RESEARCH ARTICLE OPEN ACCESS

Comparative Analysis of PSO and GA in Geom-Statistical Character Features Selection for Online Character Recognition

Fenwa O.D., Ajala F.A. and Alo O.O.
[1,2,3] Computer Science LAUTECH Ogbomoso, Oyo State, Nigeria
Corresponding Authors mail: odfenwa@lautech.edu.ng, faajala@lautech.edu.ng, ooalo@lautech.edu.ng

Abstract
Online handwriting recognition today has special interest due to increased usage of the hand held devices and it has become a difficult problem because of the high variability and ambiguity in the character shapes written by individuals. One major problem encountered by researchers in developing character recognition system is selection of efficient features (optimal features). In this paper, a feature extraction technique for online character recognition system was developed using hybrid of geometrical and statistical (Geom-statistical) features. Thus, through the integration of geometrical and statistical features, insights were gained into new character properties, since these types of features were considered to be complementary. Several optimization techniques have been used in literature for feature selection in character recognition such as; Ant Colony Optimization Algorithm (ACO), Genetic Algorithm (GA), Particle Swarm Optimization (PSO) and Simulated Annealing but comparative analysis of GA and PSO in online character has not been carried out. In this paper, a comparative analysis of performance was made between the GA and PSO in optimizing the Geom-statistical features in online character recognition using Modified Optical Backpropagation (MOBP) as classifier. Simulation of the system was done and carried out on Matlab 7.10a. The results generated show that PSO is a well-accepted optimization algorithm in selection of optimal features as it outperforms the GA in terms of number of features selected, training time and recognition accuracy.
Keywords: Genetic Algorithm, Particle Swarm Optimization, Geom-stastistical features.

I. Introduction

Online handwriting recognition today has special interest due to increased usage of the hand held devices. The incorporation of keyboard being difficult in the hand held devices demands for alternatives, and in this respect, online method of giving input with stylus is gaining quite popularity [1]. Recognition of handwritten characters with respect to any language is difficult due to variability of writing styles, state of mood of individuals, multiple patterns to represent a single character, cursive representation of character and number of disconnected and multi-stroke characters [2]. Current technology supporting pen-based input devices include: Digital Pen by Logitech, Smart Pad by Pocket PC, Digital Tablets by Wacom and Tablet PC by Compaq [3]. Although these systems with handwriting recognition capability are already widely available in the market, further improvements can be made on the recognition performances for these applications. The challenges posed by the online handwritten character recognition systems are to increase the recognition accuracy and to reduce the recognition time [4], [1]. However, selection of a feature extraction method is probably the single most important factor in achieving high recognition performance in character recognition system [5]. No matter how sophisticated the classifiers and learning algorithms, poor feature extraction will always lead to poor system performance [6]. Feature selection refers to the problem of dimensionality reduction of data, which initially consists of large number of features. The objective is to choose optimal subsets of the original features which still contain the information essential for the classification task while reducing the computational burden imposed by using many features. The focus of this paper is comparative analysis of performance of GA and PSO algorithms in selecting optimal features for online character recognition system. It compares the optimization abilities of GA and PSO in terms of number of features selected, training time and recognition accuracy.

II. Research Methodology

Hundreds of features which are available in the literature can be categorized as follows: Global transformation and Series expansion features, Statistical features and Geometrical and Topological features. Many feature extraction techniques have been proposed in literature to improve overall recognition rate; however, most of the techniques used only one property of the handwritten character. This paper focuses on developing a feature extraction technique that combined three characteristics (stroke information, contour pixels and zoning) of the handwritten character to create a global feature vector. Hence, in this paper, a hybrid feature extraction algorithm was used to alleviate the problem of poor feature extraction algorithm of online character recognition system, the optimization capability of GA and PSO was tested on Geom-statistical features extracted and MOBP was adopted as classifier.

2.1 Development of the Proposed Hybrid Feature Extraction Algorithm

The most important aspect of handwriting character recognition scheme is the selection of good feature set which is reasonably invariant with respect to shape variation caused by various writing styles. Feature extraction is the process of extracting from the raw data the information which is the most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class pattern variability [7]. Features are unique characteristics that can represent an image, i.e. a character in this case. Each character is represented as a feature vector, which becomes its identity. The goal of feature extraction is to extract a set of features, which maximizes the recognition rate with the least amount of elements. Many feature extraction techniques have been proposed to improve overall recognition rate; however, most of the techniques used only one property of the handwritten character. This paper focuses on a feature extraction technique that combined three characteristics of the handwritten character to create a global feature vector. A hybrid feature extraction algorithm was developed using Geometrical and Statistical features. Integration of Geometrical and Statistical features was used to highlight different character properties, since these types of features are considered to be complementary [8].

2.1.1 Geometrical Features

Various global and local properties of characters can be represented by geometrical and topological features with high tolerance to distortions and style variations. This type of representation may also encode some knowledge about the structure of the object or may provide some knowledge as to what sort of components make up that object. The geometrical features used in this paper were the Stroke Information and Contour pixel of the characters.

Figure 2.1: The snapshot of Genius Pen (G-Pen 450) Digitizer for Character Acquisition

2.1.1.1 Stroke Information

Stroke Information is a combination of local and global features, which are aimed to capture the geometrical and topological features of the characters and efficiently distinguish and identify the character from a small subset of characters. Stroke is storage of pen movements in online handwriting recognition. These movements appear at various positions on view point and joining these positions in first-come-first serve basis shows the appearance of drawn text. A character may consist of single or multiple strokes. The list formed in data collection includes nodes, where each node includes two fields, namely, point and stroke number. Here, the point represents a coordinate of view point and stroke number represents identity and sequential order of stroke. Higher recognition performance would be possible if online recognition methods were able to address drawing motion vector (stroke) information [9]. The feature sets consist of:

(i) Stroke Number

Stroke number helps in identifying similar points, gaps and crossings. The pen movement consists of three functions, namely, Pen-Down, Pen-Move and Pen-Up. When one presses, moves, lifts the pen up consecutively, and more than one point collected, the stroke number is incremented. Pen-Move function stores movements of pen on writing pad. An example of a digital pen for generating stroke information is as shown in figure 2.1. Figure 2.2 shows a typical example of how different stroke numbers are generated. However, only stroke is not enough because most of the time different character may get the same no of strokes. Therefore, in this research, PEN-UP is used as a feature to check how well the character matches the standard one (i.e. the average for the same character in the database). This feature is calculated by using the average strokes of a specific character as an input using the membership function as in Equation 2.1:

$$\text{PEN-UP} = e^{|\text{average} - x|} \qquad (2.1)$$

where x is the real strokes for the specific character.

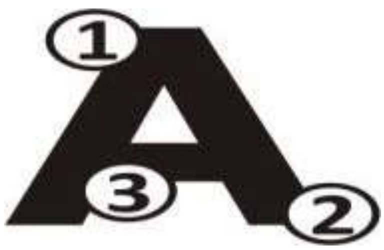

Figure 2.2: Writing character “A’’ with 3 Strokes

(ii) Pressure of the Stroke

This is the pressure representing Pen Ups and Downs in a continuous manner. The use of pen pressure as a feature is used for the improvement of a basic performance of the writer- independent online character recognition. The value of the pen pressure exerted on the writing pad was also used as feature. Moreover, recognition performance could be raised using writing pressure information of on-line writer identification systems and on-line character recognition systems [9].

(iii) Number of Junctions and their Location

A black pixel is considered to be a junction if there are more than two black pixels in its 5 by 7 neighbourhood in the resolution of the character image. The number of junctions as well as their positions in terms of 35 (5x7) quadrants are recorded. For example the character image of Figure 2.3 has 2 junctions in quadrants 2 and 17. Junctions lying within a pre-defined radial distance are merged into a single junction and the junctions associated with the headline are ignored.

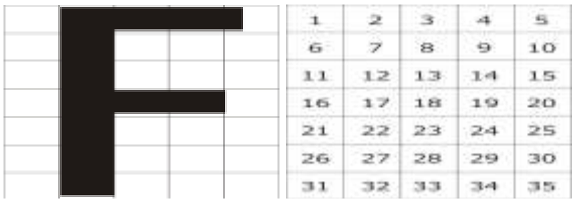

Figure 2.3: Division of character image into 35 quadrants

(iv) Horizontal Projection Count

Horizontal Projection Count is represented as HPC(i) = Σ F(i, j), where F(i,j) is a pixel value (1 for black background and 0 for white foreground) of a character image, and i and j denote row and column positions of a pixel, with the image's top left corner set to F(0,0). It is calculated by scanning the image row-wise and finding the sum of background pixels in each row (Figure 2.4). To take care of variations in character sizes, the horizontal projection count of a character image is represented by percentage instead of an absolute value and in this present work it is stored as a 4 component vector where the four components symbolize the percentage of rows with 1 pixel, 2 pixels, 3 pixels and more than 3 pixels. The components of this vector for the character image given in Figure 2.4 will be [50, 0, 10, 10], as there are 5 rows with 1 pixel; no rows with 2 pixels; 1 row with 3 pixels and 1 row with more than 3 pixels.

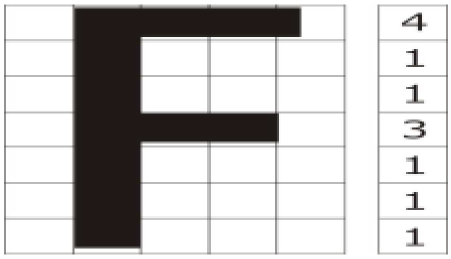

Figure 2.4: Horizontal Projection Count of character image ''F''

2.1.1.2 Contour Pixels

Correct extraction of the contour will produce more accurate features that will increase the chances of correctly classifying a given character or pattern. But the question that might arise is why first extract the contour of a pattern and then collect its features? Why not collect features directly from the pattern? One answer is, the contour pixels are generally a small subset of the total number of pixels representing a pattern. Therefore, the amount of computation is greatly reduced when feature extracting algorithms are run on the contour instead of the whole pattern. Because the contour shares a lot of features with the original pattern but has fewer pixels, the feature extraction process becomes much more efficient when performed on the contour rather on the original pattern. Contour tracing is often a major contributor to the efficiency of the feature extraction process, which is an essential process in pattern recognition [10], [11].

In order to extract the contour of the pattern, he following actions must be taken: every time a black pixel is encountered, turn left, and every time a white pixel is encountered, turn right, until the starting pixel is met again. All the black pixels traced out is the contour of the pattern. The contour tracing algorithm used in this research is based on the model developed by [12].

Contour Tracing Algorithm [12].
Input: An image *I* containing a connected component *P* of black pixels.
Output: A sequence B (b1, b2, . . . , bk) of boundary pixels, that is, the outer contour.
Begin
Set B to be empty
From bottom to top and left to right scan the cells of I until a black pixel, S, of P is found
Insert S in B
Set the current pixel, P, to be the starting pixel, S
Turn left, that is, visit the left adjacent pixel of P
Update P(ie, set it to be the current pixel)
While P not equal to S do
    If the current pixel P is black
    Insert P in B and turn left (visit the left adjacent pixel of P)
    Update P (ie, set it to be the current pixel)

Else
Turn right (visit the right adjacent pixel of P)
Update P(ie, set it to be the current pixel)
End

2.1.2 Statistical Features

Statistical features are derived from the statistical distribution of points. They provide high speed and low complexity and take care of style variations to some extent. They may also be used for reducing the dimension of the feature set. The statistical feature adopted in this research is 'Zoning'. Zone-based feature extraction method provides good result even when certain pre processing steps like filtering, smoothing and slant removing are not considered.
Image Centroid and zone-based (ICZ) distance metric feature extraction and Zone Centroid and zone-based (ZCZ) distance metric feature extraction algorithms were proposed by [13] for the recognition of four popular Indian scripts (Kannada, Telugu, Tamil and Malayalam) numerals.

In this paper, hybrid of modified Image Centroid and zone-based (ICZ) distance metric feature extraction and modified Zone Centroid and zone-based (ZCZ) distance metric feature extraction methods was used. Modifications of the two algorithms are in terms of:
(i) Number of zones being used
(ii) Measurement of the distances from both the Image Centroid and Zone Centroid
(iii) The area of application.

2.1.2.1 The Zoning Algorithm

The most important aspect of handwriting recognition scheme is the selection of good feature set, which is reasonably invariant with respect to shape variations caused by various writing styles. The zoning method is used to compute the percentage of black pixel in each zone. The rectangle circumscribing the character is divided into several overlapping, or non-overlapping regions and the densities of black points within these regions are computed and used as features as shown in Figure 2.6. The major advantage of this approach stems from its robustness to small variation, ease of implementation and good recognition rate. Zone-based feature extraction method provides good result even when certain pre processing steps like filtering, smoothing and slant removing are not considered.

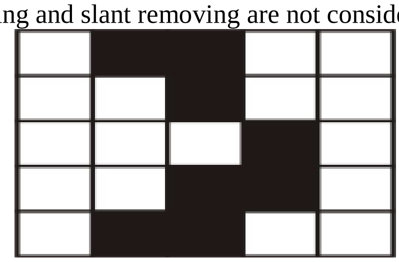

Figure 2.6: Feature Extraction using Zoning

2.1.2.2 The Developed Hybrid (Geom-Statistical) Feature Extraction Algorithm

Step 1: Get the stroke information of the input characters from the digitizer (G-pen 450)
These include:
(i) Pressure used in writing the strokes of the characters
(ii) Number (s) of strokes used in writing the characters
(iii) Number of junctions and the location in the written characters
(iv) The horizontal projection count of the character

Step 2: Apply Contour tracing algorithm to trace out the contour of the characters
Step 3: Run Hybrid Zoning algorithm on the contours of the characters
Step 4: Feed the outputs of the extracted features of the characters into the digitization stage in order to convert all the extracted features into digital forms

2.2 Image Processing

This phase is carried out in *two stages* which are: feature extraction and feature selection.

2.2.1 Feature Extraction:

This stage is carried out in two steps:
Step 1: The Stroke Information and Contour pixel of the character image are used as the geometric feature of the character. Five geometric features were extracted and these include the following:
¬ Number of the stroke
¬ Pressure of the stroke
¬ Number of Junctions and their location
¬ Horizontal projection count
¬ Contour Pixels

Step 2: Statistical details of the character area are also adopted in this paper. Ten statistical features were extracted and these are as listed below:
¬ Compute the input image centroid
¬ Centre of gravity in the horizontal direction (x–axis)
¬ Centre of gravity in the vertical direction (y–axis)
¬ Compute the distance between the image centroid to each pixel present in the zone
¬ Compute average distance between these points
¬ Compute the zone centroid for the entire pixel present in the zone
¬ Centre of gravity in the horizontal direction (x–axis)
¬ Centre of gravity in the vertical direction (y–axis)
¬ Compute the distance between the zone centroid to each pixel present in the zone

¬ Compute average distance between these points

Therefore, the total number of features extracted was fifteen and these were passed to the two selected optimization algorithms (GA and PSO) in feature selection stage.

2.2.2 Feature Selection Several optimization techniques can be used to optimize extracted features in online character recognition . Some of these include Ant Colony Optimization, Genetic Algorithm, Simulated Annealing, Particle Swarm Optimization (PSO) e t c.

2.2.2.1 Particle Swarm Optimization (PSO)

The PSO method is a member of wide category of Swarm Intelligence methods for solving the optimization problems. It is a population based search algorithm where each individual is referred to as particle and represents a candidate solution. Each single candidate solution is "an individual bird of the flock", that is, a particle in the search space. Each particle makes use of its individual memory and knowledge to find the best solution. All the particles have fitness values, which are evaluated by fitness function to be optimized and have velocities which direct the movement of the particles. The particles move through the problem space by following a current of optimum particles. The initial swarm is generally created in such a way that the population of the particles is distributed randomly over the search space. At every iteration, each particle is updated by following two "best" values, called pbest and gbest. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (fitness value). This fitness value is called pbest. When a particle takes the whole population as its topological neighbor, the best value is a global best value and is called gbest. The detailed algorithm is given as follows:

Step 1: Set the constants k_{max}, c_1, c_2, r_1, r_2, w.
Randomly initialize particle positions $x_0(i)$
for i = 1, 2………p.
Randomly initialize particle velocities $v_0(i)$
for i = 1, 2……..p.
Step 2: Set k =1.
Step 3: Evaluate function value f_k using design space coordinates $x_k(i)$
If $f_k \geq f_{pbest}$, then pbest(i) = $x_k(i)$
If $f_k \geq f_{gbest}$, then gbest= $x_k(i)$
Step 4: Update particle velocity using the following equation

$$v_{k+1}(i)= w^*(v_k(i)) + c_1r_1^*(pbest_k(i) - x_k(i)) + c_2r_2(gbest_k – x_k(i)) \quad (2.2)$$

Update particle position vector using the following equation

$$x_k + 1(i) = x_k(i) + v_k + 1(i) \quad (2.3)$$

Step 5: Increment i. If i > p, then increment k and set i = 1.
Step 6: Repeat steps 3 to 5 until k_{max} is reached.
The notations used in this algorithm are:
k_{max} = maximum iteration number
w = inertia weight factor
c_1, c_2 = cognitive and social acceleration factors
r_1, r_2 = random numbers in the range (0, 1).

In this paper, each of the fifteen features are represented by a chromosome (string of bits) with 15 genes (bits) corresponding to the number of features. An initial random population of 20 chromosomes is formed to initiate the genetic optimization. The initial coding for each particle is randomly generated. The order of position of the features in each string is stroke number, pressure of the stroke, number of junction and their location, horizontal projection count, contour pixel, image centroid, Centre of gravity in the horizontal direction (x–axis) of the image centroid, Centre of gravity in the vertical direction (y–axis) of the image centroid, the distance between the image centroid to each pixel present in the zone, the average distance between image centroid, the zone centroid for the entire pixel present in the zone, Centre of gravity in the horizontal direction (x–axis) of the zone centroid, Centre of gravity in the vertical direction (y–axis) of the zone centroid, the distance between the zone centroid to each pixel present in the zone and the average distance between the zone centroid respectively. A suitable fitness function is estimated for each individual. The fittest individuals are selected and the crossover and the mutation operations are performed to generate the new population. This process continues for a particular number of generations and finally the fittest chromosome is calculated based on the fitness function. The features with a bit value "1" are accepted and the features with the bit value of "0" are rejected. The fitness function used in this work is given by

$$\text{Fitness } (\alpha * \gamma) + \beta * \quad |c| - |r|/|c| \quad (2.4)$$

where γ = classification accuracy
c = total number of features
r = length of the chromosome (number of '1's) $\alpha \in [0, 1]$ and $\beta = 1 - \alpha$

This formula shows that the classification accuracy and the feature subset length have different significance for feature selection. A high value of α assures that the best position is at least a rough set reduction. The goodness of each position is evaluated by this fitness function. The criteria are to maximize the fitness values. An optimal solution is obtained at the end of the maximum iteration. This value is binary coded with eleven bits. The bit value of "1" represents a selected feature whereas the bit value of "0" represents a rejected feature. Thus an optimal set of features are selected from the PSO technique. **Out**

of the fifteen *features extracted,* eight *optimal set of features* are selected using PSO algorithm.

2.2.2.2 Genetic Algorithm (GA)

This is a family of evolutionary algorithms based approaches applied for optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Genetic Algorithm convert the problem into a model by using chromosomes like data structure and evolve the chromosomes using selection, recombination and mutation operator. GA begins with randomly selected population of chromosomes which represents the problem to be solved. An evaluation function is used to examine the “goodness” of each chromosome. The operation start from an initial population of randomly generated chromosomes population evolved for a number of generation and every time quality of an individual gradually increased. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, the more fit individuals are stochastically selected from the current population, and each individual's genome is modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

Basic Description of GA

Genetic algorithm is started with a set of solutions (represented by chromosomes) called population. Solutions from one population are taken and used to form a new population. This is motivated by a hope, that the new population will be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness; the more suitable they are the more chances they have to reproduce. This is repeated until some condition (for example number of populations or improvement of the best solution) is satisfied.

Outline of the Basic Genetic Algorithm

1) **Start:** Generate random population of n chromosomes (suitable solutions for the problem).
2) **Fitness:** Evaluate the fitness f(x) of each chromosome x in the population.
3) **New population:** Create a new population by repeating the following steps until the new population is complete:

a. **Selection:** Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)

b. **Crossover:** With a crossover probability, crossover the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.

c. **Mutation:** With a mutation probability, mutate new offspring at each locus (position in chromosome)

d. **Accepting:** Place new offspring in a new population.

4) **Replace:** Use new generated population for a further run of algorithm

5) **Test:** If the end condition is satisfied, stop, and return the best solution in current population

6) **Loop:** Go to step 2

Thus an optimal set of features are selected from the GA technique. **Out of the fifteen *features extracted,* eleven *optimal set of features* are selected** using GA algorithm.

III. System Design and Implementation

GA and PSO algorithms were integrated with MOBP classifier and implemented using Matlab 7.10a. The developed system was RUN under Windows7 operating system on Pentium (R) 4.00GB RAM, 2.13GH Processor. Experiments were conducted on GA-based MOBP neural network and PSO-based MOBP classifiers using 6,200 handwritten character samples (uppercase (A-Z), lowercase (a-z) English alphabet and 10 digits (0-9)) collected from 100 subjects using G-Pen 450 digitizer and the system was tested with 100 character samples written by people who did not participate in the initial data acquisition process.

IV. Results and Discussion

Experimental results show that PSO has better optimization ability than GA in terms of number features selected as indicated in table 4.1. The training time and recognition accuracy as indicated in tables 4.2 and 4.3 of PSO-based MOBP classifier is better than GA-based MOBP classifier because; PSO is a mathematical model which only uses velocity and does not support genetic operators(with higher space and time complexity) like crossover, mutation and selection.

Table 4.1: The number of features selected by the two optimization algorithms

Optimization Algorithm	**No of Features Extracted**	**No of Features Selected**
Genetic Algorithm(GA)	15	11
Particle Swarm Optimization(PSO)	15	8

Table 4.2: The Training Time (in minutes) of the two classifiers under different datasets

Dataset	**GA-Based MOBP Training Time**	**PSO-Based MOBP Training Time**
1,200	0.21	0.13
2,480	3.01	2.32
3,720	4.15	3.62
4,960	9.60	6.45
6,200	15.98	12.11

Table 4.3: Recognition Accuracies of GA-based MOBP and PSO-Based MOBP Classifiers

Dataset	**GA-Based MOBP (%)**	**PSO-Based MOBP(%)**
1,200	84	86
2,480	86	89
3,720	88	93
4,960	90	96
6,200	96	99

V. Conclusion and Recommendation

This paper focuses on a feature extraction technique that combined three characteristics of the handwritten character to create a global feature vector, a hybrid feature extraction algorithm was used to alleviate the problem of poor feature extraction algorithm of online character recognition system. However, the optimization capability of GA and PSO was tested on Geom-statistical features extracted and MOBP was adopted as classifier. PSO showed promising results in terms of the number of features selected, training time and recognition accuracy. Future work could be tailored towards simulation on system with higher memory. However, other optimization techniques such as Simulated Annealing optimization could be used for evaluation and conventional programming languages such as C#, java, and Python could also be used as a platform to check results.

REFERENCES

[1] Gupta, K., Rao, S.V. and Viswanath (2007). Speeding up Online Character Recognition. *Proceedings of Image and Vision Computing*, Newzealand, Hamilton, 41-45.

[2] Shanthi, N., and Duraiwamy, K. (2007). Performance Comparison of Different Image Sizes for Recognizing Unconstrained Handwritten Tamil Characters Using SVM. *Journal of Science*, 3, 9, 760-764.

[3] Manuel, J., Fonseca, and Joaquim, A. J. (2001). Experimental Evolution of an Online Scribble Recognizer. *Pattern Recognition Letters*, 22(12): 1311-1319.

[4] Rejean, P. and Sargur, S. N. (2000). On-line and Off-line Recognition: A Comprehensive Survey. *IEEE Transaction*

on Pattern Analysis and Machine Intelligence, 22(1): 63-84.

[5] Pradeep, J., Srinivasan, E. and Himavathi, S. (2011). Diagonal Based Feature Extraction for Handwritten Alphabets Recognition using Neural Network. *International Journal of Computer Science and Information Technology (IJCS11)*, 3, 1, 27-37.

[6] Marc, P., Alexandre, L., and Christian, G. (2001): "Character Recognition Experiment using Unipen Data'', Parizeau and al., Proceeding of ICDAR, Seattle: 10-13.

[7] Naser, M. A., Adnan, M., Arefin, T. M., Golam, S. M. and Naushad, A. (2009). Comparative Analysis of Radon and Fan-beam based Feature Extraction Techniques for Bangla Character Recognition, *IJCSNS International Journal of Computer Science and Network Security*, 9(9): 120-135.

[8] Heute, L. T. and Paquet, J. V. (1998). A Structural/Statistical feature base Vector for Handwritten Character Recognition. *Pattern Recognition Letters*, 19: 629-641.

[9] Nishimura, H. and Timikawa, T. (2003). Offline Character Recognition using Online Character Writing Information. *Proceedings of the seventh IEEE International Conference on Document Analysis and Recognition.*

[10] Liu, C. L., Sako, H. and Fujisawa, H (2003). Handwritten Chinese character recognition: Alternatives to nonlinear normalization, *In Proceedings of the 7th International Conference on Document Analysis and Recognition*, Edinburgh, Scotland, 524-528.

[11] Liu, C., Stefan, J. and Masaki N. (2004). Online Recognition of Chinese Characters: The State-of-the- Art. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(2): 198-203.

[12] Yamaguchi, T. Nakano, Y. Maruyama, M., Miyao H and Hananoi T. (2003): Digit Classification on Signboards for Telephone Number Recognition. *In Proceedings of the 7th International Conference on Document Analysis and Recognition* Edinburgh, Scotland, 359–363.

[13] Vanaja, P. R. And Rajashekararadhya, S. V (2008): Handwritten numeral recognition of three popular South Indian Scripts: A Novel Approach. *Proceedings of the Second International Conference on information processing ICIP*, 162-167.