グラフィクスとCAD　54－7
（1991. 12. 20）

アラビア文字認識のための特徴抽出

モハメドファキール　袖山　忠一

長岡技術科学大学　電気系

あらまし　アラビア文字認識のための文字の切り出しと特徴抽出の方法を提案した。切り出しの方法は、文字の水平垂直の投影された像の分析により行う。特徴抽出手法は、細線化画像から直線部の抽出のためハフ変換を使用した。直線抽出の過程として、閾値を各セルに適用することにより、高いレベルの幾何学的意味をもつセルを残すことができる。セルの座標（R，θ）は抽出された特徴である。文字は抽出した直線の組により表される。これらは、３つの特徴要素（R　θ　L）から構成される。R、θはストロークの位置と方向であり、Lは長さである。　切り出しの成功率は９８％であった。

On the Features Extraction for the Recognition of Arabic Scripts

Mohamed Fakir　　Chuichi Sodeyama

Nagaoka University of Technology, Dep. of Electrikal Engineering

Kamitomioka 1603-1, Nagaoka Niigata 940-21

Abstract In this study, segmentation and features extraction methods for the recognition of Arabic printed scripts are proposed. The segmentation method consisted by analyzing the horizontal and the vertical projection profiles of the text. From the skeleton, feature are extracted by using Hough transform . A threshold was applied to every cell and those cells whose count was greater than the threshold was selected. The threshold eliminates low count cells regarded as noise and leaves high valued cells that have geometrical meaning in the image space. The coordinates (R, θ) of the remaining cells are the extracted features. A symbol was represented by a set of strokes, and each stroke consists of three feature elements : R θ L , which forms a feature vector where R , θ describe the position and direction of the stroke and L represent its length. For the segmentation phase a 98 % rate of success was obtained.

1.INTRODUCTION

In the past two decades valuable works has been noticed in the area of characters recognition, and a larger number of technical papers and reports in the literature are devoted to the topic. This subject has attracted an immense research interest not only because of the very challenging nature of the problem , but also because it provides a means for automatic processing of large volumes of data such as postal codes, for office automation, and for a large variety of other business and scientific applications.

The difficulty of the text recognition greatly depend on the type of characters to be recognized. The difficulty varies from that needed to process relatively easy mono fonts to the that of extremely difficult cursive text. Several efforts have been devoted to the recognition of cursive script but still remains an unresolved problem. In general, however a cursive word is recognized through a hierarchical analysis, i.e., a word is decomposed into letters, letters into strokes and strokes into primitives elements [1],[2].

The difficulty involved in processing printed Arabic text is similar to that of cursive latin. This is primarily due to connectivity between letters that complicates the segmentation of each letters from the word in which it occurs. In addition to connectivity, the variant shape of Arabic characters in different word position creates another problem in recognition.

To my knowlegde only few researchers have contributed to the recognition of Arabic characters [3, 4, 5, 6]. Therefore the problem of the recognition of Arabic still an open field.

Although characters are sensive to noise. Therefore, how to extract strokes become the main problem in this field. This may be solved by the selection of the useful features customarily defined in the automatic characters recognition as two types: global and local features. The principle of global features is based on the transformation which can map the character matrix into a new domain to extract features. The selection of local features is based on geometrical and topological properties of the character, such as strokes direction , strokes density, strokes lenght and position, etc.

Unlike chinese characters, Arabic characters are formed by loops or strokes or curves. This make it difficult to globally describe a character in one parametric form.

The Hough transform (HT) method transforms the character from the spatial domain into the parametric domain to extract strokes. This a new attempt at the stroke extraction of multifont Arabic printed characters. There are two reasons for the motivation. The first one is that some strokes constutying Arabic characters are linear and can be detected by the HT as lines. The second one is that the HT method is simpler in computation than other transforms.

The process consists of five phases. After the first phase of preprocessing a word is segmented into what to be a characters in the second phase. In the third phase character main shapes are normalized in size, and classified into groupes after thinned. In the fourth phase features are extracted in the Hough transform space. The four phases are explained in the following section. But before that a brief explanation about the characteristics of Arabic characters is given.

2.CHARACTERISTICS OF ARABIC

Arabic is a very old language. Its origin lies in the Aramic [6], initially used by the Nabatean poeple. Arabic is the language of the holy book "Quran", and is the religious language of all muslims. Spoken Arabic is perhaps one of the oldest language in the word, but written Arabic was probably originated at the 4th century AD. The earliest extant Arabic writing is a trilingual inscription in Greek_Syriac_Arabic of AD 512 [6].

Arabic scripts differ from Latin and chinese characters in many structural ways. Arabic text is cursive in general,i.e., Arabic letters are normally connected on the writing line to be called "Midline".

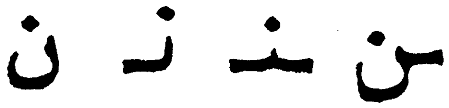
The Arabic scripts consist of 28 basic characters, which may vary in form depending on their position either in the word (at the beginning, in the middle, at the end) or in isolation (see Fig.1). Some characters differ only by the number or the position of the stress marks (ب , ت) above or below the midline.

In addition to the connectivity characteristic of Arabic scripts, the vowel diacritics are an essential part of the written Arabic. The presence or absence of vowel diacritics indicates differents meaning between what would otherwise be the same word. For example (علم) can indicated the Arabic for either "science" or "flag". If the word is isolated, diacritics are essential to distinguish between either of the meanings; though it occurs in a sentence, contextual information inherent in the sentence can be used to indicate the appropriate meaning.

In the proposed system, the issue of vowel diacritics is not threated, since it is more common for Arabic not to employ these diacritics. These are only found in old manuscripts or used in very confined areas.

3.PREPROCESSING PHASE

Four Arabic printed texts and 50 words



(a) (b) (c) (d)

Fig.1: Different shapes of character "Non". (a) Isolated, (b) Beginning of word, (c) Middle of word, (d) End of word.

isolated were entered to the microcomputer through an image scanner (300dpi). A preliminary processing operation of position normalization, smoothing, midline drift correction and lines separation are done. Before the description of these process a brief explanation about the Hough transform properties is given.

3-1. Properties of H.T.

The detection of straigh lines and other types of curves is a very operation in digital processing. One method that is frequently used for curves and lines detection is the Hough transform.

The H.T. was first proposed as method for detecting and identifying straigh lines in digital image by transforming each point in an image into sinusoidal curve in a discrete parameter space.

The normal representation of line is given by the following equation.

$$X*\cos\Theta + Y*\sin\Theta = R \qquad 0<=\Theta<180^{\circ} \qquad (1)$$

The meaning of the parameter used in the equation (1) is illustrated in Fig.2. Note that Θ is meseared with respect to the X axis. According to the equation (1), each point (x, y) can be transformed into curves in the R_Θ space, and the intersection points in many curves in the R_Θ space denotes a collinear line in the X_Y space. Hence collinear points can be detected by counting the number of intersections of curves in the R_Θ space.

The interesting properties of HT are the following.

a) A point in X_Y space (see Fig.3) corresponds to a sinusoidal curve in R_Θ space (see Fig.4).

b) Points lying on the same curve in the R_Θ space (see Fig.4) correspond to lines through the same point in the X_Y space (see Fig.5).

c) Points lying on the same straight line in the X_Y space (see Fig.6) correspond to curves through a common point in the R_Θ space (see Fig.7).

d) A point in R_Θ space (see Fig.7) corresponds to straight line in the X_Y space (see Fig.8).

3-2. Smoothing

When a patterns are scanned and

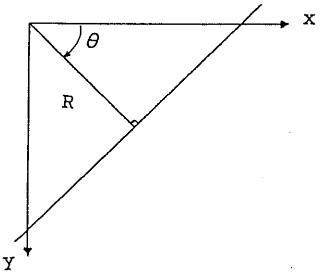


Fig.2 Normal representation of a line.

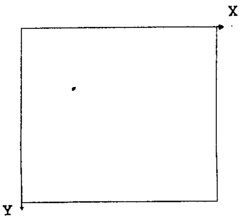


Fig.3

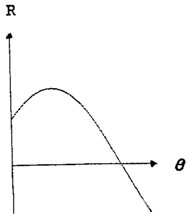


Fig.4

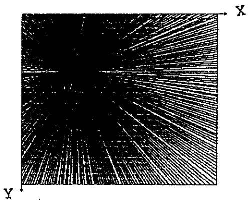


Fig.5

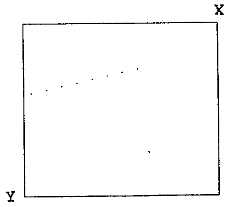


Fig.6

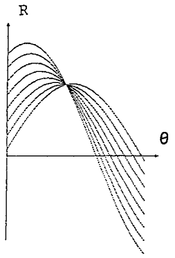


Fig.7

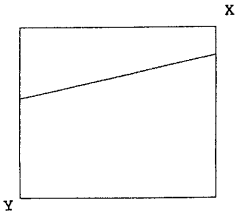


Fig.8

digitized, the raw data may carry a certain amount of noise depending on the resolution of the scanner and the scaning process employed.

Unwanted noise may cause severe distortion of the pattern which in turn will affect the accuracy of the final classification process.

The smoothing process used for Arabic text is the same as described in [7].

3-3. Position normalization

In many cases, patterns are normalized. Size, position, skew and line width are the main types of normalization used. In fact size normalization is a very technique used in pattern recognition, it facilitates computation and the feature extraction process. This will be used after character segmentation.

Here position normalization is done. Starting from the top to left corner, the image frame is scanned vertically from left to right until the first element of the word from the left is found. The distance of this element from the left side of the frame is noted as z. Next the image is scanned horizontally from the top line until the first element of the word from the the top is found. The distance of this element from the the top of the frame is noted as w. Likewise the distance of the last element of the word from the left of the frame is noted as x and that from the top of the word is noted as y. The tight boundary aroud the word is determined and lines are drawn around this boundary as shown in Fig.9.a.

The top left corner point (z, w) will be assigned a new reference point (Ø, Ø) in the following section.

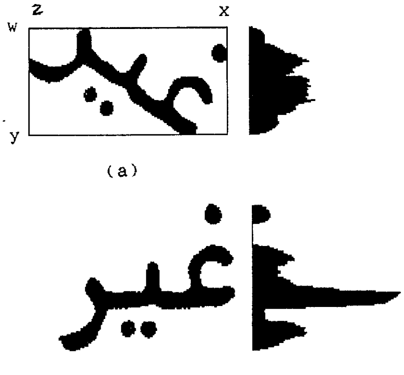
3-4. Midline drift correction [8]

The image of the text to be recognized may entered to the system slanted. This fact affect the accuracy of the segmentation and the recognition. In this case midline drift correction is needed.

The method is based on the extraction of the slop θ corresponding to midline which is defined as the line corresponding to maximum points in the horizontal projection profile (HPP) of the word. The slop θ is detected by observing heigh valued cell in the accumulative matrix in the Hough transform space. The midline drift is corrected by rotating the image of the word by θ-9Ø. Note that θ is measured with respect to the X axis. Fig.9.a, and Fig.9.b show respectively an Arabic word before and after midline drift correction.

3-5. Lines separation

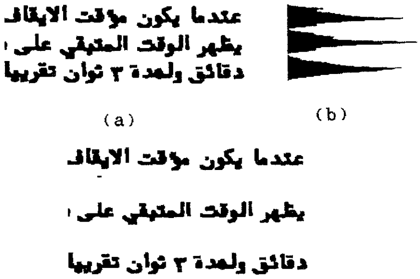
This process aims to separate the pairs of consecutive lines in the text. The process is based on the analysis of the horizontal projection profile (HPP) of the text. First the number of black points is found for each row in the HPP. This number is big at or near the center row of each text line. A fixed threshol is used to separate the pairs of consecutive lines.The threshold value is-choseen to be T=d/4, where d is the distance between two consecutive midlines. Fig.1Ø(a) &(c) illustrate respectively an Arabic text before and after this process.



(a)

(b)

Fig.9: (a) Before midline drift correction, (b) After midline drift correction.



(a) (b)

(c)

Fig.1Ø: (a) Before separation, (b) Horizontal projection profile. (c) After separation.

4.SEGMENTATION PHASE [9]

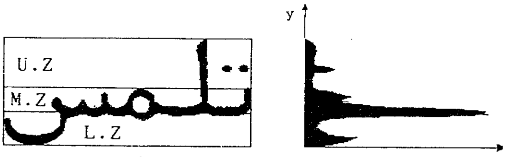
The segmentation phase is necessary in recognizing Arabic text. This phase consist of three parts: (a) Segmentation of a word into zones. (b) Segmentation of a word into characters. (c) Segmentation of a character into main part and stress marks.

a) This part consists to segment a word horizontally by horizontal lines into three zones which are the lower zone (L.Z), the middle zone (M.Z), and the upper zone (U.Z)(Fig.11). Some lines may overlap due the fact that some zones may be noexistent (Fig.12). The method consists to built the horizontal projection profile of the word. The highest density value "max" is deemed the midline, and the first next "y" such that "d(y) <= th*max" is the upper baseline ("d" is the density function and "th" is a threshold). The threshold is choseen that the stress marks are not belonging to the middle zone.

b) After the process (a) the VPP of the middle zone is built. A fixed threshold is used for segmenting a word into characters. From the threshold level the segmentation algorithm searchs for the breaks along the VPP of the middle zone of the word(see Fig.13). The character formed by loop is segmented into two parts. However if the VPP does not follow the following rule, the character remains unsegmented (see Fig.14).

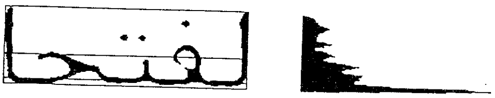
$$|D_i| < W*2/9,$$

where Di is the distance between the ith peak and the i+1 peak in the VPP, and W is the total width of the character. For more experimental result see Fig.15.



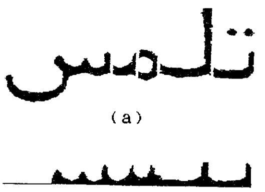
(a) (b)

Fig.11: (a) Main zones in a word, (b) Horizontal projection profile.



(a) (b)

Fig.12: (a) Word without the lower zone. (b) Horizontal projection profile.



(a)

(b)

Fig.13: (a) Result of segmentation.
(b) Vertical projection profile.

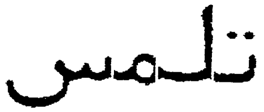
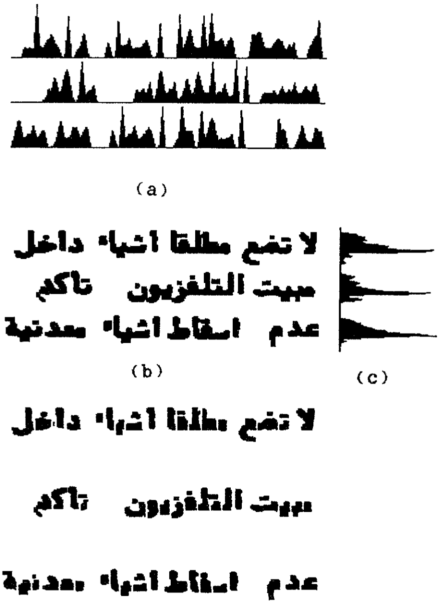


Fig.14 Result obtained after correction.

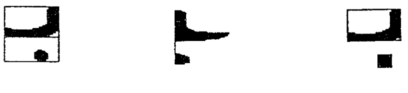


(a)

(b) (c)

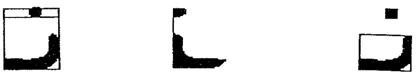
(d)

Fig.15:(a) Vertical projection profile, (b) Before segmentation, (c) Horizontal projection profile, (d) After segmentation.



(a) (b) (c)

Fig.16: (a) An Arabic character pulled from a word, (b) Horizontal projection profile, (c) After segmentation.



(a) (b) (c)

Fig.17: (a) An Arabic character pulled from a word, (b) Horizontal projection profile, (c) After segmentation.

c) For the characters which contain stress marks, a supplementary segmentation process is done. It consist to separe the main shape and the stress marks by making an horizontal projection after pulling the character from the word. This process is done by drawing lines between main shap and the stress marks if there is a white space in the horizontal projection profile. For experimental results see Fig.16 and Fig.17.

Therefore, these characters can be recognized by classifying both the main shape (ـٮ) and the stress marks. For example the following characters (ـث , ـب , ـي , ـت , ـئ) will be represented by the same shape (ـٮ).

5.SIZE NORMALIZATION PHASE

This process is used after segmentation process only for the main shape of the character. As different symbols have different hights and widths, now symbols will have the same frame through the boundary points. Note that a symbol is the name given to the main shape of a character.The size normalization process is as follows:

Let be (xi, yj) the coordinates of a point in the image before size normalization, and (x'i, y'j) the coordinates of a point in the normalized data. The symbols were normalized to 64 units such that

$$x'i = Rx(xi - xmin)$$

$$y'j = Ry(yj - ymin)$$

where Rx and Ry are the expanded rates in the x direction, and the y direction respectively given by the following equations.

$$Rx = Lx/(xmax - xmin) = Lx/lx.$$

$$Ry = Ly/(ymax - ymin) = Ly/ly.$$

xmin, xmax, ymin, ymax are known from the segmentation phase.

6.CLASSIFICATION PHASE

In this phase symbols were classified into groups after skeletonization process. The thinning algorithm which was used to obtain the skeleton of a symbol is the same as described in [1Ø] with modification in the first step conditions. The condition (a) is modifiyed from "2<=B(P1)<= 6" to "3<=B(P1)<=6". This takes care of the end points which should not be eliminated.

The task of this phase is first the reduction of the classification space dimensionality. The second reason is concerned with similar symbols and their possible mapping into the same region in the parameter space.

In this phase symbols are classified into groups by the location of end points within the skeleton frame. A end point is defined as point in the skeleton which has only one neighbor in eight neighbor notation. The symbol skeleton was enclosed in a rectangle that divided into four regions (see Fig.18. The coordinates center defined the border between quadrants. The border location varied from symbol to symbol. The resulting quadrants were labeled with a number from the set S={1,2,4,8} counterclockwise.

A search was conducted for end points in each quadrant and the symbol was assigned to some groups n by calculating $Gn = \sum_{i \in S} i * Ei$ the group code

$$Ei = \begin{cases} 1 & \text{if a least one end point exist in quadrant i.} \\ Ø & \text{otherwise.} \end{cases}$$

From the experimental results the maximun number of symbols in each groups was 8 and the number of groups was 12. Individual classification of symbols will be accomplished in the near future which will be based on the features selected in the Hough transform space.

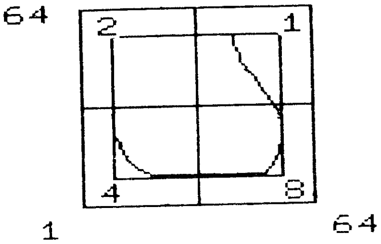


Fig.18: Zones division of a symbol matrix, with the quadrant codes imposed. The character enclosed by the interior rectangle has a group code 5.

7.FEATURES EXTRACTION IN HOUGH TRANSFORM SPACE

As mentined in section 3 the HT is a linear transforms originally developed for line detection in digital pictures. The transform is applied to all the image points and the resulting value of R given by equation (1) for each quantized value of θ in $Ø <= θ < 18Ø°$, define cells in the array. Whenever a cell is selected in this way, its content is incremented by one.

According to the structure of Arabic characters θ was quantized in step of 15°. Straight lines are then detected by observing high valued cells in the accumulative matrix H(R, θ). Peaks are found if the value of H(R, θ) exceeds a threshold value. Fig.2Ø illustrates the transform of the character illustrated in Fig.19.b. The Hough domain is the character skeleton and this image is transformed by (1); the contents of the parameter space cells after application of the transform are given in Table.1. The results in table.1 show that the goal of dimensionality reduction is attainable. Many entries in the parameter space equal zero and the number of high valued cells is relatively small.

As in the procedure for straight lines detection, a threshold T is applied to every cell and those cells whose count was greater than it was selected. Fig.21 illustrates the result obtained after application of H.T. For more experimentals results see Fig.22. The threshold eliminates low count cells regarded as noise and leaves high valued cells that have geometrical meaning in the image space. The coordinates (R,θ) of the remaining cells are the extracted features.

(a) (b)

Fig.19: (a) Before thinning,(b) Result of thinning process.

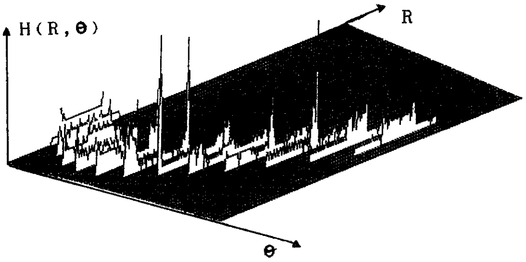
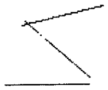


Fig.20: The array of accumulators content after transforming the character shown in Fig.19.b.



R	θ	L
0	135	19
43	75	17
80	90	21

(a) (b)

Fig.21: (a) Result obtained by HT,(b) Feature vector of Fig.19.b.

θ

R

-4:	.	.	.	.	.	.	.	.	2	2	1	1
-3:	.	.	.	.	.	.	.	.	.	.	1	1
-2:	.	.	.	.	.	.	.	.	1	1	2	1
-1:	.	.	.	.	.	.	.	.	2	4	1	1
0:	.	.	.	.	.	.	.	.	2	19	2	.
1:	.	.	.	.	.	.	.	.	2	4	1	.
2:	.	.	.	.	.	.	.	.	.	8	1	.
3:	.	.	.	.	.	.	.	1	2	8	1	.
4:	.	.	.	.	.	.	.	2	.	5	1	.
5:	.	.	.	.	.	.	.	3	2	1	1	.
6:	.	.	.	.	.	.	.	1	1	.	1	.
7:	.	.	.	.	.	.	.	2	2	1	2	.
8:	.	.	.	.	.	.	.	2	.	1	1	.
9:	.	.	.	.	.	.	.	1	2	2	1	.
10:	.	.	.	.	.	.	.	3	.	.	1	.
11:	.	.	.	.	.	.	.	3	1	1	1	.
12:	.	.	.	.	.	.	.	2	1	2	1	.
13:	.	.	.	.	.	.	.	1	1	1	1	.
14:	.	.	.	.	.	.	.	1	1	2	2	.
15:	.	.	.	.	.	.	.	2	1	1	1	.
16:	.	.	.	.	.	.	.	2	1	2	1	.
17:	.	.	.	.	.	.	.	1	1	1	1	.
18:	.	.	.	.	.	.	.	2	3	1	1	.
19:	.	.	.	.	.	.	.	2	4	2	2	.
20:	.	.	.	.	.	.	.	1	11	1	.	.
21:	.	.	.	.	.	.	.	2	6	2	.	.
22:	.	.	.	.	.	.	.	1	6	1	.	.
23:	1	.	.	.	.	.	.	1	1	1	.	.
24:	1	.	.	.	.	.	1	1	2	1	.	.
25:	1	.	.	.	.	.	5	1	2	1	.	.
26:	1	.	.	.	.	.	3	1	1	2	.	.
27:	1	.	.	.	.	.	3	1	1	1	.	.
28:	1	.	.	.	.	.	8	1	1	2	.	.
29:	1	.	.	.	.	.	1	1	1	1	.	.
30:	1	.	.	.	.	.	3	1	1	1	.	.
31:	1	.	.	.	.	.	3	2	1	2	.	.
32:	1	.	.	.	.	.	2	1	.	1	.	.
33:	1	.	.	.	.	.	3	3	1	2	.	.
34:	1	.	.	.	.	.	2	1	1	1	.	.
35:	1	.	.	.	.	.	2	2	2	1	.	.
36:	5	.	.	.	.	.	1	3	1	2	.	.
37:	3	.	.	.	.	.	1	3	2	1	.	.
38:	3	.	.	.	.	.	2	2	2	2	.	.
39:	3	.	.	.	.	.	1	3	2	2	.	.
40:	3	.	.	.	.	.	1	2	2	.	.	.
41:	3	.	.	.	.	1	1	2	2	.	.	.
42:	3	1	.	.	.	11	1	2	2	.	.	.
43:	3	1	.	.	.	17	1	2	2	.	.	.
44:	3	2	.	.	.	4	1	3	2	.	.	.
45:	3	5	.	.	.	2	2	1	2	.	.	.
46:	3	3	.	.	.	3	1	1	.	.	.	.
47:	3	3	.	.	.	1	1	2	2	.	.	.
48:	3	3	.	.	.	1	1	1	2	.	.	.
49:	3	3	.	.	.	1	1	1	2	.	.	.
50:	3	3	1	.	2	.	2	1	2	.	.	.
51:	3	3	4	.	11	1	2	1	2	.	.	.
52:	3	4	3	.	8	1	1	1	2	.	.	.
53:	3	3	3	7	5	2	2	1	2	.	.	.
54:	3	2	2	3	3	.	1	1	2	.	.	.
55:	3	3	2	4	4	1	2	1	2	.	.	.
56:	3	3	2	3	4	1	1	.	3	.	.	.
57:	3	3	3	4	5	1	2	1	1	.	.	.
58:	3	4	3	2	5	1	1	3	.	.	.	.
59:	3	3	3	2	1	1	1	.	.	.	.	.
60:	3	3	3	5	1	1	1	4	.	.	.	.
61:	3	2	3	2	1	2	2	4	.	.	.	.
62:	3	4	4	3	.	1	1	4	.	.	.	.
63:	3	3	3	1	1	.	2	4	.	.	.	.
64:	3	3	3	2	1	2	1	3	.	.	.	.
65:	3	3	4	4	1	1	1	1	.	.	.	.
66:	3	3	3	2	1	1	1	4	.	.	.	.
67:	4	3	3	5	1	1	1	4	.	.	.	.
68:	3	3	3	1	1	1	1	4	.	.	.	.
69:	5	4	4	2	1	2	1	3	.	.	.	.
70:	5	2	3	4	1	.	1	5	.	.	.	.
71:	1	3	3	1	1	1	1	1	.	.	.	.
72:	1	3	4	2	1	1	1	.	.	.	.	.
73:	1	3	4	2	.	1	1	.	.	.	.	.
74:	1	3	3	3	2	2	1	.	.	.	.	.
75:	1	3	3	1	1	.	1	.	.	.	.	.
76:	1	4	3	2	.	1	1	.	.	.	.	.
77:	.	2	4	3	2	2	1	.	.	.	.	.
78:	.	3	2	2	.	1	3	.	.	.	.	.
79:	.	3	2	3	2	.	12	.	.	.	.	.
80:	.	2	2	2	1	1	21	.	.	.	.	.
81:	.	3	3	1	3	1	.	.	.	.	.	.
82:	.	2	1	3	2	2	.	.	.	.	.	.
83:	.	2	2	2	4	3	.	.	.	.	.	.
84:	.	3	2	3	2	5	.	.	.	.	.	.
85:	.	4	2	1	3	5	.	.	.	.	.	.
86:	.	3	2	4	3	5	.	.	.	.	.	.
87:	.	.	3	2	3	6	.	.	.	.	.	.
88:	.	.	1	2	3	8	.	.	.	.	.	.
89:	.	.	2	3	3	4	.	.	.	.	.	.
90:	.	.	2	2	4	5	.	.	.	.	.	.
91:	.	.	2	3	3	7	.	.	.	.	.	.
92:	.	.	2	1	3	5	.	.	.	.	.	.
93:	.	.	2	2	3	.	.	.	.	.	.	.
94:	.	.	3	3	3	.	.	.	.	.	.	.
95:	.	.	2	2	3	.	.	.	.	.	.	.
96:	.	.	6	3	3	.	.	.	.	.	.	.
97:	.	.	2	2	3	.	.	.	.	.	.	.
98:	.	.	.	3	4	.	.	.	.	.	.	.
99:	.	.	.	2	5	.	.	.	.	.	.	.
100:	.	.	.	3	3	.	.	.	.	.	.	.
101:	.	.	.	7	.	.	.	.	.	.	.	.
102:	.	.	.	.	.	.	.	.	.	.	.	.

Table.1: The resulting parameter space after applying the transform to the character skeleton in Fig.19.b. The values signed by "○" denote the stroke extracted in Fig.19.b.

By variation of threshold value (T) during the experimement, an empirical value is selected to be T=12. Experimental results showed that the HT method can be used to detecte lines in Arabic symbols. After the peak detection a symbol was represented by a set of strokes, and each stroke consists of three features elements: R, θ and H(R,θ), which forms a feature vector where R, θ describe the position and direction of the stroke and H(R,θ) represent its length. Then symbol can be represented by a set of feature vectors V={(θi,Ri,Li)/i=1,..N} where N is the number of strokes. Then maching will be performed for recognition according this set of feature vectors.

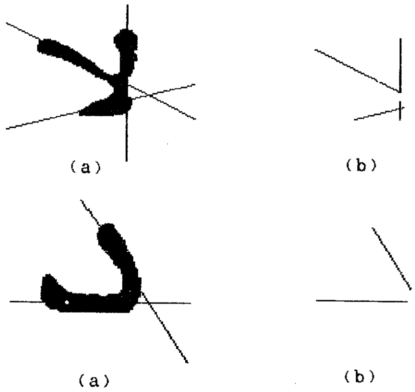


Fig.22: (a) Normalized symbols, (b) Result obtained by the application of HT.

8.CONCLUSION

This study has resolved some problems involved in the recognition of Arabic multifont text, and presented a feature extraction method using the Hough transform. Due to the linearity of some strokes composing Arabic characters the transform was selected.

Arabic texts were entered to the system through an image scanner. Texts are segmented into characters, then characters into main shapes and stress marks basing on the vertical and horizontal projection profile. A 98% rate of success was obtained.The character main shapes were normalized and thinned, then classified into groups by the location and presence of end points in predetermined zones of the symbol matrix. Thus, a problem of dimensionality reduction. Twelve groups were found to be sufficient for Arabic, with at most eight classes in each group. The transform were appliyed to the twelve groups. Each symbol were represented by a set of vectors, with which maching will be performed. The symbol(ل) and the symbol (ـٮ) are belonging to the same group, and having the same properties which resulted from the size normalization. The first symbol is a character, but the second one represents the main shape of the following characters " ـب , ـث , ـت , , ـي , ئ , ـن ". The same discussion can be done to the following symbols:" ـلـ " " ـٮـ ". The first one is a character, but the second one is the main shape of the following characters " ـبـ , ـتـ , , ـثـ , ـيـ , ـنـ ". These characters can be recognized by classifying both their main shapes and the stress marks. A future recognition system will be done based on the dynamic programing matching, and a structural classifier. The DP matching will be used to recognize the main shape of the character, and the structural classifier will be concerned to the stress marks.

REFERENCES

[1] M.Eden and M.Hall, "The characterisation of cursive writing" Proc. 4th symp. Informatics Theory, London 1961,pp:287-299.

[2] M.Berthod and S.Ayhan,"On line cursive script recognition: a structural approach with learning" in Proc,5th Int.conf. Pattern recognition, 1980,pp:723-725.

[3] A. Amin, "Machine recognition of handwritten Arabic words by the IRAC II system," in Proc.6th Int.conf.Pattern recognition, Mucnchen, Germany, 1982, pp.34-36.

[4] H.Almuallim and S.Y."A method of recognition of Arabic cursive handwriting" IEEE trans. vol.PAMI-9,No.5,1987, pp:715-721.

[5] N.Ula, et al "A variable limb-width matching algorithm for automatic recognition of Arabic characters", The 10th symposium on information theory and its application", pp:411-415, 1987.

[6] A. Dewachi, "Problem areas in the treatment of Arabic in hardware and software systems (Present and future prospects)", Workshop papers, Kuwait, Vol.1,pp:1-26,1985.

[7] J.Hasegawa,H.Koshimizu,A.Nakayama and S.Yokoi, "Image processing on personnal computer" 1986, pp:55-58.

[8] M.Fakir,C.Sodeyama, "Arabic Scripts Segmentation",Densi jyohou tuusingakki sinetsu shibu taiki,Tokushima,Japan,D-538,1991.

[9] M.Fakir and C.Sodeyama, "An approach to Arabic words recognition" Information theory and its application, ISITA 1990, pp:399-401, Vol.1. Hawaii, U.S.A.

[10] T.Y.Zhang and C.Y.Suen,"A fast parallel algorithm for thinning digital patterns", Communications of the ACM, 1984, Vol.27,Number 3, pp:236-239.