On-line Handwritten English Character Recognition Using Genetic Algorithm

Shilpa Jumanal[#1], Ganga Holi[#2]

[#1]M.Tech Student, Department of ISE, PESIT, Bangalore ,India
[#2]Associate Professor , Department of ISE, PESIT, Bangalore ,India

Abstract **- Tremendous advancement in technology has produced varieties of electronics devices such as PDAs, handheld computers where non-keyboard based method of data entry are receiving more attention in the research communities and commercial sector. The most promising options are pen-based and voice-based inputs. The increase in usage of handheld devices which accepts handwritten input has created a growing demand for algorithm that can efficiently analyse and retrieve handwritten data. This paper proposed a methodology to recognize handwritten character written on the digitizing tablet. The proposed method is based on extraction of different spatial and temporal features from strokes of the character and recognition is done by using genetic algorithm algorithm as a tool to find an optimal subset of the stroke features. The proposed system is experimented on data set consisting of 5200 samples collected from various persons for English letters and recognition rate achieved is 83.1%.**

Keywords**— On-line recognition, Genetic algorithm, spatial and temporal features, digital tablet.**

I. INTRODUCTION

There are many organizations which maintain the large number of documents for their day to day life. Processing of such documents recorded on paper or scanned form needs an efficient tool in order to extract intended information as a human would [1] and is the essential task in the organizations for varied application.

Human can see and read what is written or displayed either in natural handwriting or in printed form. A handwritten character is constructed using collection of strokes. The representation of the stroke depends on the structure or shape of the stroke in which a stroke is a string of shape features [2] [4] [5]. Using this string representation, an unknown stroke is identified by comparing it with a collection of strokes using a flexible string matching method. A full character is recognized by identifying all the strokes of the character [2] [4] [5].

Online handwritten character recognition consists of recognizing a character as it is written using an electronic stylus or a pen on a tablet. The present work describes a system for online recognition of handwritten English characters. The devices, which generate handwritten documents with online or dynamic information, needs an efficient algorithms for processing and retrieving handwritten data. So in this paper we have developed an efficient algorithm to recognize online handwritten English characters.

The goal of the paper is to develop a system which accepts handwritten character by pen tablet input device and recognizes it based on different spatial and temporal features extracted from the strokes of the character using genetic algorithm. Because a larger set of stroke features is assumed to be available in order to identify the corresponding stroke of the initial handwriting. According to this variability of combination possibilities, it is good to use the genetic algorithm as a tool to find an optimal subset of the stroke features.

The organization of the rest of the paper is as follows: In Section 2, we refer to recent related work. In Section 3, we describe in detail the proposed algorithm for online handwritten English character recognition. In Section 4, Experimental results are discussed and in Sections 5 conclusion and future work is discussed.

II. RELATED WORK

With the advent of handwriting recognition technology since a few decades applications are challenging. For example, OCR, the major application in document scanners, and is also used in many applications such as postal processing, script recognition, banking, security such as signature verification and language identification [9].

In handwritten character recognition, several different studies have shown that off-line handwriting recognition offers less classification rate compared to on-line [2]. And on-line data

offers significant reduction in memory and therefore space complexity is less because only the co-ordinates of the sampled pen point positions and possibly some other features needed to be stored [2]. Furthermore, on-line handwriting data over offline is the dynamic information on writing process whereas offline data is just static images of handwriting because image pixels do not contain any information on writing direction or writing order of strokes or state of the pen i.e. pen up or pen down state [2].

In on-line handwriting the user uses a digital pen or a digital form to write a character on a tablet device which immediately transforms handwriting into a digital representation that can be reused later without having any risk of degradation usually associated with ancient handwriting [2] [3] [4] [5].

Handwriting recognition is the ability of a computer to receive and interpret intelligible handwritten input from sources such as documents, photographs, touch sensitive screens and other devices. The image of the handwritten or printed text may be sensed "off line" from a piece of paper by optical scanning (optical character recognition). Alternatively, the motion of the pen tip may be sensed "on line", for example by a pen-based computer screen surface.

Handwriting recognition can be categories into two types based on nature of data [2] [3] [5]:

1. Off-line Handwriting Recognition and
2. On-line Handwriting Recognition.

Off-line Handwriting Recognition, user writes on paper which is later digitized by a scanner. The data is presented to the system as an image, requiring a segmentation of the writing from the image background before recognition can be done.

On-line Handwriting Recognition[2] [3] [4] [5] requires that the user write on a digitizing tablet using a special stylus, so that the user's written strokes are captured as they are being formed by sampling the pen's (x, y) coordinates at evenly spaced time intervals. The use of a pressure-sensitive switch on the pen tip indicates pen-up/pen-down status and disambiguates stroke segmentations.

Genetic algorithms, first proposed by Holland in 1962, are a class of computational models that mimic natural evolution to solve problems in a wide variety of domain. Genetic algorithms are particularly suitable for solving complex optimization problems and for applications that require adaptive problem solving strategies.

In the computer science field of artificial intelligence, a genetic algorithm (GA) is a search heuristic that mimics the process of natural evolution. This heuristic (also sometimes called a Meta heuristic) is routinely used to generate useful solutions to optimization and search problems.

The algorithm operates through a simple cycle:

1. Creation of a population of strings.
2. Evaluation of each string.
3. Selection of the best strings.
4. Genetic manipulation to create a new population of strings.

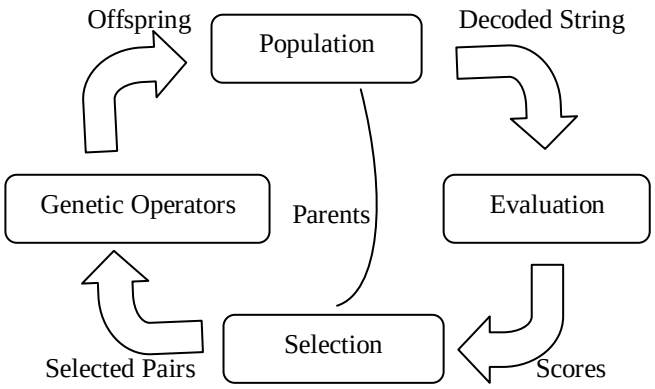


Figure 1: Reproduction Cycle

This algorithm encodes a potential solution to a specific problem on a single chromosome and applies recombination operators to them so as to preserve critical information [21]. 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 [13]. GA initially 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 such as selection, crossover and mutation are used to update the population. A repetition 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 [14] [15].

The Basic Genetic Algorithm:

- Start with a large "population" of randomly generated "feasible solutions" to a problem
- Repeatedly do the following:
  - Evaluate each of the feasible solutions
  - Keep a subset of these solutions (the "best" ones)
  - Use these solutions to make generation of new population
- Quit when you have a satisfactory solution (or you run out of time)

III. PROPOSED SYSTEM

A. SYSTEM DESIGN

This system consists of 4 modules.

1. **Data Collection**: It is first phase which collects user's input data by collecting sequence of co-ordinates points of the moving pen on a digitizer.

2. **Preprocessing and Feature Extraction**: During preprocessing, the individual strokes are resampled to make sampled points equidistant. And features are extracted from user's input pattern.

3. **Training:** During this phase prepare the training data (reference file) in a way suitable for future use by the recognition and sends the data to training dataset.

4. **Recognition**: In this phase it compares the user's input pattern after preprocessing and feature extraction with those in the training dataset.

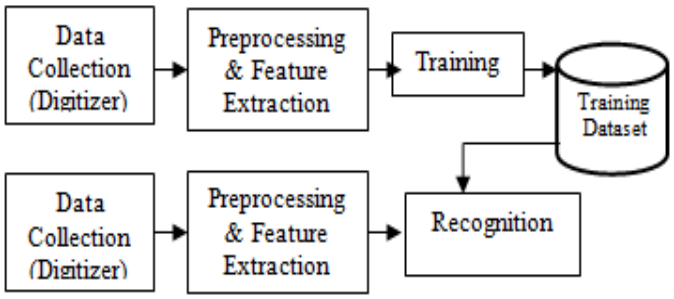


Figure 2: High Level Design of the System

B. DATASET PREPARATION & PRE-PROCESSING

Dataset is created by collecting samples from 5 different persons belongs to different age groups. 100 samples for each English characters including both uppercase and lowercase letters from various persons are collected. Total of 2600 for uppercase and 2600 for lowercase characters altogether 5200 samples are collected. Out of 5200 samples 3120 samples are taken training and remaining 2080 samples are taken for testing.

During pre-processing, the individual strokes are resampled to make the sampled points equidistant. This helps to reduce the variations in characters due to different writing speeds of the person and to avoid anomalous cases such as having a large number of samples at the same position when the user holds the pen down at a point.

The resampling distance is set to 10 pixels. And the distance between two sampled points is calculated by using Euclidian distance formula.

Figure 3: Different isolated handwritten characters for English letter 'A'.

C. FEATURE EXTRACTION TECHNIQUES

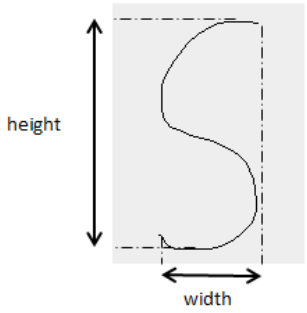
Feature extraction is a major part of any recognition system. The aim of feature extraction is to describe the pattern by means of minimum number of features that are effective in discriminating pattern classes [12].

1. **Average Stroke Length (ASL):** During preprocessing individual strokes are resampled so that sample points are placed at equidistant. Hence, the number of sample points in a stroke gives the length of the respective stroke. Therefore ASL is defined as the average length of the individual strokes in the pattern.

$$ASL = 1/n \sum_{i=0}^{n} length(strokei)$$

Where n is the number of strokes in the pattern.

2. **Width Strength:** This feature finds the strength of the horizontal line component in the pattern.



Width (stroke) = X_{min}- X_{max}
Where X_{min} - is the minimum of 'x' co-ordinate of the stroke i and
X_{max} - is the maximum of 'x' co-ordinate of the stroke i.

3. **Height Strength:** This feature finds the strength of the vertical line component in the pattern.

   Height (stroke$_i$) = $Y_{min} - Y_{max}$
   Where Y_{min} - is the minimum of 'y' co-ordinate of the stroke i and
   Y_{max} - is the maximum of 'y' co-ordinate of the stroke i.

4. **Confidence**: It computes a confidence measure for a stroke being a Shirorekha. Each stroke in the pattern is tested for three different properties of a Shirorekha; Shirorekhas are larger width of a character, almost occur at the top of the character, and are horizontal. Hence, the confidence (C) of a stroke (s) is computed as:

$$C(S) = \frac{width(S)}{width(pattern)} * \frac{\bar{Y}(S)}{height(pattern)} * \left(1 - \frac{height(S)}{width(S)}\right)$$

   Where width(S) refers to the length along the x-axis (horizontal), height(s) is the length of a stroke along the y-axis (vertical), and $\bar{Y}(S)$ is the average of the y-coordinates of the stroke points. Note that $0<C(S)<1$ for strokes with height < width.

5. **Stroke Density along x-axis**: This is the number of strokes per unit length along the x-axis of the pattern.

$$\text{Stoke Density} = \frac{n}{width(pattern)}$$

   Where n is the number of strokes in the character.

6. **Stroke Density along y-axis**: This is the number of strokes per unit length along the y-axis of the pattern.

$$\text{Stoke Density} = \frac{n}{height(pattern)}$$

7. **Aspect Ratio**: This is the ratio of the width to the height of a pattern.

$$\text{Aspect Ratio (pattern)} = \frac{width(patten)}{height(pattern)}$$

8. **Variance of Stroke Length:** This is the variance in sample lengths of individual strokes within a pattern. The value is of variance of stroke length is a non-negative integer.

D. GENETIC ALGORITHM

The genetic algorithms have the following properties:

1. Chromosome Representation

Unlike the traditional genetic algorithms (GAs) that adopt binary bit strings to encode a chromosome, a more direct representation is used in our model. The chromosome is presented by a set of feature vector which forms a character is shown in the figure 4.

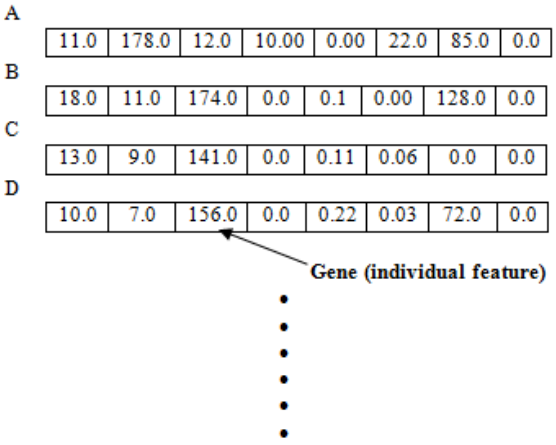


Figure 4: Chromosome Representation

2. Initialization of the Population

The initial population is randomly generated, with the number of chromosomes is set at 2600 for both uppercase and lowercase English letters. The size of each chromosome is 8.

3. Selection

Selection in genetic algorithms aims at giving a higher probability for reproduction to better individuals in a population so that their favorable characteristics can be inherited by even fitter offspring [16] [17]. This is called the principle of survival of the fittest. That is to keep the best parent (based on the fitness value) from the current population as one of the candidates in the next generation. In our experiment we have used roulette wheel selection method for selecting fittest individual from current population. The individuals selected will then go through crossover and mutation.

4. Crossover

The essence of any crossover operator is to exchange the components of two parents to form new offspring. In our experiments we have used one point crossover by cutting the chromosomes at a randomly chosen position and then swapping the segments between the two parents [23] [24]. By this operation we have increased the population from 2600 to 3900.

5. Mutation

Mutation in evolutionary algorithms is another search operator. Its main function is to introduce new genetic material and maintain a certain level of diversity in a population since crossover does not introduce any new genetic material [23] [24].

6. Fitness Evaluation

In this model the value of the fitness evaluation is the percentage of correspondence between the feature vector that establishes a chromosome and each part of the script to be addressed.

$$f_{CHAR} = \frac{N_{FC\_C}}{N_{FC}}$$

A fitness value of the character is given by ration of number of features which have a correspondence in the vector of the features composing the script and number of features forming the required character.

E. CLASSIFIER

The minimum distance classifier is used to recognize the character by finding distance between two feature vectors which forms the character input pattern using the Euclidean distance metric.

$$\text{Euclidian Distance} = \sqrt{\sum_{i=1}^{N} (Di - Qi)^2}$$

Where N is the number of features in the vector, D_i is the i^{th} feature of the training (Database) sample, and Q_i is the i^{th} feature of the testing (Query) sample.

IV. EXPERIMENTS AND RESULTS

In order to classify the on-line handwritten character and evaluate the performance of the technique, we have carried out the experiment. All experiments was performed on a Intel® core 2 duo CPU T6400 @ 2GHz with 3 GB RAM under 64 bit windows 7 Ultimate operating system. The Table 1 shows the recognition rate of the English characters from A to Z, figure 4 and figure 5 shows the result of experiment for characters A to M and N to Z respectively.

Datasets

Experiment is conducted using a combination of two different sets of data:

(i) A set of 3120 samples of English characters are taken as training samples for the recognition. Collected by 5 different persons.

(ii) A set of 2080 samples are taken for testing.

Class	Recog. Rate	Class	Recog. Rate	Class	Recog. Rate
A	100%	J	98%	S	64%
B	98%	K	54%	T	98%
C	98%	L	64%	U	98%
D	98%	M	98%	V	74%
E	54%	N	64%	W	54%
F	64%	O	100%	X	64%
G	100%	P	54%	Y	64%
H	62%	Q	98%	Z	98%
I	98%	R	98%		

Table 1: Recognition Rate of the English Uppercase letters

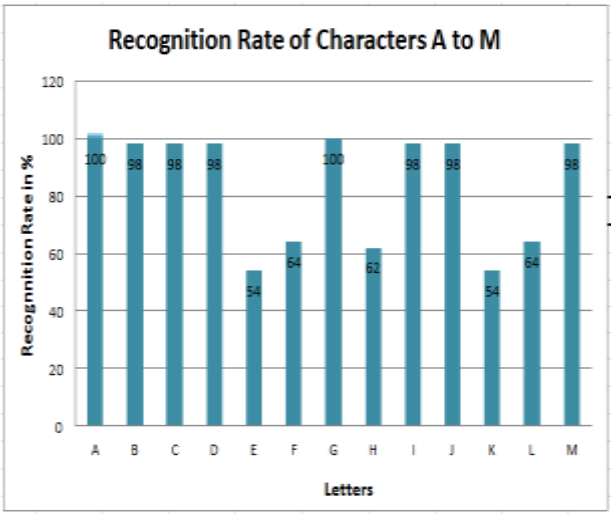


Figure 5: Results of the Experiment

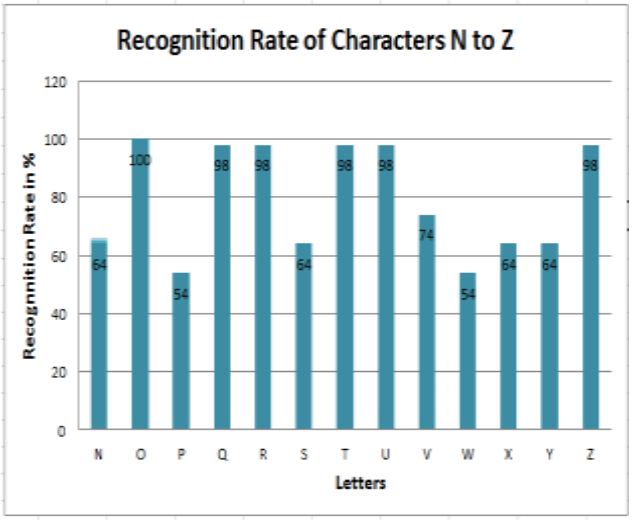


Figure 6: Results of the Experiment

V. CONCLUSION AND FUTURE WORK

In this paper an effort is made towards recognition of on-line handwritten English characters based on spatial and temporal features using genetic algorithm. Overall recognition accuracy achieved is 81.3% for English characters. This algorithm stores the feature vectors of the training pattern along with the pre-processed data points.

The proposed algorithm doesn't include the words and lines detection in a document. Other pre-processing steps, such as slant correction, are not addressed in this paper.

REFERENCES

[1]. A. Shazia and Q. Aasia, "Document Image Processing - A Review," International Journal of Computer Applications, November 2010, Volume 10, No.5.

[2]. R. Plamondon and S. Srihari, "Online and off-line handwriting recognition: a comprehensive survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000*, 63 –84.

[3]. C. C. Tappert, C. Y. Suen and T. Wakahara, "The state of the art in online handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,1990, pp.* 787–808.

[4]. A. Ashutosh , R. Rajneesh and RenuDhir, " Handwritten Devanagari Character Recognition Using Gradient Features," International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2, Issue 5, May 2012.

[5]. P. Chomtip, W. Verachad, J. Satheanpong and C. Nannaphat, "Thai Handwritten Character Recognition by Genetic Algorithm (THCRGA)," *IACSIT International Journal of Engineering and Technology, Vol.3, No.2, April 2011.*

[6]. R. Plamondon, D. Lopresti, L.R.B. Shoemaker and R. Srihari, "On-line Handwriting Recognition," Encyclopedia of Electrical and Electronics Eng., J.G. Webster, ed., vol. 15,
pp. 123-146, New York: Wiley, 1999.

[7]. X.Li, R.Plamondon, M.Parizeau, "Model-based on-line handwritten digit recognition," Proc. of 14th Intl. Conf. On Pattern Recognition, Brisbane, Australia, August, 1998,
pp.1134-1136.

[8]. U. Pal, N. Sharma, T.Wakabayashi, and F.Kimura,
"Off-line handwritten character recognition of Devanagari script," in Proc. 9th Conf. Document Analysis andRecognition, 2007, pp. 496-500.

[9]. J. Sternby, J. Morwing, J. Anderson and C. Friberg, ''On-line Arabic handwriting recognition with templates," Pattern Recognition, vol 42, 2009, pp. 3278-3286.

[10]. I. Muthumani and C.R. Uma Kumari, "Online Character Recognition of Handwritten Cursive Script," IJCSI International Journal of Computer Science Issues, vol. 9, Issue 3, No 2, 2012, pp. 352-354.

[11]. D. Connell Scott, "On Line Handwritten Recognition Using Multiple Pattern Class Models," Ph.D. Thesis, University de Michigan state, East Lansing, 2000.

[12]. M. N. Anoop and K. J. Anil, "Online Handwritten Script Recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 26, no. 1, January 2004.

[13]. L. Mahmoud, N. Sourour, B. Hala and M. A. Adel, "Genetic Algorithms for Perceptual Codes Extraction," *Journal of Intelligent Learning Systems and Applications*, 2012, 4, 255-265

[14]. F. L. Chung, T. C. Fu and R. W. P. Luk, "An Evolutionary Approach to Pattern-Based Time Series Segmentation," *IEEE transactions on evolutionary computation*, Vol. 8, No. 5, 2004, pp. 471-489.

[15]. J. P. J. Alberto and C. P. C. Juan, "Genetic Algorithms for Linear Feature Extraction," *Pattern Recognition Letters*, Vol. 27, No. 13, 2006, pp. 1508-1514.

[16]. M. S. William and A. D. J. Kenneth, "An Analysis of Multipoint Crossover," *Proceedings of the First Work- shop on Foundation of Genetic Algorithms*, Bloomington, 15-18 July 1990, pp. 301-315.

[17]. D. Delahaye, J. M. Alliot, M. Schoenauer and J. L. Farges, "Genetic Algorithms for Partitioning Air Space," *Proceedings of the Tenth IEEE Conference on Artificial Intelligence for Application*, San Antonio, 1-4 March 1994, pp. 291-297.

[18]. G. Menier, "On-Line System of Reading and Writing Cursive Handwriting: Continuous Analysis of Features and Global Interpretation Optimized by Genetic Algorithm," Ph.D. Thesis, University Rennes 1, Rennes, 1995.

[19]. G. Menier, G. Lorette and P. Gentric, "A Genetic algorithm for on-line cursive handwriting recognition", Pattern Recognition, Vol. 2, 1994, pp. 522 – 525.

[20]. R. Kala, H. Vazirani, A. Shukla and R. Tiwari, "Offline Handwriting Recognition using Genetic Algorithm," International Journal of Computer Science Issues, vol. 7, Issue 2, No 1, 2010, pp. 16-25.

[21]. K. Deb, "Genetic Algorithm in Search and Optimization: The Technique and Applications," *Proceedings of Inter- national Workshop on Soft Computing and Intelligent Systems*, Calcutta, 12-13 January 1998, pp. 58-87.

[22]. A. L. Koerich, R. Sabourin and C. Y. Suen, "Large Vocabulary Off-Line Handwriting Recognition: A Survey," *Pattern analytic application*, Vol. 6, No. 2, 2003, pp. 97- 121.

[23]. S. Budi, A. B. Muhammad and E. W. Stefanus, "A Cross Entropy-Genetic Algorithm for M-Machines No-Wait Job- Shop Scheduling Problem," *Journal of Intelligent Learning Systems and Applications*, Vol. 3, No. 3, 2011, pp. 171-180.

[24]. Z. Bahadir and O. Ibrahim, "An Improved Genetic Algorithm for Crew Pairing Optimization," *Journal of Intelligent Learning Systems and Applications*, Vol. 4, No. 1, 2012, pp. 70-80.

[25]. S. D. Connell and K. J. Anil, "Template-based online character recognition," *Pattern Recognition, 1999*, 34: 1–14.