American Journal of Applied Sciences 4 (11): 857-864, 2007
ISSN 1546-9239

Automatic Arabic Hand Written Text Recognition System

Ismael Ahmad Jannoud
Damascus University, Damascus, Syria and Al-zaytoonah University, Amman, Jordan

Abstract: Despite of the decent development of the pattern recognition science applications in the last decade of the twentieth century and this century, text recognition remains one of the most important problems in pattern recognition. To the best of our knowledge, little work has been done in the area of Arabic text recognition compared with those for Latin, Chins and Japanese text. The main difficulty encountered when dealing with Arabic text is the cursive nature of Arabic writing in both printed and handwritten forms. An Automatic Arabic Hand-Written Text Recognition (AHTR) System is proposed. An efficient segmentation stage is required in order to divide a cursive word or sub-word into its constituting characters. After a word has been extracted from the scanned image, it is thinned and its base line is calculated by analysis of horizontal density histogram. The pattern is then followed through the base line and the segmentation points are detected. Thus after the segmentation stage, the cursive word is represented by a sequence of isolated characters. The recognition problem thus reduces to that of classifying each character. A set of features extracted from each individual characters. A minimum distance classifier is used. Some approaches are used for processing the characters and post processing added to enhance the results. Recognized characters will be appended directly to a word file which is editable form.

Key words: Arabic character, classification, discrete wavelet transform, features selection

INTRODUCTION

An optical character recognition system typically consists of the following processing steps: digitization, preprocessing, segmentation, feature extraction, recognition using one or more classifiers and contextual verification or post-processing.

Recognition of Arabic characters represents an important goal, not only for the Arabic speaking countries, but also for Curds, Persians and Aurdo-speaking Indians. However, in spite of the progress of machine character recognition techniques (both printed and handwritten) of Latin, Chines and Japanese characters, research of Arabic characters has been slowly gaining momentum since the early 1980's. The main reason for such delay is the different characteristics of Arabic writing from other writings. This also results from the fact that the techniques developed for other writings cannot be easily applied to Arabic writing. Moreover, to the best of our knowledge, little work has been devoted to hand-written characters. Amin *et al.*[1] developed an on-line system called Iterative Recognition of Arabic Character (IRAC). Also, El-Sheikh and Taweel developed an on-line Arabic hand-written character recognition system[2]. Amin and Masini[3], proposed an off-line recognition of multi-font Arabic text. Almuallim and Yamaguchi[4] have developed a method for the recognition of Arabic cursive handwriting. They dealt with off-line writing and employed a structural method in which words are first segmented into strokes which are then classified using their geometrical and topological properties. Bozinovic. *et al.*[5] developed a recognition system for isolated off-line cursive script words. They presented several new techniques for low and intermediate level processing (reference line finding, letter segmentation based on detecting local minima along the lower contour and areas with low vertical profiles, simultaneous encoding of contours, extracting features and finding shape-oriented events).. Al-Yousefi and Udpa[6] introduced a statistical approach for the recognition of Arabic hand-written characters using moments of the horizontal and vertical projections of the primary character. Chen *et al.*[7] proposed a complete scheme for totally unconstrained handwritten word recognition based on a single contextual Hidden Markov Model (HMM). Abuhabiba. *et al.*[8] developed an off-line character recognition. An algorithm was developed, which yields skeletons that reflect the structural relationships of the character components. Emam[9] presented an optical character recognition system for Arabic hand-written text. He used a border transition descriptor are use as a major features. Eddahibi and Lazrek[10] contribute in the recognition of Arabic scientific document. They treated many complex forms of Arabic scientific documents including the mathematical Lateen symbols.

The proposed system shown in Fig 1, some steps in preprocessing are same as in traditional OCR and a special for the proposed AHTR system as: component

Corresponding Author: Ismael Ahmad Jannoud, Damascus University, Damascus, Syria and Al-zaytoonah University, Amman, Jordan

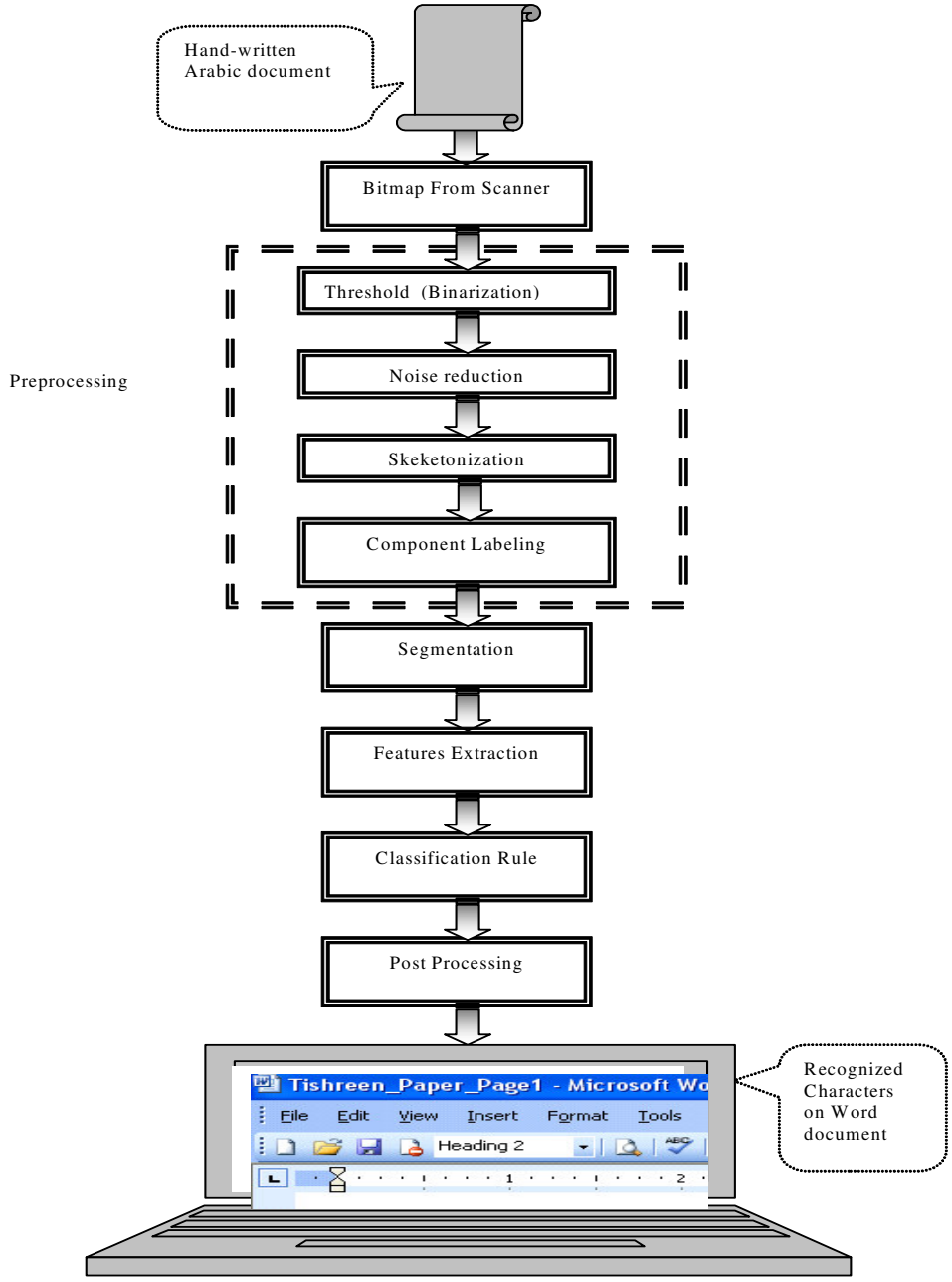


Fig. 1: Proposed AHTR system

labeling (which is leads to sub-words isolation), segmentation and the post processing stages. Post processing includes producing an electronic (editable) form as word file to display each recognized character.

Before detail speaking about the steps in the proposed system, it is important to describe the specialty of the problem of Arabic text recognition.

Some specials of Arbic text: The Arabic alphabet consists of 29 characters, where the shape of each character depends on its position within a word. Thus the characters are divided into four disjoint sets. These types are listed in Table 1 in details. The first set includes those characters which appear in an *isolated* form wherever their position are in different words. The

0	0	0
0	1	0
x	x	x

x	0	0
x	1	0
x	0	0

x	x	x
0	1	0
0	0	0

0	0	x
0	1	x
0	0	x

Fig .2: The smoothing templates

second set includes characters at the head of words, naming *beginning characters*. The third type includes the characters within words naming *middle characters*. Finally, the last type includes those characters at the tail of words naming *end characters*. Thus after segmenting a given word, it will be known a priori which character set needs to be considered.

Data acquisition and preprocessing: The handwritten Arabic text is scanned off from the input document by a scanning device and is stored as a bitmap file with a resolution of 100 dpi. Then, a number of preprocessing steps are performed for the digitized image. These steps generally include binarization, smoothing, thinning,. etc. The binarization step is achieved by comparing the gray value with a given threshold. This threshold value should not be fixed because the gray values representing the background or the characters may change from text to another. The threshold is calculated by finding the dominant gray value in the scanned text which represents the background and then choosing the threshold value to be the mid point between the dominant value and the maximum gray value[11]. Noise errors caused by the data acquisition system, needs to be eliminated from the scanned document using a smoothing step. A 3*3 window is used to examine each pixel. If the pixel under consideration has a value of "1" and its window matches any of the templates shown in Fig. 6, it is set to "0". The asteroid means "0" or "1", since its value does not affect the result.

Segmentation: Segmentation is a necessary step in order to isolate the text image objects to be passed to the recognition stage. In the proposed method, segmentation is performed in three levels: line objects, words or sub-words and character objects. Normally lines are separated by horizontal gaps and hence the projection of the whole image into a vertical axis is performed to mark line boundaries. Having an isolated line, thinning[12] and coloring[11] algorithms are applied. The later is performed to remove the overlap between cursive Arabic characters by means of coloring. In this step, we color each connected component with a different color. Some number, as shown in Fig. 3 represents each color. Then each sub-word which information pixels have different label (number), can be isolated in sub-matrix. Each sub-matrix will be fit into the segmentation stage to isolate each character and putting it in a new sub-matrix (sub-image).

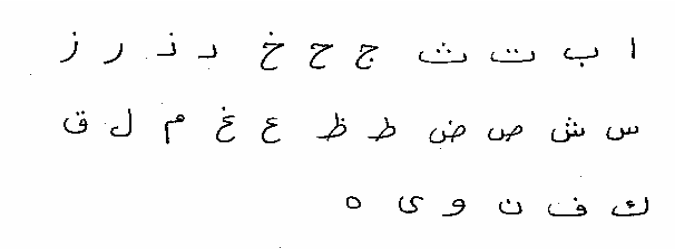
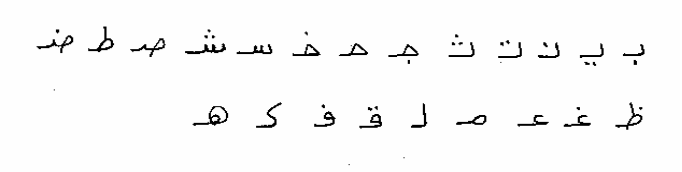
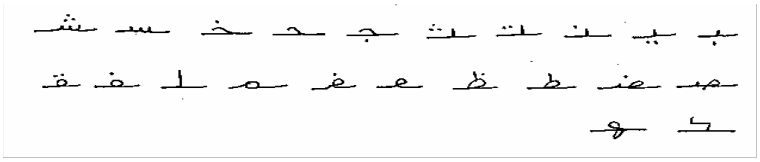
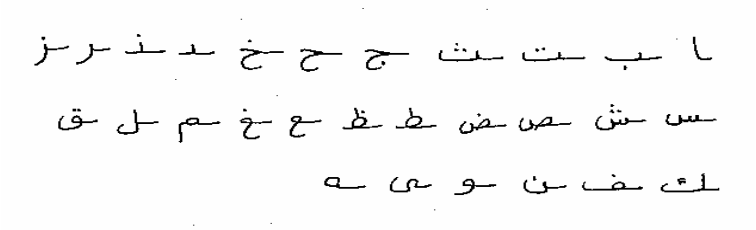
Segmentation of words: The connected components (sub-matrix) of the input line document represent either a secondary part (Dots), an isolated character, or a word (or sub-word). These sub-matrices are arranged by their order of appearance from right to left in the line document. The following segmentation algorithm is applied to these sub-matrices sequentially.

SEGMENTATION ALGORITHM

Step 1: Determine the boundaries of the current color and its base line. Base line finding is based on analysis of the horizontal histogram.
Step 2: Start from the rightmost pixel through the base line. (initialize P)
A) If the pixel has a value of "0" look up vertically:
If there are pixels having value of "1" : count the number of these pixels; N count the number of transition from "0" to "1"; T
If (N > 2 or T > 1) increment P ;
If (P > Thre1) consider this pixel as a segmentation point SP. Go to C. Else move left with writing horizontally or with an angle ± 45
If the next pixel has a value of "0" go to A.
Else go to B.
B) If a pixel has a value of "1"
If it is either (branch point or cross point Fig 4.) consider it as SP.
If (P > Thre2) Go to C.
Else move left with writing horizontally or with an angle ± 45 P=0; (A false SP.)
If the next pixel has a value of "0" Go to A
Else Go to B.
C) Determine the deletion region which is the pixels between two consecutive segmentation points except 4 columns right from the second SP and 4 columns left from the first one.
For each column of the deletion regioncount: the number of Transition between "0" and "1" T1 the number of "1's" T2
If (T1 > 1 or T2 > 3) do not delete this region (It is actually a character).
Else delete this region; C=0;
As an example for this algorithm, consider the word (كشف) resulted after applying the thinning algorithm, as shown in Fig 5. This figure illustrates the base line of this word, the segmentation points SP and the false segmentation points FSP, in addition to the segmented characters of this word.

Features extraction: The Discrete Wavelet transform (DWT) is invoked to extract efficient features of each isolated character which results from the described segmentation stage. To the best of our knowledge, this transform has not been used before for Arabic character

Table 1: Types of hand-written Arabic characters

Types of Hand-Written Arabic Characters	Details of each type	Number of characters
Isolated Characters		28
Beginning Characters		22
Middle Characters		22
End Characters		28

recognition. However, Kapogiannopoulos *et al.*[13] presented an off-line OCR system for Latin hand-written or printed characters using Biorthogonal Discrete Wavelet Transform.

The Discrete Wavelet Transform of the image function I(x,y) in two dimensions is[14]:

$$W(a,b) = \left(\frac{1}{\sqrt{a.b}}\right) \sum_{z}\sum_{v} I(v,z)\ h[(v/a),(z/b)]$$

The input vector for the Discrete Wavelet Transform is a vector which is a binary. Since the size of the input vector determines the high resolution level from which the DWT resumes, we need all our input vectors to have the same size. We used an input vector of 64*64 for our DWT. If we have an input vector of less than this size, we complete the size by adding zeros. The total energy and zero crossing of the transformed signal are calculated. Then by dividing this signal into 12 equal frequency bands and calculating the relative energy and zero-crossing for each band, we form a vector of 24 dimensions.

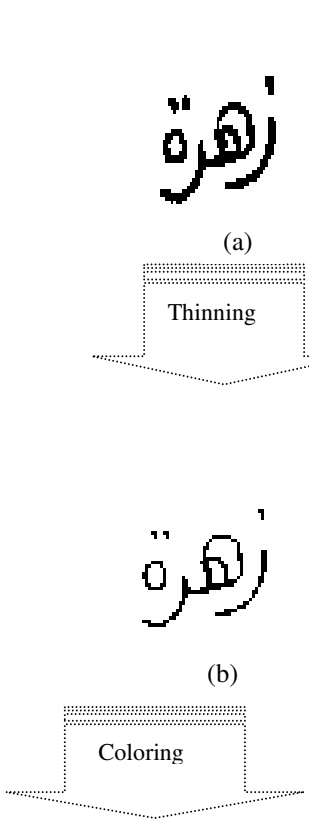


Fig 3: (a) Single line contains one word (b) After thinning step (c) After Coloring step (component labeling)

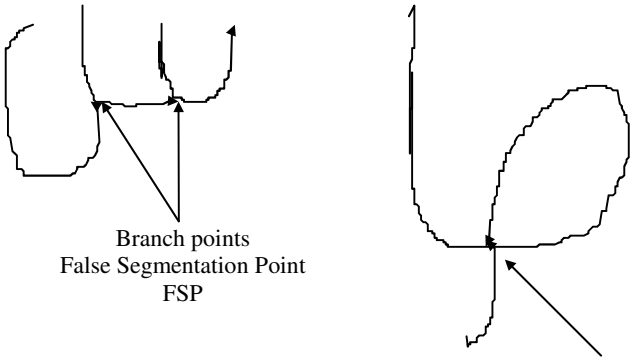


Fig 4: Types of points

(a) False Segmentation Point (FSP)

(b) Cross point

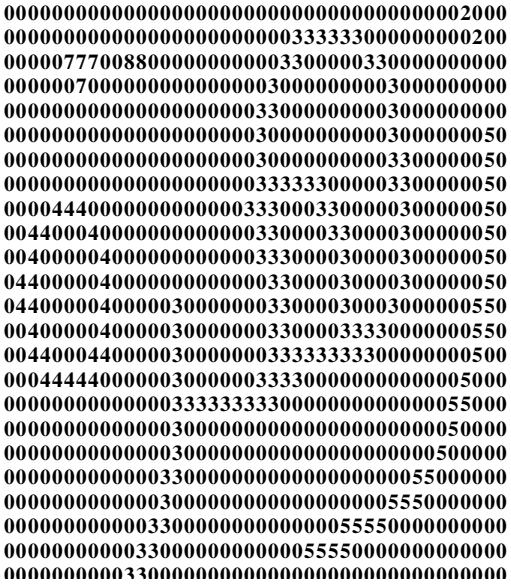


Fig 5: Distinguish between segmentation point SP and false segmentation point FSP

Classification: As mentioned before, we used a 24-dimensional pattern vector X which is defined as:

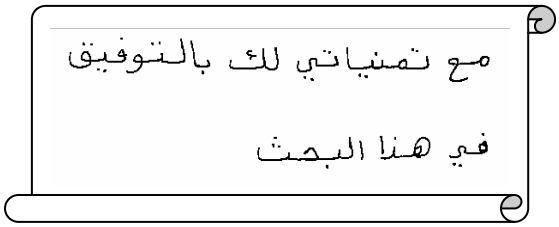
$$X = [x_1\ x_2\ .....x_D]^T \quad (1)$$

For classification, three different techniques have been developed using this set of features. In the first technique, a multi-category classifier is used. The training samples are used to estimate the mean vector $\underline{\mu}_i$ and the covariance matrix $\sum_i$ for each class ω_i

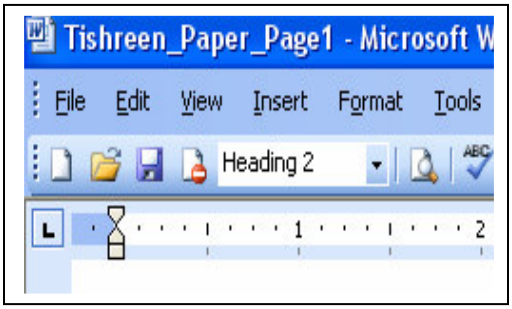
$$\underline{\mu}_i = \frac{1}{n_i} \sum_{\underline{X} \in \omega_i} \underline{X} \quad (2)$$

$$\sum_i = \frac{1}{n_i} \sum_{\underline{X} \in \omega_i} (\underline{X} - \underline{\mu}_i)(\underline{X} - \underline{\mu}_i)^T \quad (3)$$

these are known as the Maximum Likelihood Estimation (MLE) of the parameters, where n_i is the number of samples for class ω_i.



(a)



(b)

Fig 6: (a) one input document (b) the output of the AHTR system

For each set of characters, a subset of the feature set is selected. This selection is the sequential forward selection technique SFS[15,16]. This technique is based on choosing the best single feature, then the best pair including the best feature and so on. Then, we used a subset of the feature set which gives a maximum probability of character classification. This technique is simple, efficient and produces a suboptimal set of features.

To classify an unknown character, its feature vector $\underline{X}$ is calculated and a generalized distance measure between this vector and each of the average vectors is evaluated, using the selected features. This distance measure, known as the *Mahalanobis* distance, is defined as:

$$DIST_i = (\underline{X} - \underline{\mu}_i)^T \sum_i^{-1} (\underline{X} - \underline{\mu}_i) \quad (4)$$

This distance is used to classify unknown character using the following minimum-distance classification rule:

Choose Class ω_i If

$$DIST_i \leq DIST_j \quad \forall\, j \neq i \quad (5)$$

Thus, the unknown pattern is assigned to the class corresponding to the closest average vector in the feature space, after feature selection.

RESULTS AND DISCUSSION

In order to investigate the effectiveness of DWT in recognizing Arabic characters, a series of tests were performed using the three classification techniques. Data was obtained using a "Photomaker" scanner with a resolution of 100 dpi and the tests were performed using a personal computer. All the algorithms are fully implemented and tested using MATLAB programming language.

Many input documents are used which have a hundreds of characters of all types (isolated, beginning, middle and end characters) as a training data.

The AHTR system has been tested using different input Arabic documents. The outputs are directly displayed on a word file which has been opened and each recognized character appended to this file. This word file will have all the text in the Arabic hand-written input document.

The proposed system was efficient and the performance was about 90^s % differs from types of the characters. The biggest recognition rate was for the isolated characters (99%). But the middle characters has the worst recognition rate (91%). One input Arabic document is shown in Fig 6a and the output word file is shown in Fig 6b.

CONCLUSION

The main objective of this study is to develop an Arabic handwritten text recognition system. In the proposed AHTR system, smoothing is done to eliminate the noise from the input image. A three level segmentation process has been used to segment the text into lines, words (or sub-words) and finally into characters. The problem then reduced to that of classifying a set of characters. The set of Arabic characters have been divided into four main groups; the isolated, the beginning, the middle and the end characters. The recognition of each character consists of two steps: Features extraction and classification. The Discrete Wavelet transform (DWT) is invoked to extract efficient features of each isolated character which results from the segmentation stage. To the best of our knowledge, this transform has not been used before for Arabic character recognition. A 24-dimensional feature vector representing the relative energy and zero crossing was extracted from the output signal of the DWT. Training samples for characters have been used to design the multi-category classifier. It was shown that the proposed AHTR system was efficient enough and can be enhance the machine learning in the problem of Arabization.

ACKNOWLEDGEMENT

I would like to thank my friend T. Hammoud for encourage and help during the detailed of this research.

REFERENCES

1. Amin, A., A. Kaced, J.P. Haton and Moher, 1980. Handwritten Arabic character recognition by I.R.A.C. system. Proc. Fifth. Intl. Conf. Pattern Recognition, pp: 721-731.
2. El-Sheikh, T.S. and S. Taweel, 1990. Real-time Arabic handwritten character recognition. Patt. Recog., 23: 1323-1332.
3. Amin, A. and G. Masini, 1986. Machine recognition of multi-font printed Arabic text. Proc. 8th Conf. Patt. Recog. (Paris, France), pp: 392-395.
4. Almuallim, H. and S. Yamagushi, 1987. A method of recognition of Arabic cursive handwritten. IEEE Trans. PAMI, 9: 5.
5. Bozinovic, R.M. and S.N. Srihari, 1989. Off-line cursive script word recognition. IEEE Trans. PAMI, 11: 1.
6. Al-Yousefi, H. and S. Udpa, 1992. Recognition of Arabic characters. IEEE Trans. PAMI, 14: 8.
7. Chen, M., A. Kundu and J. Zhou, 1994. Off-line handwritten word recognition using a hidden Markov model type stochastic network. IEEE Trans. PAMI, 16: 5.
8. Abuhabiba, S.I., S.A. Mahmoud and R.J. Green, 1994. Recognition of handwritten cursive Arabic characters. TEEE Trans. PAMI, 16: 6.
9. Emam, A.M., 1995. Designing a reader machine for the blind. Ph.D. Thesis. University of Alexandria.
10. Mustapha, E. and A. Lazrek, 2003. Arabic scientific document composition. ICITNS 2003, Amman, Jordan.
11. Altuwaijri, M. and M. Bayoumi, 1994. Recognition of Arabic characters using neural networks. ICECS, pp: 720-725. Dec 19-22, Cairo, Egypt.
12. Rafael, C.G. and R.E. Woods, 1992. Digital Image Processing. New York.
13. Kapogiannopoulos, G. and M. Papadakis, 1994. Character recognition using a biorthogonal discrete wavelet transform. SPIE, Optical Pattern Recogn., 2825: 384-393.
14. Rashkovskiy, O., L. sadovnik and N. Caviris, 1994. Scale, rotation and shift invariant wavelet transform. SPIE, Optical Pattern Recogn., 2237: 390-401.
15. Devijver, P. and J. Kittler, 1982. Pattern Recognition: A Statistical Approach. Prentice Hall, Englewood Cliffs, N.J.
16. Robert, J.S., 1992. Pattern Recognition: Statistical, Structural and Neural Approaches. New York.