Online Handwriting Recognition for the Arabic Letter Set

MAI AL-AMMAR *
mai.alammar@gmail.com

REHAM AL-MAJED *
reham.imamu@gmail.com

HATIM ABOALSAMH *
Hatim@ksu.edu.sa

*Computer science department
King Saud University
Riyadh, SAUDI ARABIA

Abstract— Automated methods for the recognition of Arabic script are at an early stage compared to their equivalent for the recognition of Latin and Chinese languages, especially of online handwriting recognition. In this paper we describe the stages of the recognition process unique to the Arabic hand written text. We also introduce an account of Arabic online handwriting recognition methods in literature, with a rich list of references for the interested readers. We cast some light on the characteristics of Arabic writing and present an overview of the common stages normally followed by handwriting recognition systems which are: preprocessing, segmentation, feature extraction, classification, and post-processing along with the most used techniques.

Key-words: online handwriting, Arabic recognition, recognition classifications, Script recognition.

1 Introduction

Script recognition is one of the important fields of pattern recognition whether the script is isolated characters or cursive one; and whether it is handwriting or machine printed [2]. Pattern Recognition has been developed for many years, and its technology has been applied in many fields such as artificial intelligence, computer engineering, medicine, image analysis, etc. pattern recognition is defined as “a classification of input data via extraction important features from a lot of noisy data” [1].

Script recognition can be classified in two main categories according to the way that characters are fed to the system: online and offline recognition.

The offline category recognizes scripts after the writing is completed whether the script is handwritten or machine printed. It is purely a visual representation of text that does not refer to any dynamic information, such as how the character was written and in which order. The only known information is that either the pixels forming the character are on or off. The data input is fed to the computers as a digital image format captured for example by a scanner [5][6].

While online script recognition, by contrast, recognizes scripts while the user writes. That is, the user can write texts by mouse, tablet, digitizer or similar touch-sensitive device which the system captures the x-y coordinates data (of pen-tip movement) and as soon as the user writes, the system will recognize a script. It uses the digitized trace of the pen to recognize the character where the digitized samples are fed to the system as a sequence of 2D-points in real-time [5][6].

2 CHARACTERISTICS OF ARABIC SCRIPT

Arabic is spoken by 234 million people; and Arabic script is written by more than 100 million people in over 20 different countries [9]. It consists of 28 letters in addition to seven auxiliary letters (ء ، أ،إ،ؤ،ئ،ة،ى) . For each letter, there are four different shapes: stand alone, initial, medial, final shape. The choice of which shape to use depends on the position of letter within its word or sub word. Arabic script is written from right to left. Arabic word consists of one or more sub words. Each sub word consists of one character or more. There is no connection between separate words, so word boundaries are always represented by space. Arabic text, both handwritten and printed, is cursive. The letters are joined together along a writing line.

More importantly from the point of view of automated recognition, Arabic contains dots and other small marks that can change the meaning of a word, and need to be taken into account by any recognition system. There are three types of dotting [3][4][8][12]:

1. One dot ’.’ it can appear above like ’ن’, or below like ’ب’, or inside like ’ج’.
2. Two dots, they can be written either as separate dots ’..’ or as a dash ’-’, and they can appear below like ’ي’, or above like ’ت’.
3. Three dots, they can be written as three separate dots ’:.’, or as a cup ’^’ they appear only above like ‘ث’.

Also, there are different types of marks the indicate doubled consonants or different sounds:

1. Medda ’~’ written above Alef, ’آ’.
2. Hamza ’ء’ it can appear above like ’أ’, below like ‘إ’, or inside like ’ك’.
3. Shadda ‘ّ’ it can appear above letters جّ .

Arabic also uses diacritical marks (diacritics) to control the pronunciation of words and represent short vowels or other sound such as syllable endings and nunation (the addition of an “n” or “nuun”). They are usually outside the scope of handwriting recognition research since they rarely appear in handwritten documents. When some combinations of letters appear, they have unique forms. These combinations are mostly pairs of letters such as, “laam-alef,”, “laam-meem,” or “laam- heh,” all common occurrences in handwritten text. Figure 1 shows an examples of connected letters.

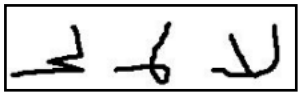

Figure 1: Connected Letters

3 Common Stages of Online Arabic Handwriting Recognition System

In this section the common stages of Arabic online handwriting recognition are presented. These stages are not mandatory in all systems. The recognition process can go without all these stages. However, some systems have integrate some of these stages with others [16][31] or pass up this stage at all as in [22][31]. Figure2 shows these stages in general.

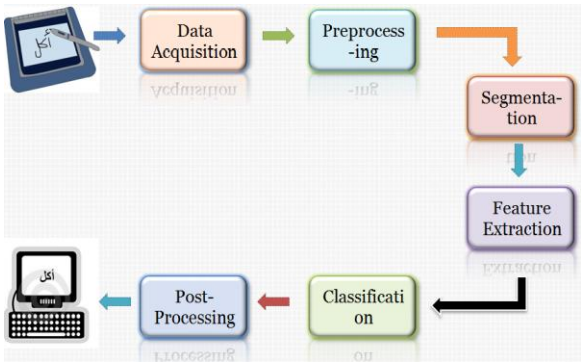

Figure 2: Common Stages of Online Handwriting Recognition

3.1 Data Acquiestion

A most important problem in developing recognition systems in general is usually the insufficiency of data. Recognition systems require different types of data sets such as training data set, testing data set and corpora for language model generation [14]. Therefore, the data acquisition stage is an important stage in the recognition process particularly in the online recognition systems.

A wide range of digitizers with different technologies are available in the market ; hence, use of digitizer tablets covers a wide range, based on their applications and reliability [18].

3.2 Preprocssing

After the input data is captured and digitized in the data acquisition stage, the recognition process goes to the second stage which is the preprocessing. In this stage the input data are prepared so that it's easier to process in the next stages. The main purpose of the preprocessing stage is to eliminate or reduce as much noise and data variation as possible of the acquired point sequences[30][15][29].

Most digitizers perform uniform temporal sampling, which often results in an oversampling of slow pen motion regions and under-sampling of fast pen motion regions [22]. However, the importance of preprocessing techniques is of course utterly dependent on the quality of the handwriting input. Nowadays, even cheaper on-line capturing devices produce fairly smooth signals and for this reason preprocessing focusing on signal noise, such as re-sampling, is no longer a crucial component of a recognition system [32]. The techniques used in the preprocessing stage include:

1. **Size Normalization**: applying this technique achieves a size independent system which means the ability of the system to recognize the input even it is written in small or large style. Of course size- independent is a good criterion when evaluating systems. The system presented in [29] scales the image so that the maximum radius for the character pixels equal to half the grid size where the radius is defined to be the length of the straight line connecting the pixel to the origin. In [17] system, The input stroke is scaled so that it fits maximally in a l00×100 box centered in the origin
2. **Translation:** [29] computes the image’s center of gravity, then translate the image such that its origin is the center of gravity.
3. **Re-sampling:** This technique performs writing-speed normalization by re-sampling the point sequences and distributing the points uniformly over the sampled curve**.** Due to the variation in writing speed, the acquired points are not distributed evenly along the stroke trajectory. There are less points in under-sampling where the speed is high and more points in

oversampling where the pen motion is slow [22][17]. Figure 3 shows the concept of re-sampling**.**

Figure 3: Re-sampling Technique

4. **Smoothing and Noise Elimination:** the input data provided by the tablet may contain a considerable amount of noise that complicates the work in next stages. This noise is caused by the digitizer as well as by a shaking hand [15][17]. This technique is necessary to eliminate duplicated date points by forcing a minimum distance between consecutive points. Moreover, [17] used weighted averaging over the range [i- k,… i+ k]. , Douglas and Peucker's algorithm was adopted to simplify the point sequences in [30]. Figure 4 shows the effect of this technique on the input data.

Figure 4: Smoothing Technique

3.3 Segmentation

Segmentation stage is an important stage in online recognition systems especially in the systems that recognize cursive script such as Arabic script [20][21]. Segmentation is the process of dividing input writing into smaller units, may be words, sub words, letters or small strokes. Although that segmentation stage is crucial in Arabic recognizer, it could be eliminated from recognition process. Therefore, there are two main approaches of recognition according to the segmentation stage: Holistic approach that eliminates segmentation process, and an analytical approach that need segmentation.

In the holistic approach, a recognizer recognizes and deals with an input word shape as a whole. It avoids the need to segmentation, therefore, it avoids errors that are generated from inefficient segmentation algorithms [22][20]. However, the holistic approach is not practical in Arabic script which has a large set of vocabulary since it needs to train the system for each word in a dictionary [22]. A system that used a holistic approach is susceptible to achieve low recognition rate particularly if it recognizes language scripts that have large dictionary size such as Arabic language but if the dictionary has small set of word, it may achieve high recognition rate[14][23]. In the sample of Arabic recognizer that we have surveyed, the [22] used holistic approach.

In the other hand, analytical approach segments an input curve, which represents a word, into individual characters or strokes, which are recognized and then assembled to identify the written word [13][22][33]. This approach has an advantage in which it requires a small set of trained model may be one model for each letter to handle large vocabulary [13]. And in case of Arabic letters which have different shapes for each letter, it is helpful to segment written words and characters into small strokes - strokes are units of recognition that are extracted through dividing letters into units. These strokes used to formed all Arabic letters and strokes number is less that Arabic letters with their different shapes. In [24], authors found that 20 strokes could represent all Arabic letters as in Figure5.

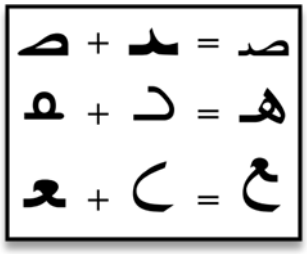

Figure5: Some Strokes that Consists Letters

Since the segmentation may produce errors which will necessarily affect the classification stage, recognition systems need to select an efficient method for segmentation. Actually, segmentation of written word in online Arabic recognition systems faced some problems such as: However, the absence of consistent baselines, large variations in writing styles, and seamless connection between letters, i.e. the connection between letters is done with almost no ligatures [22].

For cursive script, there will always be cases with such missing segmentation points due to uncertain writing. Thus, since a system needs to cope with missing segmentation points anyway, only a limited effort has been spent on the segmentation method [26].

Below we present a brief overview of some methods of segmentation used in the surveyed systems:

- **extreme points based segmentation**

These segmentation points are important because they can help in identifying ascenders and discenders in Arabic handwriting such as the medial alif (ا) , laam (ـلـ) and the

final raa (ـر). Figure6 represents the segmentation points if extreme points based segmentation is used[27].

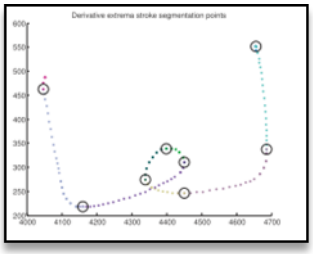

Figure 6: Extreme Points Based Segmentation

- **direction based segmentation**

The slopes of tangent lines are computed at each point, The points at which the slope of the tangent changes signs and differs by more than a threshold amount in magnitude from the slope of the previous tangent, are noted. Figure 7 represents the segmentation points if direction based segmentation is used[27].

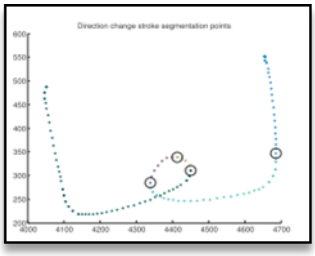

Figure7: Direction Based Segmentation

- **Intersection point based segmentation**

Points at which the handwriting stroke intersects itself are noted. Figure8 represents the segmentation points if Intersection based segmentation is used [27].

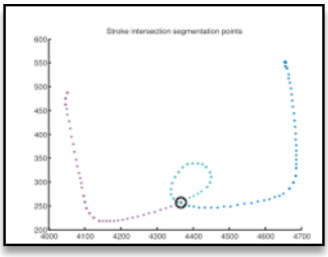

Figure8: Intersection Based Segmentation

- **Pen speed based segmentation**

In this method, pen speeds are calculated at each point according to the formula

$si = di+1 - di-1 / ti+1 - ti-1$

after that, using the following criteria to choose segmentation points: Speed minima that are lower than a threshold percentage of the average speed, and Local maxima of curvature are accepted if the speed at that point is below a threshold (even if not a minimum) and the curvature is above a threshold [27]. Figure9 represents the segmentation points if pen based segmentation is used[27].

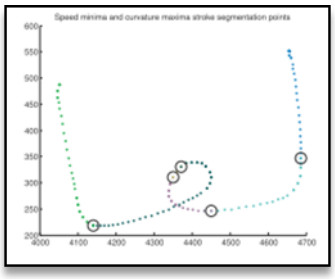

Figure 9: Pen Speed Based Segmentation

- **Merged of all above methods**

This method identify the segmentation points according to all above methods, then nearly coincident segmentation points are identified and merged with highest weight given to intersection points. However, the rate of correctly segmenting word parts into letters using this method is in the range of 60% [27], Figure 10 represents points after merging.

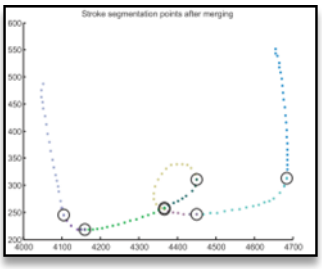

Figure 10: Points after Merging

- **genetic algorithm based segmentation**

Genetic algorithm is a class of optimization and search methods that use randomness to avoid local extreme. They are capable of adaptive and robust search over a wide range of space topologies. is an iterative algorithm that depends on the generation-by-generation development of possible solutions, with selection schemes permitting the elimination of bad solutions and the replication of good solutions that can be modified. There are three stages in a genetic search process: selection, crossover and mutation [25][26].

- **Manual (Explicit) Segmentation**

The connection between two Arabic letters usually is horizontal. This means that horizontal segments in the input are potential segmentation points. This method placed manually a segmentation point in horizontal segment

if it's long enough. Since the writing is online, the system can inform the user immediately when a segmentation point is placed. This feedback helps the user in controlling the length of the horizontal segments and hence achieving better recognition rate [17]. In [24], they used this method only in the training but not in the recognition.

- **Dynamic Time Warping approach**:

it is generalized to cursive recognition by finding the best pattern chain that match the input. This method implicitly places the segmentation points on the input since it finds the part that matches each letter in the best chain. The essence of this method is to compare all the possible letter chains against the input and to choose the nearest chain. The number of possible chains is very prohibitive. Once again, Dynamic Programming is applied to find this chain in reasonable amount of time [17].

3.4 Feature Extraction

The purpose of feature extraction stage is to only produce the relevant features of the input data because not all data points acquired by tablet are equally relevant or useful for recognition stage [24]. Input data can be represented as sets of different features.

The selected set of features should be small, whose values efficiently discriminates between patterns of different classes. A trade-off between the accuracy provided by the feature and its computational requirements must be handled carefully[29].

Freeman code is one of the simplest and common used features. It has eight directions where each direction has a specific code see figure 11. Using this feature depends on the direction of each stroke of the input where every stroke is attributed to one Freeman direction. The final representation of the input writing will be a vector of codes of strokes that form the input writing see figure 12 [15][9] use this feature in their systems.

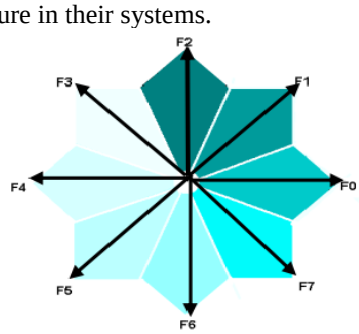

Figure 11: Freeman Code.

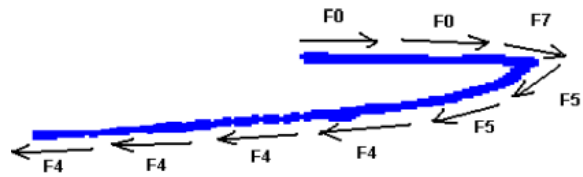

Figure 12: Applying Freeman Code on Input

Arapen [17] computes a set of features that represent the input and capture its properties that have good distinctive properties and help in discriminating between the different patterns:

1. Coordinates series: The x-y coordinates of the points that constitute the stroke
2. Tangent angles series: The angles of the line segments that constitute the stroke. See figure 13.
3. The Winding Value: The algebraic sum of direction changes. It's almost 0 for straight or s-shaped strokes and almost 360 for o-shaped ones.
4. The Aspect Ratio: The height to the width ratio.

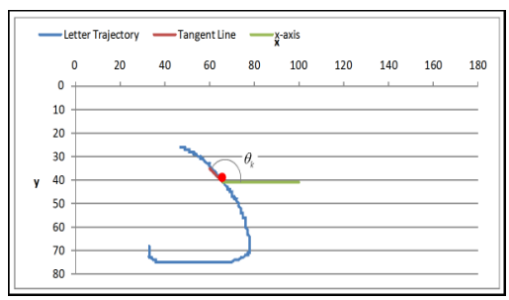

Figure 13: Tangent Line Concept

The system presented in [22] uses two layer features; global and local. The global features include loops, ascenders, and discarders. Loops are detected by inspecting the self intersection within the curve. Ascenders and descenders are defined with respect to lower and upper baselines. Figure [14] shows loops (letter Ain, Qaf) ascender (letter Alef, Lam), and descender (letter Ain) Whereas the local features characterize local relation between adjacent or nearby points on the polyline.

Figure 14: : Loop, Ascenders, and Descenders

3.4 Classification

Classification is the crucial and essential stage in the recognition process. It is a classification of each unknown object into one of a finite set of categories or classes. These categories could be a whole word, sub word, letter or even stroke according to the segmentation method used. The problem that makes the recognition of online handwriting recognition difficult is variation of shapes of the characters resulting from writing habits, styles, and the social and educational level of the writer [28][29].

3.6 Post-Processing

Post processing stage is important to increase the recognition accuracy and produce more meaningful results [24]. In this stage, some effort have been made to correct a word which is obtained from a classification stage. In most cases, a dictionary of highly frequently words is used to find the nearest string to that returned from the classification stage, if not found, the classifier will try to change the characters to find the correct one [28][24][35].

4 Results

In this section we present the results gained from this survey. Table[1] shows some recent researches in the field of online Arabic handwriting recognition. As may one thought, these systems cannot be comparable in their recognition rates since the data set , preprocessing, and the used features are not unique which play a very important role in the performance of a recognition system . Unfortunately, There is also no available large public corpus of on-line handwriting enabling performance comparison to these systems[32]. The only corpus is ADAB which is limited to 15158 Arabic words from 937 Tunisian wn/village names. Most of Arabic handwriting recognition in previous works focused on recognizing offline script [2] and most of the work in online script recognition field is done for isolated characters such as letters, digits and symbols[30]. In general, character based systems achieve higher recognition rates than word based systems where words are written in cursive manner. This may lead us to conclude that segmentation is a crucial step in almost all AOTR systems and that the main problems in Arabic text, especially its handwriting, are its cursiveness [21]. There is a tradeoff between recognizing the input word as a whole and avoiding the error-prone segmentation process and the training dataset size. We perceived that, to the best of our knowledge, there is no system that takes diacritics into consideration. All systems assume that the input data for letters or words are without diacritics. Refer to table 1.

5 Conclusion

This paper is the first survey about online Arabic handwriting recognition systems. It provides a review on the techniques that have been proposed to solve problems of Arabic online handwriting recognition along with recognition rates and description of the data set used for the approaches applied. It gives a researchers and interested people about the current states in the Arabic online recognition field which remains a challenging field even though the latest improvements of recognition methods and systems are very promising.

Table 1: Features of Available Arabic Hand Writing Recognition

Authors	Year	Data Set	Classifier	Recognition Rate
Saabni R. El-Sana J.	2009	For training: 10 writers wrote Arabic words that include all the Arabic letters in their different shapes. For testing: 6 users wrote 100 word-parts retrieved randomly from DB	Matching algorithm-Feature based technique	Max 90% Min 86%
Biadsy F. El-Sana J. Habash N.	2007	4 trainers – 800 words each 10 testers 280 words Use different data set size	Hidden Markov Model	(40k words) 89.75%
Omer M. Long Ma S.	2010	Train : 4 writers 3 times for each letter. = 336 5 writers to test the system, where each writer writes the 28	decision tree to classify a character to and then matching algorithm	97.6%
Sternby J. Morwing J. Andersson J. Friberg C.	2009	A dataset consisting of 40 persons entering words from a list of 66 Arabic words (totally 1578 samples) and each of the isolated forms of single characters has been collected testing set of 12 writers writing a total of 948 characters.	template matching	(65K words) 78.6% letters 91.9%
Monji Kherallah Lobna Haddad Adel M. Alimi	2009	24 writer 20000 word for train 14500 for test	Similarity between 2 strings (new)	91 %
Bilal Alsallakh, Hani Safadi	2006	Small corpus	2 levels 1-> mathematical matching technique 2-> DTW	50% Words 91% letters

References:

1. Liu,J. , Sun,J. and Wang S, “Pattern Recognition: An overview,” *IJCSNS International Journal of Computer Science and Network Security*, VOL.6 No.6, June 2006.
2. Plamondon, R.; Srihari, S.N.; , "Online and off-line handwriting recognition: a comprehensive survey ," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.22, no.1, pp.63-84, Jan 2000
3. Lorigo, L.M. and Govindaraju, V, “Offline Arabic handwriting recognition: a survey,” *Department of Computer Science and Engineering State University of New York*, USA, May 2006.
4. Al-Yousefi, H. and Udpa, S. S, “Recognition of Arabic Characters,” *IEEE Trans. Pattern Anal. Mach. Intell.* 14, 8, Aug.1992.
5. Connell, S. D., “Online Handwriting Recognition Using Multiple Pattern Class Models,” *Doctoral Thesis.UMI Order Number: AAI9971905., Michigan State University*, 2000.
6. Tappert, C.C. , Suen C.Y. , Wakahara T, “The State of the Art in Online Handwriting Recognition*,” IEEE Transactions on Pattern Analysis and Machine Intelligence* ,vol. 12, no. 8, pp. 787-808, Aug.1992.
7. Mahmoud, S.A.; Mahmoud, A.S.; , "Arabic Character Recognition using Modified Fourier Spectrum (MFS)," *Geometric Modeling and Imaging--New Trends*, 2006 , vol., no., pp.155-159, 16-18 Aug. 1993
8. Omer, M.A.H.; Shi Long Ma; , "Online Arabic handwriting character recognition using matching algorithm," *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference* , vol.2, pp.259-262, 26-28 Feb. 2010.
9. Jannoud A. I.; “Automatic Arabic Hand Written Text Recognition System,” *Damascus University, Damascus, Syria and Al-zaytoonah University, Amman, Jordan*, : pp 857-864, 2007
10. Srihari S., Ball G.; , “An Assessment of Arabic Handwriting Recognition Technology,” *Univ. of Buffalo, New work*, June 2007
11. El Abed, H.; Margner, V.; Kherallah, M.; Alimi, A.M.; , ", ICDAR 2009 Online Arabic Handwriting Recognition Competition," *Document Analysis and*

Recognition, 2009. ICDAR '09. 10th International Conference on , pp.1388-1392, 26-29 July 2009.

12. Al-Taani A., Al-Haj S.; ,"Recognition of On-line Arabic Handwritten Characters," *Yarmouk Univ, New Mexico State University*, Jordan, USA, 2010
13. Beigi H. at el. , "Challenges of handwriting recognition in Frasi,Arabic and other languages with similar writing styles, *an online digit recognizer*," New Work.
14. Kherallah M., Haddad L. , Alimi A.; , "A new Approach for Online Arabic Handwriting Recognition," *University of Sfax. REGIM: Research Group on Intelligent Machines*,2009.
15. Namboodiri A., Jain A.; , "Online Handwritten Script Recognition", IEEE *Trans. On pattern Analysis and Machine Inelegance*, Vol. 26, No. 1, January, 2004
16. Alsallakh, B.; Safadi, H.; , "AraPen: An Arabic Online Handwriting Recognition System," *Information and Communication Technologies*, 2006. ICTTA '06. 2nd , vol.1, no., pp.1844-1849.
17. Santosh K.C., Nattee C.;, "A Comprehensive Survey On On-Line Handwriting Recognition Technology And Its Real Application To The Nepalese Natural Handwriting ," *KATHMANDU University Journal of Science Engineering and Technology.* VOL. 5, No. I, JANUARY, 2009, pp 31-55.
18. Al-Taani A., Hammad M., ; "Recognition of On-line Handwritten Arabic Digits Using Structural Features and Transition Network," *Informatica 32 (2008) 275–281 275*
19. Khedher M. , Abandah G.; "Arabic Character Recognition using Approximate Stroke Sequence", *University of Jordan, Amman – Jordan.*
20. JUMARI K., ALI M.; ," A Survey And Comparative Evaluation Of Selected Off-Line Arabic Handwritten Character Recognition Systems," *Universiti Teknologi Malaysia, Journal Teknologi, vol.* 36 June 2002: pp.1–18.
21. Saabni, R.; El-Sana, J.; , "Hierarchical On-line Arabic Handwriting Recognition," Document Analysis and Recognition, *ICDAR '09. 10th International Conference* , pp.867-871, 26-29 July 2009
22. Shu, H. , "On-Line Handwriting Recognition Using Hidden Markov Model", *Massachuestts Institute*, Februray,1997.
23. Al-Habian, G.; Assaleh, K.; , "Online Arabic handwriting recognition using continuous Gaussian mixture HMMS," Intelligent and Advanced Systems, 2007. *ICIAS 2007. International Conference on* , pp.1183-1186, 25-28 Nov. 2007
24. Alimi, A.M.; , "An evolutionary neuro-fuzzy approach to recognize on-line Arabic handwriting," Document Analysis and Recognition, 1997., *Proceedings of the Fourth International Conference on , vol.1, no., pp.382-386 vol.1*, pp.18-20 Aug 1997
25. Kherallah M. et. al ; ," Global Recognition of the Arabic words by Genetic Algorithm and Visual Encoding ",REGIM: Research Group on Intelligent Machines, *Department of Electrical Engineering. University of Sfax*, ENIS, Tunisia.
26. Mahmoud A. Arram , Preprocessing and Segmentation of Cursive Online Arabic Script *MIT 6.870* Spring 2007
27. Al-Emami, S.; Usher, M.; , "On-line recognition of handwritten Arabic characters," Pattern Analysis and Machine Intelligence, *IEEE Transactions on* Machine Intelligence, vol.12, no.7, pp.704-710, Jul 1990
28. Khaled Al-ghoneim Onliee Arabic Character Recognition for Handheld Devices, *Department of Computer Engineering,King Saud University.(student project).*
29. Biadsy F. ;El-sana J. ; Habash N. "Online arabic handwriting recognition using hidden markov models", *The 10th International Workshop on Frontiers of Handwriting Recognition,2006*.
30. Mezghani, N.; Mitiche, A.; Cheriet, M.; , "On-line recognition of handwritten Arabic characters using a Kohonen neural network," *Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop* , vol., pp. 490- 495
31. Jakob Sternby, Jonas Morwing, Jonas Andersson, and Christer Friberg. 2009. On-line Arabic handwriting recognition with templates. *Pattern Recogn.* Vol. 42, No. 12 ,December 2009.
32. Rabiner, L.R.; "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE , vol.77, no.2, pp.257-286, Feb 1989
33. Bharath A, S.M.; , "Hidden Markov Models for Online Handwritten Tamil Word Recognition," *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference* vol.1, pp.506-510, 23-26 Sept. 2007