SELF EVOLVING CHARACTER RECOGNITION USING GENETIC OPERATORS

Shashank Mathur

***Abstract**: In this paper, a novel approach for character recognition has been presented with the help of genetic operators which have evolved from biological genetics and help us to achieve highly accurate results. A genetic algorithm approach has been described in which the biological haploid chromosomes have been implemented using a single row bit pattern of 315 values which have been operated upon by various genetic operators. A set of characters are taken as an initial population from which various new generations of characters are generated with the help of selection, crossover and mutation. Variations of population of characters are evolved from which the fittest solution is found by subjecting the various populations to a new fitness function developed. The methodology works and reduces the dissimilarity coefficient found by the fitness function between the character to be recognized and members of the populations and on reaching threshold limit of the error found from dissimilarity, it recognizes the character. As the new population is being generated from the older population, traits are passed on from one generation to another. We present a methodology with the help of which we are able to achieve highly efficient character recognition.*

***Keywords**: Genetic operators, character recognition, genetics, genetic algorithm.*

***ACM Classification Keywords**: I.2 Artificial Intelligence, I.4 Image processing and computer vision, I.5 Pattern Recognition.*

***Conference**: The paper is selected from Seventh International Conference on Information Research and Applications – i.Tech 2009, Varna, Bulgaria, June-July 2009*

Introduction

Over the years, several methods have been proposed for offline character recognition which is the translation of characters provided as images into editable textual characters. Character recognition has been previously implemented for printed Tamil text [I], a traditional south Indian language. Some previously published works in character recognition such as mathematical expression recognition [II] and intelligent form based character recognition system [III] have been successful and have laid the stepping stone for further research in the area such as application of genetic operators to provide better results. Genetic operators derive their existence from Genetics, a branch of biology dealing with the heredity and variations in living organisms. The methodology presented here projects that new generations of solutions possess traits of their predecessors, thus implementing inheritance, a significant concept of genetics proposed by Johann Gregor Mendel [IV]. Genetic algorithms have been developed on these evolutionary concepts and applied in various fields such as knapsack problem [V], structural optimizations [VI],Assembly planning [VII] and Design of Large-Scale Reverse Logistic Networks in Europe's Automotive Industry [VIII].

With the establishment of evolutionary concepts and development of genetic programming, and works of various researchers such as Price who noticed that a covariance relationship exists between the number of successful offspring that an individual produces and the frequency of any given gene in that individual [IX], we are able to achieve a highly efficient evolutionary character recognition methodology with the help of genetic operators. In this paper, haploid chromosomes which have been implemented in the form of a bit pattern of dimensions 1x315 values, which can be operated upon by genetic operators like selection, crossover and mutation. An initial population is taken from which new generations of offspring's are obtained till the error is reduced till a threshold limit and this method has been instrumental in the process of character recognition. A unique fitness function has been designed with the help of which the fittest set of offspring's are selected and ultimately lead us to the desired solution thus resulting in selection as a consequence of competition [X]. It has been previously seen that two parents result in two more offspring's, as done in evolutionary timetabling using biased genetic operators [XI].

In some previous works, Artificial Neural Networks have been used for optimizing both the architecture and the connection weights of multilayer feed forward neural networks [XII] and Arabic character recognition [XIII], but here the recognition process is carried out solely with the help of genetic operators making it a self evolving character recognition procedure. The methodology presented is significant in abridging the concepts of genetics and artificial intelligence, thereby introducing the various biological processes in programming and yielding efficient results. The various published works based on genetic algorithms like cryptanalysis method based on Genetic Algorithm and Tabu Search to break a Mono-Alphabetic Substitution Cipher in Adhoc networks[XIV] and the weight constrained shortest path problem[XV] have aided the development of the methodology presented his to ensure that results obtained from the evolutionary based algorithm provide high degree of efficiency.

The paper also presents the process of adaptation, an emergent property achieved by the Darwinian process of selection on heritable variation [XVI]. The new generations that are produced are adapted to the best solutions of the previous generations thereby providing us with a solution that is closer to the expected output as the newer generations are produced. The more the number of generations, more accurate the result is obtained as the error keeps reducing from one generation to another. The fittest solution survives at last which is provided as output. The methodology presented keeps knowledge about the fitness about all the individuals of the population of a particular generation which is lost on the development of a new generation. It is made sure that only a set of fit solutions are selected to generate a new generation so that an least error output is obtained in a optimum number of generations. Since only a set of fit values are considered to develop a new generation we get offspring's that have better fitness values as compared to the parents.

The purpose of this paper is to present new approach for character recognition based on the evolutionary theory with the application of genetic operators like selection, crossover, and mutation on sets of characters taken as an initial population. Bit patterns are used to represent haploid chromosomes which are operated upon by genetic operators and then used to generate new populations. With each new generation that is obtained, there is a decrease in the error between the pattern to be matched and the fittest offspring that was obtained in the last produced generation of solutions. The process of self evolution of recognized result from a population of patterns provides us with highly efficient results of character recognition.

Methodology

The self evolving character recognition process is carried out with the help of bit patterns of size 1x315 values which have been obtained from monochromatic images of characters of size 21x15. A set of characters for each alphabet are converted into the form of single row bit patterns to generate an initial population from which new generations of offspring's are generated till a threshold limit of error between the character to be recognized and the fittest solution of a generation is obtained. The process of creation of new generations is suspended once this threshold limit is obtained. The genetic operators like selection, crossover and mutation have been described along with a unique fitness function with the help of which only a set of fit offspring's are selected and the others are discarded, thus implementing survival of the fittest.

A. Procedure

The process of character recognition starts by reading an input monochromatic image, of any size which is resized to an image of 21x15 pixels and then converted into a single row bit pattern of dimension 1x315 which is then matched with the generated populations at various stages. A set of 130 characters, 5 characters each for all the 26 lower case alphabetical characters are considered as an initial population. The fitness function selected the set of fittest values to generate a new population which is then operated upon by the genetic operators to yield new generation on which again the fitness function is applied to generate fit solutions, this process is carried out till a threshold limit for error between the input and fittest solution is obtained. In the process of character recognition it is essential that if the error is less than the threshold limit, it can be recognized with the help of lesser number of computations, thus making the algorithm provide us a result in an optimum number of computations, thereby making it cost efficient.

B. Fitness function

This is a function with the help of which we are able to learn about the fitness of a solution, and whether that solution should be kept to generate a new population or not. With the help of this fitness function a set of offspring's are selected to generate a new generation of offspring's while the other values are discarded. We have haploid chromosomes in the form of single row bit pattern, thus in order to check whether a solution that has been generated should be kept for generating a new generation or should be discarded is based on the dissimilarity obtained between the offspring and the input. We now see the working of the fitness function, suppose we have two bit patterns, A, a single row bit pattern from the current population and B, single row bit pattern of input that is supposed to be recognized, then C is the obtained as the result of bitwise XOR of A and B, from which we obtain the dissimilarity coefficient, which is the number of '1' bits in the XOR result, which show us the number of bits for which the input and offspring differ.

A= 0 0 0 0 1 1 1 1 0 0 1 1 0 0

B= 0 1 0 0 1 0 1 1 0 0 0 1 0 0

C=A XOR B

C=0 1 0 0 0 1 0 0 0 0 1 0 0 0

Dissimilarity Coefficient = 3

As we see the result of the XOR operation in the above example, this is carried out for the entire population taken one at a time. The dissimilarity coefficients are saved and their mean is calculated. The members of the population with a dissimilarity coefficient of less than the mean value are considered fit to generate the next generation while the others are discarded.

C. Selection

This is one of the genetic operators that are useful in selecting a set of population with the help of which a new population is generated. As explained above, dissimilarity coefficients are calculated for the entire population out of which only the members having a coefficient of more than mean value are selected and the others are not a part of the new population that is generated.

D. Crossover

A genetic operator, which introduces inheritance of traits and adaptation in the new generations of offspring's. In crossover, two chromosomes overlap to provide two new offspring's which inherit the trait from their parents and are adapted towards a better fit solution. This biological crossover is implemented here with the help of two single row bit patterns which overlap at randomly varying lengths of 1 to 310 to generate new offspring's that have lower results of dissimilarity coefficients. Suppose we have two single row bit patterns, A and B of dimensions 1x315 each. Then C and D are two offspring's that are developed when the crossover takes place at a length of 5. The greater the extent of crossover, greater are the chances of getting a offspring which has a lower dissimilarity coefficient. The process of the crossover is shown in figure 1-3.

A= 0 0 0 0 1 1 … 1 1 0 1 1 1 1 0

B= 0 1 1 0 1 0 … 1 1 0 0 0 1 0 0

C= 0 0 0 0 1 0 … 1 1 0 0 0 1 0 0

D=0 1 1 0 1 1 … 1 1 0 1 1 1 1 0

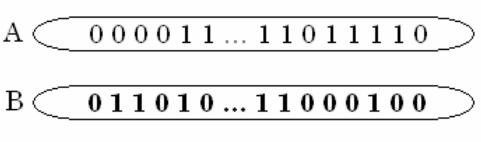

Fig. 1 Initial single row bit patterns.

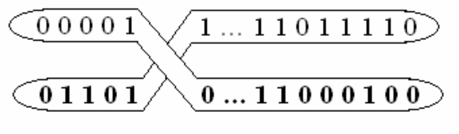

Fig. 2 Crossover at length of 5 bits.

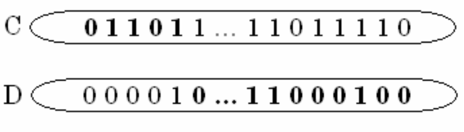

Fig. 3 Offspring's generated after crossover

E. Mutation

This is the last genetic operator used. In case we have a very low dissimilarity coefficient of the order of less than 30 bits, which would be equivalent to an error of less than 10%, we do not populate new generations, but induce mutate the selected single row bit pattern at various lengths to generate a solution such that it is able to match our threshold limit for recognition of the character. Suppose we two single row bit patterns, A and B of dimensions 1x315 each, where A Is the input to be recognized and B is the pattern with less that 10% error then mutation is carried out. The process of mutation is shown below.

A= 1 0 1 0 0 0 0 0 1 1 1

B= 1 0 1 0 1 0 1 0 1 1 1

Dissimilarity Coefficient = 2

Few mutated results:

M1= 1 0 0 0 1 0 1 0 1 1 1

M2= 1 0 1 0 0 0 1 0 1 1 1

M3= 1 0 1 1 1 0 1 0 1 1 1

M4= 1 0 1 0 1 0 1 0 0 1 1

With the application of the induced mutations, we are able to generate a new population in which each offspring differs from the parent of the previous generation by only one bit. Thus we are able to achieve a solution in lesser number of computations thereby decreasing the overall complexity of the methodology presented.

F. Algorithm

The methodology presented works according to the following algorithm shown in figure 4. We can observe that the working of the algorithm starts with the generation of an initial population. Each member of this initial population and the input image which is supposed to be recognized is applied to the fitness function from where we get the dissimilarity coefficients for each member of the population. If we get an error of less than 10%, we check if the error is less than 5%, if so then the member with the least dissimilarity coefficient is selected as the recognized. In case error is between 5% and 10% , such that character input which is quite similar to any of characters in the initial population we carry out mutation and generate a new population in which each member differs from the input by only one bit. Incase, we achieve an error of greater than 10%, we generate a new population by crossover of each member with the other members of the population at varying lengths. The new population thus generated is replaced with the initial population and we get iterations till the error reduces to 10%. The algorithm thus is capable of recognizing the characters with high efficiency.

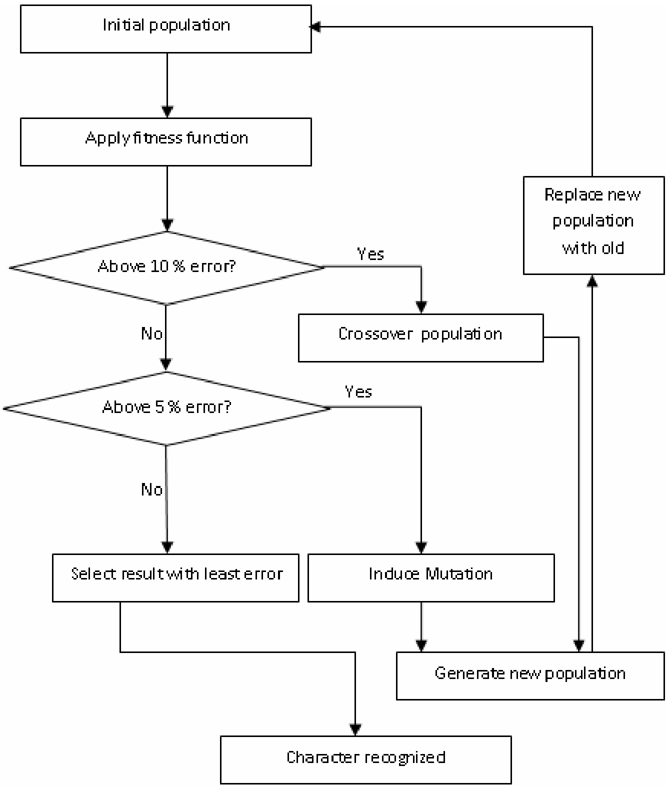

Fig. 4 Flowchart.

As the new generations are produced, the number of members of the population starts increasing. The number of individuals of a new population that is generated is found to be proportional to the number of individuals selected to generate it. The character recognition process carried out here has been implemented using a small initial population of 130 members which grows to form populations of hundreds and thousands in successive generations and leadings us to find an optimum solution.

Results

The self evolving character recognition using genetic operators was tested for 10 character inputs for each alphabet, thus a total of 260 samples and observed that a high efficiency of recognition is obtained as shown in the graph in figure 5. An overall efficiency of 79.23% was obtained for the character recognition process during the testing process. The methodology presented here was implemented on a Pentium 4 (3.4 GHz), 2GB RAM and MATLAB 7.0. We can note that the number of generations that are being created increase with the increase in dissimilarity coefficient. A character with an error of less than 5% is recognized with the help of the member of population having the least dissimilarity coefficient. Along with the genetic operators, the fitness function has been instrumental in providing a good estimate of the dissimilarity coefficient with the help of which unwanted members of the population can be discarded. Thus successfully we are able to implement concepts of genetics, a discipline of biology in character recognition process yielding high rate of efficiency.

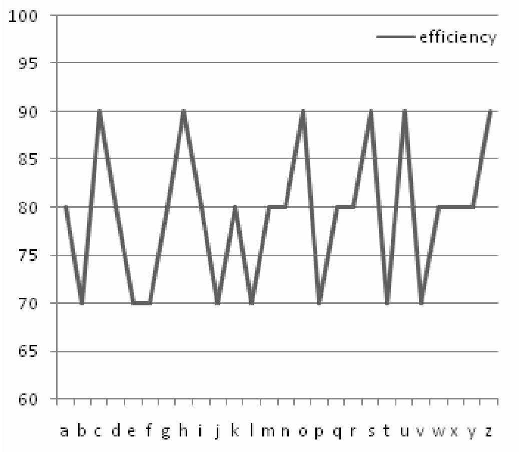

Fig. 5 Graph obtained for testing of character recognition method

Conclusion

The paper has presented a new method of character recognition using a unique and robust methodology by the application of genetic operators. The paper is significant in abridging the concepts of genetics and artificial intelligence and thus providing a stepping stone for further research in this area. The highly efficient character recognition results reflect the accuracy of the implementation of the genetic operators. The innovative fitness function used here provides us with a dissimilarity coefficient which restrains the entry of unwanted members of the population which would only increase the computational cost of the methodology presented. Character recognition task is indeed a tough task which can be carried out by the methodology presented in a biologically inspired self evolving manner implementing all the laws of genetics. The paper has provided a link of evolutionary sciences in the field of computer science.

Acknowledgement

The paper is partially financed by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA and the Consortium FOI Bulgaria (www.ithea.org, www.foibg.com).

Bibliography

[I.] Abnikant Singh, Markandey Singh, Ritwaj Ratan, Sarvesh Kumar, "Optical Character Recognition for printed Tamil text using Unicode", Journal of Zhejiang University, pp. 1297-1305, 2005.

[II.] Erik G. Miller, Paul A. Viola, "Ambiguity and Constraint in Mathematical Expression Recognition", American Association for Artificial Intelligence, 1998.

[III.] Dipti Deodhare, NNR Ranga Suri R. Amit, "Preprocessing and Image Enhancement Algorithms for a Form-based Intelligent Character Recognition System", International Journal of Computer Science and Applications Vol. II, No. II, pp.131-144, 2005.

[IV.] Weiling F,"Historical study: Johann Gregor Mendel 1822-1884", American Journal of Medical Genetics, Vol. 40, No.1, pp.1-25, 1991.

[V.] P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, "A monte carlo study of genetic algorithm initial population generation methods" , Proceedings of the Winter Simulation Conference, pp. 543-547, 1999.

[VI.] Alonso J. Juvinao Carbono, Ivan F. M. Menezes, Luiz Fernando Martha,"Mooring Pattern Optimization using Genetic Algorithms" ,6th World Congresses of Structural and Multidisciplinary Optimization, Brazil, 2005.

[VII.] Shana Shiang-Fong Smith (Shiang-Fong Chen), Yong-Jin Liu,"The Application of Multi-Level Genetic Algorithms in Assembly Planning", Journal of Industrial Technology, Volume 17, Number 4, pp.1-9. 2001.

[VIII.] Ralf Schleiffer, Jens Wollenweber, Hans-Juergen Sebastian, Florian Golm, Natasha Kapoustina, “Application of Genetic Algorithms for the Design of Large-Scale Reverse Logistic Networks in Europe’s Automotive Industry” Proceedings of the 37th Hawaii International Conference on System Sciences,2004.

[IX.] Jeffrey K. Bassett, Mitchell A. Potter, Kenneth A. De Jong, “Applying Price’s Equation to Survival Selection” Genetic and Evolutionary Computation Conference, USA, pp. 1371-1378, 2005.

[X.] D. B. Fogel1 and J. W. Atmarz ”Comparing Genetic Operators with Gaussian Mutations in Simulated Evolutionary Processes Using Linear Systems”,Biological Cybernetics., pp. 111-114, 1990.

[XI.] Daniel Danciu,”Evolutionary Timetabling Using Biased Genetic Operators” Journal of Computing and Information Technology, pp. 193-199, 2003.

[XII.] Morgan Kaufmann, “Genetic Programming of Minimal Neural Nets Using Occam’s Razor”, Proceedings of 5th international conference on Genetic Algorithms, pp. 342-349, 1993.

[XIII.] Ahmad M. Sarhan,Omar I. Al Helalat, “Arabic character recognition using artificial neural networks and statistical analysis”, Proceedings of workd academy of science, engineering and technology, Vol. 21, pp.32-36, 2007

[XIV.] K. Verma, Mayank Dave, R. C. Joshi, “Genetic Algorithm and Tabu Search Attack on the Mono-Alphabetic Subsitution Cipher in Adhoc Networks”, Journal of Computer Science, Vol.3, No.3, pp. 134-137, 2007.

[XV.] Jeffrey K. Bassett, Mitchell A. Potter, Kenneth A. De Jong, “Applying Price’s Equation to Survival Selection” Genetic and Evolutionary Computation Conference, USA, pp. 1371-1378, 2005.

[XVI.] Anthony V. Sebald and Lawrence J. Fogel,” Emergent phenomena in genetic programming”, in Proceedings of the Third Annual Conference on Evolutionary Programming, _ pp. 233-241,1994.

Authors' Information

Shashank Narain Mathur - Bachelors of Technology (Computer Science Engineering), Amity School of Engineering and Technology affiliated to Guru Gobind Singh Indraprastha University, New Delhi, India.
e-mail: shashanknarainmathur@yahoo.com