Application of Genetic Algorithms to Feature Subset Selection in a Farsi OCR

M. Soryani, and N. Rafat

***Abstract*—**Dealing with hundreds of features in character recognition systems is not unusual. This large number of features leads to the increase of computational workload of recognition process. There have been many methods which try to remove unnecessary or redundant features and reduce feature dimensionality. Besides because of the characteristics of Farsi scripts, it's not possible to apply other languages algorithms to Farsi directly. In this paper some methods for feature subset selection using genetic algorithms are applied on a Farsi optical character recognition (OCR) system. Experimental results show that application of genetic algorithms (GA) to feature subset selection in a Farsi OCR results in lower computational complexity and enhanced recognition rate.

***Keywords*—**Feature Subset Selection, Genetic Algorithms, Optical Character Recognition.

I. Introduction

ANY pattern recognition system typically consists of a section which defines and extracts useful features from a pattern and uses a classifier to classify input patterns into different classes. Depending on problems given, the number and variety of features differ according to the extracting methods and ways of representation. In many practical applications, it is not unusual to encounter problems involving hundreds of features. One can think that every feature is meaningful for at least some of discriminations. However, it has been observed in practice that, beyond a certain point, the inclusion of additional features leads to worse rather than better performance and increase the processing time [1]. Thus the selection of features, i.e. keeping suitable features and omitting unnecessary or probably redundant ones, is a crucial step in a pattern recognition system design. Feature subset selection in the context of practical applications such as character recognition presents a multi-criterion optimization function, e.g. number of features and accuracy of classification [5]. Genetic algorithms offer a particularly attractive approach for this kind of problems since they are generally quite effective for rapid global search of large, non-linear and poorly understood spaces. Moreover, genetic algorithms are very effective in solving large-scale problems [2].

Manuscript received September 30, 2006.

This work was supported in part by the Iran Supreme Council of Information and Communication Technology (SCICT).

Mohsen Soryani is with Computer Department, Iran University of Science and Technology, Tehran (e-mail: soryani@iust.ac.ir).

Najmeh Rafat is with Computer Department, Iran University of Science and Technology, Tehran (e-mail: +989123717602, n_rafat@yahoo.com).

On the other hand, characteristics of the Farsi and Arabic languages do not allow direct implementation of many algorithms used for other languages having English or Chinese like characters. Some of the features of Farsi Script are the following [3]:

- Farsi texts, unlike English are written from right to left.
- Farsi Scripts include 32 characters and each character can appear in four different shapes/forms depending on the position of the word (Beginning form BF, Middle form MF, Isolated form EF and End form EF). Table I shows some Farsi characters in their different forms.

TABLE I
DIFFERENT FORMS OF FARSI CHARACTERS [3]

IF	BF	MF	EF	IF	BF	MF	EF
أ	أ	ـا	ـا	ض	ضـ	ـضـ	ـض
ب	بـ	ـبـ	ـب	ط	طـ	ـطـ	ـط
خ	خـ	ـخـ	ـخ	غ	غـ	ـغـ	ـغ
د	د	ـد	ـد	ق	قـ	ـقـ	ـق

- The Farsi characters of a word are connected along a baseline. A baseline is the line with the highest density of black pixels. The existence of the baseline calls for different segmentation methods from those used in other unconnected scripts.
- Many Farsi characters have dots, which are positioned above or below the letter body. Dots can be single, double or triple. Different Farsi letters can have the same body and differ in the number of dots identifying them.

In this paper we applied some of the methods that have used genetic algorithms to reduce the feature dimensionality of optical character recognition systems in other languages for a Farsi OCR. Also we made some test to verify our theories.

This paper is structured as follows: Section 2 contains a general description of an OCR system. Section 3 presents a brief introduction to genetic algorithms. Details of the OCR designed for our purpose are described in section 4. Section 5 includes the experiments carried out and the results. Finally conclusions come in section 6.

II. Optical Character Recognition

Fig. 1 shows the block diagram of an OCR system. The system involves 5 stages: Preprocessing, Segmentation, Feature extraction, Classification and Postprocessing. A typical OCR system may not include some of these stages.

The recognition process starts by acquiring a digitized image. In the first stage the preprocessing of the image takes place. Recognition accuracy of the system depends on the image quality and amount of noise that exists in the image. All of the processes that improve image quality and prepare it for next stages are called preprocessing. Binarization, noise detection, smoothing and thinning are examples of these processes. Segmentation is the most important part of OCR systems especially for Farsi and Arabic languages. There are two kinds of segmentation: first, distinguishing different components of a script like paragraphs, sentences and words; and second segmentation of a word to its characters. After segmentation a set of features is required for each character. In feature extraction stage every character is assigned a feature vector to identify it. This vector is used to distinguish the character from other characters. Some of most common approaches for feature extraction are Hugh transforms, moments and characteristic loci. The step, following the segmentation and extraction of appropriate features, is the classification and recognition of characters. Many types of classifiers are applicable to the OCR problem, among which, neural classifiers and distance function achieve very good results. Finally postprocessing stage tries to improve recognition results using additional information like word dictionary.

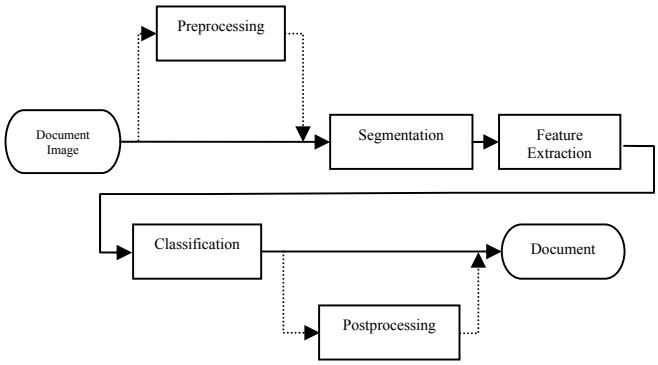

Fig. 1 Block diagram of an OCR system

III. Genetic Algorithms

Genetic algorithms are a family of computational models inspired by evolution. These algorithms encode a potential solution to a specific problem on a single chromosome and apply recombination operators to them so as to preserve critical information. GAs are often viewed as function optimizers, although the range of problems to which GAs have been applied is quite broad. The major reason for GAs popularity in various search and optimization problems is its global perspective, wide spread applicability and inherent parallelism.

GA starts with a number of solutions known as population. These solutions are represented using a string coding of fixed length. After evaluating each chromosome using a fitness function and assigning a fitness value, three different operators- selection, crossover and mutation- are applied to update the population. An iteration of these three operators is known as a generation. If a termination criterion is not satisfied this process repeats. This termination criterion can be defined as reaching a predefined time limit or number of generations or population convergence [4].

A flowchart of working principles of a simple GA is shown in Fig. 2.

As it can be seen in Fig. 2 selection is the first operator applied on a population and forms a mating pool. Crossover operator is applied next to the strings of mating pool. It picks two strings from the pool at random and exchanges some portion of the strings between them. Mutation operator changes a 1 to 0 and vice versa. Table II shows how crossover and mutation manipulate the population.

TABLE II
An Example of Crossover and Mutation Operators

Crossover			
Child1	11000	11111	Parent1
Child2	00111	00000	Parent2
Mutation			
After	11101	11111	Before

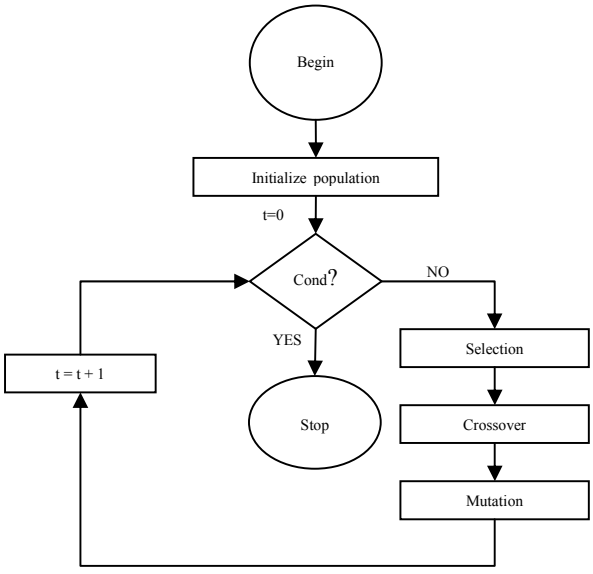

Fig. 2 A flowchart of working principles of a simple GA[4]

There are a number of factors that affect GA performance including appropriate operators, fitness function and population size. GAs are used in a variety of applications like data clustering, job scheduling and character recognition in English and Chinese languages [6] – [9].

IV. Description of Work

As we aimed to reduce feature vector of a Farsi OCR system using GA, we designed and implemented a basic OCR system to verify our tests. In this system we assume that we passed segmentation stage and use discrete characters. These characters are generated using a program so they are noise free and just need some preprocessing.

A. Preprocessing

Preprocessing stage in our OCR, includes removing character dots and placing a bounding box around character body. Then a threshold is applied and image is binarized. The resulting image goes to the next phase for extracting features.

B. Feature Extraction

We used Characteristic Loci approach to extract character features. In this approach we assign a number to each background pixel. The features are computed according to the number of intersections of lines drawn from each background pixel to right, left, up and down directions with character body. The maximum number of intersections has been limited to 3. Then, for each background pixel, a four digit number of base 4 is obtained. For example, the loci number of the point located in Fig. 3 is $(2111)_4=(149)_{10}$. This loci number is between 0 and 255 in base 10. This process is done for all background pixels. In this case, dimension of the feature vectors becomes 256. Each element of this vector represents the total number of background pixels that have loci number equal to that element. For example, 56^{th} element of this vector represents number of background pixels that have loci number equal to 56. Features are normalized by dividing by the total number of background pixels [1],[10].

Fig. 3 Calculating characteristic loci features

C. Classification

Farsi alphabets contain 32 characters. Considering a candidate for all characters with the same body, after removing dots, number of classes decreases to 18. We use a distance function to measure the distance between input character and each class center and then the character is classified into the class with the smallest distance value. Equation (1) shows the Euclidean distance that used as our classification function:

$$d = \sum_{i=1}^{n} (x_i - C_i)^2 \quad (1)$$

In which x_i refers to each feature of input character, C_i indicates each feature of class index and *n* is the total number of features.

In our system, the feature vector contains 256 features. As mentioned before, this amount of features increases the processing time whereas it might has no advantage in recognition accuracy. Therefore we are going to reduce the number of features and verify the results.

A binary mask with the length of feature vector is used for feature subset selection, in which if a bit is 1 it means that the corresponding feature is selected. Otherwise the feature is thrown away. This mask is produced by a GA. GA starts with a population of random chromosomes with the length 256 that represent binary masks. Every mask, depending on the error rate it produces in classifying the test set characters, is assigned a fitness value. Then GA operators recombine the population and create the next generation. The process continues until a stopping criterion is satisfied. Fitness function uses (2) to calculate the distance between the masked feature vector of an input character and the center of each class.

$$d = \sum_{i=1}^{n} (x_i - C_i)^2 M_i \quad (2)$$

In which x_i refers to each feature of input character, C_i indicates each feature of class index, M_i is the mask value of corresponding bit and *n* is the total number of features. Because a lot of bits in the mask have the value of zero, the feature dimension reduces, so a large amount of calculation is unnecessary that leads to recognition speed increase.

In this paper we applied two approaches to produce the mask and compare the results. In the first method, having the idea that using a different mask for each class may increase discrimination of classes, we run a GA 18 times as the number of all classes. For each class GA uses the characters which belong to the same class and generates a mask for it. In classification phase, for every input character we apply the first mask to it and calculate the distance between resulting vector and the center of the first class. Then we apply the second mask to the input character and calculate the distance between resulting vector and the center of the second class. This process continues for every class and finally the character is assigned to the class with the smallest distance value.

The second method runs just one GA using all characters in test set and creates a binary mask for whole classes. Then we calculate the distance between each input character and all class centers using (2) and assign the character to the class with the smallest distance value.

Then for each method, the recognition error rate is calculated after classification of all characters.

V. Experimental Results

To validate our proposed approach, we used a set of 1080 images of 18 Farsi alphabets typed by 12 different fonts and 5 different sizes. Fig. 4 shows all used characters in 3 fonts and sizes.

ا ب ح د ر س ص ط ع ف ق ک ل م ن و ه ی

ا ب ح د ر س ص ط ع ف ق ک ل م ن و ه ی

ا ب ح د ر س ص ط ع ف ق ك ل م ن و ه ی

Fig. 4 An example of characters in different font and sizes

These images, after removing dots and mentioned preprocessing, are divided into train and test sets randomly.

The train set is used to calculate class centers and we used the test set to estimate classification accuracy. The error rates generated by one of described methods versus the error rate of basic OCR are shown in Table III. Note that the designed OCR is not an ideal or even a very good system, but because this system have been implemented just for comparing the results of some approaches, it is not important that it produces non ideal results. After it is proved that GA is applicable in Farsi OCR, considering that GA is not problem dependent, it can be used in any OCR system.

TABLE III
CLASSIFICATION ERROR RATE

Method	No. of Features	Error rate	Classification time in test set, s
Basic OCR	256	15.18	5.15
One Binary Mask	146	11.11	3

Despite the proposed theory, generating a separate mask for each class does not produce good results. It might be because we isolated the masks and did not compare them to each other so it is likely that the best mask for one class be a good mask for other classes too. We are now working to improve this approach by using a multi objective GA that generates masks that are good for one class and bad for others.

But as it can be seen in Table III, using a binary mask produces acceptable results in decreasing feature vector size and even improving recognition accuracy and time. We run this method 10 times with different parameter settings for GA and you can see the results in Table IV.

TABLE IV
RESULTS FOR RUNNING GA FOR 10 TIMES

	Error Rate	No. of Features
1	13.51	116
2	11.29	145
3	13.51	114
4	11.48	138
5	12.04	112
6	11.29	140
7	11.11	146
8	12.03	132
9	12.77	144
10	11.29	116

VI. CONCLUSION

We present the results of two methods for selecting subset of features of a Farsi OCR system using genetic algorithms. These results show that one of these methods select features effectively, reduces the computational complexity and increases recognition rate in comparison with basic OCR. As GA is a problem independent approach, this method is easily applicable in any Farsi OCR system.

Taking advantage of GA in the context of OCR systems is not limited in this subject. This algorithm can be used in pre processing, postprocessing or designing optimized weights in classification.

REFERENCES

[1] Oliveira, L. S., Benahmed, N., Sabourin, R., Bortolozzi, F., Suen, C. Y., "Feature Subset Selection Using Genetic Algorithms for Handwritten Digit Recognition" Proc. XIV Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'01), P.362, 2001.

[2] Yang, J., Honavar, V., "Feature Subset Selection Using a Genetic Algorithm," Proc. IEEE Intelligent Systems, vol. 13, no. 2, pp. 44-49, 1998.

[3] Sarfraz, M., Nawaz, S., N., Al-Khuraidly A., "Offline Arabic Text Recognition System" Proc. 2003 International Conference on Geometric Modeling and Graphics (GMAG'03), 2003.

[4] Deb, K., "Genetic Algorithm in Search and Optimization: the Technique and Applications" Proc. International Workshop on Soft Computing and Intelligent Systems, pp. 58-87, Calcutta, India, 1998.

[5] Kudo M, Sklansky J. , "Comparison of Algorithms that Select Features for Pattern Classifiers" Pattern Recognition, Vol.33, pp.25–41, 2000.

[6] Kim, G., Kim, S., "Feature Selection Using Genetic Algorithms for Handwritten Character Recognition" Proc. Seventh International Workshop on Frontiers in Handwritten Recognition, Amsterdam, 2000.

[7] Sural, S., Das, P. K., "A Genetic Algorithm for Feature Selection in a Neuro-Fuzzy OCR System" Proc. Sixth International Conference on Document Analysis and Recognition (ICDAR'01), P.0987, 2001.

[8] Morita, M., Sabourin, R., Bortolozzi, F., Suen, C. Y., "Unsupervised Feature Selection Using Multi-Objective Genetic Algorithms for Handwritten Word Recognition " Proc. Seventh International Conference on Document Analysis and Recognition (ICDAR'03), Vol.2, P.666, 2003.

[9] Shi, D., Shu, W., Liu, H., "Feature Selection for Handwritten Chinese Character Recognition Based on Genetic Algorithms" Proc. IEEE Int. Conference on Systems, Man, and Cybernetics, vol. 5, pp. 4201-6, 1998.

[10] Ebrahimi, A., Kabir, E., "A Two Step Method for the Recognition of Printed Subwords", Iranian Journal of Electrical and Computer Engineering, Vol.2, No.2, pp.57-62, 2005 (in Farsi).