International Journal of Innovation and Scientific Research
ISSN 2351-8014 Vol. 9 No. 1 Sep. 2014, pp. 61-69

http://www.ijisr.issr-journals.org/

Printed Arabic Noisy Characters Recognition Using the Multi-layer Perceptron

R. SALOUAN, S. SAFI, and B. BOUIKHALENE

Department of Mathematic and Informatic, Polydisciplinary Faculty,
Sultan Moulay Slimane University, Beni Mellal, Morocco

ABSTRACT: In this paper, we present a comparison between two methods of features extraction; the first one is the Krawtchouk invariant moment (KIM). The second one is the Zernike invariant moment (ZIM). These moments are used for printed Arabic characters recognition in different situations: translated, rotated or resized and noisy. In the pre-processing phase we use the thresholding technique. In the learning-classification phase we use the multi-layer perceptron (MLP) that is considered as a neural network based on a supervised learning. The simulation result that we have obtained demonstrates that the KIM is more robust than ZIM in this recognition.

KEYWORDS: The noisy printed Arabic characters, the thresholding technique, the Krawtchouk invariant moments, the Zernike invariant moments, the multi-layer perceptron.

1 INTRODUCTION

Pattern recognition is a scientific domain whose goal is to classify the objects into a number of classes. Depending on the application, these objects can be images of characters, of numerals of faces or any type of measurements that need to be classified. It fact, it is composed from tree principal phases that are:

1. Pre-processing: In this phase, different techniques can be used such as binarization, noise removal in order to improve the quality of input patterns.
2. Features extraction: unique features are extracted from input patterns, which must discriminate it in an particular manner.
3. Classification: In this phase, the input pattern will be recognized.

On the other hand, many studies have been carried on Latin, Arabic numerals and characters by using the multi-layer perceptron [1-6] and the moments [7-12]. However, our study is focused in Arabic characters recognition.

In this study the pre-processing characters is carried by the thresholding technique. In the phase of features extraction from character image the KIM [13] and the ZIM[14] which are used to convert each image of character to a vector that will used as an input vector of MLP that are used to train the character images of the training database and to classify them in the test database. The last phase takes place as follows:

The weight matrix connecting the input and hidden layer of the network and that connecting the hidden layer and the output of the network should be learned by the back propagation algorithm for giving in the output of the network an identity matrix of order 28 (supervised learning) which is nothing other than the total number of Arabic characters. All the matrixes of connection (optimal matrixes) must be saved for forming a learning base.

In the classification a test vector (unknown character) is multiplied by the first optimal matrix for giving a vector which is multiplied by the second optimal matrix for producing another vector then the Euclidean distance is calculated between this last vector and each column vector of the identity matrix of order 28, the recognition will be attributed to character that numbered between 1 and 28 in the columns of identity matrix and has a smallest Euclidean distance to test character.

2 THE RECOGNITION SYSTEM

Our recognition system is presented as follow:

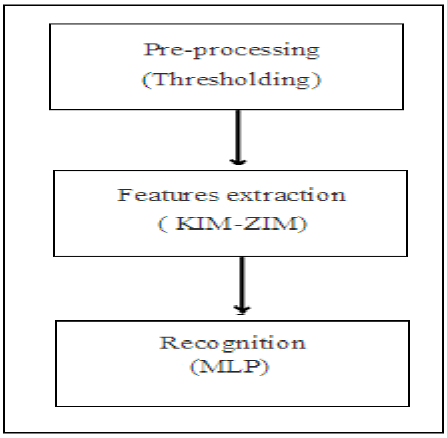

Fig. 1: System for Arabic characters recognition.

3 PREPROCESSING

Pre-processing is the first part of Arabic characters recognition system. This is used to produce a cleaned up version of the original image so that it can be used efficiently by the features extraction phase.

In our study, we preprocess the images by a thresholding technique in order to construct the images containing only the black and white colors according a preset threshold.

4 FEATURES EXTRACTION

The second phase of the printed Arabic characters recognition system is features extraction. Several methods can be used to compute the features. In this recognition system, we use the Krawtchouk invariant moments (KIMs) and those of Zernike (ZIMs).

4.1 THE KRAWTCHOUK MOMENT

4.1.1 THE KRAWTCHOUK POLYNOMIAL

The Krawtchouk polynomial of order n is given by:

$$K_n(x;p,N)=\sum_{k=0}^{N} a_{k,n,p}x^k={}_2F_1(-n,-x,-N;\frac{1}{p}) \quad (1)$$

Where $: x,n = 0,1,2\ldots N,\ N \succ 0,\ p \in [0,1].$

${}_2F_1$ is the hyper geometric function defined as :

$${}_2F_1(a,b;c;x) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{x^k}{k\,!} \quad (2)$$

And $(a)_k$ is the pochhammer symbol (called also rising factorial) defined by:

$$(a)_k = a(a+1)\ldots(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)} \quad (3)$$

The Γ function is defined by:

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt \quad (4)$$

$$And \quad : \forall n \in N , \Gamma(n+1) = n!$$

The set of $(N+1)$ Krawtchouk polynomial $\{k_n(x;p,N)\}$ forms a complete set of discrete basis functions with the weight function:

$$w(x;p,N) = \binom{N}{x} p^x (1-p)^{N-x} \quad (5)$$

And satisfies the orthogonality condition:

$$\sum_{x=0}^{N} w(x;p,N) K_n(x;p,N) K_m(x;p,N) = \rho(n;p,N)\delta_{nm} \quad (6)$$

$Where \quad : m, n = 0,1,2 \ldots N , and \ \rho(n;p,N)$ is the squared norm, which is given by:

$$\rho(n;p,N) = (-1)^n \left(\frac{1-p}{p}\right)^n \frac{n!}{(-N)_n} \quad (7)$$

And δ_{nm} is the Kronecker symbol defined by:

$$\delta_{nm} = \begin{cases} 1 & if \ . n = m \\ 0 & others \end{cases} \quad (8)$$

4.1.2 THE KRAWTCHOUK MOMENT

The Krawtchouk moment have the interesting property of being able to efficiently extract local features of an image this moment of order $(n+m)$ of an image $f(x,y)$ is given by:

$$Q_{nm} = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \bar{K}_n(x;p_1,N-1)\bar{K}_m(y;p_2,M-1) f(x,y) \quad (9)$$

The NxM is the number of pixels of an image $f(x,y)$ .The set of weighted Krawtchouk polynomials $\bar{K}_n(x;p,N)$ is:

$$\bar{K}_n(x;p,N) = K_n(x;p,N)\sqrt{\frac{w(x;p,N)}{\rho(x;p,N)}} \quad (10)$$

4.1.3 THE KRAWTCHOUK INVARIANT MOMENT

The geometric moment of an image $f(x,y)$ is given by:

$$M_{nm} = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} x^n y^m f(x,y) \quad (11)$$

The standard set of the geometric invariant moments that are independent to rotation, scaling, translation is:

$$V_{nm} = M_{00}^{-\gamma} \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} [(x-\bar{x})\cos\theta + (y-\bar{y})\sin\theta]^n * [(y-\bar{y})\cos\theta (x-\bar{x})\sin\theta]^m f(x,y) \quad (12)$$

$$Where : \gamma = \frac{n+m}{2} + 1, \ \bar{x} = \frac{M_{10}}{M_{00}}, \ \bar{y} = \frac{M_{01}}{M_{00}} \ and \ : \theta = \frac{1}{2} arctg \frac{2\mu_{11}}{\mu_{20} - \mu_{02}}$$

And μ_{nm} are the central moments defined by:

$$\mu_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} (x - \bar{x})^n (y - \bar{y})^m f(x, y) \quad (13)$$

The Krawtchouk invariant moment is:

$$\tilde{\Omega}_{nm} = \Omega_{nm} \sum_{i=0}^{n} \sum_{j=0}^{m} a_{i,n,p_1} a_{j,m,p_2} \tilde{V}_{ij} \quad (14)$$

$$\Omega_{nm} = [\rho(n; p_1, N-1).\rho(m; p_2, M-1)]^{-1/2} \quad (15)$$

$$With : \tilde{V}_{ij} = \sum_{p=0}^{i} \sum_{q=0}^{j} \binom{i}{p}\binom{j}{q} (\frac{N^2}{2})^{\frac{p+q}{2}+1} (\frac{N}{2})^{i+j-p-q} V_{pq} \quad (16)$$

$$And \quad : \binom{x}{y} = \frac{x!}{y!(x-y)!} \quad (17)$$

4.2 THE ZERNIKE MOMENT

For an image $f(x, y)$ The Zernike moment of order n and repetition m is given by:

$$A_{nm} = \frac{n+1}{\pi} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) V^*(x, y) \quad (18)$$

$$V^*(x, y) = R_{nm}(x, y) e^{jm \arctan(y/x)} \quad (19)$$

$$And : R_{nm}(x, y) = \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s (x^2 + y^2)^{\frac{n}{2}-s} (n-s)!}{s!(\frac{n+|s|}{2} - s)!(\frac{n-|s|}{2} - s)!} \quad (20)$$

$$if \ : n - |m| \ is \ even, \ n \geq |m|, n \geq 0 \quad , j = \sqrt{-1}$$

And $x^2 + y^2 \leq 1$, the symbol $*$ denotes the complex conjugate operator.

4.2.1 THE ZERNIKE INVARIANT MOMENT

The Zernike moment is invariant under rotation but sensitive to translation and scale. So normalization must be done of these moments.

$$f(x, y) = f(\bar{x} + \frac{x}{a}, \bar{y} + \frac{y}{a}) \quad (21)$$

Where $(\bar{x}, \bar{y})$ is the center of pattern function $f(x,y)$ and $a = (\beta / M_{00})^{1/2}$, β is a predetermined value for the number of object points in pattern.

5 CHARACTER RECOGNITION

5.1 THE NEURAL NETWORK MULTI-LAYER PERCEPTRON

The Neural Network [15] presented in this figure is a multi-layer perceptron that we have used in our work.

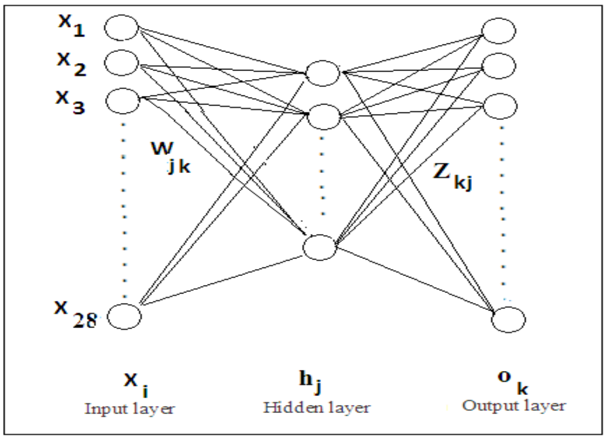

Fig. 2: The multi-layer perceptron (MLP).

This MLP contains the following elements:

- An input layer of 28 vectors, each has a 10 components (KIM vector :X_i).

- A hidden layer of 14 activations neural h_j.

- An output layer of 28 activations Neural o_k.

- 28×14 connections between input layer and hidden layer, each weighted by W_{jk}.

- 14×28 connections between hidden layer and output layers, each weighted by Z_{kj}

The operation of perceptron multi-layer learning contains in fact 5 phases:

- **Phase 1**: (Initializing randomly weights of connexions W and Z),

- **Phase 2**: (propagation of inputs):

The inputs X_i must be presented to input layer

We propagate to the hidden layer:

$$h_j = f(\sum_{i=1}^{28} X_i W_{ji}) \quad (22)$$

After for hidden layer to output layer:

$$Z_k = f(\sum_{j=1}^{10} h_j Z_{kj}) \quad (23)$$

f is the activation function which is the sigmoid or logistic function given by:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (24)$$

- **Phase 3**: (Error back propagation)

For each character of learning base of the MLP, we calculate the error at output layers that is the difference between the desired output S_k and o_k real output:

$$E_k = o_k(1 - o_k)(S_k - o_k) \quad (25)$$

Next, we propagate this error on the hidden layer; the error of each neuron of the hidden layer is given by:

$$F_j = o_j(1 - o_j)\sum_{k=1}^{28} Z_{kj} E_k \quad (26)$$

- **Phase 4**: (Correction of connections weights):

Later, we change the weights of connections:

- Between input layer and hidden layer:

$$\Delta W_{ji} = \alpha X_i F_j \quad (27)$$

- Between hidden layer and output layer:

$$\Delta Z_{kj} = \alpha Y_j E_k \quad (28)$$

Where α is the learning rate comprised between 0 and 1. This is experimentally determined (α =0.95).

- **Phase 5**:

After the learning of MLP .We use the Euclidean distance for identifying the test character.

$$d(S_{ki}, o_{ki}) = (\sum_{i=1}^{28} (S_{ki} - o_{ki})^2)^{1/2} \quad (29)$$

Where S_k is a desired output and o_k is the real output of network.

6 EXPEREMENTS AND RESULTS

In this study we used a learning base contains 140 images which represents the Arabic characters.

In the test base we have 4340 images.

We choose the size of all character images 30x30 pixels. Each character was converted to a vector of 10 components which is the KIM and ZIM values. In first once we present a test character translated, rotated or resized and not noisy, then we add increasingly a quantity of noise of type 'Gaussian' for to know the effect of noise added on the rate recognition of each character then to global rate that is to say of all characters.

We choose: The KIM parameters: p=0, 85, q=0, 75.

The values of variance σ of Gaussian noise are: [0, 0.01, 0.02, 0.03……0.30].

And its mean value is fixed to μ=0.05.

We group the values of the recognition rate τ_C for each character that we obtained in the following table:

Table 1: The recognition rate τ_C for each Arabic character

Character	$\tau_{C,ZIM}$	$\tau_{C,KIM}$	Character	$\tau_{C,ZIM}$	$\tau_{C,KIM}$
أ	58,06%	100,0%	ض	100,0%	100,0%
ب	96.77%	100,0%	ط	100,0%	100,0%
ت	70,97%	70,97%	ظ	93,55%	96,77%
ث	87,10%	93,55%	ع	96,77%	87,10%
ج	67,74%	64,55%	غ	93,55%	90,32%
ح	100,0%	90,32%	ف	96,77%	100,0%
خ	83,87%	93,55%	ق	87,10%	90,32%
د	87,10%	96,77%	ك	100,0%	100,0%
ذ	93,55%	90,32%	ل	100,0%	93,55%
ر	96,77%	100,0%	م	90,32%	87,10%
ز	100,0%	100,0%	ن	100,0%	100,0%
س	58,06%	61,29%	ه	93,55%	96,77%
ش	100,0%	100,0%	و	96,77%	100,0%
ص	96,77%	100,0%	ي	90,32%	100,0%

The associated graph to table above is:

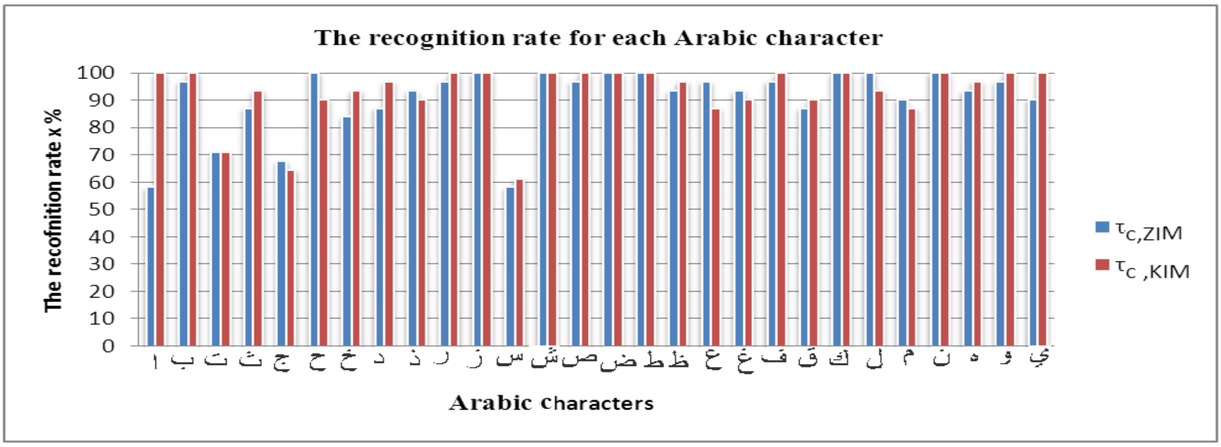

Fig. 3: The recognition rate τ_c for each Arabic character.

We present the evolution of the global rate recognition τ_g in function of noise added to characters:

Table2: The global rate recognition in function of the noise added.

Noise	$\tau_{g,ZIM}$	$\tau_{g\,,KIM}$
0.00	100,0%	100,0%
0.01	100,0%	100,0%
0.02	100,0%	100,0%
0.03	100,0%	100,0%
0.04	100,0%	100,0%
0.05	100,0%	100,0%
0.06	100,0%	100,0%
0.07	100,0%	100,0%
0.08	100,0%	100,0%
0.09	100,0%	100,0%
0.10	100,0%	100,0%
0.11	100,0%	100,0%
0.12	100,0%	100,0%
0.13	100,0%	100,0%
0.14	100,0%	100,0%
0.15	100,0%	100,0%
0.16	100,0%	100,0%
0.17	92,86%	100,0%
0.18	92,86%	96,43%
0.19	89,29%	92,86%
0.20	85,21%	92,86%
0.21	85,21%	89,29%
0.22	85,21%	89,29%
0.23	85,21%	89,29%
0.24	82,14%	89,29%
0.25	71,43%	89,29%
0.26	64,29%	82,14%
0.27	50,00%	67,86%
0.28	28,57%	57,14%
0.29	28,57%	46,43%
0.30	28,57%	46,43%

The associated graph to table below is:

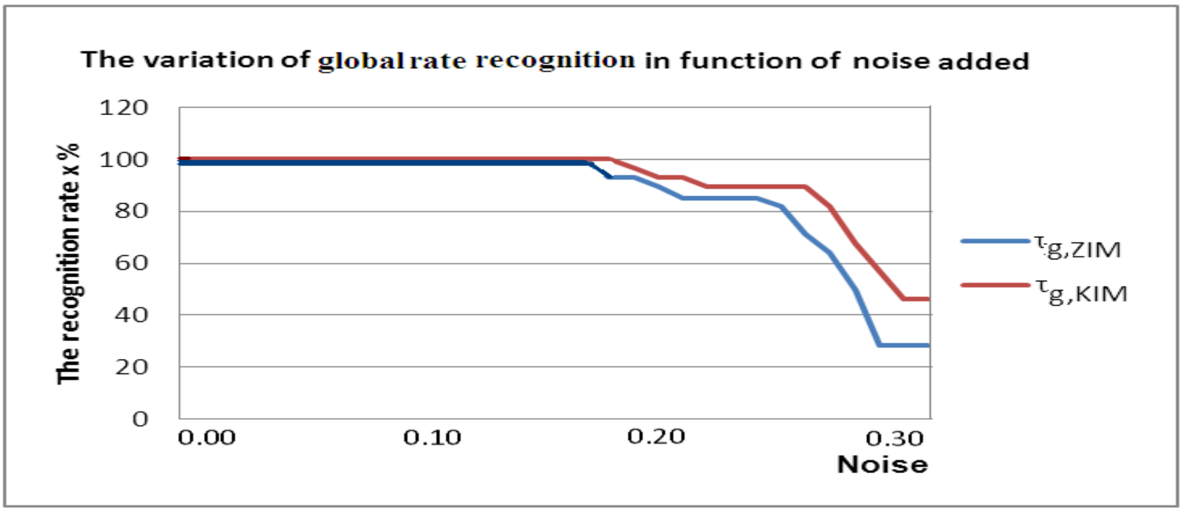

Fig. 4: The global rate recognition τ_g Arabic characters.

6.1 ANALYSIS AND COMMENT

The global rate recognition τ_g is a decreasing in function of noise dded to characters, but the important remark is that the falling of this rate of ZIM is greater than the rate of KIM, this shows that the KIM is more performing than the ZIM in recognition of noisy characters.

7 CONCLUSION

The results obtained in the recognition of noisy Arabic characters show that reliable recognition is possible by using the thresholding technique in the preprocessing phase and the KIM and the ZIM in the features extraction phase. The simulation results demonstrate that the KIM method is more robust against noise than the ZIM in this recognition.

REFFERENCES

[1] Kussul E.M., Baidyk T.N., et al., (2001) Rosenblatt perceptrons for handwritten digit recognition, Proceedings of International Joint Conference on Neural Networks IJCNN 2 1516-1520.
[2] A. A. Desai Gujarati handwritten numeral optical character reorganization through neural network. Pattern Recognition 43 (2010) 2582-2589
[3] Anuja P. Nagare. "License Plate Character Recognition System using Neural Network. " International Journal of Computer Applications, Volume 25, No. 10, July 2011, pp. 36-39.
[4] Christopher M.Bishop” Neural Networks For Pattern Recognition ” Department of computer Science and applied mathematics Aston University , Birmingham, UK Clarndon press, 1995.
[5] A. A. Desai Gujarati handwritten numeral optical character reorganization through neural network. Pattern Recognition 43 (2010) 2582-2589
[6] Anuja P. Nagare. "License Plate Character Recognition System using Neural Network. " International Journal of Computer Applications, Volume 25, No. 10, July 2011, pp. 36-39.
[7] Ibrahim A. El rube', Mohamed T. El Sonni, and Soha S. Saleh ,printed Arabic sub-word recognition using moments, World Academy of Science, Engineering and Technology 42 2010
[8] Gheith Abandah, Nasser Anssari: novel moment features Extraction for Recognizing Handwritten Arabic Letters”, Journal of Computer Science 5 (3): 226-232, 2009, ISSN 1549-3636.
[9] R.Muralidharan1, C. Chandrasekar: Object Recognition using SVM -KNN based on Geometric Moment Invariant, International Journal of Computer Trends and Technology- July to Aug Issue 2011, ISSN: 2231-2803, Page 215-220.
[10] Anass El affar Khalid Ferdous, Abdeljabbar Cherkaoui Hakim El fadil, Hassan Qjidaa: Krawtchouk Moment Feature Extraction for Neural Arabic Handwritten Words Recognition,. IJCSNS International Journal of Computer Science and Network Security, VOL.9No.1,January 2009.

[11] S.F. Bahgat, S.Ghomiemy, S. Aljahdali, M. Alotaibi, A proposed hybrid technique for recognizing arabic characters (IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 1, No. 4, 2012.
[12] P.Nagabhushan, S.A.Angadi, B.S.Anami, afuzzy statistical approach of Kannada vowel recognition based on invariant moments, NCDAR-2003, Mandy, India , pp275-285, 2003
[13] Pew-Thian Yap, Raveendran Paramesran, Senior Member IEEE, and Seng-Huat On Image analysis by Krawtchouk moments, IEEE transactions on image processing, vol. 12, NO. 11, November 2003
[14] R. Mukundan and K.R. Ramakrishnan, Moment Functions in Image Analysis- Theory and Applications, World Scientific Publishing, 1998.
[15] Dreyfus et. al (2001) : Réseaux de neurones. Méthodologie et applications. Eyrolles. 40.