# تمييز خط اليد العربي بإستخدام خصائصهُ

## الباحثون

م.م. علي شاكر محمود*

م.مبرمج صباح جواد كاظم*

**المستخلص**

ان النظام المقترح يمكن أن يصنف على أنه نظام لتمييز خط اليد العربي، وأن هذا النظام يعتمد بألاساس على مرحلة تقسيم النص العربي الى أسطر ومن ثم الى كلمات لكي يتم تمييز الاحرف بعد ذلك، أن عملية التقسيم معقدة حيث تعتمد على مجموعة من الحسابات المتعلقة بالمسافة الفاصلة بين الاحرف وتقاطع الزوايا العمودية للاحرف.

في البداية يتم تحويل النص الى صورة وهذه الصورة تعتبر هي المدخل للنظام الذي يقوم بعد ذلك بصقل (تنعيم) النص وتغيير حجمه أذا تطلب الامر.

أيضا تهتم هذه الدراسة بتمييز خط اليد المكتوب ضمن قياسات دقيقة حيث ان هنالك مجموعة من الخصائص تعتمد على نمط الكتابة نستطيع التعبير عنها باعداد صحيحة للاستفادة منها في عملية التمييز.

*الجامعة المستنصرية / كلية التربية / قسم علوم الحاسبات

# Arabic Handwritten Recognition by using His Features

**By**

**Ali Shaker Mahmood**
**and**
**Sabah Jawad Kadem**

## Abstract

*The present study is a contribution in the filed of Arabic character recognition. The first step is to enter an image from a scanner, and then some special processes are done on the image like (smoothing and thinning and resizing) before continuous to segmentation phase. Then modified moment invariant computed using the shape boundary of the character only is utilized as features.*

*Also this study is concerned with the recognition of handwritten Arabic character drawn on a graphic tablet. At first feature that are suitable for recognition are proposed. Feature that are found to be independent of the writer style are represented as a list (vector) of integer values.*

*The designed system in general view is an Arabic text recognition system basically includes a segmentation stage in order to recognize handwritten Arabic words. The segmentation process is completely based on tracing the outer of given word and calculating the output distance between the extreme points of intersection of the contour with vertical line. At the output of the segmentation stage the cursive word is presented as a sequence of isolated character contour.*

* Al-Mustansiriyah University / College of Education / Computer Science Department

## 1. Introduction

Character recognition has been an active area of pattern recognition, not only because it improves man-machine communication, but also because it provides a solution for processing large volumes of data automatically **[1]**. Many systems of character (e.g. Latin, Chinese, Korean, etc.) have been investigated for automatic recognition using both off-line (i.e. digitize images) and on-line data acquisition techniques. In general, the recognition of handwritten digitized cursive script known to be the most challenging problem due to the variations in writing styles and the lack of dynamic information related to the shape of the pattern. For more than six decades, pattern recognition field has been of tremendous importance. Therefore, related studies have increased in order to create or even to improve existing commercial applications such as credit-card readers, postal sorting machines, etc. several methods for recognizing Latin and Chinese character have been proposed, However, the machine recognition of Arabic text has not been fully explored **[2], [3]**. The major involved difficulty in processing Arabic printed text is similar to that of cursive Latin. Henceforth the connectivity between characters complicates each character's segmentation from the word in which it takes place. Furthermore, Arabic character's shape variants in various word positions create additional problems in the recognition process **[4]**.

## 2. Literature Review

Related researches on Arabic have begun for only four decades. The first work they used a structural classification method for recognition online handwritten isolated Arabic characters. Features such as the shape of the main stroke, the number of strokes, the number and position of the stress marks are then extracted from the character. Then, secondary features of the main stroke such as the frame size, the start point and the curative are extracted using a distance function in order to remove the ambiguity and provide an exact match **[5]**.

The structural recognition system of Arabic handwritten consists of four phases. The first is preprocessing, in which the word is thinned and the middle of the word is calculated. Since it is difficult to segment a cursive word into letters, words are then segmented into separate stroke and classified as strokes with a loop, and complementary characters. These strokes are then further classified using their geometrical and topological properties. Finally, the relative positions of the classified strokes are examined and the strokes are combined in several steps into the string of characters that represents the recognized word. Most errors were due to incorrect segmentation of words **[6], [7]**.

## 3. Example of Handwritten Application

Cheques processing involves all the tasks a bank officer may have to do to procession incoming cheque for client. This includes: accessing account numbers, verifying names and signatures on the cheque, verifying the date of the cheque, matching the legal amount with the courtesy amount and verifying the paying ability of the issuer **[8]**.

## 4. Properties of Arabic Printed Characters

Arabic writing can be in general, classified into typewritten (Naskh), handwritten (Ruq'a) and artistic (Kufi, Diwani, Royal and Thuluth) styles.

Arabic is written from right to left. Arabic text is cursive in general i.e. letters are normally connected on the writing line known as midline.

The Arabic characters are consisted by loops, curves and line segments. In addition, some Arabic characters have been the same shapes (Figure-1), which are distinguished from each other only by the addition of secondary parts (Figure-2), which are position above, below or inside the character. Furthermore, an Arabic character can have different shapes depending upon its position in the word. This different shapes increase the complexity of recognizing Arabic text **[9]**.

| | | |
|---|---|---|
| ح | خ | ج |

**Figure (1) Three character with the same shape**

| | | | |
|---|---|---|---|
| ٠٠ | ٠ | ∴ | ء |

**Figure (2) Different secondary parts**

Arabic is semi-cursive in nature. Out of the 28 basic letters, 22 are cursive letters while 6 are non-cursive. Within one word, a cursive letter should be connected to the succeeding letter, while non-cursive letter can not be connected to any succeeding letter. Thus, an Arabic word may be decomposed into more than one sub-word, each represents one or more connected letters with their corresponding secondary components. In addition, Arabic defines two types of secondary components which are Dots and Hamazah (a zigzag-like shape). The number and position of secondary component play a factor in identifying different letters. Due to connectivity, an Arabic letter may change significantly depending on its position within a sub-word, identity of neighboring letters, the writing font, and the way the writer connects successive letters. Arabic handwritten

letters differ in height and width. Moreover, Arabic allows the presence of diacritics that control the pronunciation of words and possibly their meanings. However, such diacritics are only used in formal documents or in cases of contextual ambiguity. Unlike Latin, Arabic is written from right to left. The vocabulary of Arabic legal amount is larger than those found in Latin languages **[10]**.

This is due to three major factors:

1. Arabic has three different forms singular, double and plural (Figure-3).

| | | |
|---|---|---|
| آلاف | الفين | الف |

**Figure (3) Singular, double and plural forms for the word "thousand"**

2. Double and plural nouns have up to four different forms according to their grammatical position see (Figure-4).

| | | | |
|---|---|---|---|
| الفا | الفان | الفي | الفين |

**Figure (4) Four grammatical forms for the same word**

3. Two forms are defined for feminine and masculine countable things see (Figure-5).

| | |
|---|---|
| ثلاثة | ثلاث |

**Figure (5) Feminine and masculine forms of the word "three**

In addition, a few common spelling mistake and/or colloquial occur in writing some Arabic numbers (Figure- 6, 7).

| | |
|---|---|
| ثلاثه | ثلاثة |

**Figure (6) Secondary components of the last letter have been ignored**

| | |
|---|---|
| مئة | مائة |

**Figure (7) Two common forms for the word "hundred"**

These factors affect the identity of letters and the number of sub-words in a word. We found more different words than sub-words in the lexical. That was one of the reasons to consider sub-word as the basic unit of Arabic legal amounts. However, this does not prevent others from using words as their basic units. In principle, Arabic allows legal amounts to be written in any order, i.e. starting from the most significant digit, from the least significant digit of from the middle. However, eloquence measurements and people habit excluded most permutations **[11]**.

## 5. Arabic Feature Class

The Arabic alphabet consists of 29 basic characters (with LAMALEF "لا") shown in (Table-1).

| | | | | |
|---|---|---|---|---|
| أ<br>**ALEF** | ب<br>**BAH** | ت<br>**TAH** | ث<br>**THEH** | ج<br>**GEAM** |
| ح<br>**HAH** | خ<br>**KHAH** | د<br>**DAL** | ذ<br>**THAL** | ر<br>**RAH** |
| ز<br>**ZEAI** | س<br>**SEEN** | ش<br>**SHEEN** | ص<br>**SAD** | ض<br>**DAD** |
| ط<br>**TAH** | ظ<br>**ZAH** | ع<br>**AYN** | غ<br>**GHAYN** | ف<br>**PHAA** |
| ق<br>**KAAF** | ك<br>**KAEF** | ل<br>**LAM** | م<br>**MEAM** | ن<br>**NOON** |
| هـ<br>**HEAH** | و<br>**WAW** | لا<br>**LAMALEF** | ي<br>**YEH** | |

**Table (1) Arabic alphabet**

These characters differ from other systems of characters in their structure and in the way they connect to form words. The same characters may take different shapes according to its position in the word. For example the Characters "ghayn" has four different shapes according to its appearance at the head in the middle at the tail of a word or if it is isolated see (Figure-8). This feature increases the number of Arabic characters to about 60 different shapes **[12]**.

| | | | |
|---|---|---|---|
| غ | ـغ | ـغـ | غـ |
| **Isolated** | **Tail** | **Middle** | **Head** |

**Figure (8) Different shapes according to appearance**

## 6. Structure of Handwritten Arabic Characters [13]

There are many characters which are

1. Presence of a main stroke

Generally, an Arabic character is written as a single stroke with no need to lift up the pen or to rewrite over a written stroke. This stroke is defined as the "main stroke", an example of that see (Figure-9).

The main stroke is the longest continues portion of the character, which is written, before lifting up the pen.

| | | | |
|---|---|---|---|
| م | ل | ع | هـ |
| **MEAM** | **LAM** | **AYN** | **HEAH** |

**Figure (9) Main stroke**

2. Dots

Some of the different Arabic characters have exactly the same shape however they are distinguished from each other by the addition of a number of dotes up to three dots in different positions relative to the main character stroke. Dots appear in about 50 % of the characters, for example e see (Figure-10).

| خـ | حـ | جـ |
|---|---|---|
| **KHA** | **HAA** | **GEM** |
| **One dot above** | **Zero dot** | **One dot below** |

**Figure (10) Dot position for the shape"حـ"**

3. Secondary stroke

A part from the dots described before there are six shapes in Arabic alphabet which the pen has to be lifted up during handwriting, these characters are present in (Figure-11).

| أ | ك | ظ | ط | كـ | لا |
|---|---|---|---|---|---|
| **ALEF** | **KAF** | **ZAH** | **TAH** | **KAF** | **LAMALEF** |

**Figure (11) Secondary stroke characters**

The secondary stroke is the longest continuous portion of the character witch is written after the main stroke, see following (Figure-12). Any Arabic character has exactly one main stroke and zero or one secondary stroke.

ا + صـ = طـ

**Figure (12) Character "TAAH" is broken-down**

4. Feature stability

In general the number of secondary stroke and the number of dots and their relative positions are stable features of Arabic characters while the structure of the main strokes is the most variant portion of the characters, see following (Figure-13).

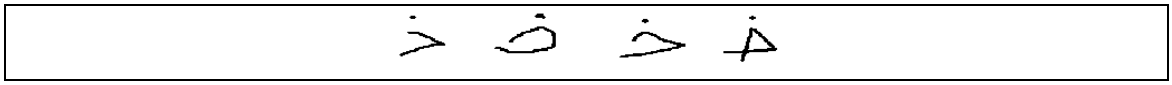


**Figure (13) Some handwritten samples of character "KHAA"**

Note that the variation of the hand written character ("KHAA", "خـ") is mainly in the main stroke while there is no variation in the number of dots (one dot) nor in its relative position with respect to the main stroke.

5. Representation of the main stroke

After preprocessing, the main stroke of the character is coded into a string of primitive in the plane .The eight primitive used are shown in (Figure-14) where each primitive is a multiple of 45, and its length is a design parameter which is adjusted to give the best performance. Example of preprocessing and primitive extraction process sees (Figure-15).

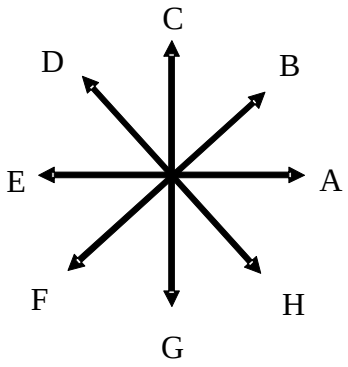



**Figure (14) Eight Primitive**

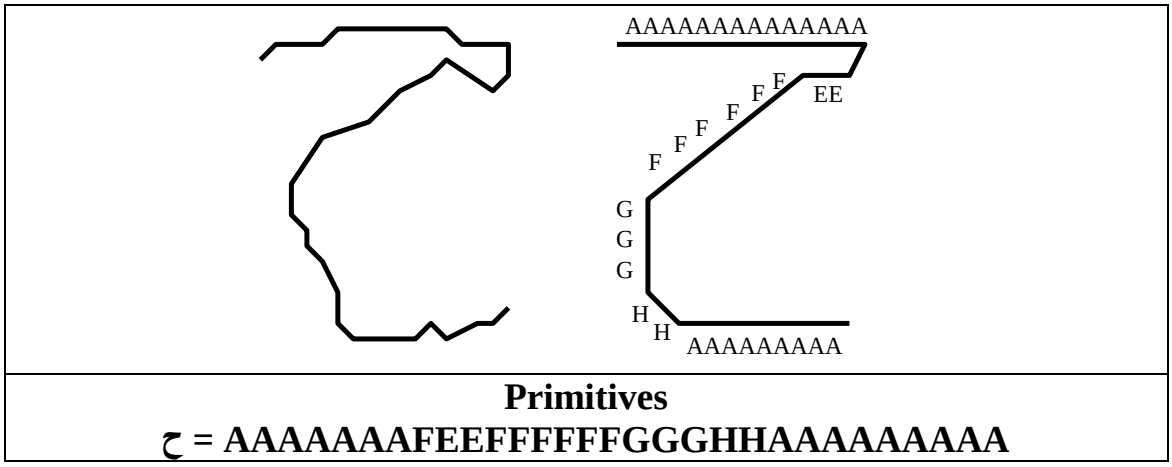



**Figure (15) Preprocessing of character "HAA"**

## 7. System Implementation

Arabic handwritten text pass throw many processes to convert it from text to separate character, these processes can be summarized in the following algorithm.

1. Image Reading Step

The process of enter text image to a system have two steps which are (digitalize image and convert image to a binary form).

1.1 Image Digitalization: There are two ways to digitalize a text (flat-bed scanner and hand-held scanner); the used way in this system is a flat-bed scanner because the output of hand-

held scanner is affected by hand motion creating thus local fluctuation.

1.2 Convert Digital Image to a Binary Form: The image is converted into (0, 1) labels. 1 and 0 represent object and the background as well as isolated black pixel over the background. Smoothing the image is carried out in order to fill up small holds and eliminate small spurs.

2. Smoothing Step

As the scanned image has different distortions and noise, thinning shape contains some false nodes. These nodes (treated as defected) do not have serious enounce on the quality of the restored image, but their description requires a lot of additional memory. The shape is smoothed to eliminate these defects.

3. Segmentation Step

The problem of separating lines or words in unconstraint handwritten is still difficult, where some other works assumes that words are already isolated by large amounts of white space or that words are written in boxes whose locations is known. However the segmentation process for the character recognition can be divided into three levels: line segmentation, word segmentation and characters segmentation. The levels of segmentation are described as follows:

1.1 **Line segmentation*:*** Text lines are segmented by pointing out the projection profile's valleys. The position between two consecutive midlines of smallest profile height denotes one boundary line.

1.2 **Word segmentation*:*** after segmenting a text into lines, it is vertically scanned if in one vertical scan two or less black pixels are encountered then the scan is subsequently denoted by 0; else the scan is denoted by the number of black pixels. In such a way, a vertical projection is built. Now, if in the profile there exist a run of a least k consecutive 0's then the midpoint of that run is considered as the boundary of a word. The value of k is taken as a half of the text line height.

1.3 **Character segmentation*:*** its main purpose is to build the vertical projection profile of the word's middle zone (note that the middle zone is chosen so that the stress marks are not belonging to it). A fixed threshold is used for segmentation a word into characters. Form the threshold level the algorithm look up the break along the vertical projection profile.

4. Repeat step 3 until the end of image text is reached.

The following figure (Figure-16) illustrates an example:

**Figure (16) Character segmentation example**

## 8. Representation of Stable Feature

The Arabic characters have many features some of this features are similar in some characters and other are not, these features are used in this system to distinguish between Arabic characters.

The shared feature are put in a vector called feature vector (**FV**), which have five fields, four of them fields are assign for stable features like (Number of Dotes and Relative Position of Dots) and the last one for unknown pattern, The table below (Table-2) describe the detail structure of feature vector.

| **FV** | | | | |
|---|---|---|---|---|
| **F1** | **F2** | **F3** | **F4** | **F5** |
| Number of Dots **(NDOTS)** | Relative Position of Dots **(PDOT)** | Number of Secondary Stroke **(NSS)** | Slop of Secondary Stroke **(SSS)** | Reject Pattern **(RP)** |

**Table (2) Feature vector structure**

The feature vector can be represented for each Arabic character, where the first column in a feature vector (**F1**) determines the Number of Dots (**NDOTS**) in the character, this column have a one digit from the set (**0**, **1**, **2 and 3**), where (**0**) refer to the character with zero dots, (**1**) refer to the character with one dots, (**2**) refer to the character with two dots and (**3**) refer to the character with three dots. The detail information about first column in a feature vector are illustrate in (Table-3).

| **F1** | **Number of Dots (NDOTS)** | **Character** |
|---|---|---|
| 0 | Zero Dot | ح د ر س ص ط و ه م ل ع |
| 1 | One Dot | ض ز غ ف ن ب ج خ ذ ظ |
| 2 | Two Dots | ت ق ي |
| 3 | Three Dots | ث ش |

**Table (3) Number of dots in Arabic character**

The second column in a feature vector (**F2**) determines the Relative Position of Dots **(PDOT)**, the relative position of the last written dot with respect to the character, this column have a one digit from the set (**1**, **2**, **3 and 4**), where (**1**) refer to the position of dots above the character, (**2**) refer to the position of dots within the character, (**3**) refer to the position of dots below the character and (**4**) refer to the position of dots within or above character. The detail information about second column in a feature vector are illustrate in (Table-4).

| F2 | Relative Position of Dots (PDOT) | Character |
|---|---|---|
| 1 | Above the Character | ف غ ز ض ش ق ت ظ ذ خ ث |
| 2 | Within the Character | ن ج |
| 3 | Below the Character | ب ج ي |
| 4 | Within or Above Character | ث ت ف ن |

**Table (4) Relative position of dots in Arabic character**

The third column in a feature vector (**F3**) determines the Number of Secondary Stroke **(NSS)** in the character, this column have a one digit from the set (**0 and 1**), where (**0**) refer to the character with zero secondary stroke and (**1**) refer to the character with one secondary stroke. The detail information about third column in a feature vector are illustrate in (Table-5).

| F3 | Number of Secondary Stroke (NSS) | Character |
|---|---|---|
| 1 | Zero Secondary Stroke | ش ه د ع ل ي |
| 2 | One Secondary Stroke | ظ ك لا ط |

**Table (5) Number of Secondary Stroke in Arabic character**

The fourth column in a feature vector (**F4**) determines the Slop of Secondary Stroke (**SSS**) in a character, this column have a one digit from the set (**0**, **1 and 2**), where (**0**) refer to the slop of secondary stroke, calculated using the first and last point in the secondary stroke, ranging from 0 to 90, (**1**) refer to the slop of the secondary stroke ranges from 90 to 180 and (**2**) refer to the unknown stroke direction (don't care). The detail information about fourth column in a feature vector are illustrate in (Table-6).

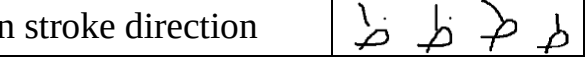
| F4 | Slop of Secondary Stroke (SSS) | Character |
|---|---|---|
| 0 | ranging from 0 to 90 | ك ك |
| 1 | ranging from 90 to 180 | لا |
| 2 | unknown stroke direction |  |

**Table (6) Slop of Secondary Stroke in Arabic character**

The last column in a feature vector (**F5**) determine reject vector which assigned to any unknown pattern that called Reject Pattern **(RP)**.

The Arabic character are classified into ten feature vectors, each one have five features explained above, the collection of these feature vectors make a feature matrix this matrix are used for matching process for each obtained feature vector. The following table (Table-7) gives a feature matrix.

| | **NDOT** | **PDOT** | **NSS** | **SSS** | **RP** | |
|---|---|---|---|---|---|---|
| | **F1** | **F2** | **F3** | **F4** | **F5** | |
| **FV1 =** | 0 | 0 | 0 | - | - | ح د ر س ص ع ل م هـ و لا ى حـ ـح سـ صـ ـع عـ ـعـ ـلـ مـ |
| **FV2 =** | 1 | 1 | 0 | - | - | خ ذ ز ض غ ف ن خـ ـخ ضـ ـغ غـ ـغـ |
| **FV3 =** | 1 | 3 | 0 | - | - | ب بـ جـ |
| **FV4 =** | 0 | 0 | 1 | 0 | - | أ ط ك كـ |
| **FV5 =** | 2 | 4 | 0 | - | - | ت تـ ق قـ |
| **FV6 =** | 3 | 1 | 0 | - | - | ث ثـ ش شـ |
| **FV7 =** | 1 | 2 | 0 | - | - | ج ـج ن |
| **FV8 =** | 0 | 0 | 1 | 1 | - | ط لا |
| **FV9 =** | 2 | 3 | 0 | - | - | يـ |
| **FV10 =** | 1 | 4 | 1 | 2 | - | ظ |
| **Otherwise** | - | - | - | - | 1 | Reject pattern |

**Table (7) Feature matrix**

## 9. Conclusion

The recognition of Arabic handwritten is difficult because an Arabic character may change depending on its position within a word, identity of neighboring letter, the writing font and letter differ in high and width therefore Arabic letter are classified into groups a according to there features in a feature vector and collect these feature vectors into feature matrix.

When the text are segmented into line, words and character, each segmented character have its own feature vector, the matching process done between feature vector and feature matrix, if has a match then go to the next character else this is unknown pattern and reject it.

## 10. References


1. M. Khorsheed and W. Clocksin, "**Structural Features of Cursive Arabic Script**", Nottingham University, UK, 1999.
2. M. Vatkin, "**The System of Handwritten Characters Recognition**", National Workshop on Signal and Image Processing, Algeria, 2002.
3. L. Oliveira, "**Feature Subset Selection Using Genetic Algorithm for Typewritten Digital Recognition**", Pontifical Catholic University of Parana, Brazil, 2005.
4. T. El-Sheikh and R. Guindi, "**Computer Recognition of Arabic Cursive Scripts**", Cairo University, Egypt, 1998.
5. M. El-Wakil and A. Shoukry, "**On-line Recognition of Handwritten Isolated Arabic Character**", Pattern Recognition Journal, Egypt, 1989.
6. A. Kundu and P. Bahil, "**Recognition of Handwritten Word by Using First and Second Order Markov Model Based**", Department of Electrical Engineering, USA, 1998.
7. F. Parhami and M. Taraghi, "**Off-line Farsi Character Recognition and Verification**", International Asian Conference on Computer Vision, India, 2003.
8. Y. Al-Ohali, M. Cheriet and C. Suen, "**Data Bases for Recognition of Handwritten Arabic Cheques**", Concordia University, Canada, 2000.
9. S. Maddouril, H. Amiril, and A. Belaid, "**Local Normalization towards Global Recognition of Arabic Typewritten Script**", Laboratory of Systems and Signal Processing, Tunis, 2001.
10. W. Clocksin and M. Khorsheed, "**Word Recognition in Arabic Typewritten**", University of Cambridge, England, 2000.
11. A. Belaid and C. Choisy, "**Human Reading Based Strategies for Off-Line Arabic Word Recognition**", University of Maryland, USA, 2006.
12. L. Meslati and M. Sellami, "**Hybrid Approach for Arabic Literal Amounts Recognition**", Badji Mokhtar University, Algeria, 2004.
13. D. Popel, " **Compact Graph Model of Handwritten Images: Integration into Authentification and Recognition**", Baker University, USA, 2002.