WALAILAK JOURNAL

http://wjst.wu.ac.th Engineering and Physical Sciences

Handwritten Arabic (Indian) Numerals Recognition using Expert System

Fahad Layth MALALLAH[1], Asem KHMAG[2,*], Ayad HUSSAIN[3] and Zaid Ahmed ALJAWARYY[4]

[1]*Faculty of Computer Science, Cihan University of Sulaimaniya, Kurdistan, Iraq*
[2]*Department of Computer Systems Engineering, Faculty of Engineering, University of Zawia, Az zawia, Libya*
[3]*Faculty of Computer Science, Mosul University, Mosul, Iraq*
[4]*Faculty of Science and Technology, University of Human Development of Sulaimaniya, Kurdistan Region, Iraq*

([*]Corresponding author's e-mail: khmaj2002@gmail.com, a.khmag@zu.edu.ly)

Received: 6 October 2015, Revised: 23 April 2016, Accepted: 29 March 2016

Abstract

The capability of a computer system to interpret intelligible handwritten data input and analyze it for many automated process system is the core of a handwriting character recognition system. Handwritten recognition plays a major role in many applications, such as cheques verification, office automation, natural human-computer interaction, as well as mail sorting. Procedures of handwritten numeral recognition have showed a remarkable contribution in order to improve the automation process and have developed human and machine interaction in several fields and applications. In this paper, the proposed approach is based on extraction of both local and global geometric features. The features are further quantified into a set of facts or conditions that are used for classification based on a set of rules as an expert system. In addition, sample handwritten images are tested with the proposed technique and the results are stated and accuracy rate is calculated based on a confusion matrix. The output showed that the recognition error rate in terms of False Rejection Rate (FRR) is 4 % or the overall successful accuracy is 96 %.

Keywords: Arabic numeral character recognition, image processing, pattern recognition, feature extraction, object segmentation, expert system

Introduction

Characters, numeral and handwriting recognition have been studied for decades due to its abilities and various applications such as bank cheque processing, postal automation, document analysis, reading postal codes and reading different forms, to name few [1-3]. However, most studies have been restricted to Chinese, Latin, and Japanese script. Very few notable works has been done on Arabic text including Arabic (Indian) characters and numerals.

The lack of research on Arabic text recognition is due to the unavailability of commonly recognized databases on public domains for Arabic numeral/ text recognition [4]. Though the application of Handwritten Character Recognition (HCR) is tremendously wide, it is considered a difficult classification task due to the privacy features of each writer, since each writer has their specific manner and fashion of writing characters. In addition, a single writer may exhibit different styles of writing. The task of feature extraction is one of the pivotal phases in successfully achieving character recognition. The feature extraction process is used to identify and extract different attributes and features from several characters which can achieve unique and distinctive representation among different characters. In the literature, a number of feature extraction techniques have been introduced which take into account several character representations. For instance, character shapes which represent different feature sets, boundaries, textures, edges and their character skeletons and strokes [5]. Due to the inherent difficulties of high variability in writing styles, several techniques have been introduced such as template matching, dynamic programming, hidden Markov model, neural networks, expert systems and hybrid techniques from the mentioned methods. In addition, due to the privacy of individual handwriting of different people; it is a complicated task to precisely recognize the handwritten characters or numbers.

Consequently, it contributes immensely to the advancement of automation steps to improve the interface between human and machine in order to get high character recognition rate [6]. From a theoretical point of view, handwritten recognition can be classified into 2 main categories which are off-line and on-line handwriting recognition techniques. Off-line handwriting character recognition is the automatic transcription by computer, where only the image of the handwriting is involved in the process. In this regard, a huge amount of research has been done especially for Latin character recognition (the approach on Arabic/Persian handwritten digit recognition is quite limited). On the other hand, in the on-line type of recognition, a computer system can trace the writing process. The strength and sequential order of each segment in handwriting can be recorded on time for recognition. Additionally, techniques of recognition of characters can be classified into 2 types, structural and statistical techniques. In structural methods, each character in handwriting script is labeled by its structural features. The recognition process in structural approaches utilizes approaches such as graphs, grammar or rules of comparisons between character features in the same script. The main merit in this kind of recognition is that it does not involve huge data training sets. However, it is a hard task to determine and define such features. In the other types, the characters are defined by their probabilistic distributions and statistical models and tools. The recognition process in this method uses techniques such as neural networks, Hidden Markov Model (HMM) and Support Vector Machine (SVM). For instance, support vector machine is a statistical learning theory based algorithm which finds an optimal separating decision plane to classify the input objects. Moreover, it is beneficial in the recognition process due to the nature of the classification problem which returns one of the 2 states of acceptance or rejection [7,8]. On the other hand, HMMs are popular in pattern recognition due to the in-depth consideration of the character relationships. In addition, HMMs adapt to the variations in size and length of handwritten scripts. However, this technique is not effective in differentiating isolated characters compared to neural networks and SVMs.

In this study, we emphasize off-line recognition of handwritten Arabic numbers in the range of 0 to 9. Handwritten digit recognition has long been an active topic in optical character-recognition and classification/learning research. Numerous studies have proposed various techniques for pre-processing, feature extraction, classification and post-processing. Also, different standard image databases are commonly used in order to evaluate the handwriting performance [9]. On the other hand, most of this research uses different procedures and methods for feature extraction such as: geometric moment's invariants, Zernike moments [10], shadow coding [11]; and different methods for classification such as: statistical, fuzzy, or neural approaches. But due to the unavailability of online standard Arabic/Persian digit recognition database, researchers have tried to build their own and use it to deal with this issue.

This paper is organized as follows. In Section 2, the related works are presented. The methodology of this study is introduced in Section 3. The simulation and experiments implementation are presented in Section 4. In section 5, the results and discussion are addressed. The final section contains the conclusion and future works.

Related work

In the last 3 decades, Arabic characters, numerals and script recognition has received a significant amount of research interest. AI-Muallim and Yamaguchi [12] proposed a technique based on structural recognition for Arabic handwritten scripts which were segmented into strokes. The strokes were categorized and integrated into characters based on their features. However, the method is not effective in many cases due to inaccurate segmentation of each word. Another study [13] utilized a binary tree procedure in order to segment Arabic printed text to characters. In [14], Arabic and handwritten text recognition system is implemented using multi-layer neural networks. This technique is not expensive for feature extraction; and also the execution time does not rely on font or the size of the script. Aburaiba [15] dealt with the difficulties that are faced in the processing of handwritten text documents in binary image format, such as extracting lines, circles or geometrical shapes from handwritten script, which is found to be useful in official signature retrieval and matching. Moreover, look-up tables were used for the recognition of isolated hand-written Arabic numbers introduced by [16]. Saleh *et al*. [17] described an efficient algorithm for coding handwritten Arabic characters. This technique was mainly based on separation of Arabic handwritten characters prior to the recognition process. In the study presented in [18], a novel off-line cursive Arabic script recognition method was proposed to recognize off-line handwritten cursive script which has very clear variability according to a segmentation based approach. Single component strokes were extracted and the recognition process was performed based on the geometric features of each character. Khorsheed [19] proposed an algorithm for off-line recognition of handwritten Arabic script, in which the segmentation of characters is not mandatory. The method decomposes the skeleton of the word into an observation sequence, and by applying a statistical approach with structural features; the classification of each character can be achieved. In this method, a single HMM was used for the classification task. In the study presented in [20], HMM was used for unconstrained Arabic handwritten word recognition. Furthermore, a novel procedure for isolated Arabic (Indian) numerals handwritten by HMM has been introduced in [21]. Four features containing circle, angle, horizontal and vertical were created according to digit pixel image segmentation, and then the calculation of each segment ratio of black pixels to total size of each segment was computed [21]. Furthermore, the features which resulted were utilized in order to train and evaluate the HMM models [21]. Word recognition using HMMs can be extended easily to include the lines and paragraphs recognition by applying model languages [22,23]. Recently, techniques based on abductive networks properties have become a dominant technique in pattern recognition; feature extraction, artificial intelligence and classification, which are used in several fields [24,25].

A comprehensive survey on HMM systems for handwriting recognition along with their performance can be found in [26]. On the other hand, recognition techniques which utilized feature analysis of primitive characteristics of patterns and curve tracing were adopted in [27]. Template matching techniques have been avoided due to their sensitivity to variations in a character's geometric shape and position in the script [28]. Accordingly, Artificial Neural Networks (ANNs) are the most useful discriminative model which is applied in character recognition procedures. The techniques based on ANN can perform a very accurate task in the context of a pattern classifier. Moreover, some research effort has been devoted to the analysis of the use of hidden layers in pattern classifications [29,30]. In this regard, biologically inspired approaches such as Multi-layer neural networks [31] and convolutional neural networks [32] have exhibited their superiority in the overall recognition rate compared with probability density function approaches such as Gaussian Bayesian methods, Gaussian mixture models and non-parametric techniques like K-nearest neighbor classifiers. Recently, a study based on computational intelligence models for the accurate recognition of Arabic text was presented in [33]. The method is based on using an average template-matching approach for recognizing Arabic (Indian) numerals, with the classification performed using the Euclidean distance between the vector of features of the samples and the generated models. In the same way, Sadri *et al*. in [34] utilized SVM for isolated handwritten recognition of Arabic/Persian numerals. Their method stated each digit from 4 different positions, and then extracted 64 features to train SVM in order to separate the different digit modules. An average recognition rate of 94.14 % was obtained in this method. Building a full system for character recognition is almost impossible and most recognized sections should be tested by human verifiers, which is not appropriate and available. Thus, practically, it is customary some of the unrecognizable samples are rejected due to verification process of the machine recognition is resulted by the human operator then the final result is made by automatic machine [35]. Detecting a restricted set of handwritten keywords, such as in [36], is valuable tool for handwritten mails sorting, signatures and authorized scripts. However, for instance, a complete transcription of the majority of words presented in mails would be much more robust and give more accuracy rate of recognition. In this paper, we focus on Arabic numeral recognition where we introduce a new recognition approach of Arabic characters. Sample handwritten images are tested with the proposed method and the results are illustrated. The proposed approach is based on extraction of both local and global geometric features. The method mainly uses a preprocessing step which includes conversion to a binary image, morphological filtering (closing operation), and segmentation before entering the data to the recognition system. The accuracy value is calculated based on receiver operating characteristics and the confusion matrix. The value is calculated for each node in the network. The final result shows that the proposed method provides a recognition accuracy of more than 96 %.

Methodology

Framework design and preprocessing

The steps of the proposed method are illustrated in **Figure 1**. The process begins with an input image that contains object numbers. Firstly, the image is converted to black and white. Then, the image complement process is applied to the black & white image to enable further processing on image pixels with a value of 1 as shown in **Figure 2**.

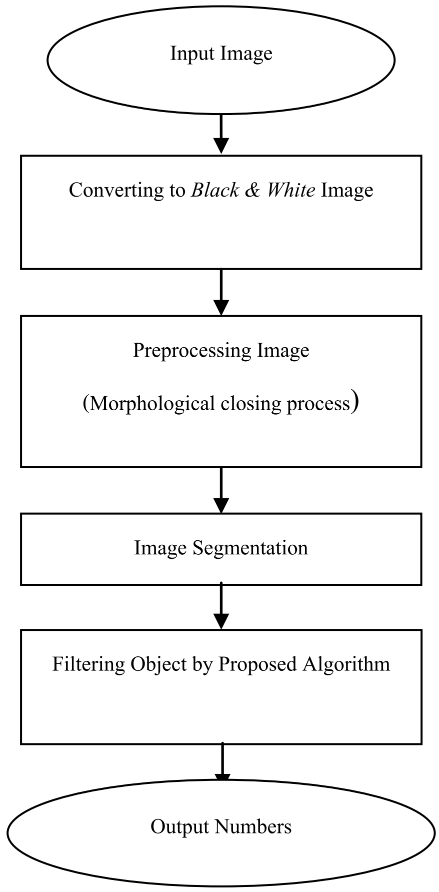


Figure 1 Proposed hand-written flowchart for number recognition system.

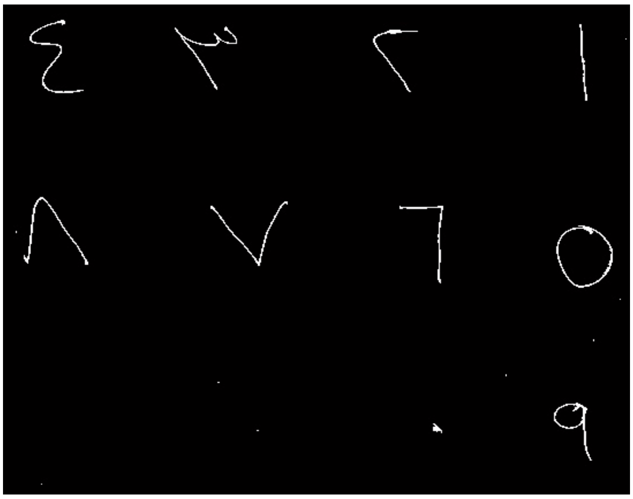


Figure 2 Hand-written Arabic numbers image after black & white and complement processing.

Obviously, there is noise in the sample depicted **Figure 2** that could be removed by applying a morphological filtering process as shown in **Figure 3**. However, noise has not been removed yet, but in case a pen was fluctuating during writing; it is ensured that pixels of each object has been connected thoroughly to represent one object.

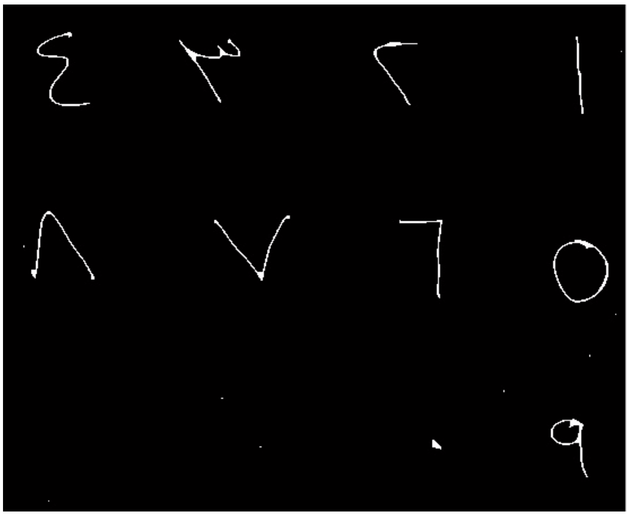


Figure 3 Hand-written Arabic numbers after morphological filtering processing.

The rationale for performing morphological closing filtering is to smooth and ensure an object without any incisions that can later have a negative effect on the recognition. In order to remove the noise completely, an image processing tool is applied for removing any object with pixels that are equal to or less than 15 pixels as an area of that object. As depicted in **Figure 4** the objects now are ready to be segmented into objects for identification.

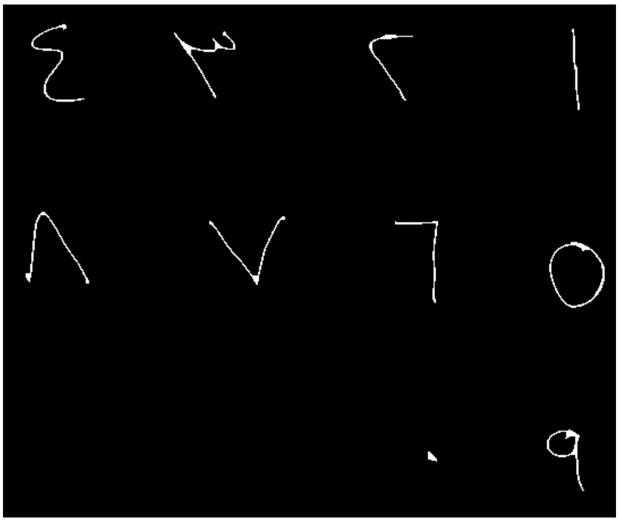


Figure 4 Hand-written Arabic numbers after object smoothing and removing noise thoroughly.

Afterward, as shown in **Figure 5**, objects have been segmented into many separated objects. Image segmentation is applied using a region labeling method [37], and finally, the proposed algorithm for number recognition is run for decision making of each number's identity.

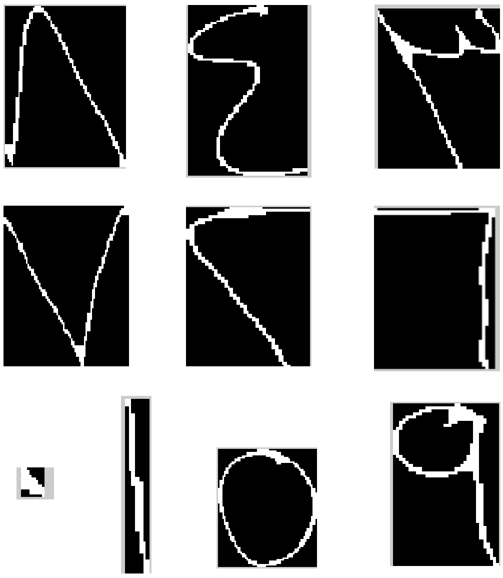


Figure 5 Some examples of the segmented Arabic object numbers.

Expert system

An expert system is defined as a computer system that emulates the decision-making ability of a human expert [38]. Expert systems are suitable tools for implementing structural pattern recognition techniques and they help to solve difficult pattern recognition problems. More rules and human experience can be added easily using rule-based systems, especially in closed-system applications with precise inputs and logical outputs [39,40]. Expert systems have a number of major system components and interface with individuals who interact with the system in various roles as shown in **Figure 6**. In rule-based expert systems, there are 2 basic techniques; forward chaining and backward chaining inference. The domain knowledge is represented by a set of IF-THEN production rules and the data is represented by a set of facts about the current situation. The matching of the rule IF parts to the facts produce inference chains. The inference chain indicates how an expert system applies the rules to reach a conclusion. The inference engine must decide when the rules have to be fired [40]. An inference engine using forward chaining searches the inference rules until it finds one, where the IF clause is known to be true. Forward chaining is used in this paper because of the similarity to the methodology that depends on the data-driven reasoning. The reasoning starts from the known data and proceeds forward with that data. Each time, only the topmost rule is executed, and when fired, the rule adds a new fact to the database. Any rule can be executed only once and the match-fire cycle stops when no further rules can be fired [41,42].

Feature extraction and recognition

In the field of the recognition, the most important and critical part of the computer vision system is how features of an object are going to be built, because smarter object features extraction results in better system accuracy. This section discusses the features that a play major part in the recognition of each numerical object, as well as how the features are obtained or extracted.

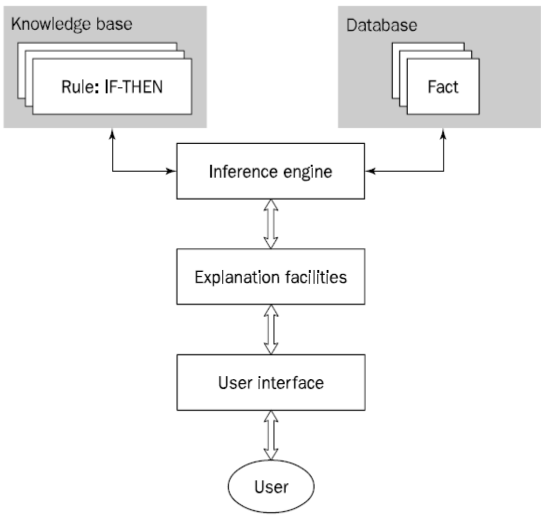


Figure 6 Components of an expert system [24,25].

Prior to that, it is essential to mention that the recognition operation is performed by processing each object once at a time. Therefore, the object segmentation operation is considered very crucial in the proposed method. Details of the segmentation operation are elaborated in [43]. For instance, **Figure 5** shows some examples of segmented Arabic numbers for feature extraction, where each number will be processed separately. The feature extraction and recognition of each number are performed sequentially.

In other words, we extract the features of a particular number using the proposed algorithm, and then, the decision of the number's identity is performed based on the set of rules in the expert system. The process examines every possible *match* of the *facts* provided in the inference engine to determine the expected identity of the number. For instance, a random number can be predicted by first checking whether the feature properties *align* with the *facts* about number 5 in the inference engine. If an alignment is not found, the system examines the possibility with the *facts* about number 9. The process is repeated till it reaches number 6, but if a match is found then the expected identity will be presented to the user.

The following recognition order of the numbers is designed deliberately in the following order according to the proposed idea, since sometimes priority issue has been taken in the consideration of the recognition.

Number 5 Recognition

In order to assign an identity '5' to an object, 3 conditions (facts) should be satisfied. Firstly, the Euler number should be zero. Euler number describes the relation between the number of contiguous parts and the number of holes on a shape. Let ***S*** denote the number of contiguous parts and ***N*** be the number of holes in a shape. Thus the Euler number is determined as in (1);

$$Euler\ No. = S - N \tag{1}$$

For example, the Euler number for Shape (B) is -1, Shape (9) is 0, and shape (3) is 1. Secondly, the number of flips, resulting from horizontal middle scanning, should be greater than or equal to 3 flips. The flips number is computed by scanning the object from left to right at the middle-level of the image. The third condition is the number of flips resulting from the horizontal lower part scanning that flips number must be greater than 2, this condition is important to discriminate between number 5 and number 9, while in number 9 this condition must be equal to 2. The pseudo code for number 5 is as follows;

If [(Euler Number No. ==0) && (Flip_number_middle_horizonatally >= 3) && (Flip_number_lower_horizontal > 2)]
Then
The Object is '5';

In **Figure 7**, it can be seen that the upper arrow indicates more than 3 flips by scanning from left to right at the mid-level of the image; also the lower scanning it is more than 2 flips.

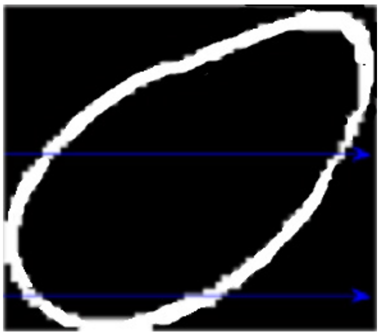


Figure 7 View of the scanned Arabic object number 5 to count the number of flips.

The number of flips is simply a count of the alternating transitions of pixel values from '1' to '0' or vice versa. Below is a pseudo code for extracting the number of flips.

```
first_value=(1,mid_y);
for x=min: x_max
    if (first_value ~= Object_array(x,mid_y))
       first_value= Object_array(x,mid_y);
       flip_num=flip_num+1;
    end
end
```

Number 9 Recognition

In case the conditions for number '5' are not satisfied, the object will be examined with the conditions for number '9'. The first condition is that the Euler number should be zero. The second condition is the flips count resulting from object scanning from left to right direction at the lower part of the object, should be less than or equal to 2. As illustrated in **Figure 8**, the only one arrow indicates the scanning direction to count the number of flips, which in this case is equals to 2.

Pseudo code for number 9 recognition is;

If [(Euler Number ==0) && (Flip_number_lower_horizontal <=2)]
Then
The Object is '9';

Figure 8 View of the scanned Arabic object number 9 to count the number of flips.

Number 7 Recognition

Four conditions are also considered to determine whether the preprocessed object is number 7, when the conditions of '5' and '9' are not satisfied. The Euler Number of the object should be equal to one. As shown in **Figure 9**, the width of the object at the upper segment should be greater than the width of the middle and lower segments. Then, the width at the middle segment should be larger than the lower segment width. Finally, flips number horizontally of the top quarter of the object must be less than 6. The benefit of the fourth condition is to differentiate 7 and number 3. The pseudo code is as follows;

If [(Euler Number =1) && (Lower_dist < Middle_dist < Upper_dist) &&
Max(Flip_number_top_quarter < 6))]
Then
The Object is '7';

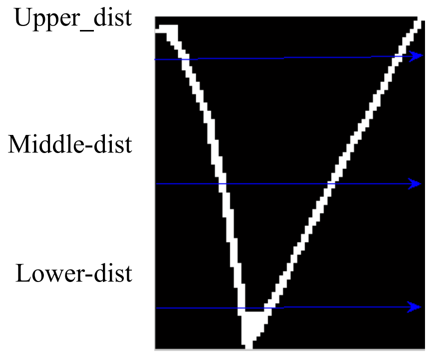


Figure 9 View of the 3 scanning positions of Arabic object number seven.

Number '8' Recognition

Three conditions are used to examine whether the object is number '8' shown in **Figure 10**, when the conditions for number '5', '9' and '7' are not satisfied. Firstly, the Euler Number should be equal to one. Secondly, the widths of the object at the upper, middle, and lower segments are checked. Basically, the width at the upper segment should be less than the width at the middle and lower segments. The final condition is that the middle width should be less than the width at lower segment of the object. The pseudo code is as follows;

If [(Euler Number No.=0) && (lower_dist > middle_dist > upper_dist)]
Then
The Object is '8';

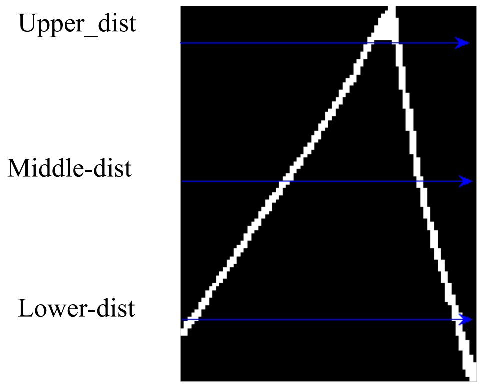


Figure 10 View of the 3 scanning positions of Arabic object number eight.

Number 3 Recognition

When the conditions for number '5' '9', '8' and '7' are not satisfied, the object will be examined with the following conditions to determine whether the number is '3'. Firstly, the Euler Number should be equal to 1. Then, the algorithm calculates the number of flips for 2 separate parts. Firstly, the part located in the top quarter of the number object, as highlighted with the upper arrow in **Figure 11**. Here, the number of flips must be greater than or equal to 6. This kind of feature is quite discriminating in comparison to the features of other number since 3 is the only object whose number of flips is equal to 6 in the mentioned position. The third condition is achieved by calculating the flips in the lower quarter of the object and the result should be less than or equal to 2. The pseudo code for the number 3 object is as follows;

If [(EulerNumber= 1) && (flip_top_quarter >=6) && (flip_bottom_quarter<=3)]
Then
The Object is '3';

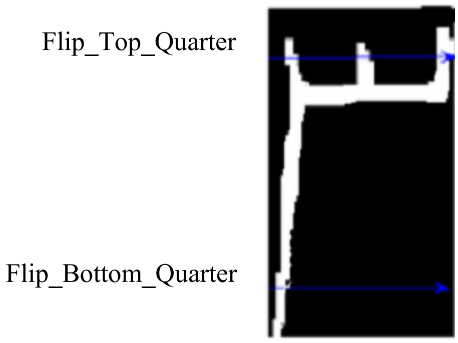


Figure 11 View of the 2 scanning positions of Arabic object number three.

Number 2 Recognition

We now proceed to describe the facts or conditions to determine the identity of the input image as 2, when the conditions for number '5' '9', '8', '7' and '3' are not satisfied. However, prior to that, it is imperative to mention that morphological processing based on skeletonization [44] as shown in **Figure 12**, is adopted in order to extract features that are specific only to number 2, and also enhance the processing of further feature computations.

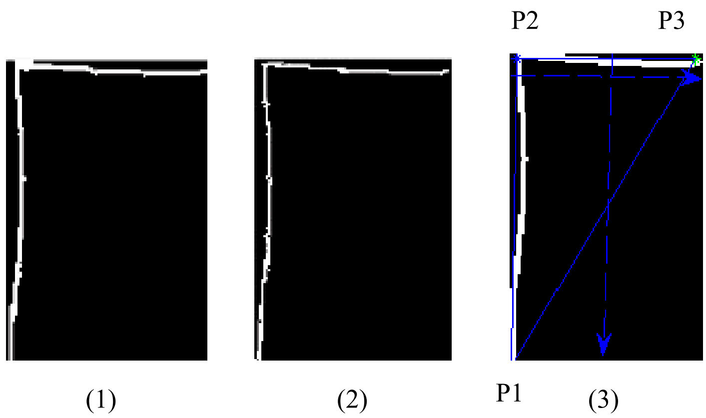


Figure 12 Views of the steps to recognize Arabic object number two.

The 6 conditions are as follows: the Euler Number should be equal to one. The ratio fraction of the object should be greater than or equal to 0.25. This factor is determined by dividing the width (*W*) distance by height (*H*) distance. In order to compute each of the mentioned distances, 3 basic points are located on the object as illustrated in **Figure 12**, which are denoted by the following: *P1* is positioned at the bottom of the object, *P2* is positioned at the upper left, and *P3* lays on the upper right. Since, each point has its coordinates *x* and *y* as *P(x,y)*, Height (*H*), width (*W*) and the slope distance are determined according to the Euclidean distance;

$$w_{dist} = \sqrt[2]{(y_3 - y_2)^2 + (x_3 - x_2)^2} \tag{2}$$

$$H_{dist} = \sqrt[2]{(y_2 - y_1)^2 + (x_2 - x_1)^2} \tag{3}$$

$$slop_{dist} = \sqrt[2]{(y_3 - y_1)^2 + (x_3 - x_1)^2} \tag{4}$$

Then the ratio is computed as;

$$Width\_Height\_Ratio = \frac{W\_dist}{H\_dist} \tag{5}$$

The third condition is that, the angle between the slope (*P1*-*P3*) and x-axes must be positive (larger than zero) to indicate that the object is number '2', let $P3(x_3,y_3)$ and $P1(x_1,y_1)$ denote 2 points of the slope line as shown in **Figure 12**, then the angle is computed using Eqs. (6) and (7);

$$\theta = \frac{abs(y_3 - y_1)}{x_3 - x_1} \tag{6}$$

$$Angle = tan^{-1}(\theta) \tag{7}$$

The fourth condition is achieved by counting the number of flips (horizontally top quarter scanning), which should be equal to 2 in order to differentiate the number from number 3. The fifth condition also computes the number of flips but at a different position where the scanning is in the top middle vertically of the object, the number of flips should be equal to or less than 4, the fifth condition is useful for discriminating number 2 and 4. The sixth condition is the solidity computation which must be less than 0.9, as the areas of the number 2 is intuitively small compared with its background, the benefit of this condition is also for discriminating number 2 and 0 (more details on solidity is explained in Number 0 Recognition). The following is the pseudo code for the 6 conditions;

If [Euler Number = 1 && Width_height_ratio > 0.25 && Slop_orientation > 0 && Flip_num_top_horiz <=2 && Flip_num_middle_vertical <=4 && Solidity < 0.9]
Then
The Object is '2';

Number 0 Recognition

In case the conditions for number '5' '9', '8', '7' '3' and '2' are not satisfied, the object will be examined with the conditions for number '0' which are described as follows: Firstly, the Euler Number should be equal to one. Secondly, solidity (S) factor should be greater than 0.9, $(S > 0.9)$. Solidity (S) is a scalar specifying the proportion of the pixels object in the convex hull that are also in the region as shown in **Figure 13**, it is computed in (8) as follows;

$$S = \frac{Area\_S}{Convex\ Hull\ Area} \tag{8}$$

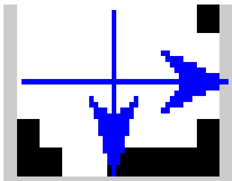


Figure 13 View of the Arabic object number zero.

where the area of object ($Area\_S$) is calculated according to the conditional equation in (9);

$$Area\_S = \begin{cases} \sum_{y=1}^{m} \sum_{x=1}^{n} pixel(x,y) & , pixel(x,y) = 1 \\ 0 & , pixel(x,y) = 0 \end{cases} \tag{9}$$

and the convex hull is calculated as follows;

$$Convex\ Hull\ Area = \sum_{y=1}^{m} \sum_{x=1}^{n} pixel(x,y) \tag{10}$$

The third condition is that, the aspect ratio, which is the division of the minor axis by the major axes, should be more than 0.5. This factor is chosen as both of the mentioned axes sum up to 1, thus 0.5 is considered to take the worst case. The pseudo code of the 3 conditions is;

If [Euler Number = 1 && Solidity > 0.9 && Aspect_Ratio > 0.5]
Then
The Object is '0';

Number 4 Recognition

In case the conditions for number '5' '9', '8', '7', '3', '2' and '0' are not satisfied, the object will be examined with number '4' conditions using 2 conditions. Firstly, the Euler number should be equal to one. Secondly, the number of flips should be greater than or equal to 5. To retrieve the number of flips, the object is scanned from the top middle point to the bottom of the image, as shown in **Figure 14**.

The pseudo code is as follows;

If [Euler Number =1 && flip_num_middle_vertically >= 5]
Then
The Object is '4';

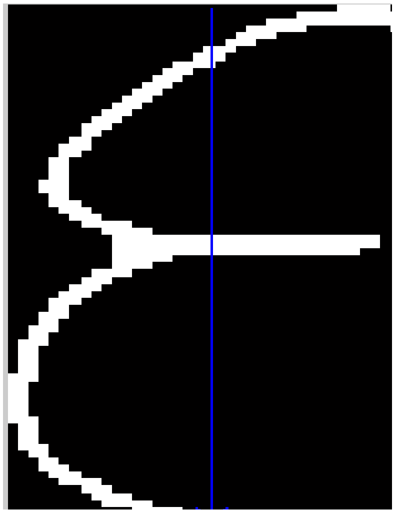
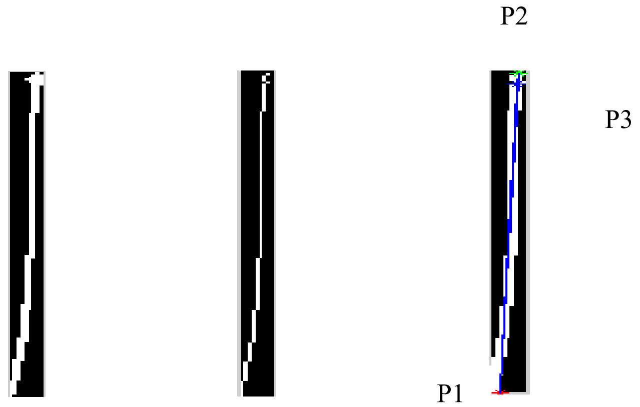


Figure 14 View of the Arabic object number four.

Number '1' Recognition

By examining the object with the aforementioned conditions, if the output of the object cannot be accurately decided, then the following conditions describing the facts about number '1' will be considered. Similarly, in this case skeletonization is utilized to enable extraction of detailed information about number 1. Afterwards, the 2 conditions considered are: firstly, the Euler Number must be equal to one. Secondly, the ratio fraction of the object shall be less than the factor (0.25) (the opposite of the number 2 and 6 recognition). This factor is determined by dividing the width distance by the height distance. In order to compute each mentioned distance, 3 basic points have to be located in the object as shown in the **Figure 15**: *P1* is positioned in bottom of the object, *P2* is positioned upper left, and *P3* lays upper right. As usual each point has its trajectories *x* and *y* as *P(x,y)*. Height, width and slope distances are determined according to Euclidean distance and depicted in **Figure 14**. The determinations have been explained as in Eqs. (2) - (5). Now, the pseudo code for the 3 conditions is as follows;

If [(Euler Number =1 && Width_Height_Ration < 0.25]
Then
The Object is '1';



(1) (2) (3)

Figure 15 Views of the Arabic object number one.

It is important to mention that, there is no need to compute and depend on the slope as the hand-written number "1" is often originated by the writer as a plus or minus slop, therefore, the slope condition has been ignored as it is an irrelevant condition.

Number 6 Recognition

Finally, if conditions for numbers '5' '9', '8', '7', '3', '2', '0', '4', and '1' have not been achieved successfully, the object will be examined with facts about number '6' which are described as follows. Again, skeletonization is initially used to preprocess the object. Afterwards, the 3 conditions are: firstly, the Euler Number must be equal to one. Secondly, the ratio fraction of the object should be greater than or equal to a factor (0.25) (the opposite of number '1' recognition). This factor is determined by dividing the width and height distances. In order to compute each mentioned distance, 3 basic points are located on the object as depicted in the **Figure 16**. *P1* is positioned in bottom of the object, *P2* is positioned upper left, and *P3* lays upper right. Also, since each point has its trajectories *x* and *y* as *P(x,y)* thus the height distance (H), width distance (W), slope distance, width height ratio are determined based on the Euclidean distance using the following Eq. (11) - (14);

$$H_{dist} = \sqrt[2]{(y_3 - y_1)^2 + (x_3 - x_1)^2} \tag{11}$$

$$W_{dist} = \sqrt[2]{(y_2 - y_3)^2 + (x_2 - x_3)^2} \tag{12}$$

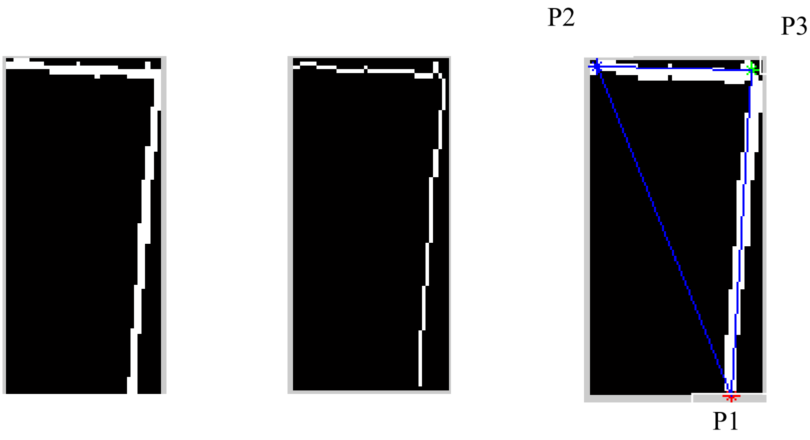
$$slop_{dist} = \sqrt[2]{(y_2 - y_1)^2 + (x_2 - x_1)^2} \tag{13}$$

$$Width\_Height\_Ratio = \frac{W\_dist}{H\_dist} \tag{14}$$

The third condition is that, the angle between the slope (*P1-P2*) and x-axes should be negative (less than zero) to indicate that this object is number '6', which is computed as follows;

$$\theta = \frac{abs(y_2 - y_1)}{x_2 - x_1} \tag{15}$$

$$Angle = tan^{-1}(\theta) \tag{16}$$



(1) (2) (3)

Figure 16 Views of the Arabic object number six.

The pseudo code for the 3 conditions is as follows;

If [Euler Number =1 && Width Height Ratio > 0.25 && slop orientation < 0]
Then
The Object is '1';

Eventually, if the object has been matched with any specified number conditions, then the object will be classified as un-recognized object or object without identity. This case happens due to many reasons due to a distorted or deformed or unclear object.

Experiment and implementation

In order to test the proposed algorithm and ascertain its ability to generalize to any random input number, we consider evaluating the algorithm with random handwritten numbers taken randomly from 15 individuals in this experiment. Each individual has written 10 Arabic numbers from 0 through 9. Overall, the tested numbers are 150 different numbers ranging from 0 to 9. In other words, each number has been repeated 15 times originating from a different writer. The method of acquisition of the numbers to personal computer to be processed by using Matlab platform is either by using a scanner device to input numbers which are written on a paper or by using tablet or PDA devices to input the handwritten numbers directly.

Normally, in the verification or identification comparison, there are 2 possible error measures: False Accept Rate (FAR), which results from the forged template that is accepted by the computer system falsely during testing and False Rejection Rate (FRR), which results from the genuine template that the system recognizes as the query template wrongly [45,46]. Finally, the total accuracy of the system is calculated by subtracting the average error rate from 100 % as in Eq. (17);

$$Accuracy\% = 100\% - \frac{FAR+FRR}{2} \tag{17}$$

In this kind of research, the FAR error does not exist, since there are no forged templates in this experiment. Therefore, FAR is mainly deemed to be zero. However, FRR is largely used for the testing the measure to assess the recognition rate, because the Arabic numbers are considered as genuine templates, if they are wrongly recognized by the computer system, then the FRR increases. For example, number '7' is deemed as genuine template, if the computer system recognized it as 7, FRR is going to be zero, otherwise FRR will be increased. Finally, the equations that are used to estimate the accuracy of this research are in Eqs. (18) and (19);

$$Accuracy\% = 100\% - FRR\% \tag{18}$$

$$FRR\% = \frac{Total\ False\ Reject}{Total\ True\ attempt} \times 100\% \tag{19}$$

Result and discussion

Two types of results are reported as the outcomes of this research. The first result is the recognition error for each distinct Arabic number among the 150 sample set, which is attained by calculating the total output of a specific numeral type divided by the total input (queried) of the same numeral type that have been randomly collected in the dataset. For example, as shown in **Table 1**, the numeral number 9 has been iterated 15 times with no error in the recognition output (FRR=0). For the remaining numeral types, **Table 1** shows the details as dataset iteration times with their system output and also shows the successful accuracy. It can be observed in **Table 1** that the highest False Reject Rate error (FRR) is 20 %, which is specifically related to the Arabic number 2. By looking at the worst FRR, it can be noticed that number 2 has 20 % error, number 3 has 13.33 and number 6 has FRR of 6.66 %. These results are mainly due to the high competence of the writing such that the writing style that has a high degree of rotation or unclear number that the algorithm could not recognize this kind of distorted object as a specific number. Regarding the rest of the numbers zero, one, 4, 5, 7, 8, and 9 have no errors during testing.

Table 1 The number types, computer system outputs, FRR and successful accuracy using the proposed system.

Numerical type	Dataset numerical attempt	System recognized falsely	FRR as in (19) %	Accuracy as in (18) %
0	15	0	0.00	100.00
1	15	0	0.00	100.00
2	15	3	20.00	80.00
3	15	2	13.33	86.66
4	15	0	100.00	100.00
5	15	0	100.00	100.00
6	15	1	6.66	93.33
7	15	0	0.00	100.00
8	15	0	0.00	100.00
9	15	0	0.00	100.00

The second report style from this research is obtained by calculating the recognition rate and FRR error for each individual (user writer), where each individual has written 10 numeral numbers from 0 - 9. Therefore the computation of FRR will be the number of wrongly recognized over 10, the result of FRR will be subtracted from 100 %. **Figure 17** shows a bar chart which describes each attempt among the 15-individuals in *x* axes with their successful accuracy as in 100 % in *y* axis of the chart.

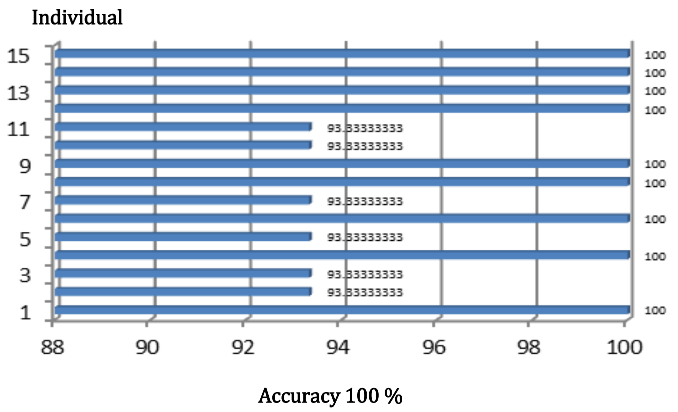


Figure 17 Illustration of the accuracies (%) in x axis with their corresponding 15 individual attempts in y axis.

Here, the overall successful accuracy is 97.3 %, which is the average recognition rate for 15 individuals. It is clear in **Figure 17** that the following individuals: 2, 3, 5, 7, 10, and 11 have 93.3 % accuracy because one of the numbers is not recognized correctly. This rate (93.3 %) is calculated as follows;

$$100\ \% - \left(\frac{1}{15} \times 100\ \%\right) = 93.3\ \% \tag{20}$$

In this research method, there is no dataset training to be matched against as a matching operation. However, it works by extracting facts by using geometric feature extraction in order to be applied to the set of rules by using if-else statements as an expert system works.

Conclusions

An efficient algorithm for Arabic handwritten number recognition is proposed in this paper. Several preprocessing operations are initially applied to the input image such as conversion to binary image, morphological filtering (closing operation), and segmentation before entering the data to the recognition system. The proposed approach is based on extraction of both local and global geometric features. The features are further quantified into a set of facts or conditions that are used for classification based on a set of rules as an expert system. The experiment has been conducted on a 150 random numbers collected from 15 individuals as user writers. The output showed that the recognition error rate in terms of FRR is 4 % or the overall successful accuracy is 96 %. In future work, we would improve and investigate the possibility of increasing the recognition rate by making the algorithm much smarter against difficult or distorted handwritten numbers.

References

[1] R Palacios, A Gupta and PS Wang. Handwritten bank check recognition of courtesy amounts. *Int. J. Image Graph.* 2004; **4**, 203-22.

[2] W Ding, CY Suen and A Krzyżak. A new courtesy amount recognition module of a check reading system. *In*: Proceeding of the 19th International Conference in Pattern Recognition. Tampa, USA, 2008, p. 1-4.

[3] M Wienecke, GA Fink and G Sagerer. Toward automatic video-based whiteboard reading. *Int. J. Doc. Anal. Recogn.* 2005; **7**, 188-200.

[4] B Al-Badr and SA Mahmoud. Survey and bibliography of Arabic optical text recognition. *Signal Process.* 1995; **41**, 49-77.
[5] D Trier, AK Jain and T Taxt. Feature extraction methods for character recognition-a survey. *Pattern Recogn.* 1996; **29**, 641-62.
[6] J Pradeep, E Srinivasan and S Himavathi. Diagonal based feature extraction for handwritten alphabets recognition system using neural network. *In*: Proceeding of the 3rd International Conference on Electronics Computer Technology. Kanyakumari, India, 2011.
[7] L Ferrer, E Shriberg, S Kajarekar and K Sonmez. Parameterization of prosodic feature distributions for SVM modeling in speaker recognition. *In*: Proceeding of the IEEE International Conference in Acoustics, Speech and Signal Processing. Honolulu, USA, 2007.
[8] A Khmag, AR Ramli, SJB Hashim and SAR Al-Haddad. Additive noise reduction in natural images using second-generation wavelet transform hidden Markov models. *IEEE J. Trans. Electr. Electron. Eng.* 2016; **11**, 339-47.
[9] CL Liu, K Nakashima, H Sako and H Fujisawa. Handwritten digit recognition using state-of-the-art techniques. *In*: Proceeding of the 8th International Workshop in Frontiers in Handwriting Recognition. Ontario, Canada, 2002, p. 320-5.
[10] MH Shirali-Shareza, K Faez and A Khotanzad. Recognition of handwritten farsi numerals by zernike moments features and a set of class specific neural network classifiers. *In*: Proceeding of the International Conference on Signal Processing Applications, and Technology. Dallas, Texas, 1994, p. 998-1003.
[11] MH Shirali-Shahreza, K Faez and A Khotanzad. Recognition of handwritten Persian/Arabic numerals by shadow coding and an edited probabilistic neural network. *In*: Proceeding of the International Conference on Image Processing. Washington DC, 1995, p. 436-9.
[12] H Almuallim and S Yamaguchi. A method of recognition of Arabic cursive handwriting. *IEEE Trans. Pattern Anal. Mach. Intell.* 1987; **9**, 715-22.
[13] A Amin and HB Al-Sadoun. A new segmentation technique of Arabic text. *In*: Proceeding of the 11th International Conference on Pattern Recognition Methodology and Systems. The Hague, 1992, p. 441-5.
[14] A Amin and HB Al-Sadoun. Hand printed Arabic character recognition system. *In*: Proceeding of the 12th Conference on IAPR International. Jerusalem, 1994, p. 536-9.
[15] I Abuhaiba, MJ Holt and S Datta. Processing of binary images of handwritten text documents. *Pattern Recogn.* 1996; **29**, 1161-77.
[16] S Saadallah and S Yacu. Design of an Arabic character reading machine. *In*: Proceeding of the Computer Processing of Arabic Language. Kuwait, 1985.
[17] A Saleh. A method of coding hand-written Arabic characters and its application to context-free grammar. *Pattern Recogn. Lett.* 1994; **15**, 1265-71.
[18] IS Abuhaiba, MJ Holt and S Datta. Recognition of off-line cursive handwriting. *Comput. Vis. Image Understand.* 1998; **71**, 19-38.
[19] MS Khorsheed. Recognising handwritten Arabic manuscripts using a single hidden Markov model. *Pattern Recogn. Lett.* 2003; **24**, 2235-42.
[20] S Almaadeed, C Higgins and D Elliman. Off-line recognition of handwritten Arabic words using multiple hidden Markov models. *Knowl. Base. Syst.* 2004; **17**, 75-9.
[21] HA Al-Muhtaseb, SA Mahmoud and RS Qahwaji. Recognition of off-line printed Arabic text using Hidden Markov Models. *Signal Process.* 2008; **88**, 2902-12.
[22] H Bunke, S Bengio and A Vinciarelli. Offline recognition of unconstrained handwritten texts using HMMs and statistical language models. *IEEE Trans. Pattern Anal. Mach. Intell.* 2004; **26**, 709-20.
[23] Y Kessentini, T Paquet and AB Hamadou. A multi-stream HMM-based approach for off-line multi-script handwritten word recognition. *In*: Proceeding of the 11th International Conference on Frontiers in Handwriting Recognition. Montreal, Quebec, Canada, 2008, p. 147-52.
[24] E El-Alfy and RE Abdel-Aal. Using GMDH-based networks for improved spam detection and email feature analysis. *Appl. Soft Comput.* 2011; **11**, 477-88.

[25] R Fox and W Hartmann. An abductive approach to hand-written character recognition for multiple domains. *In*: Proceeding of the International Conference on Artificial Intelligence. CSREA Press, 2006, p. 349-55.
[26] T Plötz and GA Fink. Markov models for offline handwriting recognition: A survey. *Int. J. Doc. Anal. Recogn.* 2009; **12**, 269-98.
[27] A Amin and JF Mar. Machine recognition and correction of printed Arabic text. *IEEE Trans. Syst. Man Cybern*. 1989; **19**, 1300-6.
[28] LD Harmon. Automatic recognition of print and script. *Proc. IEEE* 1972; **60**, 1165-76.
[29] FN Said. 1996, Recognition of Symbols and Characters on Engineering Drawings and Maps. Ph.D Dissertation, Concordia University, Canada.
[30] N Indurkhya and SM Weiss. Heuristic configuration of hidden units, in backpropagation neural networks. *In*: Proceeding of the International Joint Conference on Neural Networks. Seattle, WA, 1991.
[31] M Gardner and S Dorling. Artificial neural networks (the multilayer perceptron): A review of applications in the atmospheric sciences. *Atmos. Environ.* 1998; **32**, 2627-36.
[32] YL Cun and Y Bengio. *Convolutional Networks for Images, Speech, and Time Series*. *The Hand Book of Brain Theory and Neural Networks*. MIT Press Cambridge, MA, USA, 1998, p. 255-8.
[33] F Al-Omari. Handwritten Indian numeral recognition system using template matching approaches. *In*: Proceeding of the IEEE International Conference on Computer Systems and Applications. Beirut, Lebanon, 2001, p. 83-8.
[34] J Sadri, CY Suen and TD Bui. Application of support vector machines for recognition of handwritten Arabic/Persian digits. *In*: Proceeding of the 2nd Iranian Conference on Machine Vision and Image Processing. Tehran, Iran, 2003, p. 300-7.
[35] J Arlandis, JC Perez-Cortes and J Cano. Rejection strategies and confidence measures for a k-NN classifier in an OCR task. *In*: Proceeding of the 16th International Conference on Pattern Recognition. Quebec, Canada, 2002, p. 576-9.
[36] JA Rodrıguez and F Perronnin. Local gradient histogram features for word spotting in unconstrained handwritten documents. *In*: Proceeding of the International Conference on Frontiers in Handwriting Recognition. Montréal, Québec, Canada, 2008.
[37] RM Haralock and LG Shapiro. *Computer and Robot Vision*. Addison-Wesley Longman Publishing, Boston, USA, 1992.
[38] P Jackson. *Introduction to Expert Systems*. Addison-Wesley Longman Publishing, Boston, USA, 1986.
[39] M Ahmed and RK Ward. An expert system for general symbol recognition. *Pattern Recogn.* 2000; **33**, 1975-88.
[40] M Negnevitsky. *Artificial Intelligence: A Guide to Intelligent Systems*. Addison-Wesley, Boston, USA, 2005.
[41] M Sasikumar, S Ramani, M Raman, KSR Anjaneyulu and R Chandrasekar. *A Practical Introduction to Rule Based Expert Systems*. Narosa Publishing House, New Delhi, India, 2007.
[42] T Sharma, N Tiwari and D Kelkar. Study of difference between forward and backward reasoning. *Int. J. Emerg. Tech. Adv. Eng.* 2012; **2**, 271-3.
[43] LG Shapiro and RM Haralick. *Computer and Robot Vision*. Vol I. Addison-Wesley, USA, 1992.
[44] TY Kong and A Rosenfeld. *Topological Algorithms for Digital Image Processing*. Vol I. Elsevier, North Holland, 1996.
[45] F Malallah, SMS Ahmad, WAW Adnan, OA Arigbabu, V Iranmanesh and S Yussof. Online handwritten signature recognition by length normalization using up-sampling and down-sampling. *Int. J. Cyber Secur. Digit. Foren.* 2015; **4**, 302-13.
[46] F Malallah, SMS Ahmad, WAW Adnan and S Yussof. Non-invertible online signature biometric template protection via shuffling and trigonometry transformation. *Int. J. Comput. Appl.* 2014; **98**, 4-17.