International Journal of Computer Theory and Engineering, Vol. 2, No. 1 February, 2010
1793-8201

Fuzzy Logic approach to Recognition of Isolated Arabic Characters

Majida Ali Abed, Hamid Ali Abed AL-Asadi, Zainab sabah Baha Al-Deen, Ahmad Naser Ismail

Abstract—In In recent years, fuzzy logic has been increasingly used to improve conventional methods especially in pattern recognition fields. This research present proposed system to recognize isolated Arabic characters, using a approach of fuzzy logic. The proposed system does not requires segmentation of the Arabic words or Arabic texts to characters but entered three different shape for every isolated Arabic character saved in many templates, after then applied laws and stages of fuzzy logic, save the results in different files, enter four shape for the character we wanted to recognize different from the three shapes were saved then compare the results recognition with the saved results of characters ,we used for testing the characters (Alif , Baa ,Geam ,Dal ,Raa,Seen ,Dhad ,T'ta, Ain ,Faa ,Kaf , Lam , Mem ,Haa ,Waw ,Yaa).Recognition results shown at many figures in the paragraph the experimental results for two different shapes Arabic isolated characters(Ain, Faa). The proposed system was tested on 96 different shape of characters and a 88% recognition success rate was obtained. The proposed system has been implemented and tested on Matlab R2008b environment.

***Index Terms*— Pattern Recognition, Fuzzy Logic (FL), Isolated Arabic Recognition (IAC), Template Comparison.**

I. Introduction

Arabic language is a widely used languages as more than 1 billion people use Arabic in either their daily activities or religion-related activities, but it received little attention especially on the researches interested on recognition the Arabic characters, the reason is the nature of writing Arabic characters, mixed Arabic characters and the complexities of writing and direction from right to left, and other specifications. [1],[2]. Very little research has gone into character recognition in Arabic due to the difficulty of the task and lack of researchers interested in this field [3]. There are two types of character recognition systems "on-line and off-line" systems. Each system has its own algorithms and methods. The main difference between them is that in an on-line system the recognition is performed in time of writing while the off-line recognition is performed after the writing is completed. The ultimate goal of character recognition is to simulate the human reading capabilities. Character recognition system can improve the interaction between man and machine in many applications [4],[5].Several works were dedicated to the isolated Arabic characters recognition.

Some classifiers were based on neuronal networks [6],[7], on hidden Markov models [8], on affixal approach or the K-nearest neighbours [9].We propose a system on fuzzy logic for the recognition isolated Arabic characters (table 1) models [8], on affixal approach or the K-nearest neighbours [9]. We propose a system on fuzzy logic for the recognition isolated Arabic characters (table 1).

TABLE.IThe set of isolated Arabic characters that Testing by our proposed system.

ط	ض	س	ر	د	ج	ب	أ
8	7	6	5	4	3	2	1
ي	و	هـ	م	ل	ك	ف	ع
16	15	14	13	12	11	10	9

The first section presents the proposed algorithm of our proposed system .The second section basic concepts of fuzzy logic and applications .The third section classification stage of the characters. The forth section experimental result. we finished by conclusion and future work.

II. Proposed Algorithm

In this section we explain the general steps of the proposed algorithm which used in this research (Figure 1) show the block diagram of this algorithm.

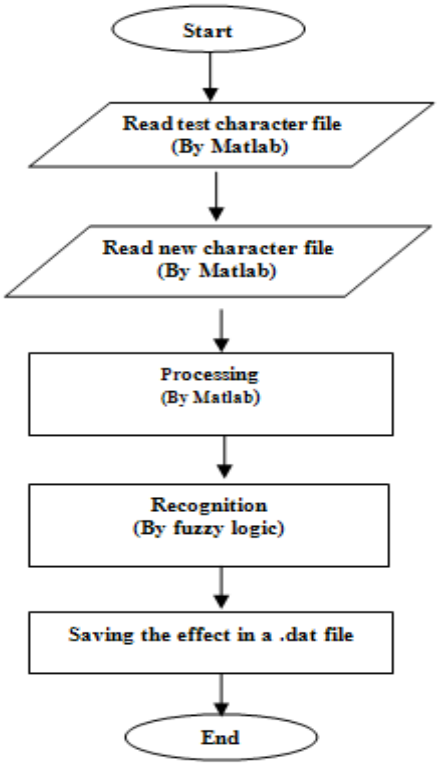

Fig.1 Block diagram of proposed algorithm.

Fuzzy logic was first developed by Lotfi Zadeh [10]. It was developed for solving decision making problems with

'IF-THEN' rules later ,it was used to deal with uncertainly and imprecise data management. For several years fuzzy logic has been applied in different filed, such as automated control ,consumer products, [11] industrial systems, automotives, decision analysis, medicine, geology ,ccontrolling aircraft flight, chemical reactor and processes, and applications of artificial intelligence such as expert systems ,pattern recognition and robotics. Fuzzy logic is much more general than binary logical systems. The generality of fuzzy logic is needed to deal with complex problems in the realms of search. Fuzzy logic provides a foundation for the development of new tools for dealing with natural languages and knowledge representation. Such as precipitated natural language, theory of hierarchical definability, unified theory of uncertainty [12]. Its advantages not much different than the binary logic, except the binary Logic working on only two levels ((0,1) but the fuzzy Logic treat with hypotheses to be correct confirm the degree of reality between (0,1) The real values of the assumptions indicate the degree of emphasis on the premise that is correct. Fuzzy logic easy way to reach definitive conclusions based on vague, and ambiguous and imprecise and noise, therefore, it makes sense to control the individual to take a quick decision [13],[14]. There are many motives that prompted scientists to develop fuzzy logic with the development of computer software and created a desire to invent or programming systems could deal with the information of others as human minute but this problem was born as the computer can not deal only with specific and accurate information, has resulted in this trend known as expert systems and artificial intelligence, fuzzy logic is one theory in which to build these systems. There are many concepts of fuzzy logic such as fuzzy set theory, fuzzy logic operations (complement operation ,union operations and the intersection operation)these operations can be also applied on binary sets [15]. we consider that the set membership is defined on the universe of discourse U. For a fuzzy set M, the membership of an element u ∈ U is given by μ M (u) ∈ [0, 1].

- Set Equality

$$M = N \Leftrightarrow (u(U, \mu M(u) = \mu N(u)$$

- Set Union (corresponds to OR):

$$\mu AUB(u) = \max[\mu A(u), \mu B(u)], (u(U$$

- Set intersection (corresponds to AND):

$$\mu A \mid B(u) = min[\mu A(u), \mu B(u)], (u(U$$

- Set complement (corresponds to NOT):

III. Classification Stages

The classification is the important and difficult stage of the recognition, the process start by acquiring the files contain many records of isolated Arabic characters, these files are saved as in a computer running Matlab.Matlab is used to read these files and convert it to black and white format ,the pixel values then used to create a header file to represent the character in the main program.The classification stages include many operations with many equations [16].

A. Sum of pixels

Determine the sum of black Pixels and sum of white pixels in the binary matrix, which represent the isolated Arabic character, these compute operations represented by equations(1) and (2),

$$MBP = \sum_{i=1}^{n} \sum_{j=1}^{m} A(n,m) \qquad \begin{matrix} i = 1,2,3,\dots,n \\ j = 1,2,3,\dots,m \end{matrix} \tag{1}$$

$$MWP = \sum_{i=1}^{n} \sum_{j=1}^{m} A(n,m) \qquad \begin{matrix} i = 1,2,3,\dots,n \\ j = 1,2,3,\dots,m \end{matrix} \tag{2}$$

Where,
A (n, m): binary matrix of n rows and m columns.
MBP: the number of black point in a matrix of the character.
MWP: the number of white point in a matrix of the character.

Then, sum of pixels (SS) is calculated by equations (3)

$$SS = \frac{\left| MBP \mid MWP \right|}{L} \tag{3}$$

Where,
SS: Sum of pixels.
L: the number of shape used for a single character.

We obtain from this stage binary matrices of different shapes of the isolated character. Saved these matrices in different files, then for every matrix saved in the file we compute sum of every row in it, and saved the result in one dimension matrix (SOR), which can be represented by equation (4),

$$SOR(j) = \sum_{j=1}^{m} SS_j \tag{4}$$

B. Templates comparison

The closeness of match between every row of the template can be found by comparing every template against every other saved template .This process is defined by the equation (5),

$$DOCT(i) = \sum_{i=1}^{m} \frac{KM(i)}{L} \tag{5}$$

Where,
KM: the number of templates that matched of every row for the testing template with each row of the saved templates.
DOTC: Degree Of Templates Comparison.

C. *Fuzzy logic recognize*

The matrices SOR,DOTC represent the input of the fuzzy system, which depended on Mamdani fuzzy model and simple fuzzy if-then rule were used to classify characters [17], then applied the various components of fuzzy system. The fuzzy logic system consists of a fuzzification, fuzzy inference system, defuzzification.

- Fuzzification: Using a set of input membership function for the extracted features to values from 0 to 1,the result of this operation eventuating fuzzy values for the matrices SOR, DOTC.
- Fuzzy Inference System: Using fuzzy set theory to map features to classes. To compute the output of the fuzzy inference we must determine a set of rules, using the input membership functions, There are three membership functions used with this system(Gaussian, Trapezoidal, Triangular) and the fuzzy linguistic variables facing to these membership functions are SC (Small Characters), MC(Middle Characters), BC (Big Characters).We choose Trapezoidal membership function in our proposed system because it is a famous, simple membership, take less time and little memory in computing, which can be represented by Figure(2), equation(6), [14],

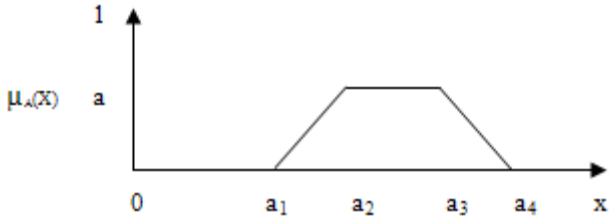

Fig.2 Trapezoidal membership function.

Fig.3

$$A_{ij}(x) = \begin{cases} \frac{x_i - a_1}{a_3 - a_1} & a_1 \le x_i \le a_3 \\ 1 & a_3 \le x_i \le a_4 \\ \frac{x_i - a_2}{a_4 - a_2} & a_4 \le x_i \le a_2 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

table(2) explain the rules of fuzzy logic, then combining the fuzzified inputs (SOR,DOTC) according to the fuzzy rules to establish a rule strength, finding the consequence of the rule by combining the function, combining the consequences to get an output distribution.

TABLE.II Fuzzy rules.

SOR / DOTC	SC	MC	BC
SC	SC	SC	NSC
MC	SC	MC	MC
BC	SC	MC	BC

- Deffuzzification: Using to convert inputs of the fuzzy system after treatment with the inference rules. We applied the method Center-of-Gravity (COG) for solving the deffuzzification as in equation (7) ,

$$GCOG = \frac{\sum_{i=1}^{i=m}(u_i * g_i)}{\sum_{i=1}^{i=m} u_i} \quad (7)$$

Where,
1<= i <=m,
m :number of rules.
g: centroid of the backend membership function corresponding for each rule.

From the previous description, the total matching procedure is divided into two phases. The first phase determines the output real values of fuzzy stages system, which represented the contribution of each row in the process of matching and saved the resulting in a new matrix. After the first phase is finished, the proposed system repetition of these stages for each row in the group of saved templates with test character ,which is the second phase. Finally, according to previous phases, collection the summation of the total rows to represent the amount of matching the character under test with each set of saved characters.

IV. Experimental Results

Arabic language consists of 28 characters. Each character may have up to four forms depending on its location within a word. . Our experiments are performed on 16 isolated characters (Alif , Baa ,Geam ,Dal ,Raa ,Seen ,Daa ,T'ta , Ain ,Faa ,Kaf , Lam , Mem ,Haa ,Waw ,Yaa) with four different shapes ,three shapes Templates saved in many files and the forth shape template used for testing by proposed system . For examples the following represent the applied steps of our proposed system on two isolated characters "ع"and "ف ".

1) The three shapes templates of the saved character " ع" (Initial ,Medium ,Final) and three shapes templates of tested character"ع" (Initial ,Medium ,Final)

shapes templates of saved character "ع"

Initial template saved	Medium template saved	Final template saved
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000000000	000000000000	000011111000
000011111000	000011110000	000100000100
000100000100	000100001000	000100000000
001000000000	001000000100	000011110000
000100000000	000100001000	000100000000
011111110000	011111111110	000100000000
000000000000	000000000000	000100000000
000000000000	000000000000	000011111100
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000

Shapes templates of tested character "ع"

000000000000	000000000000	0000000000000
000000000000	000000000000	0000000000000
000011111000	000011110000	0000111111000
000110001100	001100001100	0011000000000

001100000000	011000000110	0011000000000
000110000000	001100001100	0000111110000
011111111000	111111111111	0011000000000
011111111000	111111111111	0011000000000
000000000000	000000000000	0011000000000
000000000000	000000000000	0000111111100
000000000000	000000000000	0000000000000
000000000000	000000000000	0000000000000
Initial template saved	Medium template saved	Final template saved

a. Sum of pixels:

Sum of pixels of ع Initial

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ع Medium

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 3.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ع Final

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

b. Sum of row:

SOR(i) Initial "ع" =
[0.0 0.01.51.2 0.6 0.6 2.4 2.4 0.0 0.0 0.0 0.0]^T

SOR(i) Medium"ع" =
[0.0 0.01.21.20.6 1.2.3 .63.6 0.0 0.0 0.0 0.0]^T

SOR(i) Final "ع" =
[0.0 0.01.50.6 0.6 1.2 0.6 0.6 0.61.8 0.0 0.0]^T

c. Templates comparison

DOCT(i) Initial "ع" =
[0.4 0.4 0.8 0.8 0.6 0.6 0.4 0.81.0 1.01.01.0]^T

DOCT(i) Medium " ع " =
[0.4 0.3 0.7 0.7 0.6 0.6 0.3 0.71.0 1.01.01.0]^T

DOCT(i) Final " ع" =
[0.5 0.4 0.9 0.9 0.6 0.6 0.5 0.91.0 1.01.01.0]^T

2) The three shapes templates of the saved character "ف" (Initial ,Medium ,Final) and three shapes templates of the tested character "ف" (Initial ,Medium ,Final)

Shapes templates of saved character "ف"

000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000110000	000001100000	000000110000
000001111100	000000000000	000001111100
000001000100	000001110000	000001000100
000001111100	000010001000	000001111100
000000000100	000100000100	010000000100
000000000100	001000000010	010000000100
011111111100	011111111100	011111111100
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
Initial template saved	Medium template saved	Final template saved

Shapes templates of tested character "ف"

000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000110000	000001100000	000000110000
000011111110	000000000000	000011111110
000011000110	000011110000	000011000110
000011111110	000111111000	000011111110
000000000110	001100001100	110000000110
000000000110	001100001100	110000000110
011111111110	011111111110	111111111100
011111111110	011111111110	111111111100
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
Initial template saved	Medium template saved	Final template saved

a. Sum of pixels:

Sum of pixels of ف Initial

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.0
0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ف Medium

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ف Final

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0
0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

b. Sum of row:

SOR(i) Initial "ف "=
[0.0 0.0 0.6 2.1 0.9 1.8 0.6 0.6 3.0 3.0 0.0 0.0]^T

SOR(i) Medium "ف "=
[0.0 0.0 0.6 0.0 1.2 1.8 1. 2 1.2 3.0 3.0 0.0 0.0]^T

SOR(i) Final "ف" =
[0.0 0.0 0.6 2.1 0.9 2.1 0.9 0.9 3.0 3.0 0.0 0.0]^T

c. Templates comparison

DOCT(i) Initial "ف " =
[0.5 0.6 0.6 0.6 0.8 0.5 0.7 1.01.0 1.01.01.0]^T

DOCT(i)Medium"ف " =
[0.4 0.4 0.8 0.8 0.6 0.6 0.4 0.81.0 1.01.01.0]^T

DOCT(i) Final "ف " =
[0.4 0.4 0.8 0.8 0.6 0.6 0.4 0.81.0 1.01.01.0]^T

3) The three shapes templates of the saved character "ف" (Initial ,Medium ,Final) and three shapes templates of the tested character "ف" (Initial ,Medium ,Final)

Shapes templates of saved character "ف"

000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000110000	000001100000	000000110000
000001111100	000000000000	000001111100
000001000100	000001110000	000001000100
000001111100	000010001000	000001111100
000000000100	000100000100	010000000100
000000000100	001000000010	010000000100
011111111100	011111111100	011111111100
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
Initial template saved	Medium template saved	Final template saved

Shapes templates of tested character "ف"

000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
000000110000	000001100000	000000110000
000011111110	000000000000	000011111110
000011000110	000011110000	000011000110
000011111110	000111111000	000011111110
000000000110	001100001100	110000000110
000000000110	001100001100	110000000110
011111111110	011111111110	111111111100
011111111110	011111111110	111111111100
000000000000	000000000000	000000000000
000000000000	000000000000	000000000000
Initial template saved	Medium template saved	Final template saved

a. Sum of pixels:

Sum of pixels of ف Initial

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.0
0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ف Medium

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Sum of pixels of ف Final

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.0 0.0 0.0 0.0 0.3 0.0
0.0 0.0 0.0 0.0 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0
0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0
0.3 0.3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.3 0.0
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0
0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.3 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

b. Sum of row:

SOR(i) Initial "ف "=

$[\ 0.0\ 0.0\ 0.6\ 2.1\ 0.9\ 1.8\ \ 0.6\ 0.6\ 3.0\ 3.0\ 0.0\ 0.0\ ]^T$

SOR(i) Medium "ف "=

$[\ 0.0\ 0.0\ 0.6\ 0.0\ 1.2\ 1.8\ 1.\ 2\ 1.2\ 3.0\ 3.0\ 0.0\ 0.0\ ]^T$

SOR(i) Final "ف" =

$[\ \ 0.0\ 0.0\ 0.6\ 2.1\ 0.9\ 2.1\ 0.9\ 0.9\ 3.0\ 3.0\ \ 0.0\ 0.0\ ]^T$

c. Templates comparison：

DOCT(i) Initial "ف " =

$[\ 0.5\ 0.6\ 0.6\ 0.6\ 0.8\ 0.5\ 0.7\ 1.01.0\ 1.01.01.0\ ]^T$

DOCT(i)Medium"ف " =

$[\ 0.4\ 0.4\ 0.8\ 0.8\ 0.6\ 0.6\ 0.4\ 0.81.0\ 1.01.01.0\ ]^T$

DOCT(i) Final "ف " =

$[\ 0.4\ 0.4\ 0.8\ 0.8\ 0.6\ 0.6\ 0.4\ 0.81.0\ 1.01.01.0\ ]^T$

The proposed system described above has been fully implemented. The working of the system is as follows: We used in our experiment on 16 isolated Arabic characters and take for example two isolated Arabic characters "ع" and "ف" with 12 different shapes. It is divided evenly into Templates set and testing set. In our proposed system, entering the values of matrices (SOR, DOTC) to fuzzy system. We used Trapezoidal membership function in our proposed system . Repeat the steps of the proposed algorithm of our system,saving the result in many files ,then comparing the results obtained with the saved results of the templates, choose the maximum value for templates comparison which represent the recognize of isolate character .The Fuzzy System has a highly developed rules system for these characters. We performed the experiments based on this algorithm that had 88% accuracy.

V. Conclusions and Future Work

This paper describes proposed system to recognize isolated Arabic characters, based on a approach of fuzzy logic. This approach is very suitable for the recognition of a complex data such as Arabic characters.The fuzzy logic implementation uses very little memory, making it possible to provide full functionality. Experiment results showed the effectiveness of the proposed system with isolated Arabic characters. As a first result of using this system, we obtained an overall recognition rate of 88%.Developing an automatic selection method of thresholds and an invariant method for the size of the input character This system will be improved in our future works by integrating other characters based on geometrical measures (example: the size of the input character, ratio height/ width or line of basis of character).

Reference

[1] Amin, A., 1998. Offline Arabic character recognition: The state of art. Pattern Recognition. 31.
[2] Adnan Amine ,Humoud Al-Sadoun and Stephen Fischer, (1996) " Hand-printed Arabic character Recognition system using an neural network "pattern recognition, Vol. 29.
[3] Sami S.E., R. Ramsis and A., 1990."Arabic Character Recognition System, A Statistical Approach for Recognizing Cursive Typewritten Text". Pattern Recognition. 23.
[4] Lorigo, L., Govindaraju, V., 2006. Offline Arabic handwriting recognition: A survey. IEEE Trans.on Pattern Anal. Machine Intell. 28.
[5] Margner, V., Pechwitz, M., EI Abed, H., 2005. ICDAR 2005 Arabic handwriting recognition competition. In: The8th International Conference on Document Analysis and Recognition,vol.1.
[6] H.AL-Yousefi and, S. S.Udpa "Recognition of Arabic Characters",IEEE Transactions on pattern analysis and Machine Intelligence, Vol.14.
[7] Ahmed M. Zeki and Mohamad S. Zakaria, 2004, "Challenges in Recognizing Arabic Characters" International Islamic University alaysia(IIUM),Kuala Lumpur .Malaysia)
[8] Abdel wahab zramdini and rolf ingold, 1993,"Optical Font recognition from projection profiles" electroic publishing, Vol 6.
[9] Korsheed, M. S. 2002,"Offline Arabic character recognition –review. Pattern Anal.Appl.5.
[10] Zadeh ,Lotfi A. ,1998 , "Fuzzy logic Computer ", vol 22, issue 4.
[11] Zadeh ,Lotfi A. 2004 ,"Fuzzy logics systems: origin concepts ,and trends)
[12] Chen, G., Pham T., 2001 "Introduction to fuzzy sets, fuzzy logic, and fuzzy control systems).
[13] Ruspini, E., 1992 "Introductio n to Fuzzy set theory basic concepts And structures", IEEE.
[14] Brigette Krantz, A "Crisp" Introduction to Fuzzy Logic, Colorado University.
[15] Mohammed Abdel-Hadi, Ibrahim al-Hamdi Al-Hosni, Ahmed Hamid Cairo, 1999 "Fuzzy logic in the decision-making", Arab Journal of Administrative Sciences Vol. 6, Kuwait.
[16] M. Katrthigayan, R. Nagarajan, Sazali Yaacob, 2005," Artificial intelligence in IC chip marking inspection " Koiej university Kejuruteraan Utara Malaysia.
[17] N. Ben Amor,n.Essoukri Ben Amare,2005,"Wavelet transform and fuzzy approach in the recognition of Arabic characters "Ai-Isra Private University, Amman-Jordan.

Majida Ali Abed received his B.S. in mathematic science from Basrah university, 1978,the M.S. in computer science from Basrah University in 1986. She is an assistant professor in the Department of Computer Science at the Tikrit University. She is the author of more than 30 publications in technical journals and conferences. She served on the several universities and computer centers in Iraq Her research interests are in the areas of Artificial intelligence (expert systems, pattern recognition, processing of natural languages),image processing.