New Approaches for Cursive Languages Recognition: Machine and Hand Written Scripts and Texts

KHALID SAEED
Computer Engineering Department
Faculty of Computer Science
Bialystok University of Technology
Wiejska 45A, 15 351 Bialystok
POLAND (*)
aidabt@ii.pb.bialystok.pl

Abstract: - Three different approaches are considered in this paper to deal with the methods of Pattern Classification and Recognition. The main patterns considered are images representing the alphabet of cursive-scripts languages, particularly Arabic alphabet. The practical results of written scripts recognition led to the possibility of applying the main ideas and criteria to written and spoken texts and hence to generalise the worked out algorithms and approaches and extend them to test other kinds of images.

Key-Words: - Scripts and Texts Recognition, Toeplitz Matrices, Neural Networks Approaches.

1 Introduction

A lot has been done on Pattern Recognition methods and their different applications. However, the language recognition field still has its importance in research [1,2,3,4,5,6] and remains open. This comes from the fact that people are still looking for a simple and easy way of communication between them. If the common language has not been reached, yet, then let us find a fast economical approach of natural languages recognition and translation, or even processing and converting them, one to another, to simplify the ways of communication between people. Moreover, let the blind also join us in understanding what we write, through a fast way of converting a written text to a spoken one for them, to be on line with us.

In this paper shown some approaches of using the possible ways of contribution in the world of Pattern Recognition, trying to give some better solutions to the problem of finding a general and more economical algorithm to recognise both written and spoken texts of cursive, and sometimes fuzzy, character. If we succeed in reaching our aim, then we will also be able to extend the methods to apply to some defaced documents taking place, very often, in signed cheques or sealed certificates, being very difficult for classification and recognition. We have chosen Arabic handwriting because it represents variety of images and different graphical shapes can be obtained and formed by handwritten texts. Moreover, the Arabic handwriting is one of the most difficult ones, and therefore, working out an easy fast general algorithm for its recognition would be most required in signal and language processing.

In this work three different approaches are presented. They involve and are based on the following approaches and methods of classification and description:

1. *Toeplitz Matrices Approach* [1, 2, 7],
2. *Projection Approach and Neural Networks* [1,3,8,9],
3. *Classification without Segmentation* [2,10].

1.1 Some Features of Arabic, Farsi and Urdu Alphabets

Arabic alphabet is used by over 30% population of the world, for example in Arabic, Farsi (Persian) and Urdu languages. Simple and economical methods of their recognition are, therefore, certainly, as important as in other languages.

(*) *This work is sponsored by The Rector of Technical University of Bialystok.*

Unfortunately, it is not necessary that the algorithms, used on other languages, also apply on this alphabet with the same efficiency. This comes from the fact that Arabic alphabet has its specific features [11], some of which are the following:

1. Some of the scripts have loops in their structure: dhad -ض, ttaa -ط, zhaa -ظ , qqaf - ق

2. And (or) they have dots (one, two or three), above or below the letter. In many cases the only difference between similar letters is the number or position of these dots, as shown below.

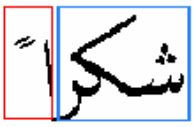
- baa - ب , taa - ت, thaa - ث
- jeem - ج , hhaa - ح , khaa - خ
- dal - د , thal - ذ , raa - ر, zay - ز
- seen -س, sheen - ش
- ssad - ص, dhad - ض
- ttaa - ط, zhaa - ظ
- ain – ع, ghain - غ
- faa - ف, qqaf - ق

3. In both hand and machine written words, letters are joined either in a word or a subword. For example, the word *shukran* (Arabic for thanks) consists of two subwords (Fig.1).

Fig.1 *The word 'shukran' in its two subwords.*

2 Preprocessing Algorithms

For all approaches the stage of *Preprocessing* is the same - a general structural algorithm [1,12],



worked out to prepare the script or the word for *Feature Extraction*, is used. As recognition implies thinning and segmentation before classification, some of the obtained results in these fields are also introduced.

2.1 Segmentation

The main problem arising in segmentation is the possibility of overlapping of scripts in a word or subword which occurs quite often especially in hand written texts of Arabic language. Fig.2 shows this concept in a machine written word. In hand written texts, the problem is even bigger.

Fig.2 *Overlapping of letters in the word of Fig.*1.

Although the possibility of classification and recognition of words without their segmentation is also considered in this paper, it is still of great importance and interest to find a simple solution to the problem of segmentation. Some aspects and solutions are suggested and given in *Section* 3.3.

2.2 Thinning

This implies image preparation for classification. It transforms the script lines with different widths to a one-pixel width skeleton and that is why thinning is sometimes called *pixelization* [13] or *skeletonization* [14]. As mentioned above, a simple algorithm has been done [7,12] for this purpose. Fig.3 shows the result of applying the algorithm to thin and detect lines and dots of the letter ج - *J*.

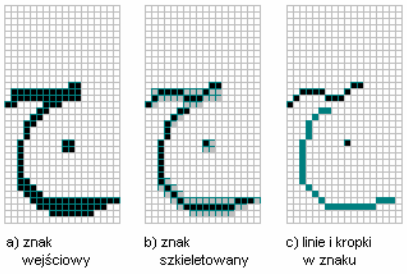


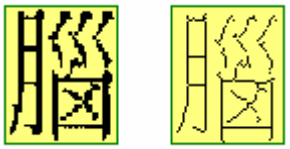
Fig.3 *Thinning and line detection of the letter* ج

2.3 Samples of Thinning Results

Fig.4 shows the letter *B* thinned by this algorithm and three other ones, given for comparison.

Fig.4 *Thinning by:* (a) *the algorithm worked out and used in this paper;* (b), (c) *and* (d) *algorithms of other authors* [15].

Also, this algorithm has been applied on other languages, as well. Fig.5 shows its results on the



Chinese word '*brain*'.

Fig.5 *Thinning the Chinese word 'brain'.*

3. Three Classifying Algorithms: Features Extraction

In this section three methods of classification are introduced with some examples and most important characteristics.

3.1 Toeplitz Approach

Here is given the basic part of the describing and classifying system, the algorithm of verification. Its main idea comes from a criterion [15,16] used to test digital filter transfer functions for realizability. According to this algorithm the minimum eigenvalues spectrum of a given digital filter transfer function is determined. For example, the spectrum of a digital *realizable* filter shown in Fig.6 has the following transfer function:

$$H(z) = \frac{4.93 - 2.06z^{-1} - 0.53z^{-2} + 2.2z^{-3}}{1 - 0.06z^{-1} + 0.06z^{-2} + 0.06z^{-3}} \quad (1)$$

The spectrum shows a monotonically decreasing series of minimal eigenvalues, λ's.

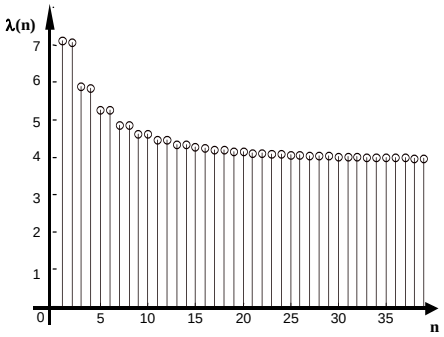


Fig.6 λ-spectrum for the eigenvalues of Eq.(1).

Now, consider the following digital filter transfer function. It is still decreasing monotonically, but tends to a negative value at some value of *n* (Fig.7). This means that the function does not represent any realizable digital filter.

$$H(z) = \frac{1 + z^{-1} + 0.4z^{-2}}{1 - 1.2z^{-1} + 0.9z^{-2}} \quad (2)$$

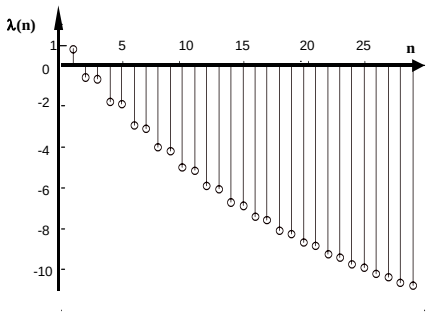


Fig.7 λ*-spectrum for the eigenvalues of Eq.* (2).

This, however, is not within the topics of this paper although it shows very important results [16]. In this paper, we are most interested in the way the eigenvalues of Toeplitz matrices behave, as they form very useful, standard and stable vectors and very easy for comparison, description and classification. These vectors are the main feature vectors in our algorithms. To achieve that and define the feature vectors, the distances $|w_i|$ from origin (0,0) to some selected pixels of the image (Fig.8) are calculated, first. Then the differences between the successive vectors are determined:

$$r_i = |w_{i-1}| - |w_i| \quad \text{for} \quad i = 1,2, ..., n \quad (3)$$

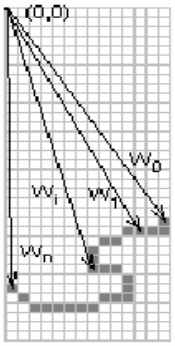


Fig.8 *Distances* $|w_i|$ *to image are calculated.*

Now, Toeplitz matrices and their determinants D_i are found for each of which the minimal eigenvalues $\lambda_{\min}$ are calculated.

$$D_i = \begin{vmatrix} r_1 & r_2 & r_3 & \dots & r_i \\ r_2 & r_1 & r_2 & \dots & r_{i-1} \\ r_3 & r_2 & r_1 & \dots & r_{i-2} \\ \dots & \dots & \dots & \dots & \dots \\ r_i & r_{i-1} & r_{i-2} & \dots & r_1 \end{vmatrix}; \quad i=1,2,\dots,n. \qquad (4)$$

The feature vectors are hence defined:

$$a_j = \left[\lambda_{1_{\min}}, \lambda_{2_{\min}}, \dots, \lambda_{n_{\min}}\right] \qquad (5)$$

For each detected line in a script there exists a feature vector. Therefore, a letter of two lines has two such vectors.

3.1.1 Classifying Algorithm

The most essential steps of the classifying algorithm are the following:

1. The number of lines per script is determined. They may be one (the letter د - *D* or ر - *R*, for example), two (the letter ح - *Hh*, or ط - *Tt*, for example) or three (س - S and ش - *Sh*).
2. The script pixels of the skeleton (or its lines) are registered and their distance from the origin (0,0) is measured (Fig.8). Sometimes, four or five such points and distances are sufficient to give the main features required for verification of the image.
3. If exist, dots are checked for position (above the letter like in ت - *T* or ز - *Z*, or below it as in the letters ب - *B* or ج - *J*, for example) or number (one dot, two dots or three, as well seen in the same examples above).

According to the results obtained, the letter is classified and hence recognized [1,8]. The same mathematical basis was used successfully in the method of recognition based on Muqla model [2].

3.1.2 Algorithm Results - Example

All Arabic letters were checked for recognition purpose using this algorithm. They also were compared with other known ones. Fig.9 shows five different fonts of the letter د - *D* together with their feature vectors.

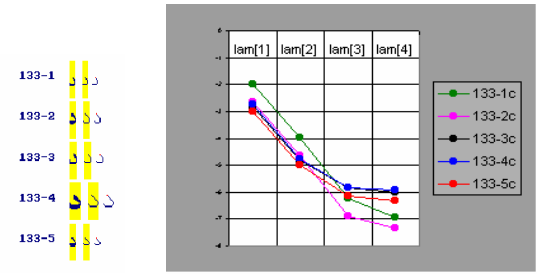


Fig.9 *The letter* د - *D in five different fonts and their corresponding feature vectors.*

3.2 Projection Approach

This approach resolves each pixel of the image onto two of twelve axes to form some kind of bars [3,8]. The feature vector's elements are the lengths of the resulting bars, where a zero element is also a possibility. Fig.10 shows the letter *B* thinned and projected according to this approach. The feature vector is found from the figure; given by:

$$V = [\mathbf{64, 64, 79, 36, 57, 64, 73, 86, 29, 71, 64, 73}].$$

For comparison and classification an artificial neural network is used [13,17,18].

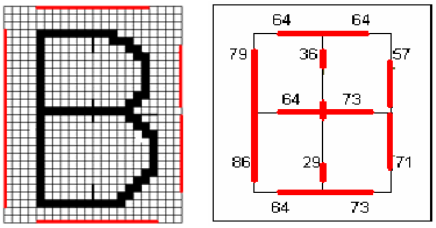


Fig.10 *The letter B and its projection map.*

The applied multi-layer NN is trained by supervised-learning method of back-propagation [18], following the RPROP (Resilient backPROPagation) algorithm presented by Riedmiller and Braun [17]. The change in weights vector is given by:

$$\Delta W_{ij}(k) = -\eta_{ij}^{(k)} signum(\frac{\partial E(W(k))}{\partial W_{ij}}) \qquad (6)$$

where $E(W)$ – objective function,

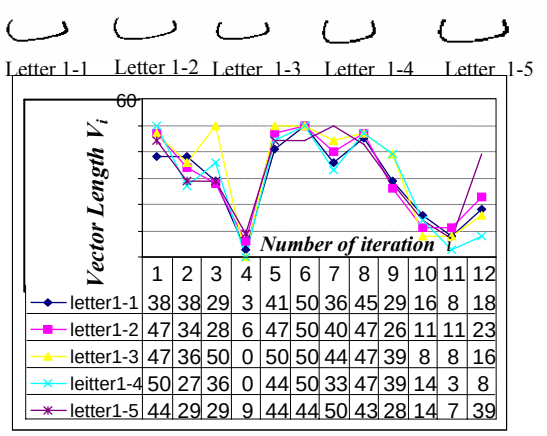
$$\eta_{ij}^{(k)} = \begin{cases} \min(a\eta_{ij}^{(k-1)}, \eta_{\max}) & for \quad S_{ij}(k)S_{ij}(k-1) > 0 \\ \min(b\eta_{ij}^{(k-1)}, \eta_{\min}) & for \quad S_{ij}(k)S_{ij}(k-1) < 0 \\ \eta_{ij}^{(k-1)} & otherwise \end{cases}$$

with $a = 1.2$, $b = 0.5$, $\eta_{min} = 10^{-6}$, $\eta_{max} = 50$, and

$$S_{ij}(k) = \frac{\partial E(W(k))}{\partial W_{ij}}$$

3.2.1 Example

The letter ب - *B,* in its five different fonts, is projected and processed for feature vectors and given as an example in Fig.11.



	1	2	3	4	5	6	7	8	9	10	11	12
letter1-1	38	38	29	3	41	50	36	45	29	16	8	18
letter1-2	47	34	28	6	47	50	40	47	26	11	11	23
letter1-3	47	36	50	0	50	50	44	47	39	8	8	16
leitter1-4	50	27	36	0	44	50	33	47	39	14	3	8
letter1-5	44	29	29	9	44	44	50	43	28	14	7	39

Fig.11 *Feature Vectors of the letter* ب *- B; they are the same for* ت*-T,* ث*Th and* ن*- N.*

3.3 Ibn Muqla Approach

This method [2] is based on a model that assumes the image to be laid on a circle. Then the mathematical model of Toeplitz approach is applied. Also, the method of projection can be used here. The main advantage of this method is the simplicity of its application to word recognition without segmentation [10]. The algorithm of classification used in this approach easily finds the scripts connecting points (the points *A* and *B* in Fig.12).

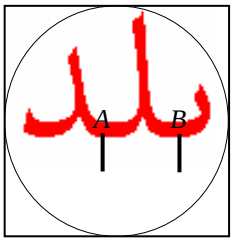


Fig.12 *The word* بلد *- balad (Arabic for country) processed without dots, thinning or segmentation.*

These points are the places where cutting for scripts separation takes place which helps for fast classifying as their places show high flexibility. Moreover, as was mentioned in *Section* 2.1, the problem of overlapping can be solved by this model, since the connecting points can be modified to completely isolate the last pixel of the letter on left of the point from the first pixel of the letter on its right at the base line of the word. Fig.13 shows how the problem of overlapping is overcome by this method.

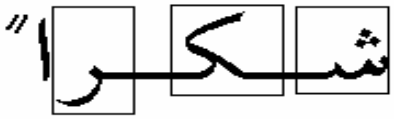


Fig.13 *The image of Fig.*1*, after overcoming the problem of overlapping shown in Fig.*2*.*

4 Conclusions

New general algorithms on cursive scripts recognition, based on new mathematical models, are presented in this paper. The results are shown through some examples. A solution, which seems to be practical from experimental point of view, to the problem of overlapping of letters in a word, occurring frequently, especially in handwritten texts, has been achieved and proposed in this work. The mentioned in this paper methods and

other ones are being tested on spoken texts, with encouraging results.

References:

[1] K. Saeed, "Three-Agent System for Cursive-Scripts Recognition," *Proc. CVPRIP'2000 Computer Vision, Pattern Recognition and Image Processing - 5th Joint Conf. on Information Sciences JCIS'2000*, Vol.2, pp. 244 -247, Feb. 27 - March 3, N. Jersey 2000.

[2] K. Saeed, A. Dardzińska, "Cursive Letters Language Processing: Muqla Model and Toeplitz Matrices Approach," *FQAS'2000 – 4th Int. Conference on Flexible Query Answering Systems, pp. 326-333*, Oct. 25 – 27, Warsaw 2000. (P*rinted as*: *Flexible Query Answering Systems; Recent Advances,* Springer-Verlag, 2000)**.**

[3] K. Saeed, "A Projection Approach for Arabic Handwritten Characters Recognition, "*Proc. of ISCI - International Symposium on Computational Intelligence,* pp.106-111, Aug. 31 – Sep. 1, Kosice, Slovakia. (P*rinted as*: *Quo Vadis Computational Intelligence? New Trends and App. in Comp. Intelligence,* Springer-Verlag, 2000)**.**

[4] M. Ghuwar, "Modelling and Recognition of Arabic Scripts," Ph.D. Thesis, *Institute of Computer Science, Polish Academy of Sciences*, Warsaw, 1997.

[5] D. Burr, "Experiments on Neural Net Recognition of Spoken and Written Text," *IEEE Transactions on Acoustics, Speech, and Signal Proc.,* Vol.36, No.7, July 1988**.**

[6] M. Dehgham, K. Faez, "Handwritten Farsi Character Recognition Using Evolutionary Fuzzy Clustering," *Proceedings of Eusipco'98, 9th European Signal Processing Conference*, Sep. 8-11, pp. 423-426, Rhodes, Greece 1998.

[7] K. Saeed, R. Niedzielski, "A Fast Recognition Structured Algorithm of Typewritten Cursive Scripts," *Proceedings of 6th Conference of PTSK on Simulation in Research and Development* (in Polish), pp. 67-71, Bialystok – Bialowieza, 1999.

[8] K. Saeed, M. Nalewajko, "Recognition of Cursive Scripts by Projection - A Neural Approach," *Proceedings of 6th Conf. of PTSK on Simulation in Research and Development* (in Polish), pp. 65-67, Aug. 25-27, Bialystok-Bialowieza 1999.

[9] M. J. Nalewajko, "Universal system for recognition of scripts using artificial neural networks," M.Sc. Thesis, *Faculty of Computer Science, Technical University of Bialystok*, Bialystok, 1999.

[10] K. Saeed, A. Dardzińska, "Language Processing: Word Recognition without Segmentation, "Accepted for publication and presentation in *WSES'2000 Multi-conference on Applied and Theoretical Mathematics*, 1-3 Dec., Vravrona, Greece 2000.

[11] M. Ghuwar and W. Skarbek, "Recognition of Arabic Characters - A Survey," *Polish Academy of Science*, Manuscript No.740, Warsaw 1994.

[12] K. Saeed, R. Niedzielski, "Experiments on Thinning of Cursive-Style Alphabets," *Inter. Conf. on Information Technologies ITESB'99*, June 24-25, Mińsk 1999.

[13] J. A. Freeman, D. M. Skapura, "*Neural Networks - Algorithms, Applications, and Programming Techniques*," Addison Wesley, MA 1991.

[14] K. Saeed, "Experimental Algorithm for Testing the Realization of Transfer Functions," *Proc. of 14th IASTED Conf. on Modelling, Identification and Control*, pp. 114-116, Feb. 20-22, IGLS, Austria 1995.

[15] Y.S. Chen, "The Use of Hidden Deletable Pixel Detection to Obtain Bias-Reduced Skeletons in Parallel Thinning," *Proceedings of 9th ICPR'96 - IEEE,* Vol.2, pp. 91-95, Vienna 1996.

[16] K. Saeed, "On the realization of digital filters," *Proceedings of 1st International Conference on Digital Signal Processing and its Applications– DSPA'98,* Vol.2, pp. 141-143, Moscow 1998.

[17] M. Riedmiller, H. Braun, "RPROP – A Fast Adaptive Learning Algorithm," *Technical Report, Karlsruhe University*, 1992.

[18] S. Osowski, "Neural Networks - Algorithmic Approach," (in Polish), *WNT*, Warsaw 1996.