Fractal Features Classification for Texture Image Using Neural Network and Mathematical Morphology

Mohammed Talibi-Alaoui and Abderrahmane Sbihi

***ABSTRACT.* This work proposes a new method for unsupervised texture image classification, which is based on both Kohonen maps and mathematical morphology. Various features obtained from the fractal dimension computed using differential box counting method, are extracted from the texture images and then applied and projected into a Kohonen map. This map is represented by the underlying probability density function (pdf) estimated, by a non-parametric technique in the n-dimensional space, from the weight vectors resulting of the learning process. Under the assumption that each modal region of the underlying pdf corresponds to a one homogenous region in the texture image, the second step of the process understanding consists to an extraction, in the Kohonen map, of the modal regions of the pdf as connected components without using any thresholding procedure. That is done by making concepts of morphological watershed transformations suitable for modal domains detection. The observations falling in the so localised homogenous region in the image are considered as prototypes and are then used in the clustering procedure by means of an assignment rule.**

Index Terms—**Texture Image, Classification, Kohonen Network, Watershed Transformation, Fractal dimension, Fractal features.**

I. Introduction

THE quality of interpretation of a texture image depends heavily on the segmentation. Among segmentation method, some are searching the related homogeneous regions in the image. This low level treatment is used for the identification of classes present in the image, and therefore the classification by assigning each pixel of the image to one of the classes identified.

Several algorithms have been proposed [1, 2, 3, 4, 5]. Some require a thresholding of the histogram or adjustment of parameters, others are limited to cases where the different classes correspond to separable clouds in the measurement space, while others do not take into account the geometric relationships in the image.

In this paper, we propose a new approach for unsupervised classification of texture images based on morphological and connectionist concepts. We first calculate local fractal features from the whole image, then place the feature vector of each pixel into the feature space which form a cloud of observations. To help discover the different classes present in this cloud of N-dimensional observations, we propose as first treatment phase, the projection of these observations on a two-dimensional Kohonen map [6, 7]. The information in each cell of this map is represented by the probability density function value (pdf) estimated by a nonparametric procedure [8], from the distribution in multidimensional space of weight vectors resulting from the learning of the neural network.

Under the assumption that each regional maximum of the pdf [9, 10] is a modal area of this function, which corresponds to a homogenous region in the image, the second step of treatment consists in extracting automatically regions of the fdp modal regions into connected and individualized components. This extraction is based on the exploitation of the watershed technical, generally used for segmentation of digital images using mathematical morphology [11, 12, 13].

Finally, the classification phase is to take the weight vectors corresponding to the modal regions detected as prototypes of homogeneous regions in the image. Weights from each of these prototypes are the basis of the assignment of any pixel of the image to one of the classes extracted.

II. Representation of the Image Texture Information on the Kohonen Map

Every classification process begins with an acquisition step of observations which consists in determining relevant attributes that characterize better the objects. The sample of observations is constituted by fractal features of a texture image.

A. Self-Organising feature Map (SOM) and learning process

Let's $\Gamma = \{X_1, X_2, X_3, ..., X_Q\}$ be a sample of Q observations in a N-dimensional space such as

Manuscript received March 18, 2012; revised April 3, 2012.
M. Talibi Alaoui is with the LARI Laboratory, Department of Mathematics and Computer Science, FSO, BP. 717, 60050, Oujda, Morocco (corresponding author to provide phone : 212-0536-500601; fax: 212-0536-500603; e-mail: m.talibialaoui@fso.ump.ma).
A. Sbihi is with the LIRF Laboratory, Department of physics, FSK, University of Ibn Tofail, BP. 133, 14000, Kénitra, Morocco (e-mail : sbihi@ensat.ac.ma).

$X_q = [x_{q,1}, x_{q,2}, .., x_{q,n}, .., x_{q,N}]^T, \qquad q = 1, 2, .., Q.$

The Kohonen network is made of a two layers. The first one, the input layer is composed of N neural units representing the N attributes of the observation X_q. The output layer, or competitive layer, is composed of M neural units regularly distributed on the map which elaborates prototypes of the data (cf. Figure 1). The neural units of the first layer are connected to the units of the second layer. Each interconnection from an input unit j to an output unit m has a weight $W_{m,j}$. That means that each output unit m has a corresponding weight vector $W_m = [W_{m,1}, W_{m,2}, .., W_{m,n}, .., W_{m,N}]^T$ (cf. figure 1).

Each neural unit in the output layer is assigned with a specific position and a weight vector. When an input $X_q(t)$ is presented to the network, the neural unit whose weight vector is the closest to this observation wins the competition and is allowed to learn it even better. The neural units of the second layer are so interconnected to elaborate the winning neural units by inhibiting the other units. The output of the winner is then equal to 1 while the outputs of all the other output units are set to 0, such as :

$$y_m = \begin{cases} 1 \ if \ \ d(X_q(t), W_m(t)) \le d(X_q(t), W_{m'}(t)) \ \ m' \ne m \\ 0 \ \ else \end{cases} \tag{1}$$

Where $d(X_q(t), W_{m'}(t))$ is the Euclidean distance between the observation $X_q(t)$ and the weight vector $W_{m'}(t)$ of the unit m' in the output layer.

The winning neural unit and its neighbours are updated. The size of the neighbourhood is decreased as the training goes on. The weight vector of this winning unit, noted m^*, and its neighbours m are modified according to equations :

$$\begin{aligned} &W_m(t) = W_m(t-1) + a(t).[X_q(t) - W_m(t-1)] && if \ m = m^* \\ &W_m(t) = W_m(t-1) + a(t).h(m^*,t).[X_q(t) - W_m(t-1)] && if \ m \in V(m^*, r(t)) \\ &W_m(t) = W_m(t-1) && if \ m \notin V(m^*, r(t)) \ and \ m \ne m^* \end{aligned} \tag{2}$$

where :

- m^* is the winning unit defined by :

$$m^* = Arg \min_{m} [d(X_q(t), W_m(t))] \tag{3}$$

- $r(t)$ is the interaction radius which depends on the number t of the iteration.
- $a(t)$ is the learning coefficient at the time t. This coefficient can be an hyperbolic, exponential or linear function of t.
- $V(m,r)$ is the neighbourhood of a neural unit m with a radius r, defined by :

$$V(m,r) = \{m' \in [0, M[, \ m' \ne m \ / \ d(U_m, U_{m'}) \le r\} \tag{4}$$

- $d(U_m, U_{m'})$ denotes the Euclidean distance between the position vectors U_m and $U_{m'}$ of the m and m' neural units.
- $h(m^*,t)$ is the interaction function that depends on the proximity radius $r(t)$ defined by :

$$h\ (m^*, t) = \exp\left(-\frac{d(U_m, U_{m'})^2}{2\, r(t)^2}\right) \tag{5}$$

B. Application to texture image classification

In this application, we used a texture image (cf. figure 9).

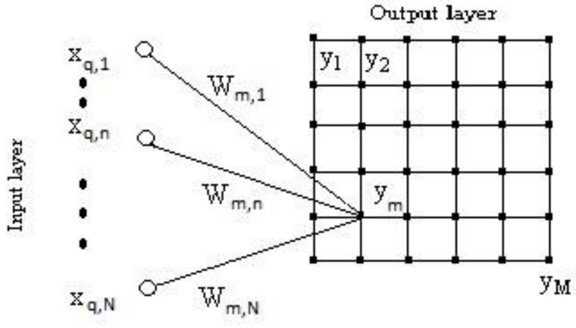

Fig. 1. Kohonen Network

a. Fractal dimension

The concept of fractal is used in a large number of applications including image analysis, classification pattern recognition, segmentation etc [14]. Fractal objects have irregular shapes and complex structures that cannot be represented adequately by the traditional Euclidean dimension. The concept of fractal dimension (FD) is used as an indicator of surface roughness [15].

Of the wide variety of methods for estimating the fractal dimension that have so far been proposed, the box-counting method [14], as it can be computed automatically and can be applied to patterns with or without self-similarity[16].

The box counting method consists in partitioning the image space into square boxes of equal size. The box covers the image space of the function or pattern of interest and the number of boxes that contain at least one pixel of the function is counted. The process is repeated with different box sizes. The fractal dimension is obtained from the slope of the best fitting straight line to the graph plotting the log of the number of boxes counted versus the log of the magnification index for every stage of partitioning as shown in figure 2. For example, an image measuring size $M \times M$ pixels is scaled down to $s \times s$, where $1 < s < M/2$, and s is an integer. Then, $r = s/M$.

Fractal dimension D is given by, $D = \dfrac{\log(N_r)}{\log(1/r)}$ (6)

In this paper the differential box counting method is used to calculate the FD and then different fractal features are derived from this fractal dimension.

b. Differential Box Counting Method

N. Sarkar and Chaudhuri had proposed the differential box counting (DBC) method and have compared it with other conventional four methods in [17].

Consider an image of size $M \times M$ pixels. Let it be scaled down to a size $s \times s$ where $M/2 > s > 1$, where s is an integer. Then, $r = s/M$. Now consider the image to be in a 3D space with (x, y) denoting the spatial co-ordinates, while the z axis denotes the gray level. The (x, y) space is partitioned into grids of size s x s. On each grid there is a column of boxes of size $s \times s \times s'$. Figure 2 shows the schematic for computing FD using differential box counting method.

If the total number of gray level is G, then $[G/s'] = [M/s]$. Numbers from 1, 2, …are assigned to the boxes starting from the lowest gray level value. Let the minimum and the maximum gray level of the image in the $(i, j)^{th}$ grid fall in box number k and l, respectively. The contribution of Nr in $(i, j)^{th}$ grid is given by:

$$n_r(i, j) = l - k + 1 \tag{7}$$

Due to the differential nature in computing n_r this method is called differential box counting method. The contributions from all grids are found by :

$$N_r = \sum_{i,j} n_r(i, j) \tag{8}$$

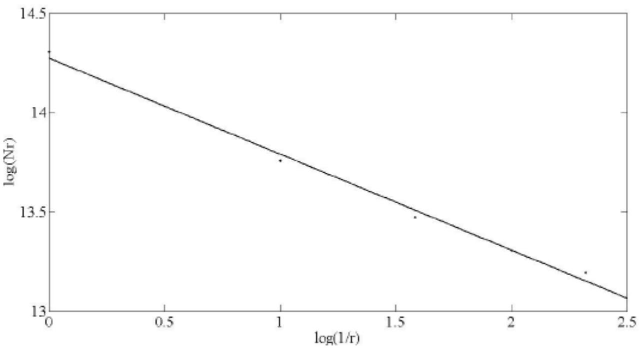

Fig. 2. Plot of $\log(N_r)$ versus $\log(1/r)$

N_r is computed for different values of s i.e. different values of r. Using equation (6) D, the fractal dimension can be estimated, from the least square linear fit of $\log(N_r)$ along $\log(1/r)$. The slope of the best fitting curve will give the fractal dimension. Figure 2 shows the plot of $\log(N_r)$ versus $\log(1/r)$ from which the FD is computed. A random placement of boxes is applied in order to reduce quantization effects.

c. *Fractal features*

In this paper the differential box counting method is used to calculate the FD and then different fractal features are derived from this fractal dimension which constituted the sample of observations used in the proposed approach.

Five features derived from [15, 18] based on fractal dimension are the FD of original image (f_1), high gray valued image (f_2), low gray valued image (f_3), horizontally smoothed image (f_4) and vertically smoothed image (f_5).

d. *Learning phase*

We try here to model the learning coefficient (cf. figure 3) by an exponential function which decreases towards zero when t increases such as :

$$a(t) = \alpha_1 \, e^{-\alpha_2 t} + \alpha_3 \tag{9}$$

Let itermax denotes the fixed number of iterations for the learning phase. By choosing the values 0.8 and 0.05 respectively as the maximum and the minimum values of $a(t)$ and that this coefficient reaches the values 0.1 in (2/3*itermax), this learning coefficient is given by :

$$a(t) = 0.75 * \exp\left(-(8.12) * (10^{-5}) * \mathrm{t}\right) + 0.05 \tag{10}$$

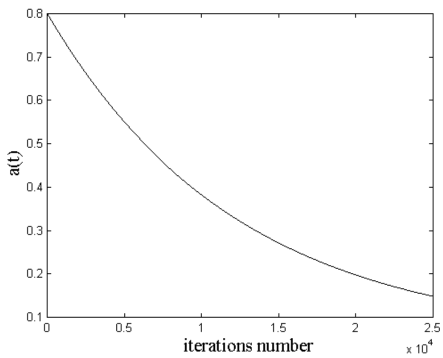

Fig. 3. Learning coefficient

The proximity radius $r(t)$ introduced in the interaction function $h(m^*, t)$ depends on the t^{th} iteration such as :

$$r(t) = \begin{cases} r(t-1) - 1 & \text{if} \quad \mathrm{t \bmod (n_r Q) = 0 \ and \ r(t) > 1} \\ \varepsilon & \text{if} \quad \mathrm{t \bmod (n_r Q) = 0 \ and \ r(t) \le 1} \\ r(t) & \text{otherwise} \end{cases} \tag{11}$$

This proximity radius decreases every $n_r Q$ iterations, where n_r is the epoque number with a constant radius and Q is the number of the observations in the sample. Note that one epoque corresponds to one scan of the total data involved in the learning process of the network. $x \bmod(y)$ denote the remainder after division of x by y.

In the learning phase, the observations are presented sequentially one by one to the network randomly and without putting back to be sure that in each epoch, all the observations were “learned” by the network.

e. Visualisation of the pdf on the Kohonen Map

This first step of the process concerns the self-organising and the learning of the network which permit to represent the Kohonen map. Once the learning phase is processed, the

ISBN: 978-988-19251-3-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

determined weight vectors in the multidimensional data space are used to estimate the underlying probability density function (pdf). For this purpose, we use the non-parametric Parzen estimate defined by [8][9] :

$$p(W_m) = \frac{1}{\mathrm{Q}} . \sum_{q=1}^{\mathrm{Q}} \frac{1}{V[D(W_m)]} . \Omega\left(\frac{\mathrm{W_m} - X_q}{\mathrm{h_Q}}\right) \qquad (12)$$

where :

- $\Omega(X) = \frac{1}{\sqrt{2\pi}} . \exp(-\frac{1}{2} \mathrm{X}^{\mathrm{T}} . \mathrm{X})$
- $D(W_m)$ is the domain estimation. When it corresponds to an sphere with $\mathrm{h_Q}$ radius centered in W_m,

$$V[D(W_m)] = \frac{\pi^{\frac{\mathrm{N}}{2}}}{\Gamma(\frac{\mathrm{N}}{2}+1)} . \mathrm{h_Q^N} \quad \text{with} \quad \Gamma(\frac{\mathrm{N}}{2}+1) = \frac{(\mathrm{N}+1)!\sqrt{\pi}}{2^{(\mathrm{N}+1)}(\frac{\mathrm{N}+1}{2})!} \qquad (13)$$

and $\mathrm{h_Q} = \mathrm{h_0} . \sqrt{\mathrm{Q}}$

The parameter $\mathrm{h_0}$ has a great effect on the quality of the estimation. If it is large, the little maxima of the pdf are indeterminable. Inversely, if $\mathrm{h_0}$ is too small, we obtain an estimation with many non significant maxima.

The visualisation of the pdf permits to display the Kohonen map as a digital image where each unit of the map is represented by a gray value pixel which corresponds to the pdf value. These pixel intensities permit to visualize the clusters frontiers. The visualisation of the pdf estimated with $\mathrm{h_0} = 0.02$ is displayed in figure 4 and figure 5. We can observe that the map is constitued by four regions where the pdf presents high values, separated by valleys where the pdf presents low values. We consider that a region is a set of connected pixels in the map with relatively higher values of the pdf.

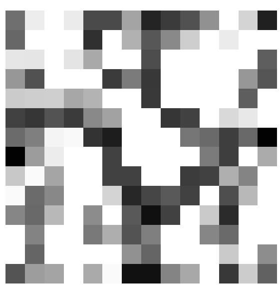

Fig. 4. Kohonen Map

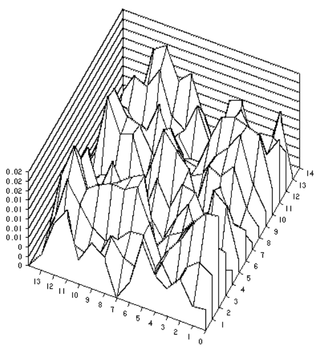

Fig. 5. pdf graph

This data projection method provides a planar display of the high dimensional data set. So, it can be assumed that clusters on the Kohonen map are images of clusters in the raw multidimensional data space, and we can usually analyse graphic displays without conscious use of any analytical model of clusters, or any mathematical decision rule. However this technique, used alone, doesn't allow an automatic data classification. To automate this process, and to give a powerful tool to detect, to extract and to give a number of the clusters from the Kohonen map, we propose to apply the watershed morphological transformation. The following step concerns the problem of modal regions detection in the Kohonen map.

III. Modal Regions Extraction with Watershed Technical

This principal phase of the procedure consist in the localisation of the modal regions of the underlying pdf by means of the watershed algorithm based on homotopic thinnings of the function [13].

Prior to mode detection, some kind of pre-processing is needed to enhance significant local variations of the density function. Opening operation, tends to smooth the function by filling up small holes and removing insignificant peaks in the function, while preserving the global shape of the function (cf. Figure 6) [13].

As we use the watershed approach, which is well suited for determining the catchment basins corresponding to the regional minima of a function [19], we introduce the additive inverse $f(W_m)$ of the function $p(W_m)$. Thanks to this simple transformation, The maxima of $p(W_m)$ become minima of $f(W_m)$.

The watershed of a function can be constructed through consecutive homotopic thinnings of this function. The homotopic thinning is a transformation commonly used in mathematical morphology for image skeletonization (Figure 7) [14]. In the watersheds resulting graph, modal regions of the pdf that are homotopic and geodesic extensions of regional maxima are separated by lines and are easily extracted in connected components through a simple morphological transformation.

Figure 8 shows the graph of the function resulting from this technical [20]. The same result is schematically represented by a binary representation of the Kohonen map, where modal regions of the pdf are well represented by an individualized connected components.

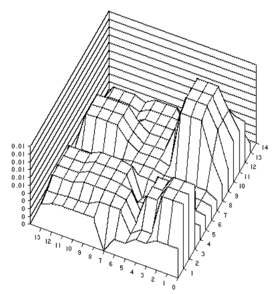

Fig. 6. Opening smoothing

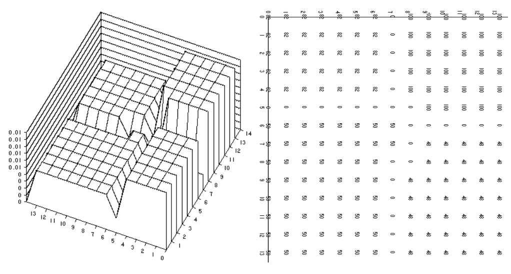

Fig. 7. pdf modal regions

ISBN: 978-988-19251-3-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

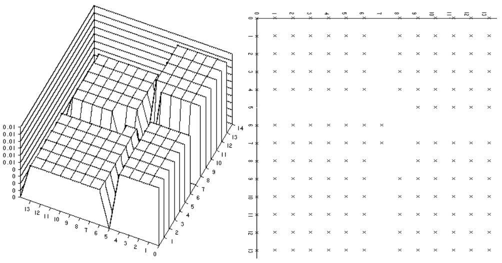

Fig. 8. Classes identified on the Kohonen map

IV. CLASSIFICATION RESULTS

Under the assumption that the detected modal regions correspond to homogenous regions in the image and a one class can be represented on the Kohonen map by one or more homogeneous regions in the image, the points constitute modal regions are considered as prototypes of classes present in the image (cf. Figure 9) and are the basis of the assignment of any pixel of the image to one of the classes, through the Euclidean distance on the map (cf. Figure 10).

Fig.9. Original image

Fig. 10. Classified Image

V. CONCLUSION

In this work, we proposed an approach to unsupervised classification of textured image, based on the combination of an algorithm of Mathematical morphology in a Kohonen map. In this approach, we represent at first the Kohonen map by the pdf underlying the sample of observations.

Modal regions of the pdf are then extracted into connected components by the watershed method which corresponds to a homogeneous region in the image. Finally, in classification phase, the weights vectors corresponding to the extracted modal region are taken as prototypes of classes present in the image, and are used for the assignment of each pixel in the image to one of the classes identified. This approach shows that in an unsupervised context, the tools of mathematical morphology associated with the Kohonen map allows a good automatic classification of the textured image without using any thresholding procedure. As perspective, we search to apply our approach on 3D image.

REFERENCES

[1] T. Uchiyama et M.-A. Arbib. "Color image Segmentation using competitive learning". *Pattern Analysis and Machine Intelligence*. Vol. 16, N 12, 1994.

[2] Tremeau, P. Golantoni and B. Laget, "On colour segmentation guided by the cooccurrence matrix", OSA Annual Conference on Optics and Imaging in the Information Age, pp. 30-38, 1996.

[3] A. Verikas, K. Malmqvist, L. Bergman, "Colour image segmentation by modular neural network", Pattern Recognition Letters, 1996.

[4] P. Scheunders, "A Genetic C-Means Clustering Algorithm applied to Color Image Quantization", *Pattern Recognition*, Vol. 30, N6, pp. 859-866, 1997.

[5] T. Géraud, P.-Y. Strub et J. Darbon, "Segmentation d'Imags Couleur par Classification Morphologique non supérvisée", *Proceeding of the International Conference on Image and Signal Processing ICISP'2001*, Agadir, Morocco, mai 2001.

[6] T. Kohonen, "Self-Organisation and Associative Memory", *Springer-Verlag, 2nd Edition*, New York, 1984.

[7] E. Claridge. "Shape Recognition using the Kohonen Self-Organising.Feature Map", *Pattern Recognition Letters*. March 1992.

[8] E. Parzen. "On Estimation of a probability density function and mode", *Ann. Math. Stat*., Vol.33 pp. 1065–1076, 1962.

[9] JP Asselin de Beauville, "l'estimation des modes d'une densité de probabilité multidimensionnelle", *Revue statistique et Analyse des Données*, Vol. 8, No 7, pp 16-40, 1983.

[10] JP Asselin de Beauville, "Panorama sur l'utilisation du mode en classification automatique". *RAIRO-APII, 23, pp* 113-137, 1989.

[11] S. Beucher, "Extrema of Gray Tone Fonctions and Mathematical Morphology". *Rapport du C.G.M.M., Ecole des Mines, Fontainebleau*, n°793, 1983.

[12] S. Beucher et C. Lantuejoul, "Use of Watersheds in Contour Detection". *Int. Workshop on Image Processing, CETT/IRISA*, Rennes, 1979.

[13] S. Beucher, "Segmentation Tools in Mathematical Morphology", *Image Algebra and Morphological Image Processing*, SPIE, Vol. 1350, pp. 70-84, 1990.

[14] Rosana Esteller, George Vachtsevanos, Javier Echauz, Brian Litt, "A Comparison of Waveform Fractal Dimension Algorithms", IEEE Trans. On Circuits and Systems -I: Fundamental Theory and Applications, Vol.48, No. 2, pp 177-183 Feb 2001.

[15] B. B. Chaudhuri , Nirupam Sarkar , "Texture Segmentation Using Fractal Dimension", IEEE Transactions on Pattern Recognition and Machine Intelligence, Vol. 17, No. 1, Jan 1995

16] R. Fazel-Rezai and W. Kinsner, "Texture analysis and segmentation of images using fractals," in Proc. IEEE Canadian Conf. Electrical Computer Engineering, May 9–12, 1999, pp. 786–791.

[17] Nirupam Sarkar, B. B. Chaudhuri, "An Efficient Differential Box-Counting Approach to Compute Fractal Dimension of Image", IEEE Trans. on Systems, Man and Cybernetics, Vol. 24, No. 1. Jan 1994, pp-115-120

[18] Harsh Potlapalli , Ren C. Luo, " Fractal-Based Classification of Natural Textures", IEEE Trans. On Industrial Electronics, Vol. 45, No. 1, pp 142-150,Feb 1998.

[19] S. Beucher, and C. Lantuejoul, " Use of Watersheds in Contour Detection " . Int. Workshop on Image Processing, CCETT / IRISA, Rennes, September (1979).

[20] T. M. Cover, and P.-E. Hart, " Nearest Neighbor Pattern Classification ", I.E.E.E. Trans. Info. Theory, vol. IT-13, n°1, pp. 21-27, (1967).

ISBN: 978-988-19251-3-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)