12

Applying Genetic Algorithm in Multi Language's Characters Recognition

Hanan Aljuaid
Faculty of Computer Science and Info System, Taif University, Taif, Saudi Arabia

1. Introduction

The character recognition (CR) mechanization is being intensively investigated in the pattern recognition research area. CR automation means translating images of characters into a text; in other words, it represents an attempt to simulate the human reading process. CR is very difficult to accomplish owing to various issues such as the inconsistency of human writing, the segmentation of words into characters, high variability in terms of handwriting styles and shapes, the size of the lexicon, and the writing skew or slant.

There are two main classifications of the problem of handwriting recognition: online recognition and offline recognition; these terms refer to the format of the input handwritings image. Temporal information is available in online recognition, for instance pen tip coordinates as a time function, while in offline recognition, just the handwritings image is obtainable. Several applications require offline handwriting recognition capabilities; these include commercial form reading, bank processing, document archiving, office automation, mail sorting, etc. Up to the present time, offline handwriting recognition is still an open problem, and has been dealt with by several researchers in this field (Benouaretha *et al.*, 2008;Plamondon and Srihari, 2000; Koerich *et al.*, 2003; Vinciarelli, 2002).

While many different methods of solving the OCR problem have been explored, the use of a genetic algorithm to recognize characters has been growing in popularity. "Genetic algorithms offer a particularly attractive approach for this kind of problems since they are generally quite effective for rapid global search. Moreover, genetic algorithms are very effective in solving large-scale problems"(Oliveira *et al.*, 2001).This begs the question of what genetic algorithms (GAs) actually are.

A GAs is an optimization and search technique utilized in computer science to find approximate solutions to problems. It is inspired by processes in biological evolution such as natural selection, inheritance, recombination, and mutation. GAs are generally realized in a computer model, in which a population of runner solutions to an optimization problem progress to better solutions. The evolution starts from a population of completely random individuals and occurs in generations. In each generation, the fitness of the whole population is evaluated, and multiple individuals are selected from the current population based on their fitness. These are modified, mutated, or recombined to create a new population, which becomes current in the next iteration of the algorithm, as shown in Figure

1. Usually, the solutions are represented in strings of 0s and 1s, although different encodings are also possible. So, evolutionary algorithms play on populations, instead of coming to one solution.

This chapter will give an extended illustration of offline character recognition. The system is based on feature extraction and the genetic algorithms approach.

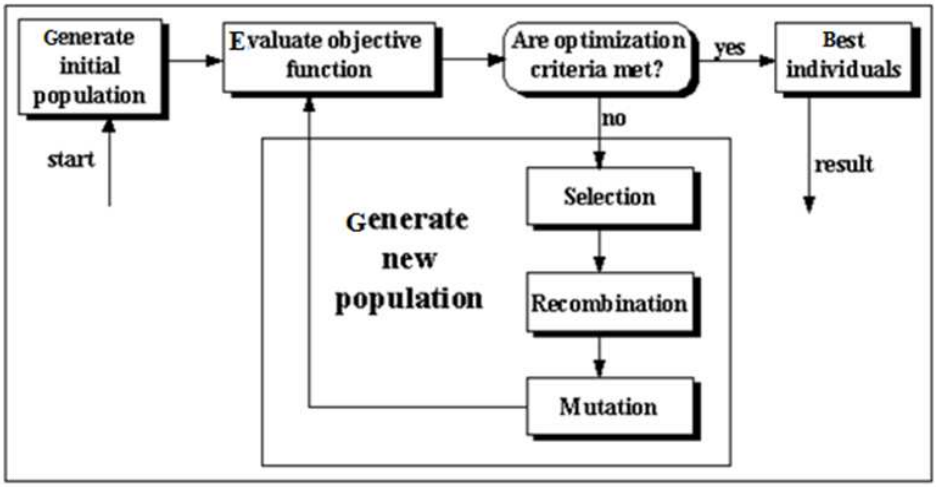


Fig. 1. Structure of a single population evolutionary algorithm

2. Character recognition stages

The character recognition system can be decomposed into several stages: preprocessing, representation, character segmentation, feature detection, and character recognition. Some systems use a subset of these stages. These stages are described in Table 1.

Preprocessing	Noise removal, text detection, etc.
Representation	Skeletons, contours, baseline detection
Segmentation	Segments words, sub-words, characters, strokes, or other units
Features	Information is passed to the recognizer such as shape attributes, pixels
Recognizer	Algorithm that identifies letters

Table 1. Components of OCR recognition

When we focus on these stages, we can see that the recognition method starts by cleaning the image through the use of image processing techniques in the preprocessing stage. Then, the image can be improved to form an extra-short representation. Features of words or characters are identified after the segmentation stage. Finally, from these features, the recognizer proceeds to recognize the wording. The expression "features" refers to any

quantities that are recognized. They can be pre-computed through segmentation and computed for individual characters, as shown in Figure 2. The importance of these stages, their impact on the recognition method, and the relation between them will be discussed in the next sections.

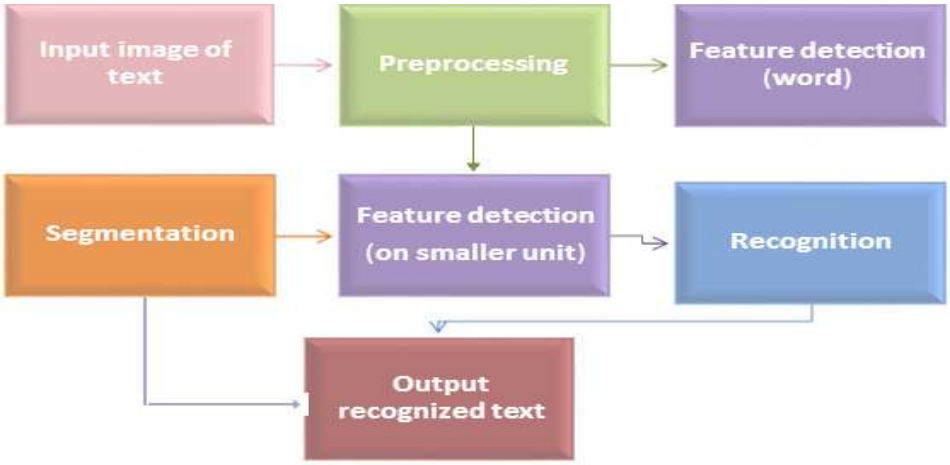


Fig. 2. Character recognition stages

2.1 Pre-processing and representation

The methods of the preprocessing phase are divided into two types of methods according to their function. First, the methods function to construct clean and usable raw data like noise reduction. Second, the smoothing and normalization methods function to prepare the data image to be segmented and recognized in terms of features, for instance: contour tracing, skeleton extraction, and vertical and horizontal projection. These methods represent the preprocessing of the recognition process. However, the segmentation approach can be defined as preparing a short and snappy representation of the word image to be segmented. The common techniques in the representation processes are: vertical and horizontal projection, contour tracing, and skeleton extraction which discussed in the following section.

2.2 Vertical and horizontal projection

The vertical projection methods importance appears in detecting the junction lines between the adjacent characters and the white spaces by counting the black pixels in every column of the image. However, the vertical projection method is not efficient in handwritten recognition owing to overlapping and skew problems. To explore the word image into lines, words, and characters it has been used with horizontal projection,. These techniques take into account that the link between characters is not always as wide as the letters. The projections profile of the image is best achieved in these techniques.

The definition of horizontal projection is as follows:

$$h\left(i\right) = \sum p\left(i, j\right)$$

For vertical projection, it is:

$$v\ (j) = \ \Sigma\ \mathrm{p}\ (i, j)$$

where i is the row number and j is the column number. P is the pixel value. The value is 0 for a white pixel (background) and 1 for a black pixel (foreground).

Figure 3 illustrates the horizontal and vertical projection profiles of an Arabic sentence after eradicating the secondaries. The highest point in Figure 3(c) represents the baseline. The thickness of the baseline is determined by computing the thickness of the longest spike, taking the most repeated column height (Timsari and Fahimi, 1996), or considering the position of loops as a reference, as these are always close to the baseline (Olivier *et al.*, 1996). Among other essential information in the projection profile that can be subtracted is the width, height, and amount of connected components of sub-words (Al-Yousefi and Udpa 1992; Mohammed 2006).

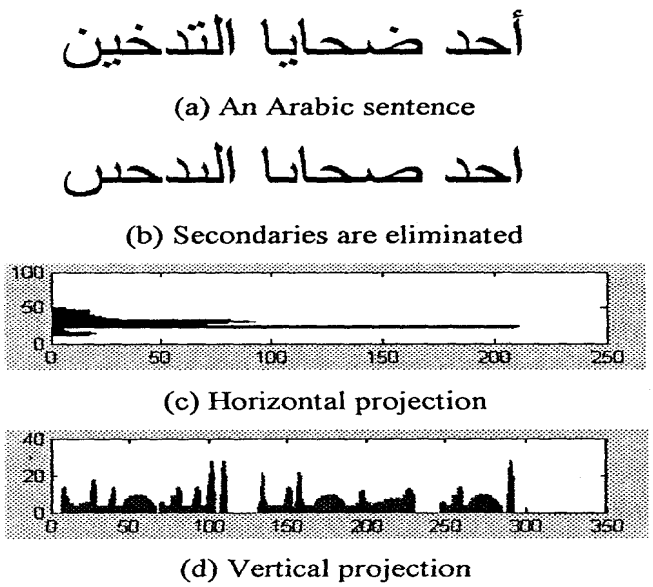


Fig. 3. Horizontal and vertical projections (Mohammed, 2006)

However, more researchers use the vertical and horizontal projection profile for different functions to prepare the image for recognition. For example, they use it to detect the baseline or to segment the word into characters.

One of the most important functions of projection profiles in character recognition is detecting the baseline. The researcher used dissimilar algorithms to detect the baseline according to the projection profile methods. One of these methods to divide the image into zones according to the intensity of pixels in each zone; the highest intensity zone of black pixels is the baseline. The middle zone is dealt with using the vertical projection in Sarfraz *et*

al. (2003), where four regions or zones are identified: the baseline, upper, middle, and lower regions. The zones are defined according to the baseline zone, which has the most black pixels, the middle region is the region on top of the baseline and double the width of the baseline, and has a constructed vertical projection (Sarfraz *et al.*, 2003). The point that all researchers agree on is that the baseline has the highest number of black pixels in the horizontal projection profile (Altuwaijri and Bayoumi, 1995; Hashemi *et al.*, 1995).

On the other hand, the importance of the projection profile appears in the segmentation method. Different methods are used to segment the word into characters using the projection profile. One approach uses a permanent entrance to do so (Nawaz *et al.*, 2003). The connection region between two letters is detected when the projection rate of the middle region is lower than two-thirds of the baseline width. If the outline is larger than one-third of the baseline the start of a new character is detected and the current region tracks the connection region with a better rate. However, other works have applied the vertical projection profile in the direction of determent the pixels of the baseline and secondaries, as in Altuwaijri and Bayoumi (1995).

The necessary information of a shape in character recognition is stored in its skeleton (Zeki, 2005). In other words, the thinning process means creating the skeleton of the image. A skeleton created by highlighting the centerline of the word image, it is a one-pixel-width image. This helps to bring out fundamental information about the word. Figure 4 shows an example of image thinning where the skeleton of the word has one pixel of width.

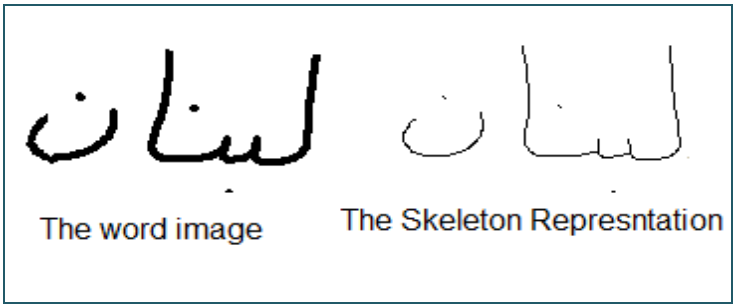


Fig. 4. A word image and its skeleton

The thinning operation improves the ability to recognize and segment characters. However, in the segmentation process, extracting segments from the skeleton graph is more trustworthy than finding the real linked points in a word (Abuhaiba *et al.*, 1994; Khorsheed and Clocksin, 1999). The skeleton can be segmented into strokes. Each stroke begins and ends with a feature point. A feature point could be the end point, branch point, or cross point (Amin *et al.*, 1996; Khorsheed and Clocksin, 1999). Other segmentation techniques begin by determining the baseline of word; after that, the segmentation points are detected by measuring the columns with pixels around the baseline. This point will appear at the middle of the association stroke (Khaly and Sid-Ahmed, 1990). Then, the explore for the starting tip is set in the region of the baseline to extract the stroke; after this, the search for the stroke end point starts by tracing the curve. An end point may be a branch point, a cross point, a point with unexpected modify in the curve following a horizontal movement close to the baseline, or a line end point (Almuallim and Yamaguchi, 1987; Zeki, 2005).

Vertical projection detects the beginning and ending points of characters; these points might be true or applicator points. The true beginning point will be detected, if there is a transformation from zero to nonzero in the vertical projection; the true ending point is detected, if there is a transformation from nonzero to zero (Jambi, 1991; Abandah and Khedher, 2004). This method needs further processing in the presence of vertical overlaps. Other methods have positioned character graph models to recognize isolated letters. Every character's skeleton is transformed to a hierarchy structure that is coordinated to a model through a rule-based recognizer. Every model becomes a state machine through a conversion analogous to the instructions of segments inside the character and with other "fuzzy" restrictions to differentiate some characters from others (Abuhaiba et al., 1994; Lorigo and Govindaraju, 2006).

2.3 Contour tracing

The tracing of the contour is conducted to detect border line pixels or when the contour contains significant information about an item (Khorsheed, 2002). It aims to transform the edge of the word into a series of codes to explore the images features. The coding method begins by discovering the location of an initial pixel then identifies the relations of the following pixels on the contour until it reaches the initial pixel.

RPCT stand for Regional projection contour transformation, it is projected the image in a number of directions which is vertical and horizontal. This was presented with the chain code contour of each projection. The contour was modeled and features were achieved for every segment using a two-dimensional model that took into account the amount of dynamic pixels, slope, and curvature. Individual HMMs were used to create and reproduce split-feature vectors from; the contours of the horizontal and vertical projections were compliant with two HMMs per character. Through recognition, individual categories were incorporated to improve performance (Dehghan, 2001; Lorigo and Govindaraju, 2006).

Tracing the contour also plays a main role in segmentation. It is used in the Segmentation and Recognition of Arabic Printed Text (SARAT) system (Margner, 1992), a segmentation method stand on the external contour of the words. The algorithm starts by detecting the end points of the upper contour. Next, the upper contour segmented into pieces takes place, including a curve of the same symbol.

Techniques depending on tracing the contour keep away from the problems resulting on the thinning procedure since they study the structural of the characters forms according to the scanning. Nevertheless, they are affected by noise on the contour; hence, the contour needs to be thinned first.

2.4 Baseline detection

The baseline is an imaginary line used to connect the characters of the word. It is usually detected in the segmentation stage and helps in characterizing the strokes of the characters. Several methods have been published for detecting the baseline. Kanai *et al.* (1998) used the projection profile technique to detect the fiducial points by interpreting the lowest resolution layer of the image. Detecting the baseline is an ordinary step in many offline handwritten Arabic recognition OACR systems, and it is often an essential step before the segmentation and the feature extraction steps.

For instance, El-Hajj *et al.*'s (2005) system depends on the upper and lower baselines. These are contained by the HMM recognizer in the context of frame-based features, which are integrated features measuring densities, transitions, and concavities in zones defined by the detected baselines. The IFN/ENIT database was used to test the system. For every experiment, three images of the four image sets was used to train the system and tested on the excluded set. In the experimentation, the addition of baseline-dependent features to similar measurements that do not use those zones significantly improved recognition (El-Hajj *et al.*, 2005; Lorigo and Govindaraju, 2006).

2.5 Segmentation

The segmentation stage is necessary when it comes to recognizing Arabic wording. Each mistake in the segmentation method the main form of the characters will generate another representation of the character's form (Amin, 1998). One of the types of recognition strategies needed in the segmentation stage is an analytical strategy, as discussed in Section 2.3. Analytical strategies are sub-classed into two methods that have affected the segmentation mechanism of printed and handwritten Arabic text into individual characters: implicit and explicit segmentation (Amin, 1998).

i. In explicit or external segmentation, the word is segmented into characters after that recognized each character independently. This method is typically more costly because of the greater difficulty of locating the best word.
ii. In implicit or internal segmentation, words are segmented to characters and recognized all together. This category of segmentation is generally considered to facilitate recognition of all the characters through rules. Various rules need to be created manually to obtain excellent accuracy. Thus, the higher the number of rules, the better the recognition.

More researchers have used explicit segmentation to prepare a letter for recognition. In most of their work, the segmentation stage depends on the representation stage. Haraty and Ghaddar use this method to segment the characters and identify ligatures. Their method is based on a thinning technique, along with structural and other features such as the statistics of corner points, loops, division points, and endpoints, and the density and quantity of black pixels. In addition, double neural networks are presently being employed. In these systems, the projected breakpoints were confirmed or rejected by the neural network (Haraty and Ghaddar, 2003).

Conversely, another type of segmentation strategy called (recognition based segmentation), and is an implicit technique. This is unlike the methods discussed above, which were regarded as explicit segmentation techniques. In implicit techniques, the segmentation and recognition of the characters occur at the same time. The basic theory of this approach is that provisional segmentations can be presented through exploiting a changeable window with variable widths that are or are not completed by the categorization (Cheung *et al.,* 2001).

One example of the implicit type of segmentation is that of El-Dabi *et al.* (1990), where the invariant moments are used and checked in terms of their alignment with the feature space of the font. If a character is not found, another column is affixed to the original portion of the word and moments are calculated and checked again. This procedure is repeated until a character is recognized or the end of the word is reached. This method allows the system to

handle overlapping and to isolate the connecting baseline between connected characters. The segmentation rate resulting from this method was 83% (El-Dabi *et al.*, 1990).

As can be noticed from the above discussion, the segmentation approach aims to solve the most serious problems of traditional segmentation. Hence, no accurate character segmentation path is necessary. In principle, any of the other approaches can be used here as long as they have some recognition capabilities (Cheung *et al.,* 2001).

2.6 Feature extraction

Feature extraction is used to characterize the tokens that can be applied for a special recognition of every character. Features categorized into two types. First, there are local features; these are typically *geometric* (e.g., curved in their convex parts and types of junctions: intersections, T-junctions, endpoints, secondaries , loops, height and width of stocks, etc.). Second, there are global features that are typically statistical (invariant moments, Fourier transform, etc.) or *topological* (number of connected components, connectivity, number of holes, etc.) (Amin, 1997).

For instance, the Freeman chain code is commonly used for feature extraction moreover than segmentation process or the recognition process (Abdullah, 2007). On the other hand, invariant moments are also used as a feature in some studies, as in El-Dabi *et al.* (1990).

As can be observed from the discussion in previous sections and below, these techniques all use feature extraction and the work cannot succeed without it. This work will be use six local features of characters length, width, loop, left connection, right connection and complementary.

3. Genetic algorithm

GAs can be defined as a "class of optimization and search methods that use randomness to avoid local extreme solutions"(Kherallah *et al.*, 2009). A GA is an iterative algorithm based on many generations of probable solutions, among selection schemes authorization the removal of bad solutions and the reproduction of good ones that can be modified (Kherallah *et al.*, 2009).

Genetic algorithm works with a special fitness to evaluate each individual or string in the population. The evolution starts from population of randomly generations. There are other operation that helps in selected the individuals or strings of the population based on their fitness. The individuals may modify or mutated to form a new population, which used in the next iteration. The algorithm terminates when reached fitness level for the population. If the algorithm has concluded due to a maximum number of generations, a suitable solution may or may not have been reached.

The present chapter depends on the use of GA techniques for character recognition. However, when we focus on research that used GAs for character recognition, there is some works used GAs in character recognition. Wherever, these studies used GAs to recognize online handwriting. Only one study used GAs with NNs, and to recognize offline handwriting, and this was for Latin script. The handwriting recognition model described in this work progresses in three stages: (1) the segmentation of the handwritten text, (2) the

recognition of segmented characters with the help of ANNs, and (3) the selection of the best solution from the four ANN outputs with the help of the GA. A strong algorithm for handwriting segmentation has been described, with the help of which individual characters can be segmented from a word selected from an image of a paragraph of handwritten text, which is given as input in the model. The algorithms were tested with 200 handwritten samples; out of these, 142 samples were correctly recognized, representing an overall efficiency of 71% (Mathur *et al*., 2008).

On the other hand, GAs have used with good results for online Latin character recognition. Menier *et al.* (1994) have presented a genetic algorithm for the online recognition of cursive handwriting. The GAs work with a population of solutions called strings. Each string has a lexical picture and a graphic primitive list that describes how the word is written. Each string is made with construction blocks called allographs. The GAs are used to find the best reconstruction of the word to be analyzed, based on graphic primitives and using the allograph list. This can be seen as an alternative analysis method for word recognition that does not require the definition of a scanning strategy. This system achieved 84% recognition in a manuscript test with a lexical set of 150 words and a small allograph set. The recognition subset consists of 160 words, including ten extra words not belonging to the lexicon (Menier *et al*, 1994).

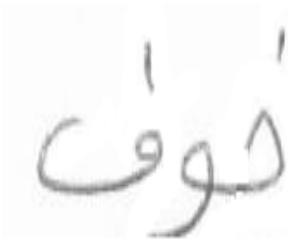
Ramin Halavati *et al.* (2006) employed a narrative Multiple States Machine as a general tool for elastic pattern recognition and utilized an evolutionary method to generate this machine. The most important scheme after the machine is to expand and sustain special hypotheses about the specified sequence of segments and increasingly confirm or reduce them to reach a single ending result. This is applied using the Persian language, employing a typical feature set and a specific tailored GA. The identification and calculation times were contrasted with a dynamic programming comparison approach. This approach achieved an 89% recognition rate without a dictionary and 96.1% with a dictionary for Persian language groups. Moreover, it was compared with pruned dynamic programming (DP), and shown to present an almost constant recognition speed, whereas DP's computational time increases exponentially when the number of segments increases. Thus, this system gave results more than 10 times faster than DP for 9 segment words, and 100 times faster for words with 13 segments (Halavati *et al.* , 006).

4. Method of feature extraction

In this work, there is a special strategy to segment the word into characters and recognize each character: the whole word is scanned and its features extracted. Each word has eight types of features: (1) quantity of sub-words, (2) quantity of peaks for every sub-word, (3) number of loops for every peak, (4) number and location of complementary characters, (5) the width and (6) height of every peak, and (7) whether there is left and (8) right connection.

After the segmentation of the word, the feature of each character must be detected to recognize the shape of the character. The recognition algorithm is based on six features of each character shape. These features are (1) the length of the character, (2) its width, (3) whether or not it has a loop, (4) whether there is a right character connected to it, (5) whether there is a left character connect to it, and (6) whether there is a complementary character like the hamza, one point, two points, or three points, etc.

(a)



(b)

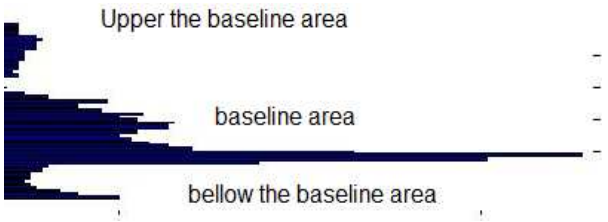


Fig. 6. Detecting the baseline by the horizontal projection profile: (a) handwritten word, (b) horizontal projection profile to detect the baseline.

This algorithm apply in the characters are written cursively and all the characters in the word are based upon the baseline. The feature extraction algorithm depends on this, so in this algorithm the character image is divided to three areas: The first is the baseline area that was detected by the horizontal projection profile; the longest spike represents the baseline, as shown in Figure 6. The second is the area above the baseline; this area usually contains the upper complementary and the upper length of a character, as in ,كـ in Arabic characters and l, h in Latin characters. The third area is the area below the baseline, which usually contains the lower complementary character and the lower length of the character, as in ، وج, g ,y See Figure 6.

The vertical and horizontal projection profiles are calculated for each area individually, to help in detecting the features of the character. Each character shape has six features that are unique to it; no other shape has that feature combination.

After defining the area of image and its projection profile, the feature extraction algorithm starts by tracing the boundary of the image and traveling around the eight neighbourhoods. One vector array has six cells called feature vectors, where each cell represents one feature of the character's shape using an integer value, as shown in Figure 7, that describes the feature vector of character shape ـه . This character is short, has a small width, has a loop in the baseline, has no right connection, has a left connection, and has no complementary character. This vector is unique for the character shape ـه .

Each feature is detected according to the following strategy:

Length: This is denoted by 0 if the character shape is short or 1 if the character shape is long. The length of the character is detected by the baseline area; if it is in the baseline area only, it is short; otherwise, it is long.

Width: This is denoted by 0 if the character shape has a small width and 1 if the character shape has a large width. The width of the character shape depends in the vertical projection profile of the image and the thinning algorithm where all the images have one pixel width.

Loop	Length	Width	Left connection	Right connection	Complementary character
-1	0	0	1	0	0

Fig. 7. Feature vector representing the character ـه

Loop: This is denoted by 0 if the character shape has no loop; 1 if the character shape has one loop in the baseline area above the baseline like in ـه ,a ; -1 if the character shape has one loop below the baseline, ـه ; 2 if the character shape has two loops above the baseline as in Bهـ; 3 if the character shape has an open loop on the left side and it is standing on the baseline, as in ـه ,z ; or -3 if the character shape has an open loop on the right side and it is standing on the baseline, as in ـه ,c,G . The type of loop is detected according to the following algorithm:

a. If the subtraction of the entire nearest pixel row in the same column is >1, and the subtraction of all the nearest pixel column in the same row is >1, the loop type is 1 if the last row is the baseline. Otherwise it is -1.
b. *If the subtraction of the entire nearest pixel row in the same column is >1,* except the first column, which has one pixel only and the second, which has two pixels behind one another, the loop type is 3. Otherwise the loop type is -3.
c. *If there are more than two pixels in the same column, the subtraction of it is >1*, and the subtraction of all the rows in the same column and all columns in the same row is >1, the loop type is 2.
d. Otherwise the loop type is 0.

The right connection: is denoted by 0 if there is no character on the right side; otherwise it is 1. This is detected by the vertical projection profile of the baseline area; if there are ones on the right side of the character width, it denoted by 1; otherwise it is 0.

The left connection; is denoted by 0 if there is no character on the left said; otherwise it is 1. This is detected by the vertical projection profile of the baseline area; if there are ones in the left side of the character width, it is denoted by 1; otherwise it is 0.

The complementary character: is denoted by 0 if there are no complementary character, 1 if there is one dot above the baseline area, -1 if there is one dot below the baseline area, 2 if there are two dots above the baseline area, -2 if there are two dots below the baseline area, if there are three dots above the baseline area, and 4 if there is a hamza above the baseline area. This is detected using the vertical projection profiles of the areas. The devised feature algorithm is illustrated in Figure 8.

5. Applying GA

When the features of the characters in the sub-word are determined, the next phase is to recognize the characters of the sub-word. The genetic algorithm approach will be used for this purpose. The GAs will recognize the character depending on the following strategies:

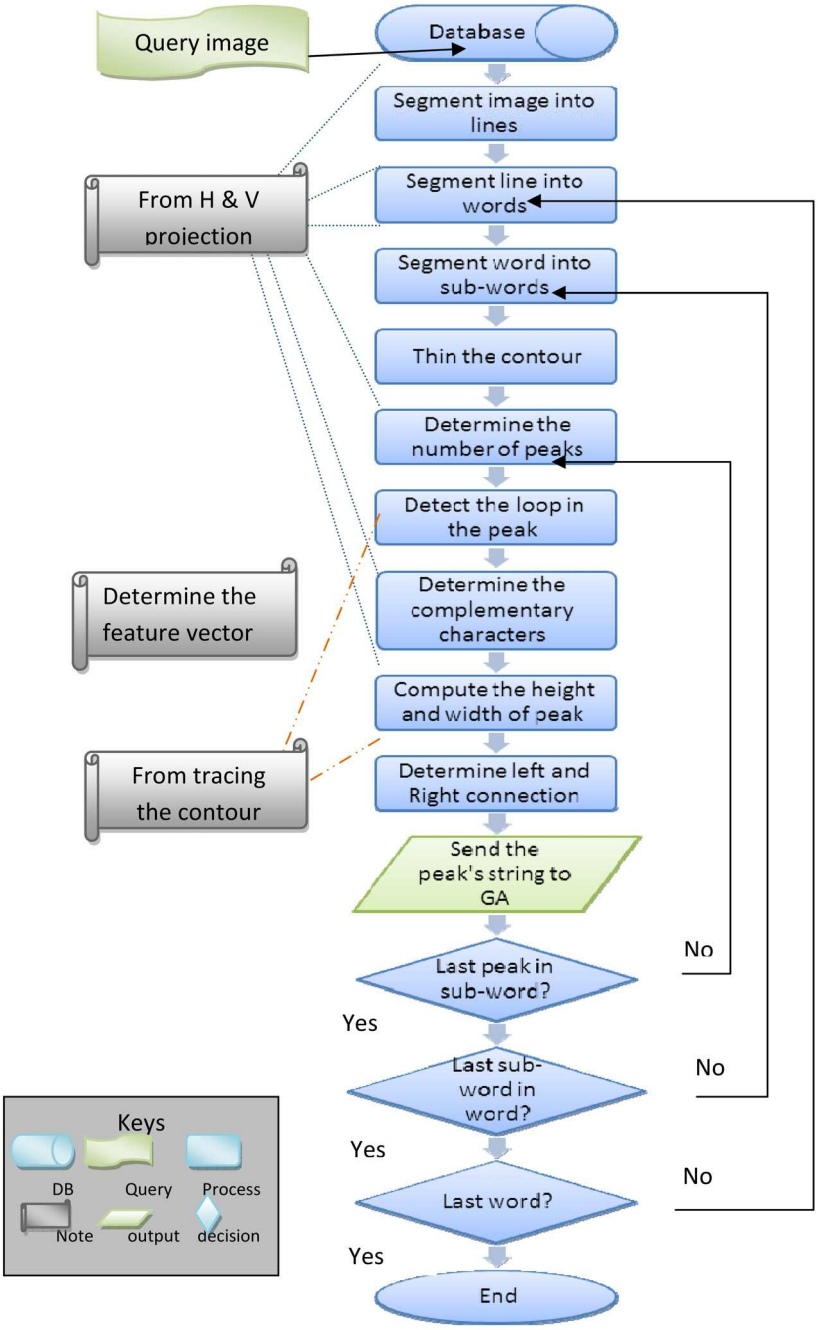


Fig. 8. Feature extraction algorithm

5.1 The fitness function

The value returned from the fitness function for one gene represents the degree of matching between the feature vector of the character represented by that gene and the feature vector of the real character. The fitness function here will be counted as the same number of bits in two feature vectors and returns the value of the same features. If this number is 6, it is the optimal solution. For example, to recognize the characters of the sub-word من, there are two feature vectors shown in Figure 9 for the character shapes ـه and ـن that are used to calculate the fitness function for the character shape ـن . Table 2 shows the summed feature vector values of a character shape with random population values; if the number of one's in the answer is equal to 6, this is the best gene. In Table 2, there are three random genes: the first has a fitness value of 3, the second has a fitness value of 5, and the third has a fitness value of 5. Table 3 shows the fitness function of the population genes of the character shape ـه.

-1	0	0	1	0	0

Feature vector representing the character ـه

0	1	1	0	1	1

Feature vector representing the characterـن

Fig. 9. Feature vectors of ـه and ـن

ـن Main feature vector	Random population feature vector	Calculation	F
011011	101010	001110	3
	011010	111110	5
	111011	011111	5

Table 2. Fitness function of the character ـن

Main feature vector	Random population feature vector	Calculation	F
-100100	101010	110001	3
	011011	000000	0
	111011	100000	1

Table 3. Fitness function of the character ـه

5.2 Selection reproduction operator

The importance of this operator is that it reproduces an opposition among diverse feature vectors. The Tournament selection type of selection operation used, where this selection reproduction operator makes sure that better feature vectors are found while the poorer feature vectors are marked as redundant. It establishes new feature vectors within the population, along with the other operators, specifically the crossover operator and the mutation operator. In our example in the character " ـن " in Table 2, the gene with fitness

value 3 is the worst one, so it is discarded; the best function has a value of 5, but we have two such feature vectors, so the two feature vectors will be duplicated, as show in Figure 10. In the character "ﻫ " in Table 3, the fitness value 0 is the worst one, so it is discarded, whereas 3 is the best function, and will be duplicated.

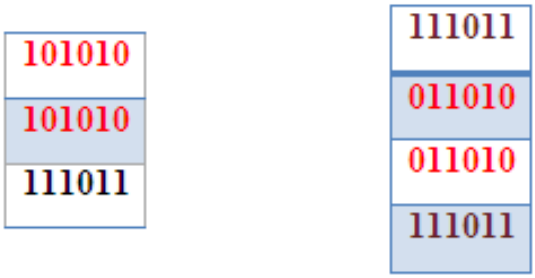


Fig. 10. Example of reproduction operator

5.3 Crossover operator

A single crossover point is applied. Feature vectors are duplicated at random with a higher probability for the feature vectors undergoing crossing. For each parent, a sub–feature vector is chosen where the resultant feature vector is constructed by concatenation of the sub–feature vectors, as shown in Figure 11.

Fig. 11. Example of crossover operator

5.4 Mutation operator

To avoid the recombination of the same feature vectors and to expand the explored solution space, as well as to alternate between some chromosomes, the mutation operator is used in combination with the crossover operator. This gives new and possibly improved solutions, as shown in Figure 12, where the 1 in the fifth index changes to 0.

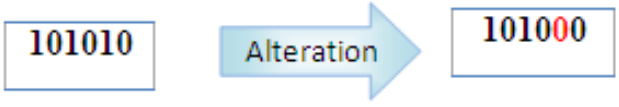


Fig. 12. Example of mutation operator

6. Evaluation

The evaluation of the algorithm depends on the number of words, sub-words, and characters that are recognized, and on the number of words, sub-words, and characters that are unrecognized, out of the number of tested words, sub-words, and characters. The percentage of words recognized will be calculated as:

$$W = \frac{Number\ of\ Sub-words\ that\ are\ recognized}{Number\ of\ tested\ Sub-words} \times 100$$

And the percentage of sub-words recognized will be calculated as:

$$SW = \frac{Number\ of\ Sub-words\ that\ are\ recognized}{Number\ of\ tested\ Sub-words} \times 100\ .$$

Finally, the percentage of characters recognized will be calculated as:

$$L = \frac{Number\ of\ letters\ \ that\ \ are\ recognized}{Numbers\ of\ letters\ tested} \times 100$$

7. Experiment

In order to determine the correction rate of the developed system, various criteria should exist and be tested in the GA. These include the population size, number of generations, crossing over probability, and mutation probability. These criteria were optimized in the three experiment types and the execution time was calculated for each character. The recognition problem is solved by the GA, which may yield a different solution for the same word each time, depending on the population and number of iteration. However, there is currently high similarity between the original solution and the anticipated one. We remark that if the population size is less than or equal to 100, the tested letters will be better recognized. Nevertheless, in other cases, the population size is about 100 and the executed time augments. The correction rate is 87.81 for all the tested word from Arabic and Latin Characters as show in table 4.

We also re-created the execution with a population of a different range (10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 1000) to analyze how the population affected the recognition score. The recognition rate among a population size of 10 is show in Figure 13; if the population size decreases, the recognition rate increases.

Tested words	Words with correct features	Words with incorrect features	Correction rate
109300	95975	13325	87.81

Table 4. Results of experiments in the recognition stage

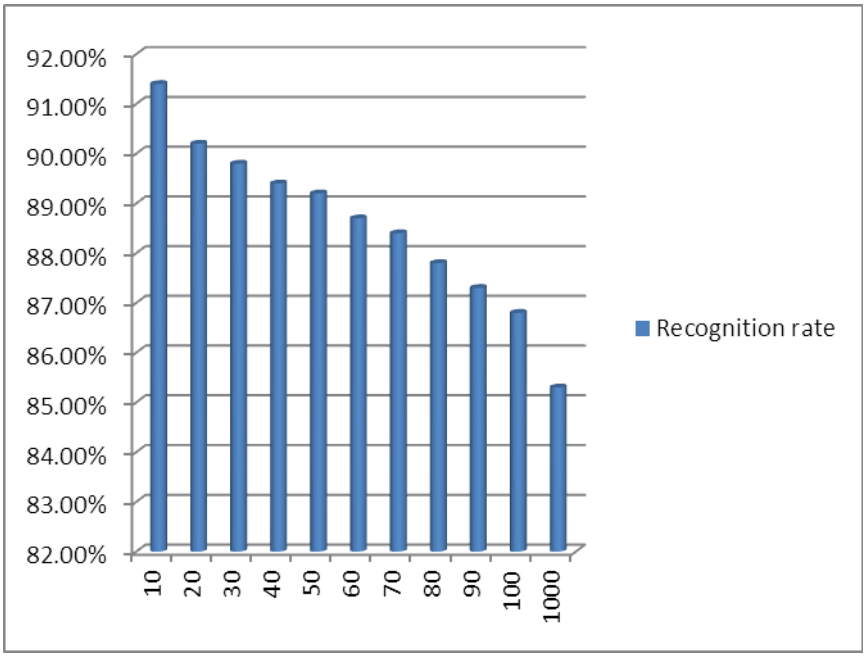


Fig. 13. Relation between population size and recognition rate.

8. References

Abandah, G.A. and Khedher, M.Z. (2004). *Printed and handwritten arabic optical character recognition-initial study.* Technical Report, University of Jordan. Amman, Jordan: August.

Abdullah, S. A. (2007). *Off-line Handwritten Arabic Characters Segmentation using Rotation Invariant Segment Feature (RISF).* Master of Science,USM.

Abuhaiba, I. (2003). A Discrete Arabic Script for Better Automatic Document Understanding. *The Arabian Journal for Science and Engineering, 28* (1B), 77-94.

Al-Hajj, R., Likforman-Sulem, L., and Mokbel, C. (2009). Combining Slanted-Frame Classifiers for Improved HMM-Based Arabic Handwriting Recognition. *IEEE transactions on pattern analysis and machine intelligence, 31. Retreived from http://www.biomedsearch.com/nih/Combining-Slanted-Frame-Classifiers-improved/19443916.html*

Aljuaid, H., Mohamad, D., Sarfraz, M. (2009). "Arabic Handwriting Recognition Using Projection Profile and Genetic Approach". The 5th International conference in Signal Image Technologies and Information based system(SITIS'09). Marrakech: IEEE CPS.

Aljuaid, H., Mohamad, D., Sarfraz,M. (2010)."Evaluation Approach of Arabic character Recognition", International Journal of Computer Vision and Image Processing (IJCVIP), IGI .

Aljuaid,H., Mohamad, D., Sarfraz,M. (2010). "A Tool To Develop Arabic Handwriting Recognition System Using Genetic Approach". Journal of Computer Science, 6(5): 490-495.

Aljuaid, H., Mohamad, D., Sarfraz,M. (2009). ICRIIS09. "Recognition of Arabic Handwriting Using Genetic Approach.

Alimi, A., and Ghorbe, O. (1995). The Analysis in an On-line Recognition System of Arabic Handwritten Characters. *Proc. 3rd Int. Conf. on Document Analysis and Recognition.* Canada, 890-893.

Alimi, A. M. (1997). An Evolutionary Neuro-Fuzzy Approach. *IEEE*, 0-8 186-7898-4/97.

Alma'adeed, S., Higgens, C., and Elliman, D. (2002). Recognition of Off-line Handwritten Arabic Words using Hidden Markov Model Approach. *Proc. 16th International Conference on Pattern Recognition, 3* , 481-484.

Alma'adeed, S., Higgens, C., and Elliman, D. (2004). Off-line Recognition of Handwritten Arabic Words using Multiple hidden Markov models. *Knowledge-Based Systems, vol. 17* , pp. 75-79.

Almuallim, H., and Yamaguchi, S. (1987). A Method of Recognition of Arabic Cursive Handwriting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 9* (5), 715-722.

Al-Sadoun, H. A. (1995). A New Structural Technique for Recognizing Printed Arabic Text. *International Journal of Pattern Recognition and Artificial Intelligence, 9* (1), 101-125.

Altuwaijri , M., and Bayoumi, M. (1995). A New Thinning Algorithm for Arabic Characters using Self-organizing Neural Network. *IEEE International Symposium on Circuits and Systems (ISCAS'95)*. Seattle, WA, 3, 1824-1827.

Amin, A. (2000). Recognition of printed arabic text based on global features and decision tree learning techniques. *Pattern Recognition Society* , 1309-1323 , doi:10.1016/S0031-3203(99)00114-4.

Amin, A. (2001). Segmentation of Printed Arabic Text. In *Advances in Pattern Recognition – ICAPR 2001* , Springer Berlin / Heidelberg , 2013/2001, 115-126 , DOI:10.1007/3-540-44732-6).

Amin, A. (2003). Recognition of Hand-printed Characters based on Structural Description and Inductive Logic Programming. *Pattern Recognition Letters, 24*, 3187-3196.

Ben Amara and Najoua Essoukri. (2003). Classification of Arabic Script using Multiple Sources of Information: State of the Art and Perspectives. *International Journal on Document Analysis and Recognition, 5*, 195-212.

Benouareth, A., Ennaji, A., and Sellami, M. (2006). HMMs with Explicit State Duration Applied to Handwritten Arabic Word Recognition. *18th Int. Conf. on Pattern Recognition (ICPR'06)*. *2*, 897-900.

Benouaretha, A., Ennajib ,A.,and Sellamia,M. (2008). Semi-continuous HMMs with explicit state duration for unconstrained Arabic word modeling and recognition. *Pattern Recognition Letters , 29* (12), 1742-1752 ,doi:10.1016/j.patrec.2008.05.008.

Cheung, A., and Bennamoun, N.W. (2001). An Arabic Optical Character Recognition System using Recognition-based Segmentation. *Pattern Recognition, 34* (2), 215-233.

Cowell, J., and Hussain, F. (2001). Thinning Arabic Characters for Feature Extraction. *IEEE Conference on Information Visualization.* 25-27 July. London, UK, 181-185.

Dehghan, M. K. F. (2001). Handwritten Farsi (Arabic) Word Recognition: A Holistic Approach using Discrete HMM. *Pattern Recognition, 34*, 1057-1065.

El-Dabi, S.S., Ramisis, R., and Kamel, A. (1990). Arabic Character Recognition System: A Statistical Approach for Recognizing Cursive Typewritten text. *Pattern Recognition, 5* (23), 485-495.

El-Hajj, R., Likforman-Sulem, L., and Mokbel, C. (2005). Arabic Handwriting Recognition using Baseline Dependent Features and Hidden Markov Modeling. *Proc. International Conference on Document Analysis and Recognition*. Seoul, Korea, 893-897.

El-Khaly, F., and Sid-Ahmed, M.A. (1990). Machine Recognition of Optically Captured Machine Printed Arabic Text. *Pattern Recognition, 23* (11), 1207-1214.

Fahmy, M. M. M., and Al Ali, S. (2001). Automatic Recognition of Handwritten Arabic Characters using their Geometrical Features. *Jurnal of Studies in Informaticas and Control, 10* (2).

Farah, M. G., Rygh, J.H, Steen, T.W, Selmer, R., Heldal, E., and Bjune, G. (2006). Patient and Health Care System Delays in the Start of Tuberculosis Treatment in Norway. *BMC Infectious Diseases, 6*, 1186/1471-2334-6-33.

Hashemi, M.R, Fatemi, O., and Safavi, R. (1995). Persian Cursive Script Recognition. *3rd international Conference on Document Analysis and Recognition (ICDAR'95)*. Montreal, Canada, *2*, 869-873.

http://ar.wikipedia. (2009). Retrieved from http://ar.wikipedia.org/wiki/

Jambi, K. (1991). *Design and Implementation of a System for Recognizing Arabic Handwritten Words with Learning Ability*. Master of Science, Illinois Institute of Technology.

Kandil, A. H., and El-Baily, A. (2004). Arabic OCR: A Centerline Independent Segmentation Technique. *International Conference on Electrical, Electronic and Computer Engineering (ICEEC'04)*. 5-7 September. 412-415.

Kherallah, M., *et al.* (2009). On-line Arabic Handwriting Recognition System Based on Visual Encoding and Genetic Algorithm. *Engineering Applications of Artificial Intelligence, 22* (1), 153-170, doi:10.1016/j.

Khorsheed, M. S. (2003). Recognising Handwritten Arabic Manuscripts using a Single Hidden Markov Model. *Pattern Recognition Letters, 24*, 2235-2242.

Khorsheed, M. S., and Clocksin, W. F. (1999). Structural Features of Cursive Arabic Script. *10th British Machine vision Conference (BMV'99)*. September. University of Nottingham, UK, *2*, 422-431.

Kim, G., and Govindaraju, V. (1997). A Lexicon Driven Approach to Handwritten Word Recognition for Real-time Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19* (4), 366-379.

Klassen, T. (2001). *Towards Neural Network Recognition of Handwritten Arabic Letters.* MASTER thesis: Dalhousie University.

Koerich, A. S. (2003). Large Vocabulary Off-line Handwriting Recognition: A Survey. *Pattern Anal., 6*, 97–121.

Lee, S.-W., and Kim, Y-J. (1995). A New Type of Recurrent Neural Network for Handwritten Character Recognition. *Proc.3rd Int. Conf. On Document Ananlysis and Recognition*. Canada, 38-41.

Madhvanath, S. G. (2001). The Role of Holistic Paradigms in Handwritten Word Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23* (2), 149-164.

Margner, V. (1992). SARAT-A System for the Recognition of Arabic Printed Text. *11th IAPR International Conference on Pattern Recognition Methodology and Systems (ICPR'92)*. 30 August – 3 September. Horgue, Netherlands, *2*, 561-564.

M¨argner, V., and El Abed,H. (2009). ICDAR 2009 Arabic Handwriting Recognition Competition. *10th International Conference on Document Analysis and Recognition* , IEEE Computer Society ,(pp. 978-0-7695-3725-2).

Maroy, M. B. (1979). Learning in Syntactic Recognition of Symbols Drawn on a Graphic Tablet. 166-182.

Mohammed, A. M. (2006). *Segmentation of Arabic Characters using Voronoi Diagrams.* Doctor of Philosophy. UKM, Bangi.

Mostafa, M. (2004). An Adaptive Algorithm for the Automatic Segmentation of Printed Arabic Text. *17th Natinal Computer Conference*. Madinah, Saudi Arabia, 437-444.

Nawaz, S. N, Sarfraz, M., Zidouri, A., and Al-Khatib. (2003). An Approach to Offline Arabic Character Recognition using Neural Networks. *10th IEEE International Conference on Electronics, Circuits and Systems (ICECS'03)*. *3*, 1328-1331. W.G.

Oliveira, L. S., and Sabourin, R. (2003). A Methodology for Feature Selection using Multiobjective Genetic Algorithms for Handwritten Digit String Recognition. *International Journal of Pattern Recognition and Artifical Intelligence, 17* (6), 903-929.

Pernkopf, F., and Bouchaffra, D. (2005). Genetic-based EM Algorithm for Learning Gaussian Mixture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27 (8)*, 1344-1348.

Plamondon, R., and Srihari, S. N. (2000). On-line and Off-line Handwriting Recognition: A Comprehensive Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22* (1), 63-84.

Sarfraz, M., Nawaz, S. N., and Al-khuraidly, A. (2003). Offline Arabic Text Recognition System. *International Conference on Geometric Modeling and Graphics (GMAG'03)*. London, England, 30-36.

Sari, T., Souici, L., and Sellami, M. (2002). Off-line Handwritten Arabic Character Segmentation and Recognition System: ACSA. *8th International Workshop on Frontiers in Handwriting Recognition (IWFHR'8)*. Niagara-on-the-lake, CA, 452-457.

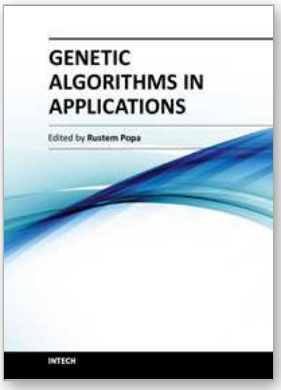
Souici-Meslati, L., and Sellami, M. (2004). A Hybrid Approach for Arabic Literal Amounts Recognition. *The Arabian Journal for Science and Engineering, 29*, 177-194.

Vinciarelli, A. B. (2004). Offline Recognition of Unconstrained Handwritten Texts using HMMs and Statistical Language Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26* (6), 709–720.

Wang, Y.-K., and Fan, K.-C. (1996). Applying Genetic Algorithms on Pattern Recognition: An Analysis and Survey.*Proceeding of ICPR'96, 2,740,* HYPERLINK "http://doi.ieeecomputersociety.org/10.1109/ICPR.1996.546921" \t "_blank" *doi.ieeecomputersociety.org/10.1109/ICPR.1996.546921*

Zeki, A. (2005). The Segmentation Problem in Arabic Character Recognition The State Of The Ar. *International Conference on Information and Communication Technologies* , 11 - 26 .

Wikipedia. (2009). Retrieved from http://www.wikipedia.org/



Genetic Algorithms in Applications

Edited by Dr. Rustem Popa

ISBN 978-953-51-0400-1
Hard cover, 328 pages
Publisher InTech
Published online 21, March, 2012
Published in print edition March, 2012

Genetic Algorithms (GAs) are one of several techniques in the family of Evolutionary Algorithms - algorithms that search for solutions to optimization problems by "evolving" better and better solutions. Genetic Algorithms have been applied in science, engineering, business and social sciences. This book consists of 16 chapters organized into five sections. The first section deals with some applications in automatic control, the second section contains several applications in scheduling of resources, and the third section introduces some applications in electrical and electronics engineering. The next section illustrates some examples of character recognition and multi-criteria classification, and the last one deals with trading systems. These evolutionary techniques may be useful to engineers and scientists in various fields of specialization, who need some optimization techniques in their work and who may be using Genetic Algorithms in their applications for the first time. These applications may be useful to many other people who are getting familiar with the subject of Genetic Algorithms.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Hanan Aljuaid (2012). Applying Genetic Algorithm in Multi Language's Characters Recognition, Genetic Algorithms in Applications, Dr. Rustem Popa (Ed.), ISBN: 978-953-51-0400-1, InTech, Available from: http://www.intechopen.com/books/genetic-algorithms-in-applications/applying-genetic-algorithm-in-multi-language-s-characters-recognition

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821