two matrixes; one which contain the indices of classes of each block and another matrix which represents the weights of the found classes respectively. This phase constitutes the compression of the image, considering which one has a profit in the face of the image.

Huffman coding is a variable length coder in which the length of a code word is based on the probability of the symbol (i.e. decorrelated image value). This method is optimal, given the severe constraint that an integral number of bits are assigned to each symbol. The bit rate has a lower bound of 1 bit/pixel.

At the decompression phase, for each index, a lookup process is performed in the input vector to obtain the corresponding weight representative of the original block. The obtained weights are placed, in sequence, at the decompressed file.

1. Load the compressed file.
2. Select, in order, an index i from the indices of 'classes' tables.
3. Using i as an address, access the corresponding vector to obtain the weight and store it in the same order of index i into the decompressed file.

4. IMAGE PRE-TREATMENT WITH WAVELET THRESHOLDING DENOISE

4.1. Wavelet

An image is often corrupted by noise in its acquisition and transmission. Image denoising is used to remove the additive noise while retaining as much as possible the important signal features. In the recent years there has been a fair amount of research on wavelet thresholding and threshold selection for signal de-noising [5], [17]-[10], [11], because wavelet provides an appropriate basis for separating noisy signal from the image signal. The motivation is that as the wavelet transform is good at energy compaction, in details the small coefficient are more likely due to noise and large coefficient because of important signal features [19]. These small coefficients can be thresholded without affecting the significant features of the image.

Thresholding is a simple non-linear technique, which operates on one wavelet coefficient at a time. In its most basic form, each coefficient is thresholded by comparing against threshold, if the coefficient is smaller than threshold, set to zero; otherwise it is kept or modified. Replacing the small noisy coefficients by zero and inverse wavelet transform on the result may lead to reconstruction with the essential signal characteristics and with less noise. Since the work of Donoho & Johnstone [5], [12], [7], [10], there has been much research on finding thresholds, however few are specifically designed for images. In our work, we would study wavelet thresholding technique to image compression.

4.2. Wavelet Thresholding

Let the original image be {fij , i, j= 1,2,...M}, where M is some integer power of 2. The goal is to remove the noise, or denoised {fij}, and to obtain an estimate {f ij} of {f ij} which minimizes the mean squared error (MSE),

$$\text{MSE}(\hat{f}) = \frac{1}{M^2}\sum_{i,j=2}^{M}(\hat{f}_{ij} - f_{ij}) \qquad (2)$$

Let f = {fij}i,j; that is the boldfaced letters will denote the matrix representation of the signals under consideration. Let v and v-1 denote the two dimensional orthogonal discrete wavelet transform (DWT) matrix and its inverse respectively. Then Y = v f represents the matrix of wavelet coefficients of f having four subbands (LL, LH, HL and HH) [13], [21]. The subbands HHk, HLk, LHk are called details, where k is the scale varying from 1, 2...J and J is the total number of decompositions. The size of the subband at scale k is M/2k × M/ 2k. The subband LLJ is the low resolution residual. The wavelet thresholding denoising method processes each coefficient of Y from the detail subbands with a threshold function to obtain Xˆ. The denoised estimate is then fˆ = v-1 Xˆ.

<table>
<tr><td>LL3</td><td>HL3</td><td rowspan="2">HL2</td><td rowspan="3">HL1</td></tr>
<tr><td>LH3</td><td>HH3</td></tr>
<tr><td colspan="2">LH2</td><td>HH2</td></tr>
<tr><td colspan="3">LH1</td><td>HH1</td></tr>
</table>

Fig 2 : Subbands of the 2-D orthogonal wavelet transform

4.3. Denoising by threshoding

The problem boils down to finding an optimal threshold such that the mean squared error between the signal and its estimate is minimized. The wavelet decomposition of an image is done as follows: At the first level of decomposition, the image is split into 4 subbands, namely the HH, HL, LH and LL subbands. The HH subband gives the diagonal details of the image; the HL subband gives the horizontal features while the LH subband represents the vertical structures. The LL subband is the low resolution residual consisting of low frequency components. Subband as such is in turn split at higher levels of decomposition. A basic procedure remains the same:

1. Calculate the DWT of the image.

2. Threshold the wavelet coefficients. (Threshold may be universal)

3. Compute the IDWT to get the denoised estimate.

Soft thresholding is used for all the algorithms because of the following reasons: Soft thresholding has been shown to achieve near minimax rate over a large number of Besov spaces [8]. Moreover, it is found yielding more pleasing visual images.

4.4. Visushrink [25]

VisuShrink is thresholding by applying the Universal threshold proposed by Donoho and Johnstone [7]. This threshold is given by $\sigma \sqrt{2\log M}$ where σ is the noise variance and M is the number of pixels in the image. It is proved in [7] that the maximum of any M values iid as N(0, σ^2) will be smaller than the universal threshold with high probability, with the probability approaching 1 as M increases. Thus, with high probability, a pure noise signal is estimated as being identically zero. However, for denoising images, VisuShrink is found to yield an overly smoothed estimate.

5. EXPERIMENTAL RESULTS

Following are the results of running the denoising algorithm for the two compression methods FCM and ISOM, where the performance of the two method is compared. In order to simulate the experiments, we took the well-known "cameraman.tif" and "peppers.bmp" images of 256*256 pixels of 256 gray levels.

Fig 3: 'Cameraman.tif'

Fig 4: 'Peppers.bmp'

The final codebook quality was evaluated by using the distortion measure (MSE) for an image of size N*N given in equation (3), this measure is used to determine the reconstructed image quality and the compression ratio (τ), is defined as in equation (4).

$$\text{MSE}=\|y - x\|^2 = \frac{1}{N^2}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{i,j} - y_{i,j})^2 \qquad (3)$$

$$\tau = (1 - \frac{Tc}{To}) \times 100 \qquad (4)$$

Where x^{ij} and y^{ij} are the pixel intensities for the original and the reconstructed image. The control parameter in compression process is size of the square blocks. Figure 5, 6, 7, and 8 shows mean square error and compression ratio for compression methods by FCM and ISOM with image pre-treatment by thresholding denoise.

D: distance between nodes; ε : learning coefficient; τ : compression ratio; MSE : Mean Square Error; c : number of classes.

In figure 5 and 6, we apply the FCM compression technique and we notice that by using the Visushrink filter we have the same compression ratio and best mean square error for the two block size decomposition 8*8 and 4*4.
Figure 7 and 8, shows for a decomposition block size 8*8 and by applying the filter Visushrink, the compression ratio increases and the mean square error decreases in the reconstructed image compressed by ISOM. For the same figure and by 4*4 block size decomposition, we notice that the compression ratio and the mean square error increases after using the Visushrink filter.
Finally, image compression provide us better results in compression ratio and mean square error by using a pre-treatment filter Visushrink on image. ISOM compression shows that there are more performant results than FCM compression.

Number of classes = 304 τ = 68.26%
MSE = 81.40 Block size : 8 × 8

a. Image compressed by FCM without filter

Number of classes = 304 τ = 68.21%
MSE = 67.98 Block size : 8 × 8

b. Image compressed by FCM with VisuShrink thresholding denoise

Number of classes = 372 τ = 84.57 %
MSE = 78.1 Block size : 4 × 4

c. Image compressed by ISOM without filter

Number of classes = 372 τ = 84.48%
MSE = 58.41 Block size : 4 × 4

d. Image compressed by ISOM with VisuShrink thresholding denoise

Fig 5: Reconstructed image 'Cameraman.tif', at left: compression by FCM; at right: denoised image by Visushrink and compression by FCM

Number of classes = 371 τ = 61.32 %
MSE = 60.24 Block size : 8 × 8

a. Image compressed by FCM without filter

Number of classes = 371 τ = 61.32%
MSE = 56.28 Block size : 8 × 8

b. Image compressed by FCM with VisuShrink thresholding denoise

Number of classes = 442 τ = 81.69 %
MSE = 47.06 Block size : 4 × 4

c. Image compressed by FCM without filter

Number of classes = 442 τ = 81.74 %
MSE = 39.44 Block size : 4 × 4

d. Image compressed by FCM with VisuShrink thresholding denoise

Fig 6: Reconstructed image ‘Peppers.bmp’, at left: compression by FCM; at right: denoised image by Visushrink and compression by FCM

D = 70 ε= 0.01 τ = 68.75 %
MSE = 103.5 Block size : 8 × 8

a. Image compressed by ISOM without filter

D = 70 ε = 0.01 τ = 70,8 %
τ = 104.14 Block size : 8 × 8

b. Image compressed by ISOM with VisuShrink thresholding denoise

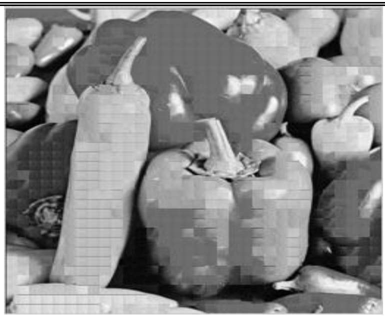
D = 35 ε = 0.01 τ = 72.7295 %
MSE = 141.2407 Block size : 4×4

c. Image compressed by ISOM without filter

D = 35 ε = 0.01 τ = 75.63 %
MSE = 147.07 Block size : 4×4

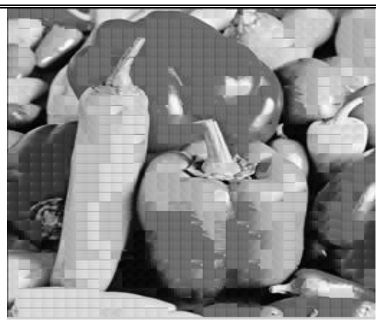
d. Image compressed by ISOM with VisuShrink thresholding denoise

Fig 7: Reconstructed image 'Cameraman.tif', at left: compression by ISOM; at right: denoised image by Visushrink and compression by ISOM



D = 70 ε = 0.01 τ = 62.20%
MSE = 124.39 Block size : 8×8

a. Image compressed by ISOM without filter

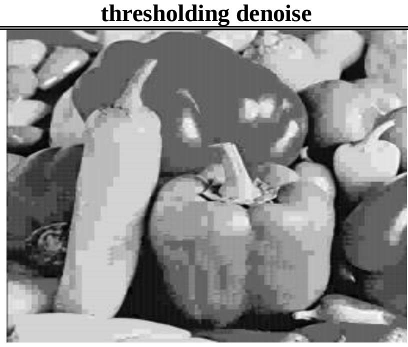


D = 70 ε = 0.01 τ = 63.96%
MSE = 117.96 Block size : 8×8

b. Image compressed by ISOM with VisuShrink thresholding denoise

D = 35 ε = 0.01 τ = 76.19%
MSE = 78.66 Block size : 4×4

c. Image compressed by ISOM without filter



D = 35 ε = 0.01 τ = 77.85%
MSE = 82.08 Block size : 4×4

d. Image compressed by ISOM with VisuShrink thresholding denoise

Figure 8. Reconstructed image 'Peppers.bmp', at left: compression by ISOM; at right: denoised image by Visushrink and compression by ISOM

6. CONCLUSION

We verify the compromise that exists between the compression ratio and the quality of the rebuilt image. Indeed, we apply the wavelet thresholding denoise, when we use Visushrink filter to the compression technique. We note that the incremental self organizing map compression method provide us better results than the fuzzy c-means compression method. When we apply Visushrink to the fuzzy c-means compression technique the compression ratio is the same and the mean square error decreases. We note also that by 8*8 decomposition block size, incremental self organizing map compression method provide us increasing compression ratio and decreasing mean square error by applying wavelet thresholding denoise to the original image and by 4*4 decomposition block size increasing compression ratio and mean square error.

7. REFERENCES

[1] A. Grossman, Wavelet Transform and Edge Detection, In Stochastic Processes in Physics and Engineering, M. Hazewinkel. Ed, Sodrecht: Reidel, 1986

[2] S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, IEEE Trans. on Pattern Analysis and Machine Intelligence, 11(1989)7, 674-693

[3] S. Mallat, W. L. Hwang, Singularity detection and processing with wavelets, IEEE Trans.on Information Theory, 38(1992)2, 617-643

[4] D. L. Donoho, Wavelet Thresholding and W.V.D.: A 10-minute Tour, Int. Conf. on Wavelets and Applications, Toulouse, France, June 1992

[5] D.L. Donoho, De-Noising by Soft Thresholding, IEEE Trans. Info. Theory 43, pp. 933-936, 1993

[6] Barnsley M.F., Ervin V., Hardin D. and Lancaster J., (1993), Solution of an inverse, Fractal image compression. AK Peters Ltd.,Wellesley.

[7] D.L. Donoho, I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Biometrika, 81(1994), 425-455

[8] D. L. Donoho, De-Noising by Soft-Threshholding, IEEE Trans. Information Theory, Vol. 41(3), May 1995

[9] Hamzaoui R., (1995), Codebook clustering by self-organizing maps for fractal image compression, NATO ASI Conference On Fractal Image Encoding and Analysis, July.

[10] D.L. Donoho and I.M. Johnstone, Wavelet shrinkage: Asymptopia?, J.R. Stat. Soc. B, ser. B, Vol. 57, no. 2, pp. 301-369, 1995.

[11] M. Lang, H. Guo and J.E. Odegard, Noise reduction Using Undecimated Discrete wavelet transform, IEEE Signal Processing Letters, 1995.

[12] D.L. Donoho and I.M. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, Journal of American Statistical Assoc., Vol. 90, no. 432, pp 1200-1224, Dec. 1995.

[13] M. Vetterli and J. Kovacevic, Wavelets and Subband Coding. Englewood Cliffs, NJ, Prentice Hall, 483 pp. 1995.

[14] G. Chag, et al. Image denoising via lossy compression and wavelet thresholding, In Proc. IEEE Int. Conf. Image Processing, Santa Barbara, CA, 1997, 604-607

[15] A. Chambolle, R. A. DeVore, N-Y Lee and B. J. Lucier, Nonlinear Wavelet Image Processing: variational problems, compression and noise removal through wavelet shrinkage, IEEE Trans. Image Processing, Vol. 7, pp. 319-335, 1998.

[16] X. Shao and V. Cherkassky, Model Selection for Waveletbased Signal Estimation, Proc. IEEE Int. Joint Conf. on Neural Networks, Anchoradge, Alaska, 1998

[17] S. Grace Chang, Bin Yu and M. Vetterli, Adaptive Wavelet Thresholding for Image Denoising and Compression, IEEE Trans. Image Processing, Vol. 9, pp. 1532-1546, Sept. 2000.

[18] Mas Ribès J.-M., Simon B. and Macq B., (2000), Combined Kohonen neural networks and discrete cosine transform method for iterated transformation theory, Signal Processing: Image Communication, Accepted 15 May.

[19] Maarten Jansen, Noise Reduction by Wavelet Thresholding, Springer –Verlag New York Inc.- 2001.

[20] Lakhwinder Kaur, Savita Gupta, and R. C. Chauhan, Image Denoising using Wavelet Thresholding, Third Conference on Computer Vision, Graphics and Image Processing, India Dec 16-18, 2002.

[21] Savita Gupta and Lakhwinder kaur, Wavelet Based Image Compression using Daubechies Filters, In proc. 8th National conference on communications, I.I.T. Bombay, NCC-2002

[22] Luger F., (2002), Artificial Intelligence: Structures and Strategies for complex problem solving, 4th edition, Addison-Wesley, 856p.

[23] Negnevitsky M., (2002), Artificial Intelligence: A guide to Intelligent Systems, Addison- Wesley, 394p

[24] Tsekouras E. G., (2005), A fuzzy vector quantization approach to image compression, Elsevier, Applied Mathematics and Computation, N° 167, pp. 539-560.

[25] Y. Yang, Image denoising using wavelet thresholding techniques 2005

[26] Zumray D., (2006), A unified framework for image compression and segmentation, Expert Systems with Applications.

[27] I. Chaabouni, Compression d'images par les cartes topologiques de Kohonen, Diplôme de Master, Ecole Nationale d'Ingénieurs de Sfax, Juin 2007