



Development of a Genetic based Neural Network System for Online Character Recognition

J. O. Adigun

Department of Computer
Technology, Yaba College of
Technology, Lagos
Nigeria

E.O. Omidiora

Department of Computer
Science and Engineering
Ladoke Akintola University of
Technology, Ogbomoso,
Nigeria

S.O. Olabiyisi

Department of Computer
Science and Engineering
Ladoke Akintola University of
Technology, Ogbomoso
Nigeria

O.D. Fenwa

Department of Computer
Science and Engineering
Ladoke Akintola University of
Technology, Ogbomoso
Nigeria

O. Oladipo

Centre for Information
Technology Management
Yaba College of Technology
Lagos., Nigeria

M.M Rufai

Department of Computer
Technology, Yaba College of
Technology
Lagos, Nigeria

ABSTRACT

Character Recognition has been one of the most intensive research during the last few decades because of its potential applications. However, most existing classifiers used in recognizing online handwritten characters suffer from poor feature selection and slow convergence which affect training time and recognition accuracy. Hence, this paper focused on integrating an optimization (genetic algorithm) into modified backpropagation neural network to enhance the performance of character recognition. This paper proposed a methodology that is based on extraction of features using stroke number, invariant moments, projection and zoning. Genetic algorithm was used as feature selection to optimize the subset of the character for classification. A Modified Genetic Algorithm (MGA) was modified to reduce character recognition errors using fitness function and genetic operators. However, an integration of optimization algorithm (modified genetic algorithm) into an existing modified backpropagation (MOBP) learning algorithm was employed as classifier. For further enhancement of classifier, three classifiers (C1, C2 and C3) were formulated from MGA-MOBP model and evaluated using training time and correct recognition accuracy. C3 performed better than C1 and C2 in terms of convergence rate, correct recognition accuracy and feature selection (its ability to remove irrelevant features of character images). The results of the developed system achieved a false recognition of 0.56% and 99.44% overall recognition accuracy compared with existing models.

General Terms

Pattern Recognition

Keywords

Artificial Neural Network, optical backpropagation, genetic algorithm, character recognition, feature extraction, feature selection, genetic operators.

1. INTRODUCTION

Development in Information Technology has produced varieties of electronics devices such as Personal Digital

Assistants (PDAs), handheld computers where non-keyboard based method of data entry are receiving more attention in the research of pattern recognition. The increase in usage of handheld devices has created a growing demand for algorithm that would reduce the processing time and increase recognition accuracy [8][10][12]. A character is recognized by identifying all the strokes of the character [1][4]. Handwriting recognition can be classified into two types as offline and online handwriting recognition methods. Offline recognition for which recognition takes place on a static image captured once the writing process is over [9][1] whereas in online system, handwriting data are captured during writing and information on the ordering of strokes are also available [16]. However, optimization techniques was integrated into classification techniques in character recognition systems to further improve the overall performance of the recognition system. The backpropagation neural network is a popular algorithm for training a multilayer network [Adigun et al, 2015], despite this, it has a lot of drawbacks: slow convergence and easily be trapped in local minimal. Hence, this paper integrates the MGA and MOBP to achieve a better performance of recognition accuracy. The performance evaluation was carried out on C1 (MGA and MOBP) C2 (MGABased MOBP) and C3 (MGA and MGABased MOBP) are the three classifiers used to check the effect of the optimization algorithm on the performance of the recognition system. Section two describes related work, while section three contains the material and methods, with discussion of results in section four and draw conclusion in section five.

2. RELATED WORK

Online handwritten character recognition has been one of the most challenging research areas in the field of pattern recognition in the recent years. Several works have been focusing on new techniques that would reduce the processing time and increase recognition accuracy [1]. The selection of appropriate feature extraction method is undoubtedly the single most important factor in achieving high recognition performance [11][7]. [8] proposed a method based on the extraction of different spatial and temporal features from

strokes of the character and recognition was done by using genetic algorithm as a tool to find an optimal subset of the stroke features. Recognition rate of 83.1% was achieved. [14] developed English character recognition system using the hybrid of standard backpropagation and genetic algorithm for recognition of uppercase alphabets. Recognition rate performance was 91.1%. [9] designed a novel and robust hybrid recognition system for Odia handwritten character, based on the algorithm of feed forward BPNN combined with Genetic Algorithm (GA) to perform the optimum feature extraction and recognition. [4][5] a hybrid feature extraction technique was developed using geometrical and statistical features. However, a multiple classification character recognition scheme was employed using modified counter propagation and modified optical backpropagation neural network. [17] presented a backpropagation network algorithm combined with genetic algorithm to achieve both accuracy and training swiftness for recognizing alphabets. Recently, researchers have invested in improving backpropagation algorithm in many ways such as adjustment of learning rate [3], and error adjustment [4][5], However, these works were yet to deliver the desired results. Hence, the aim of this paper is to integrate modified genetic algorithm into modified backpropagation neural network to improve the performance of character recognition system in terms recognition accuracy and recognition time.

3. MATERIALS AND METHODS

Figure 1 shows the block diagram of the character recognition system. The system consists of five phases, Acquisition, preprocessing, feature extraction, training and classification phase (using integration of modified genetic algorithm and) testing. In this paper, a modified genetic algorithm (MGA) and modified backpropagation(MOBP) neural network was developed for better online character recognition. To substantiate the effectiveness of the developed technique, it was compared with three types of classifiers. C1, C2 and C3 classifiers. The character images were classified into correct recognition, false recognition and recognition failure. The feature selection was used to reduce the number of features by elimination of insignificant and redundant features which improves the classification accuracy. The performance results were evaluated using images sizes, and database sizes based on training time, recognition time and recognition accuracy.

3.1 Data Acquisition and Data Preprocessing

The dataset were collected using digitizer table with an electric pen with sensing writing board. A programming interface was developed to acquire a set of character information from individuals. Characters considered were 26 upper case (A-Z), 26 lower case (a-z) English alphabets and 10 digits (0-9) making a total of 62. 6200 characters were collected from 50 individuals written twice for each character. All these entries were stored and serve as the training data set which was the input data that was fed into neural network for character recognition system. Figure 2 shows a sample of dataset used.

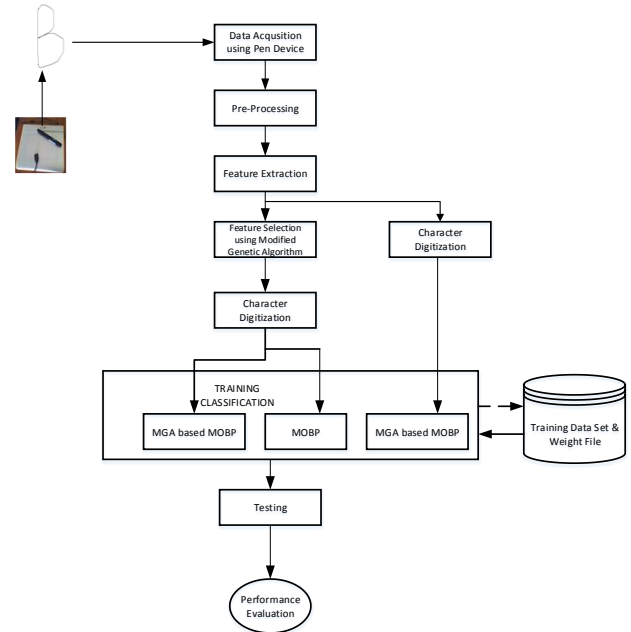


Figure 1: Block diagram of the Developed Character Recognition System

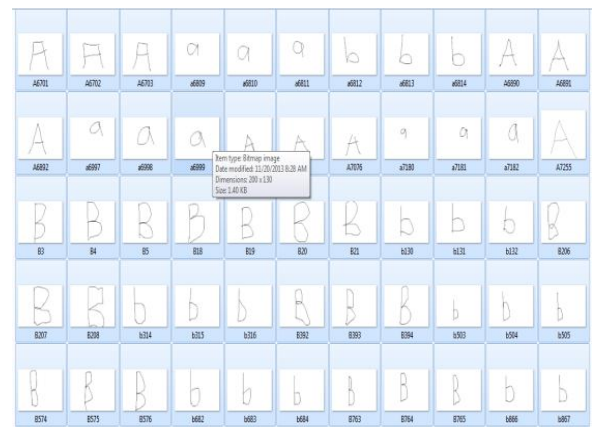


Figure 2: example of sample data collected using Genius easy i4405x

The data collected were subjected to a number of preliminary processing steps to make it usable in the descriptive stages of character analysis. The output is converted into the binary form by assigning "1" to black and "0" to white portion of the character image. The output was normalized by reducing the size and all redundancy errors from the character were removed without losing any important information and shape the input image (trained and tested) into suitable matrix size of 5 by 7, 10 by 14 and 20 by 28 for all input characters for feature extraction. Hence, three processing techniques were used in this work: binarization, extreme coordinate measurement and grid resizing.

3.2 Feature Extraction

In this paper the process of feature extraction were employed and tested on training set to determine the most optimal set of features for character recognition system. The features used were based on stroke information, invariant moments, projection and zoning of the handwritten character to create a global feature vector. Most of the techniques used only one property feature. This research combines feature to complement each other, handle style variations and

distinguish one input pattern from another pattern to achieve better performance. Hence, a hybrid structural and statistical feature was developed to highlight different character properties that effectively identify a character as shown in Figure 3.

Developed Hybrid (Struct-Statistical) Feature Extraction Algorithm

The following stages were taken in the development:

Step 1: Stroke information of the image characters was obtained this include: pressure used, number of strokes and projection count of the character.

Step 2: Invariant moments to determine position, size and orientation of the character was applied

Step 3: Run Hybrid Zoning Algorithm on the character.

Step 4: Feature selection to reduce number of features by elimination of insignificant features was performed.

Step 5: The outputs of the extracted and selected features of the characters was fed into the digitization stage in order to convert all the extracted features into digital forms.

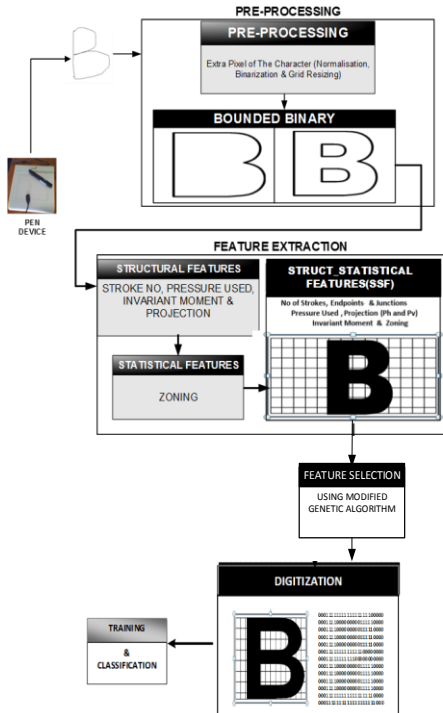


Figure 3: Developed Feature Extraction Model

Hybrid Zoning Algorithm

In this paper a hybrid of modified ICZ and modified ZCZ proposed by Fenwa *et al*,2012 was used:

Input: Pre-processed character image

Output: Features for Classification and Recognition

Begins

- Step 1: Divide the input image into 25 equal zones
- Step 2: Compute the input image centroid
- Step 3: Compute the distance between the image centroid to each pixel present in the zone
- Step 4: Repeat step 3 for the entire pixel present in the zone

Step 5: Compute average distance between these points

Step 6: Compute the zone centroid

Step 7: Compute the distance between the zone centroid to each pixel present in the zone.

Step 8: Repeat step 7 for the entire pixel present in the zone

Step 9: Compute average distance between these points

Step 10: Repeat the steps 3-9 sequentially for the entire zones

Step 11: Finally, 2*n(50) such features are obtained for classification and recognition.

Ends.

Modified Genetic Algorithm (MGA)

The training of Neural network by backpropagation is much time consuming and slow convergence due to local minimal. The loss of potential solution by standard genetic algorithm (SGA) is a common drawback. To overcome these problems, standard genetic algorithm was modified to speed up the convergence and reduce computational time for enhancing the training capacity of the network. Two unique strategies were modified to improve the GA optimization speed.

(a) The main idea of Genetic Algorithm is to mimic the processes in nature, there is a possibility that SGA may lose some potentially useful information. Therefore mutation was employed to prevent falling of all solutions in local population into local optimum of character recognition. The genetic algorithm mutation operator was processed first, followed by selection and crossover. This led to a new modification denoted as MSC (Mutation, Selection Crossover). After mutation, the fitness value for offspring are calculated and best fittest individuals are selected to replace the parent and reproduction of new chromosome.

(b) Modification of Fitness function. To determine the optimum subsolution, the fitness function was modified returned as (equation 3.1) is the number of selected zero or approximately zero features from the algorithm and multiplied with the percentage of feature used. The selection of parent was based on best fittest (i.e. smaller fitness values) because the optimization purpose was to reduce the percentage of recognition error while reducing the number of features used [11] and the fitness function was modified as:

$$\text{Fitness} = (\text{no of selected zero features}) * \% \text{FeatureUsed} \quad 3.1$$

where $H = H_{\text{projection}}$, $DH = \text{Downsample_Height}$

$DW = \text{Downsample_Width}$

$$\text{FeatureUsed} = \left(\frac{H}{DH * DW} \right) * 100$$

The MGA is terminated when some criteria are satisfied, e.g. when a particular point in the search space is encountered.

The MGA is terminated when some criteria are satisfied, e.g. when a particular point in the search space is encountered.

3.3 Feature Selection (FS)

FS ensures that only relevant features of the character are used in the character recognition process. The aim is to reduce the number of features by eliminating insignificant and redundant features which improve the classification accuracy. This reduces the computational burden imposed by using many features and improve the training time and testing efficiency.



In this paper, modified genetic algorithm is proposed for feature selection. MGA 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 mutation selection, and crossover are used to update the population. A repetition of these three operators is known as a generation. The new chromosome will then replace the chromosome with the lowest survival rate. This process will be iterated until the desired error rate is achieved. The features with the bit value "1" are selected and the features with the bit value of "0", the corresponding one are rejected

Feature Selection (Using Modified Genetic Algorithm)

1. Start
2. Get character features set
3. Initialize parameters (set gen<= number of generation ; set n <= population size, mutation prob)
4. Generate randomly distributed chromosomes to form initial population
5. Encode features by a chromosome {using bit strings encoding}
6. Gencount <= 0
7. Rank chromosomes based on its uniqueness {i.e. first chromosome occurrence = 1, subsequence occurrence = 0 }
8. Arrange chromosomes based on their fitness value{ Accept features with bit value = 1 and reject features with bit value = 0}
9. Mutate selected chromosomes based on mutation probability
10. Select chromosome with best fitness value
11. Perform crossover on parent chromosomes to form new offspring and replace the weak chromosomes with the new offspring
12. gencount <= gencount + 1 and repeat until total number of generation
13. if gencount = gen then goto 14 else 7
14. Output chromosome with the highest fitness value(feature selected)

3.4 Architecture of Artificial Neural Network Used

The architecture of artificial neural network used describes the properties of individual neurons such as the input layer, how they are combined, number of hidden layers, number of hidden neurons, its training function, performance function, error function and output neurons. The input layers take the input from character image extraction algorithm. The hidden layer used in this research was calculated by 2/3 of the input layer size, plus the size of the output layer with image sizes of 5 by 7 (35 pixels), 10 by 14 (140 pixels) and 20 by 28 (560 pixels) respectively. The output layer consists of 62 neurons; this is due to the fact that there are 62 characters to be identified. Thus each output neuron corresponds to every character and a binary vector of size 6 was used to represent the output values. The error function is proportional to the

square of the Euclidean distance between the desire output and the actual output of the network for a particular input pattern. As an alternative, other error functions whose derivatives exist and can be calculated at the output layer can replace the traditional square error criterion (Haykin, 2003). In this paper, error of the third order developed by Fenwa *et al.* (2012) was adopted to replace the traditional square error while the performance function used is number of errors. The equation of the cubic error is given as:

$$\delta_{pk}^o = (Y_{pk} - O_{pk})^2 \cdot f'_{pk}(\text{net}_{pk}^o) \quad 3.2$$

The cubic error will be manipulated mathematically in order to maximise the error of each output unit which will be transmitted backward from the output layer to each unit in the intermediate layer as shown below:

$$\text{Modified } \delta_{pk}^o = 3((1 + e^t)^2 \cdot f'_{pk}(\text{net}_{pk}^o))$$

$$\text{If } (Y_{pk} - O_{pk})^2 \geq 0 \quad 3.2a$$

$$\text{Modified } \delta_{pk}^o = -3((1 + e^t)^2 \cdot f'_{pk}(\text{net}_{pk}^o))$$

$$\text{If } (Y_{pk} - O_{pk})^2 \leq 0 \quad 3.2b$$

where Y_{pk} = Desired output

O_{pk} = Network output

$$T = (Y_{pk} - O_{pk})^2$$

Transfer functions are mathematical formula that determine the output of a processing neuron. They are referred to as transformation, activation or threshold functions. Transfer function is how the inputs and the weights are combined to produce output. The transfer functions adopted in this research were Euclidian distance and sigmoid function.

3.5 Development of Hybrid MGA and MOBP Model

In this paper, hybrid of MGA and MOBP neural network was developed as shown in Figure 3.6. The online character written is first captured using digitizer to get new image for processing. Then the characters are resized for training the neural network which has undergone various processes like converting the grayscale, binarizing, finding edge for feature extraction etc. During the preprocessing and feature extraction, the individual strokes are resampled to make sample data equidistant (standard size). This helps to reduce the variations, gaps and blurred in the characters and to avoid anomalous cases. Features are extracted from user's input pattern. The MGA was used for optimal feature selection. The initial inputs of the characters provided are trained with different parameters and different sets of characters. The characters are classified using three classifiers: C1, C2, and C3.

During training, the weights of the network are iteratively adjusted to minimize the error function. The set of outputs obtained are fed into the genetic algorithm to select the best fittest and best solution. The outputs of the genetic algorithm are sent to the neural networks as input. The recognition model involves the usage of artificial neural networks which have been exhaustively trained to recognize different types of handwritings. This is achieved by using highly efficient supervised learning algorithm. The output obtained from the trained are stored as files. Match the introduced character with the one in the database template and classify the given

character or pattern image. The extracted features are classified according to identify pattern to which it has been trained.

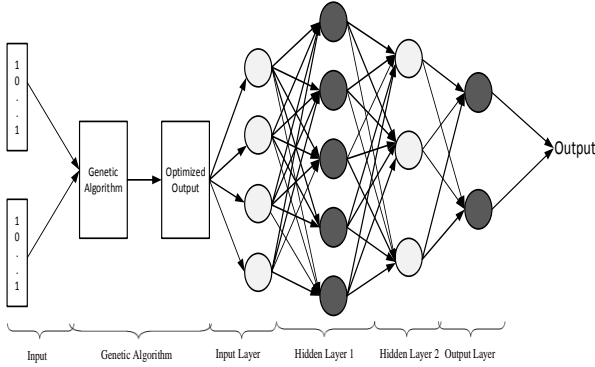


Figure 3.6: The Hybrid of GA and MOBP Model Used

3.5.1 The Genetic and Neural Network Algorithm

This research work employed hybrid of modified Genetic Algorithm and Modified Optical Backpropagation neural networks for the training and classification of the input pattern. The training algorithm involves the following two stages:

Stage A: Performs the training of the character sample generated using Genetic Algorithm.

Step 1: Select character samples collected from various people (Generate random population of n chromosomes this corresponds to initial population for the genetic Algorithm).

Step 2: Evaluate the fitness function of the population equation

Step 3: Mutate new offspring at each locus based on mutation probability.

Step 4: Select parent chromosomes from the population according to their fitness function (better fitness, bigger chance to be selected).

Step 5: Perform crossover on two strongest parents to create a new offspring and replace the weakest parent with the new child created. Insert the new offspring in the new population and use new generated population for a further run of the algorithm.

STAGE B: Performs the training of the input from the genetic algorithm to the output layer. According to [16]. Assume there are m input units, n hidden units, and p output units. W_{ji}^h weight on the connection from the i th input unit to j th hidden unit.

1. Input vector $X_p = X_{p1} + X_{p2} + X_{p3} \dots + X_{pm}$ to the input units. Each value in input layer receive X_{pi} and forward to values in hidden layer.

2. Calculate the net-input values to the hidden layer units. Each value in hidden layer sum up the input values weight Equation (3.5)

$$\text{net}_{pj}^h = (\sum_{i=1}^n W_{ji}^h \cdot X_{pi}) \quad 3.5$$

where net_{pj}^h is net input to the j th hidden unit, W_{ji}^h is weight on the connection from the i th input unit to j th hidden unit.

3. Calculate the outputs from the hidden layers. Use the activation function to measure the outputs Equation (3.6)

$$i_{pj} = f^h_j(\text{net}_{pj}^h) \quad 3.6$$

where i_{pj} is net input to the j th hidden unit and f^h is derivative of activation function and forward those outputs to every value to output layer. Each value in output layer sum up the input values weight Equation (3.7).

4. Calculate the net-input values to the output layer units.

$$\text{net}_{pk}^o = (\sum_{j=1}^L W_{kj}^o \cdot i_{pj}) \quad 3.7$$

where W_{kj}^o is weight on the connection from the j th hidden unit to k th output unit

5. Calculate the outputs from the output units. Use the activation function to calculate output value Equation (3.8)

$$O_{pk} = f^o_k(\text{net}_{pk}^o) \quad 3.8$$

where O_{pk} is actual output for the k th output unit

6. Calculate the error term for the output units

$$\text{Modified } \delta_{pk}^o = 3((1 + e^t)^2 \cdot f'^o_k(\text{net}_{pk}^o))$$

$$\text{If } (Y_{pk} - O_{pk})^2 \geq 0 \quad 3.9a$$

$$\text{Modified } \delta_{pk}^o = -3((1 + e^t)^2 \cdot f'^o_k(\text{net}_{pk}^o))$$

$$\text{If } (Y_{pk} - O_{pk})^2 \leq 0 \quad 3.9b$$

using cubic error adjustment where:

$$Y_{pk} = \text{Desired output}$$

$$O_{pk} = \text{Network output}$$

$$T = (Y_{pk} - O_{pk})^2$$

7. Calculate the error term for the hidden units, through applying Modified δ_{pk}^o also Modified

$$\delta_{pj}^h = f^h_j(\text{net}_{pj}^h) \cdot (\sum_{k=1}^M \text{Modified } \delta_{pk}^o \cdot W_{kj}^o) \quad 3.10$$

δ_{pk}^o is signal error term for the k th output unit, δ_{pj}^h is signal error term for the j th output unit.

8. Update weights on the output layer with learning rate η

$$W_{kj}^o(t+1) = W_{kj}^o(t) + \mu W_{kj}^o(t) + (\eta \cdot \text{Modified } \delta_{pk}^o \cdot i_{pj}) \quad 3.11$$

9. Update weights on the hidden layer.

$$W_{ji}^h(t+1) = W_{ji}^h(t) + (\eta \cdot \text{Modified } \delta_{pj}^h \cdot X_i) \quad 3.12$$

where η is the learning rate. Repeat steps from step 1 to step 9 until the error $(Y_{pk} - O_{pk})$ was acceptably small for each of the training vector pair. The developed algorithm is stopped when the cubes of the differences between the actual and target values summed over units and all patterns were acceptably small.

3.5.2 Classification and Testing Stage

The classification phase is the last stage of the developed character recognition system. This phase determined the overall performance of the developed algorithm. Three types of classifier were implemented for online character recognition.



- (a) C1 (MGA and MOBP)
- (b) C2 (MGABased MOBP)
- (c) C3 (MGA and MGABased MOBP).

At C1 there was optimization at feature selection and no optimization at classification level, C2 there no optimization at feature selection and optimization at classification level and C3 optimization at feature selection level and optimization at classification level. The performance of each C1,C2 and C3 was evaluated using training time, false recognition and correct recognition accuracy. The objective for using the optimization algorithm at feature selection level is in two parts: (i) feature reduction which improves the convergence time and (ii) elimination of insignificant features which improves the classification accuracy. In this research work modified genetic algorithm was used for optimal feature selection. The extracted features were subjected to optimization which finally yields the optimal feature set. The number of neurons used in the input layer for this classifier was reduced since the number of optimal features was lesser than the complete feature set. Also, the mathematical calculations are minimized because of the reduced size of the weight matrix. MGA was used during the training to solve the problem of local minima in backpropagation and finds a nearly globally optimal set weights on a relatively short time. it also finds values close to the global optimum.

Hence, a significant reduction in the time period was achieved for the weight adjustment of the hidden layer neurons. C3 is better than the other 2 classifiers (C1 & C2). This is because C3 has dual capabilities of removing irrelevant features of character images and easy escape from being trapped in local minima. The characters are classified according to similarity of their shapes and features from the data set collected of different person's handwritten notes using C3. The genetic feature output was projected into image space and stored in the database(template), were used for training and testing the classifier. This was retrieved during testing as test images were projected to image space. In this stage the feature extracted from the desired output was compared with those of actual output of the network for a particular input pattern. The character images were classified as Correct Recognition, False Recognition and Recognition Failure.

4. RESULTS AND DISCUSSIONS

This section discusses the phases of developing online character recognition system. All the algorithms were implemented using C# programming language, 64bits operating system, 8.00GB RAM and run under Windows 8 Operating system on Hp Pavilion i5-3230M CPU @2.60GHz processor and different interfaces representing the phases of the system development were presented and discussed. Experiments were performed with 6200 handwriting character samples (uppercase (A-Z) lowercase (a-z) English alphabet) and digits (0-9) collected from 50 persons using a digitizer and the system was tested with 540 character samples. The performance of the system was evaluated based on training time, recognition time and recognition accuracy. Figure 4.2 shows the result of recognition process displaying recognition status of character A.

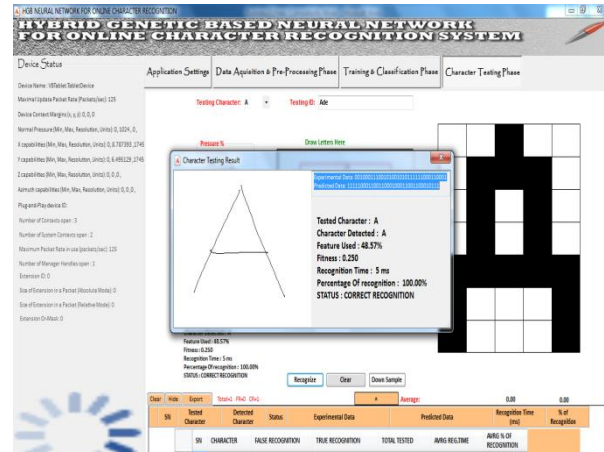


Figure 4.1: Result of recognition process displaying recognition status of character A

Results Discussion

The results of the evaluation were presented in tables and bar charts. Table 4.1 showed the effect of variation of database size and image size on recognition accuracy. While the character recognition between image sizes had an inversely proportionality relation to the recognition accuracy, there is decrease in the variation within the database size and images size (implies as image size increases, the recognition accuracy decreases). Increase in database size had a directly proportional relation to recognition accuracy because of higher database sizes. There was an increase in the variation between the database size and image sizes (i.e. when considering each database size across the different image sizes). However, rate of increment in recognition accuracy with respect to database size was considerably small.

Table 4.1: Effect of variation of Database size and Image size on Average Recognition Accuracy

| Database Size | 5 by 7 Image Size | 10 by 14 Image Size | 20 by 28 image |
|---------------|-------------------|---------------------|----------------|
| 1240 | 97.04 | 96.36 | 94.81 |
| 2480 | 97.41 | 96.95 | 95.37 |
| 3720 | 98.15 | 97.67 | 96.45 |
| 4960 | 98.52 | 98.22 | 97.07 |
| 6200 | 99.44 | 98.77 | 97.63 |

Table 4.2 showed the effect of variation of Database size on Average recognition Accuracy for three classifiers C1, C2 and C3. The recognition accuracy of C3 for database size 1240 to 6200 reveals an increase in proportion and higher than the other classifiers.

Table 4.2: Effect of variation of Database size on Average Recognition Accuracy for three classifiers

| Database size | C1 | C2 | C3 |
|---------------|-------|-------|-------|
| 1240 | 95.74 | 96.30 | 97.04 |



| | | | |
|------|-------|-------|-------|
| 2480 | 96.11 | 96.67 | 97.41 |
| 3720 | 96.67 | 97.39 | 98.15 |
| 4960 | 97.59 | 98.15 | 98.52 |
| 6200 | 97.96 | 98.52 | 99.44 |

Three classifiers were tested in this work. From Table 4.3, it was observed that recognition accuracy of C2 had better results than C1, while C3 with feature selection was superior to C2. A relative improvement in classification accuracy was obtained for the C3 over the other classifiers which were evident from Table 4.3 results. Since the insignificant features were eliminated, the accuracy has been relatively increased. Table 4.4 lists the overall classification accuracy of the classifiers. It was also evident from the table that the C3 was superior to other classifiers in terms of classification accuracy.

Table 4.3: Performance Evaluation of Classification Accuracy of the Three Classifiers

| Character Samples | C1 | | | C2 | | | Developed System (C3) | | |
|-------------------|--------|--------|--------|--------|--------|--------|-----------------------|--------|--------|
| | CR (%) | FR (%) | RF (%) | CR (%) | FR (%) | RF (%) | CR (%) | FR (%) | RF (%) |
| 1240 | 95.7 | 3.9 | 0.4 | 96.3 | 3.3 | 0.4 | 97.1 | 2.6 | 0.4 |
| 2480 | 96.1 | 3.7 | 0.2 | 96.7 | 3.2 | 0.2 | 97.4 | 2.6 | 0.0 |
| 3720 | 96.7 | 3.3 | 0.0 | 97.4 | 3.0 | 0.0 | 98.2 | 1.7 | 0.2 |
| 4960 | 97.6 | 2.4 | 0.0 | 98.2 | 1.9 | 0.0 | 98.5 | 1.5 | 0.0 |
| 6200 | 99 | 2.1 | 0.0 | 98.5 | 1.5 | 0.0 | 99.4 | 0.6 | 0.0 |

Table 4.4 represents performance evaluation of the developed system with related works in literature. It can be deduced from the table that system developed was higher than Fenwa *et al* (2012). The best recognition performance was achieved in the developed system because genetic algorithm was used to optimize the optical backpropagation with 99.44% recognition accuracy. No recognition failure was recorded, which was an indication that the developed system was able to recognize characters irrespective of the writing styles.

Table 4.4: Performance Evaluation of the Developed System with Related Works

| Author | Feature Used | Method Applied | Image Used | %Accuracy |
|-----------------------------|--------------|----------------|-----------------|-----------|
| Yeremia <i>et al</i> (2012) | - | SBPN, GA | Characters | 90.77 |
| Padhi <i>et al</i> (2012) | Zoning | SBP, SGA | Odia Characters | 94 |
| Fenwa <i>et al</i> | StrokeNo | MCPN, | Characters | 99 |

| | | | | |
|------------------------|---------------------------------------|---------------|---------------------|-------|
| (2012) | Counter pixel, Projection & Zoning | MOBP | & Digits | |
| Developed System(2015) | StrokeNo projection, Moments & Zoning | FS, MGA, MOBP | Characters & Digits | 99.44 |

5. CONCLUSION AND FUTURE WORK

A system for online character recognition which reduced the recognition failure was developed using hybrid of structural and statistical features used for extracting features from character images. MGA was used for feature selection to reduce the feature space which enhanced recognition accuracy and reduced training time. Three types of classifiers were used. C1, C2 and C3 were implemented for character image recognition. This paper investigates the necessity for optimization algorithms to enhance the performance of the classifiers. The results obtained at the recognition accuracy for the developed system signified a better accuracy to other classifiers. Hence, an optimization technique was highly essential irrespective of the classifiers used. Thus, the application of MGA for performance improvement of the neural classifiers was explored in the context of character recognition. Finally, irrespective of the modifications and the system used, this paper highlights the significance of optimization algorithm for accurate and better character recognition systems. No recognition failure was recorded and 99.44% recognition accuracy was obtained. Future work could be geared towards investigating the performance of other optimization techniques on optical backpropagation neural network. The System should be extended to cursive characters, word recognition and local languages recognition.

6. REFERENCES

- [1] Agnihotri, V. P. (2012): "Offline Handwritten Devanagari Script Recognition", Information Technology and Computer Science, (8): 37-42. (<http://www.mecs-press.org>)
- [2] Ayyaz, M. N., Javed, I. and Mahmood, W. (2012): "Handwritten Character Recognition Using Multiclass SVM Classification with Hybrid Feature Extraction", Pak. J. Engg. & Appl. Sci. 10: 57-67.k
- [3] Fenwa, O D, Omidiora, E, O and Fakolujo, O. A. (2012): "Development of a Feature Extraction Technique for Online Character Recognition System", Innovative System Design and Engineering ISSN 2222-2871(Online) 3(3):10-23
- [4] Fenwa, O. D., Omidiora, E, O., Fakolujo, O. A., Ajala, F. A., Oke, A. O. (2012): "A Modified Optical Backpropagation Neural Network Model in Online Handwritten Character Recognition System", International Journal of Computer Application 2 4(2): 190-201.
- [5] Haykin, S. (2003): "Neural Networks: A comprehensive Foundation", PHI New-Delhi, India
- [6] Ibrahim A. Adeyanju, Olusayo D. Fenwa, Elijah O. Omidiora(2014): "Effect of Non-image Features on recognition of handwritten alpha-



- numeric characters" *International Journal of Computers and Technology(IJCT)*.13(11):5155-5161.
- [7] Jumanal Shilpa and Holi, Ganga (2013): "On-line Handwritten English Character Recognition Using Genetic Algorithm" *International Journal of Computer Trends and Technology (IJCTT)*. 4(6): 1885-1890.
- [8] Muhammad, F. Z., Dzulkifli, M. and Razib, M. O. (2006): "Writer Independent Online Handwritten Character Recognition Using a simple Approach", *Information Technology Journal* 5(3): 476-484.
- [9] Omidiora E. O., Oyediran, G. O., Olabiyisi, S. O. and Arulogun, O. T. (2008): "Classification of Soils of Central Western Nigeria using Neural Network Rule Extraction and Decision Table, *Agricultural Journal* 3(4):305-312
- [10] Omidiora, E. O., Oladipo, O., Oyeleye, C. A and Ismaila W. O. (2013) A Study of Genetic Principal Component Analysis (GPCA) in feature extraction and recognition of face images, *Journal of Computer Science And Engineering* Vol. 19, issue 1.
- [11] Omidiora, E. O., Adeyanju, I. A. Oladipo., Fenwa, O. D. (2014) "Comparison of Machine Learning Classifiers For Recognition of Face Images", *Journal Of Computer Science And Engineering* 19(1):1-5
- [12] Padhi, D (2012): "Novel Hybrid Approach for Odia handwritten Character Recognition System", *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(5):150-157
- [13] Ranpreet, K. and Singh, B. (2011): "A Hybrid Neural Approach for Character Recognition System", (*IJCSIT*) *International Journal of Computer Science and Information Technologies*, 2 (2):721-726.
- [14] Razzak, M. I., Hussain, S.A. and Mirza, A.A (2012): "Bio-Inspired Multilayered and Multilanguage Arabic Script Character Recognition System", *International Journal of Innovative Computing, Information and Control* . 6(4):2681-2691.
- [15] Yerima Hendy, Niko Adrianus Yuwono, Pius Raymond and Widodo Budiharto (2013): "Genetic Algorithm and Neural Network for Optical character recognition" *Journal of computer science* 9 (11): 1435-1442.
- [16] Sutojo, T. E. Mulyanto and V. Suhartono, 2011. *Keedarsan buatan . ANDI, Yogyakarta*