International Journal of Computer Engineering Science (IJCES)
Volume 2 Issue 3 (March 2012) ISSN : 2250:3439
https://sites.google.com/site/ijcesjournal http://www.ijces.com/

Zernike moments and neural networks for recognition of isolated Arabic characters

Mustapha Oujaoura[1] , Rachid El Ayachi[1], Mohamed Fakir[1], Belaid Bouikhalene[1], and Brahim Minaoui[1],

[1] Laboratory of Information Processing and Telecommunication, Department of Computer Science, Faculty of Science and Technology, Sultan Moulay Slimane University.
PO Box. 523, Béni Mellal, Morocco
{oujaouram, rachidieea, fakfad, bbouikhalene, bra_min}@yahoo.fr

Abstract. The aim of this work is to present a system for recognizing isolated Arabic printed characters. This system goes through several stages: preprocessing, feature extraction and classification. Zernike moments, invariant moments and Walsh transformation are used to calculate the features. The classification is based on multilayer neural networks. A recognition rate of 98% is achieved by using Zernike moments.

Keywords: Character recognition, Zernike moments, invariant moments, Walsh transformation and neural networks.

1 Introduction

The Optical Character Recognition (OCR) is a field of pattern recognition that focuses on character shapes. The goal is to assign an identifier to a form of reference prototypes.

Research works in optical Arabic characters recognition had a great expansion in recent years. Nemouchi and Farah [1] presented an hybrid approach for the development of an offline Arabic handwriting recognition system. Hachour [2] proposed a system for recognition of Arabic characters dedicated to the automatic reading of characters. Touj and al [3] used the generalized Hough transform for the recognition of Arabic printed script.

In this work, the objective is to recognize isolated Arabic printed characters using the system shown in Figure (Fig. 1).

Fig. 1. Recognition system of Arabic characters.

In the preprocessing phase, normalization is applied to remove unwanted areas using the method of histogram; In this phase, we first calculate the horizontal and vertical histograms, then the histogram is scanned horizontally in both directions: respectively from the top to the bottom and from the bottom to the top until finding the first black pixels, thereafter, the vertical histogram is traversed in both directions: respectively from the left to the right and from the right to the left until finding the first black pixels. Finally, after determining the positions of the first black pixels, we eliminate unwanted areas.
The feature extraction phase uses Zernike and invariants moments for calculating the parameters of image.
In the classification phase, neural networks are adopted due to their effectiveness in the recognition field.

2 Feature extraction

The feature extraction is the second phase to be applied in the Optical Character Recognition (OCR); it plays a very important role for the character recognition, as it must consider the representation of the character in some situations such as: translation, rotation and change of scale. This is the reason that justifies the use of Zernike moments [6] and invariant moments [7] in the system treated in this work. Also, the Walsh transform [8] gives a detailed description of the image and it has been applied in several cases for various purposes, but little used for character recognition.

2.1 Zernike Moments

Zernike moments are constructed using a set of complex polynomials which form a complete orthogonal set on the unit disk with $(x^2 + y^2) = 1$:

$$Z_{mn} = \frac{m+1}{\pi} \int_x \int_y I(x,y)[V_{mn}(x,y)]dxdy \qquad (1)$$

Where m and n define the order of moment and $I(x,y)$ the gray level of a pixel of image I on which the moment is calculated.
The Zernike polynomials $V_{mn}(x,y)$ are expressed in polar coordinates as follows:

$$V_{mn}(r,\theta) = R_{mn}(r)e^{-jn\theta} \qquad (2)$$

Where $R_{mn}(r)$ is the orthogonal radial polynomial:

$$R_{mn}(r) = \sum_{s=0}^{\frac{m-|n|}{2}} (-1)^s \frac{(m-s)!}{s!\left(\frac{m+|n|}{2}-s\right)!\left(\frac{m-|n|}{2}-s\right)!} r^{m-2s} \quad \textbf{(3)}$$

Moments Z_{mn} are invariant under rotation and scale changes.

2.2 Invariants Moments

Let f be a function defined by: $f(x, y) = 1$ on a closed and bounded region R and $f(x, y) = 0$ elsewhere.

We define the moment of order (p,q) as follows:

$$m_{pq} = \iint_R x^p y^q f(x, y)dxdy \quad \text{, for } p,q = 0,1,2,... \quad \textbf{(4)}$$

The central moments can be expressed by:

$$\mu_{pq} = \iint_R (x-\overline{x})^p (y-\overline{y})^q f(x, y)dxdy \quad \text{with } \overline{x} = \frac{m_{10}}{m_{00}} \text{ , } \overline{y} = \frac{m_{01}}{m_{00}} \quad \textbf{(5)}$$

For a digital image, the equation (5) becomes:

$$\mu_{pq} = \sum\sum_{(x,y)\in R} (x-\overline{x})^p (y-\overline{y})^q f(x, y) \quad \textbf{(6)}$$

One can easily verify that the central moments of order $p+q \leq 3$ can be calculated by the following formulas:

$$\begin{aligned}
&\mu_{00} = m_{00} \\
&\mu_{10} = 0 \quad , \quad \mu_{01} = 0 \\
&\mu_{11} = m_{11} - \overline{y}m_{10} \\
&\mu_{20} = m_{20} - \overline{x}m_{10} \quad , \quad \mu_{02} = m_{02} - \overline{y}m_{01} \\
&\mu_{12} = m_{12} - 2\overline{y}m_{11} - \overline{x}m_{02} + 2\overline{y}^2 m_{10} \\
&\mu_{21} = m_{21} - 2\overline{x}m_{11} - \overline{y}m_{20} + 2\overline{x}^2 m_{01} \\
&\mu_{30} = m_{30} - 3\overline{x}m_{20} + 2\overline{x}^2 m_{10} \\
&\mu_{03} = m_{03} - 3\overline{y}m_{02} + 2\overline{y}^2 m_{01}
\end{aligned} \quad \textbf{(7)}$$

International Journal of Computer Engineering Science (IJCES)
Volume 2 Issue 3 (March 2012) ISSN : 2250:3439
https://sites.google.com/site/ijcesjournal http://www.ijces.com/

The central moments are invariant by translation. They can be normalized to preserve the invariance by scaling and we obtain the normalized central moments:

$$\alpha_{pq} = \frac{\mu_{pq}}{\mu_{00}{}^{\gamma}} \text{ , with } \gamma = \frac{p+q}{2} + 1 \text{ , for } p+q = 2,3,... \tag{8}$$

The following invariants moment were obtained by Hu (1962) and frequently used as features for pattern recognition.

$$\begin{aligned}
\varphi_1 &= \alpha_{20} - \alpha_{02} \\
\varphi_2 &= (\alpha_{20} - \alpha_{02})^2 + 4\alpha_{11}^2 \\
\varphi_3 &= (\alpha_{30} - \alpha_{12})^2 + (3\alpha_{12} - \alpha_{03})^2 \\
\varphi_4 &= (\alpha_{30} + \alpha_{12})^2 + (\alpha_{21} + \alpha_{03})^2 \\
\varphi_5 &= (\alpha_{30} - 3\alpha_{12})(\alpha_{30} + \alpha_{12})[(\alpha_{30} + \alpha_{12})^2 - 3(\alpha_{21} + \alpha_{03})^2] \\
&\quad + (3\alpha_{21} - \alpha_{03})(\alpha_{21} + \alpha_{03})[3(\alpha_{30} + \alpha_{12})^2 - (\alpha_{21} + \alpha_{03})^2] \\
\varphi_6 &= (\alpha_{20} - \alpha_{02})[(\alpha_{30} + \alpha_{12})^2 - (\alpha_{21} + \alpha_{03})^2] \\
&\quad + 4\alpha_{11}(\alpha_{30} + \alpha_{12})(\alpha_{21} + \alpha_{03}) \\
\varphi_7 &= (3\alpha_{21} - \alpha_{30})(\alpha_{30} + \alpha_{12})[(\alpha_{30} + \alpha_{12})^2 - 3(\alpha_{21} + \alpha_{03})^2] \\
&\quad + (3\alpha_{12} - \alpha_{03})(\alpha_{21} + \alpha_{03})[3(\alpha_{30} + \alpha_{12})^2 - (\alpha_{21} + \alpha_{03})^2]
\end{aligned} \tag{9}$$

Hu has shown that these quantities $\varphi_i, (1 \le i \le 7)$ are invariant under scaling, translation and rotation.

2.3 Walsh Transformation

The Walsh transformation W (u, v) can be calculated using the following formula:

$$W(u,v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) g(x,y,u,v) \tag{10}$$

Where f(x, y) is intensity of the pixel with the coordinates (x, y) in the original binary image, u, v = 0, 1… N-1. The size of the image f is N x N, and function g is the kernel function of the transformation and has the following form:

$$g(x,y,u,v) = (1/N) \prod_{i=0}^{n-1} (-1)^{b_i(x) b_{n-i-1}(u) + b_i(y) b_{n-i-1}(v)} \tag{11}$$

Where $b_i(x)$ is the ith bit in the binary expansion of x (so it is equal either 0 or 1), and N=2^n.

The Walsh transform is unique in the sense that if we consider two different binary images, the corresponding feature vectors are also different.
In our case, since the Walsh transformation is invariant under size changes, to perform and calculate the Walsh transformation, first we have to change the original image to the size of $2^n * 2^n$ for n=8.

3 Neural Networks

The figure (Fig. 2) represents an example of the neural network used in this paper; it is a multilayer neural network that contains a hidden layer.

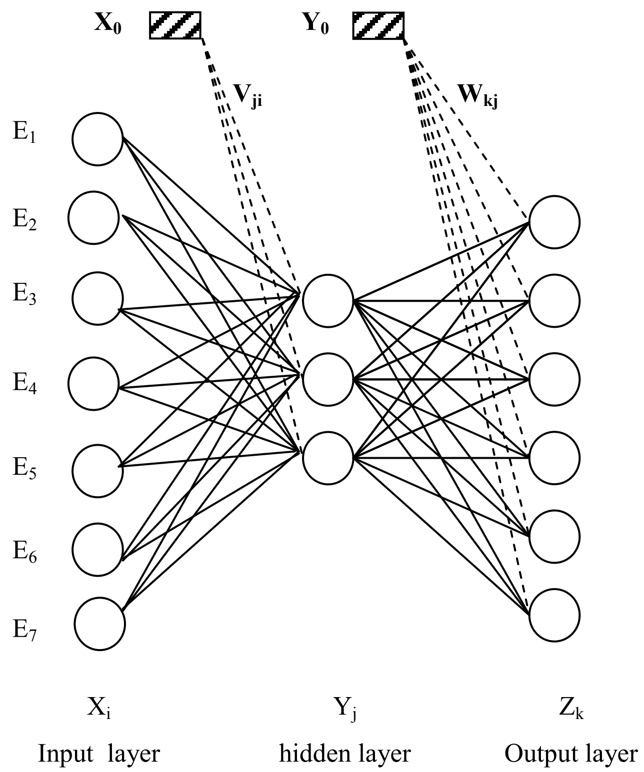

Fig. 2. Neural network in the case of using invariant moments.

This neural network has:
- An input layer with 7 input cells (vector of invariant moment) $E_i = X_i$
- A hidden layer of 3 activation neurons

- An output layer of 6 activation neurons
- 7 × 3 connections between the input and the hidden layers, each weighted by V_{ji}
- 3 × 6 connections between the hidden and the output layers, each weighted by W_{kj}
- X_0 et Y_0 are scalars.

The operating principle of the neural network (Fig. 2) is based on a set of steps:

Step 1: (Initialization of the connection weights)
The weights are taken randomly.

Step 2: (Propagation of the inputs)
The inputs E_i are presented to the input layer: $X_i = E_i$
Propagation to the hidden layer is done using the following formula:

$$Y_j = f\left(\sum_{i=1}^{7} X_i V_{ji} + X_0\right) \tag{12}$$

Then from the hidden layer to the output layer:

$$Z_k = f\left(\sum_{j=1}^{3} Y_j W_{kj} + Y_0\right) \tag{13}$$

X_0 and Y_0 are scalars.
f is the activation function (sigmoid function) :

$$f(a) = \frac{1}{1 + \exp(-a)} \tag{14}$$

Step 3: (Backpropagation of error)
At the output layer, the error between the desired output and the actual output is calculated by:

$$E_k = Z_k (1 - Z_k)(S_k - Z_k) \tag{15}$$

The calculated error is propagated to the hidden layer using the following formula:

$$F_j = Y_j (1 - Y_j) \sum_{k=1}^{6} W_{kj} . E_k \tag{16}$$

Step 4: (Adjusting the connections weights)
The connections weights between the input layer and the hidden layer are adjusted by:

$$\Delta V_{ji} = \eta . X_i . F_j \quad \text{and} \quad \Delta Y_0 = \eta . F_j \tag{17}$$

Then, the connections weights between the hidden layer and output layer are changed by:

$$\Delta W_{kj} = \eta . Y_j . E_k \quad \text{and} \quad \Delta X_0 = \eta . E_k \tag{18}$$

η a parameter to be determined empirically.

Step 5: (loop)
Loop back to step 2 until a defined stopping criterion (error threshold, maximu number of iterations).

After learning and performing the OCR, the Euclidean distance is used to identify characters:

$$d(t_k, o) = \left(\sum_{i=1}^{6} (t_{ki} - o_i)^2 \right)^{1/2} \tag{19}$$

With, t_k the desired output and o the output of the neural network.
However, the input layer of the neural network has nine input cells in the case of adoption of Zernike moments as a features extraction method and it has forty nine input cells when Walsh transformation is used as feature extraction method.

4 Results

The image of the figure (Fig. 3) represents the above mentioned reference character used in this recognition system.

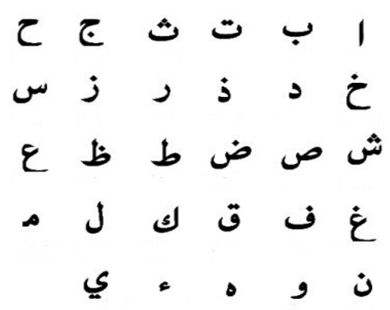

Fig. 3. Reference character.

The Graphical User Interface is illustrated in figure (Fig. 4)

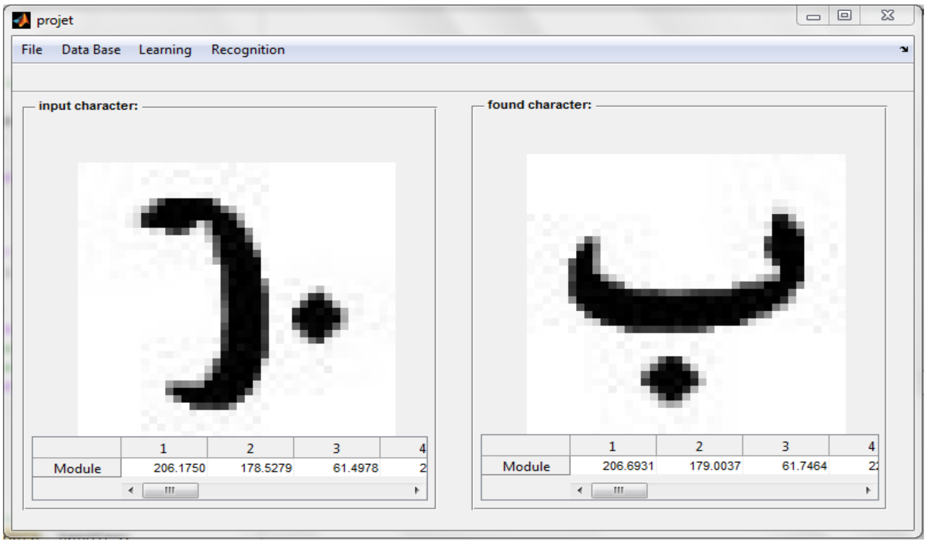

Fig. 4. Graphical User Interface (GUI).

The comparison between the Zernike moments, invariant moments and Walsh transformation, used as feature extraction methods, is illustrated in the following table:

Table 1. Recognition and error rates.

feature extraction methods	Recognition rate	Error rate
Zernike Moments	98.27%	1.73%
Invariant Moments	94.82%	5.18%
Walsh Transformation	93.11%	6.89%

The results in Table 1 show that:

- The computed recognition rate of Zernike moments is greater than the recognition rate found by invariant moments and Walsh Transformation.
- The error rate obtained using Zernike moments is less than the error rate calculated in the case of using invariant moments and Walsh Transformation.

All the tests are performed using database containing 116 images. The proposed system has been implemented and tested on a core 2 Duo personnel computer using a Matlab software.

5 Conclusion

The system developed in this work is an offline recognition system of isolated Arabic printed characters. In the preprocessing, we applied the normalization to eliminate unwanted areas and reduce the execution time. Then, the Zernike moments, invariant moments and Walsh Transformation are used to calculate the parameters in the feature extraction phase. Finally, we adopted neural networks as a method of classification.

References

1. S. Nemouchi and N. Farah, « Reconnaissance de l'Ecriture Arabe par Systèmes Flous », Département Informatique, Université Badji Mokhtar Laboratoire de Gestion Electronique du Document (LABGED), Algérie.
2. O. Hachour, « Reconnaissance hybride des caractères arabes imprimés », JEP-TALN 2004, Traitement Automatique de l'Arabe, Fès, 20 avril 2004, Maroc.
3. S. Touj, N.E. Ben Amara et H. Amiri, « Reconnaissance de l'écriture Arabe imprimée par transformée de Hough généralisée », Laboratoire des systèmes et de traitement du signal-ENIT, Tunisie.
4. S. Haitaamar, « Segmentation de textes en caractères pour la reconnaissance optique de l'écriture arabe », thèse, 2007, Université El Hadj-Lakhdhar Batna, Algérie.
5. H. Miled, C. Olivier, M. Cheriet, K. Romeo-Pakker, « Une Méthode Rapide de Reconnaissance de l'Écriture Arabe Manuscrite », Seizième Colloque Gretsin, 15-19 Septembre 1997, Grenoble.
6. A. Choksuriwong, H. Laurent et B. Emile, « Etude Comparative de Descripteurs Invariants d'Objets », ENSI de Bourges - Université d'Orléans, Laboratoire Vision et Robotique - UPRES EA 2078, France.
7. R. El Ayachi, M. Fakir et B. Bouikhalene, « Transformation de Fourier et Moments Invariants Appliqués à la Reconnaissance des Caractères Tifinaghe », SITACAM'11, Agadir-Maroc, 06-07 Mai 2011.
8. Attila Fazekas and Andr´as Hajdu, «Recognizing Typeset Documents using Walsh Transformation», Journal of Computing and Information Technology - CIT 9, 2001, 2, 101–112.