Performance Evaluation of Neural Networks in Bangla OCR

Sanjit Kumar Saha
Lecturer
Department of Computer Science & Engineering,
Jahangirnagar University, Savar,
Dhaka-1342, Bangladesh

Md. Shamsuzzaman
Lecturer
Department of Computer Science & Engineering,
Begum Rokeya University,
Rangpur-5400, Bangladesh

Dulal Chakraborty
Assistant Professor
Department of Information and Communication Technology,
Comilla University, Comilla,
Bangladesh

Md. Mahfuzur Rahman
Assistant Professor
Department of Computer Science & Engineering,
Comilla University, Comilla, Bangladesh

Md. Al-Amin Bhuiyan, PhD
Professor
Department of Computer Science & Engineering,
Jahangirnagar University, Savar, Dhaka-1342,
Bangladesh

ABSTRACT
Optical Character Recognition is of great interest in machine learning and computer vision. Recognition of Bangla Character is a fast forwarded leap to this journey. Neural Network is the field of study in computer learning and its emerging day by day. Time and accuracy is the first concern in learning by machine. Many research works have been accomplished in recognizing Bangla text (both hand written and printed) to achieve high accuracy rate. Neural network is not out of this research work. Back Propagation Neural Network (BPNN) is one of the mostly adopted neural network methodologies in learning and training OCRs. In this research, a comparison is asserted between BPNN and BPNN+BAM (a hybrid network). The hybrid network cuts down the no. of iterations in training the characters awfully in comparison with BPNN. Various number of (2, 4, 6) training images are considered to get the image feature matrix in feeding to the network. Number of iterations and error are observed while the weights are being updated in a optimized level for better recognition of characters with a high accuracy. The iterations in training depends on number of hidden layer used in the network. So, 50% and 70% of hidden layer are used for observation. The iteration decreases more than half of the iteration in BPNN while using BPNN and BAM as hybrid network for dimension reduction of feature matrix.

General Terms
Bangla Character Recognition

Keywords
Bangla Character Recognition, Bangla Text Segmentation, Matra Detection, Neural Network, hybrid Neural Network

1. INTRODUCTION
Bangla scripts are moderately complex patterns. Unlike simple juxtaposition in Roman scripts, each word in Bangla scripts is composed of several characters joined by a horizontal line (called '*Matra*' or head-line) at the top. The concept of upper and lower case (as in English) character is absent here.

There are some basic features or properties of any Bangla printed script:

i.Writing style of Bangla is from left to right with 11 independent vowel and 39 consonant characters.

ii.The concept of upper and lower case (as in English) is absent in Bangla. We may call these characters as basic character as shown in Fig. 1. From Fig. 1 it is noted that most of the characters have a horizontal line at the upper part. This horizontal line is called headline. In Bangla language, we call this line '*Matra*'.

iii. There are some other types of characters used in Bangla dictionary, called suffix-prefix characters. We can call these as type II (suffix-prefix determiner) characters, as shown in Fig. 2.

The Bangla character image database employed for this research contains a total of 128 mainstream characters: 11 of them are vowels, 39 are consonants, 10 are numerals, 62 are composite characters, 6 are suffix-prefix determiner. These are being trained by the backpropagation neural network.


(a) Shorbama

(b) Benjonbamo

(c) Bangla numerals

(d) A few Bangla composite characters

Fig 1. Some Bangla mainstream characters used for images recognition.

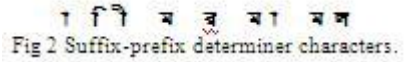


Fig 2 Suffix-prefix determiner characters.

For recognition, each Bangla character image is represented as 24x24 pixels, as shown in Fig. 3. The middle 16x16 (row 5 to 20 and column 5 to 20) pixels contain type 1 or mainstream character of Bangla dictionary. Left 16x4 (row 5 to 20 and column 1 to 4) pixels of 24x24 image occupies left part of type 2 character. The right part of type 2 character occupies 16x4 matrix (row 5 to 20 and column 21 to 24) but right side of main stream character. For top and bottom parts of characters, they contain 4x16 (row 1 to 4 and column 5 to 20) and 4x16 (row 21 to 24 and column 5 to 20) matrices. According to the picture as shown in Fig. 3, we can find:

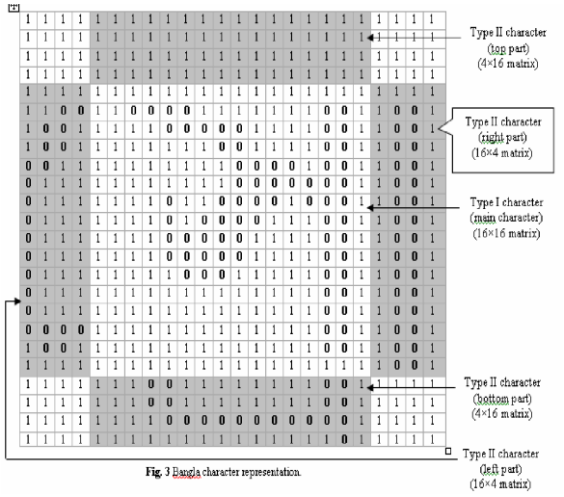


Fig. 3 Bangla character representation.

Table 1. LRTB suffix-prefix parts of a character

Bangla Character	Left	Right	Top	Bottom
ক	1	0	0	0
ত	0	0	1	0
ফ	1	1	0	0
ঐ	0	1	1	0
ক্ত	1	1	1	0
দ্ম	1	0	1	1
ু	0	0	0	1
ূ	0	0	0	1
ৃ	0	0	0	1
ৄ	0	0	0	1
ঁ	0	0	0	1
ঃ	0	0	1	0
়	0	0	1	0

With this rule a truth table can be formed using left (L), right (R), top (T) and bottom (B), LRTB suffix-prefix parts of a character, as shown in **Table I**.

From **Table I** it is evident that the down part of type 2 characters is always zero. So there is no way to find the discrepancy between these 6 characters.

Two exceptional characters are considered that do not follow exactly the type 2 LRTB suffix-prefix rule. To find these characters left and left corner is checked, and left, right and right-corner, respectively.

2. SEGMENTATION

Bangla characters are being segmented by following a sequence of operations: Scanning, skew correction, binary image conversion, and histogram analysis.

2.1 Scanning

Acquisition of a word of Bangla texts are done by using a scanner (Canon CanoScan 600). In this investigation, some popular Bangla words have been employed for image database in BMP type file. During scanning operation, standardization and geometrical normalization (size and direction) of the images have been performed.

During analysis images are being made as 640x480 pixel resolution, as shown in Fig. 4.

Fig 4. Scanning a Bangla word image

2.2 Conversion to Binary Image

The original image is obviously a color image. It is first converted into grayscale image and then binarized because it is easy to match patterns in binary format for character recognition.

2.3 Matra Detection

The 'matra' is described by two rows with the highest number of black pixels on it. The first row of the 'matra' will have highest histogram than the previous white row, as shown in Fig. 5. The second row of 'matra' will also have a highest histogram than the next one although the next row will contain some black pixels than the white one [15]. We represented the histogram of the first row as M_1 and the histogram of the second row as M_2. These two rows will combine the 'matra'. Now, M is a function of M_1, M_2 i.e. $M=(M\ 1\ +\ M\ 2\ )\ /\ 2$. The determination of 'matra' is the key point in the segmentation. Since, depending on the determination of 'matra', the lines will be separated. Then each line will be considered separately to again segmenting as words and then the characters will be segmented from the word.

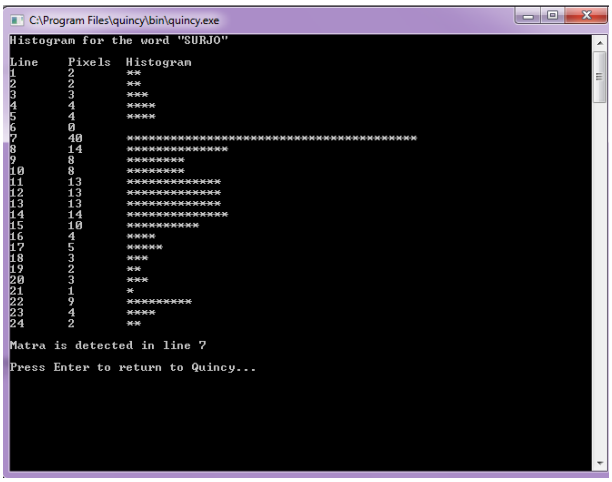


Fig 5. Histogram for the word "Surjo"

As, the highest frequency determines the "matra". The word "Surjo" looks like the following after elimination of "matra":

Fig 6. "Surjo" without Matra

2.4 Baseline Detection

Baseline detection is achieved by a special type of histogram construction by counting black pixel followed by white pixel of a single line, as shown in Fig. 6. If we analyze the lower 30% of this histogram then we can find a row which contains a very high number of pixels then others. That row indicates the baseline of that line. Now we detect all the baselines of the total image and then find the maximum frequency of the baselines and set the final height of the characters.

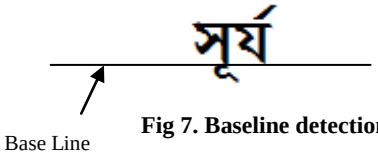


Fig 7. Baseline detection

2.5 Character Segmentation

After segmenting the words we cut a word from 'matra' to baseline and made a horizontal histogram. Analyzing this histogram we can see that each character is separated by one or more white pixels. Using these white pixels we separate the characters. Sometimes some neighboring characters cannot be separated, in these cases we use depth-first-search technique to separate the two characters. After separating a character we save that and also separate the top (upper side of 'matra') and bottom (under the baseline) of that character. The character segmentation process is illustrated in Fig. 7.

Fig 8. Character Segmentation

3. HYBRID NEURAL NETWORK

A hybrid neural network is composed of two neural networks. One is BAM NN and the other is Bidirectional NN. Bidirectional Associative Memory (BAM) has used for dimensional reduction of the feature matrix to make the recognition faster and more efficient. The network architecture for the hybrid neural network is shown in Fig. 8.

After preprocessing, the Bangla characters are engraved in a 16×16 grid and fed into the input layer of the neural network as the input feature vectors or training patterns. The grid patterns are, therefore, represented as vectors of 256 components (if the pixel in the grid is shaded, the vector component is 1 otherwise it is 0). The number of neurons in the hidden layers are 60 (60% of the input layer) for each. Since a total of 128 characters including composite characters are being classified, so the number of neurons in the output layer is 128. With 128 neurons in the output layer, we can represent a 7 bit code (2^7=128) to classify each target output. So the target outputs are 0000000 to 1111111, corresponding to each character.

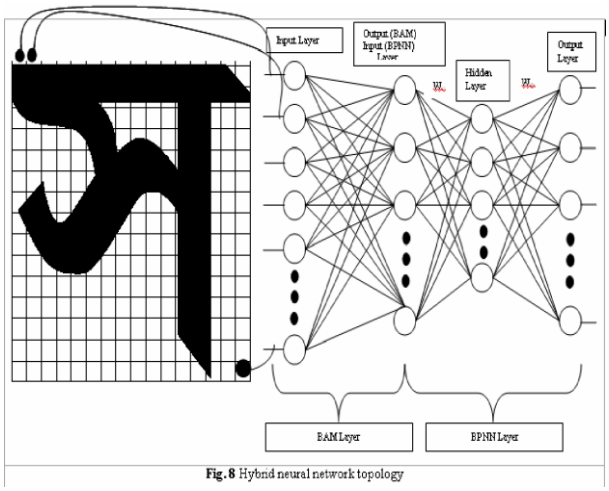


Fig. 8 Hybrid neural network topology

Processing Hybrid NN

After training, we have to do some simple calculation in order to reorganize Bangla words. In the process of reorganization we need not train all the characters of Bangla dictionary. We need to train all the type 1 or mainstream characters. But in case of type 2 characters we need to train only 6 bottom layer characters and 2 top layer characters. The rest of the characters can be evaluated using LRTB suffix-prefix rule.

4. EXPERIMENTAL RESULTS & PERFORMANCE

The character recognition phase is most important and complicated phase. Hence, the basic character is 16×16 pixels image. Each character has a feature matrix of 256 elements in it. Each element is nothing but binary values (0 and 1).

Hence, in this experiment the number of neurons in input layer is 256, neurons in input layer for BPNN and output layer for BAM is 16, number of neurons in hidden layer is 10, and finally the neurons in output layer is 7. The number of neurons in hidden layer can vary from 50% to 70% of its input neurons.

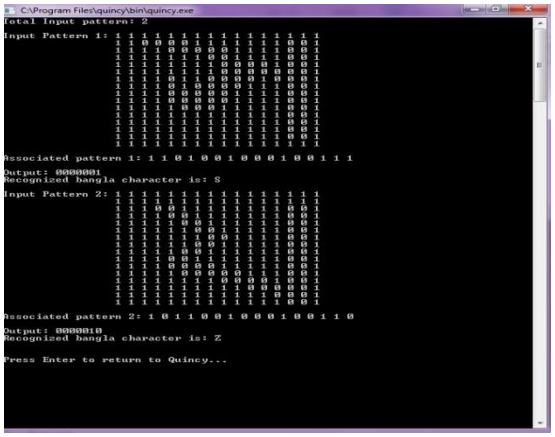
In order to justify the performance of the neural network, various experiments were carried out. All experiments were performed with 5 training images and 2 test images for each character. There was no overlap between the training and test image sets. The back-propagation neural network was trained using the default learning parameter settings (learning rate 0.3, threshold 1) for 75 epochs. Finally, the neural network was used to recognize characters separately. Fig. 9 shows the snapshot of the program output.

Now, we can get the resultant word সূর্য as follows:

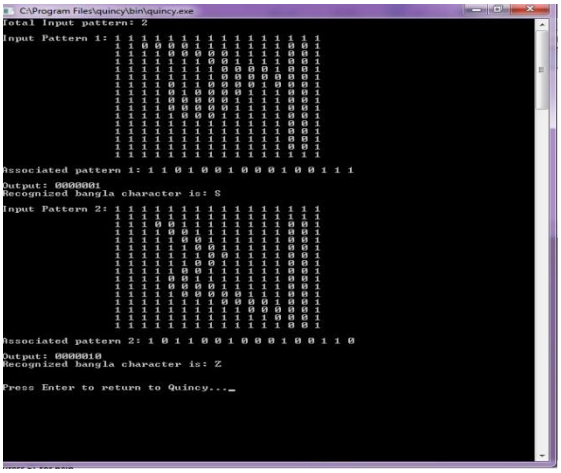
From the truth tables we can recognize the category II symbols in the testing word. Hence, ূ and ্.

The hybrid network helps us to recognize two basic characters in the test word. After the recognition of these result is স

and র as the two basic characters.



(a) Recognition using 70% hidden layer



(b) Recognition using 50% hidden layer

Fig 10. Recognition of two test images (স and ঘ)

Sample input: জোনাকি
Output: জো না কি

Error Minimization

In the training process, the program executed until the error comes to a minimum level. We can show how the error decreases per iteration by the following two graphs (Graph I & Graph II). Hence, at first we a task is implemented by using the BPNN algorithm and secondly, the BAM and BPNN algorithm is merged to train and recognize the Bangla OCR. The results is observed and found that, using hybrid network (BAM and BPNN) rather than BPNN takes less iteration to train and less time to recognize characters.

Anumber of training input patterns (2, 4, 6) is taken and an error level (suppose 0.001) to stop the training process. The no. of iterations depend on the no. of hidden layer and also on what algorithm we used for the training process. Hence, when 2 characters is taken as training image the number of iterations are 39518 and 7660 for BP decreases to 3372 and 2477 in hybrid network, for 4 characters training image the number of iterations are 75652 and 39518 for BP decreases to 11000 and 9876 in hybrid network, for 6 characters training image the number of iterations are 150452 and 125013 for BP decreases to 52310 and 32000 in hybrid network in 50% hidden layer and 70% hidden layer respectively.

Table 2. Iterations and Error (50% hidden layer and 2 characters)

Iterations	Error BP with BAM	Error BP
1	2.75381	4.764672
500	1.0000238	3.333325
1000	0.9999932	3.134522
1500	0.5523913	3.00403
2000	0.499992	2.99914
2500	0.49792	2.980321
3000	0.4888012	2.85321
3372	0.0009023	2.53421
5000		2.01103
10000		1.99931
15000		1.521006
20000		0.553487
25000		0.533332
30000		0.51783
35000		0.50021
39518		0.000121

Table 3. Iterations and Error (70% hidden layer and 2 characters)

Iterations	Error BP with BAM	Error BP
1	1.663217	4.652317
500	0.0987382	3.433321
1000	0.0821432	2.7221468
1300	0.00706421	1.52759431
1500	0.0056879	1.49912
2000	0.00205162	1.499123
2477	0.0009143	1.493113
2600		1.4881632
3900		1.481021
5200		1.470103
6500		0..4990123
7660		0.001321

Table 4. Iterations and Error (50% hidden layer and 4 characters)

Iterations	Error BP with BAM	Error BP
2	2.314675	5.414325
1000	0.623655	5.2232
2000	0.559574	5.119857
3000	0.330815	5.002372
4000	0.119082	4.510815
5000	0.05217	4.212505
6000	0.030912	4.100618
7000	0.021368	4.015081

8000	0.01598	4.00401
9000	0.013088	3.889118
10000	0.011036	3.330201
11000	0.009472	3.102137
12000		3.021368
13000		3.001308
14000		2.71391
15000		2.3126
20000		2.0031
25000		1.51036
30000		1.33338
35000		1.111689
40000		1.00301
50000		1.0011
60000		0.080001
75652		0.009472

Table 5. Iterations and Error (70% hidden layer and 4 characters)

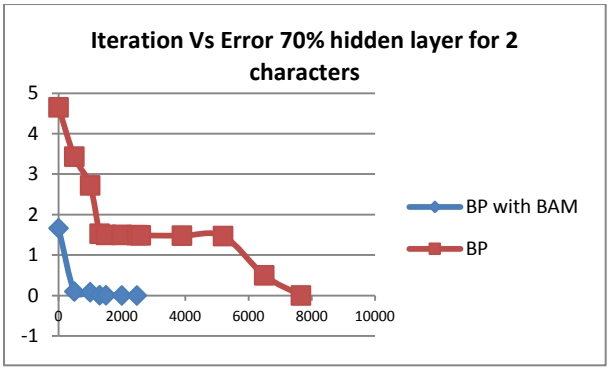
Iterations	Error BP with BAM	Error BP
2	2.155355	5.555355
1000	0.628756	4.628756
2000	0.53733	4.32733
3000	0.521256	4.121256
4000	0.51327	3.51327
5000	0.509651	3.309651
6000	0.507397	3.107397
7000	0.505231	3.005231
8000	0.499972	2.499972
9000	0.01341	2.201341
9876	0.008989	2.008989
10000		1.655355
15000		1.328756
20000		1.13733
25000		0.521256
30000		0.111327
35000		0.010651
39518		0.008397

Table 6. Iterations and Error (50% hidden layer and 6 characters)

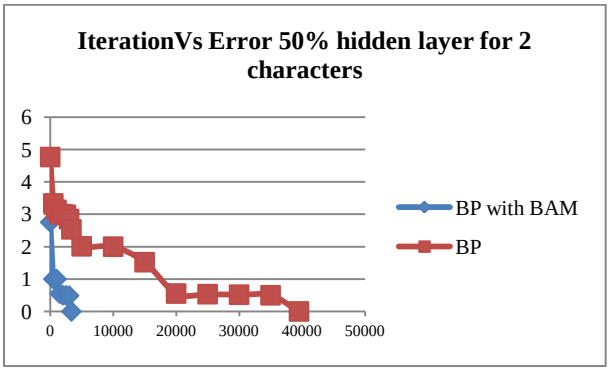
Iteration	Error BP with BAM	Error BP
1	3.116488	6.531415
5000	1.551328	5.121578
10000	1.113236	4.444417
15000	0.990902	4.313242
20000	0.511379	4.112145
25000	0.333232	3.321355
30000	0.113243	3.232576
35000	0.003214	3.21251
40000	0.002312	3.000321
45000	0.001903	2.999931
50000	0.000145	2.99313
52310	0.000101	2.113213
60000		2.00421
70000		2.00004
80000		1.5534
90000		1.5532
100000		1.499212
110000		1.489829
120000		1.00302
130000		0.05321
140000		0.00212
150452		0.000121

Table 7. Iterations and Error (70% hidden layer and 6 characters)

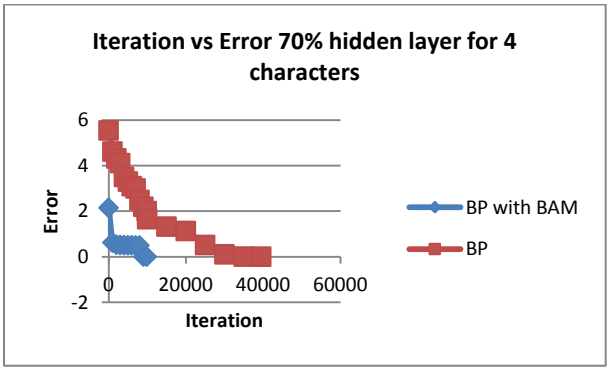
Iteration	Error BP with BAM	Error BP
1	2.7731	5.112275
3000	1.5521	4.475971
6000	1.441419	4.453143
9000	1.000042	3.33342
11000	0.909321	3.310121
13000	0.552132	3.290903
16000	0.33232	3.28821
20000	0.29132	2.555123
25000	0.0089821	2.514551
30000	0.0001323	2.49055
32000	0.0001001	1.52094
40000		1.500013
50000		1.49902
60000		0.55412
70000		0.50051
80000		0.490902
100000		0.3321
120000		0.003293
125013		0.000132



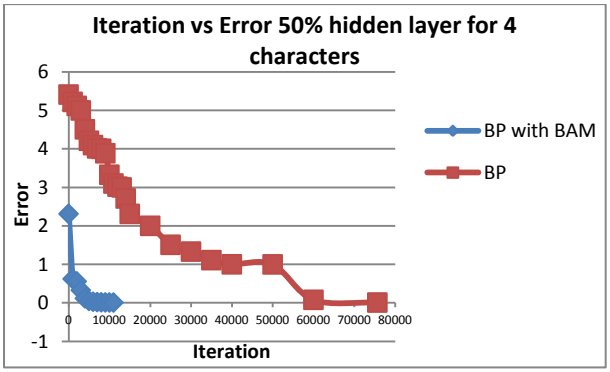
Graph 1. Iteration vs. Error (70% hidden layer and 2 character)



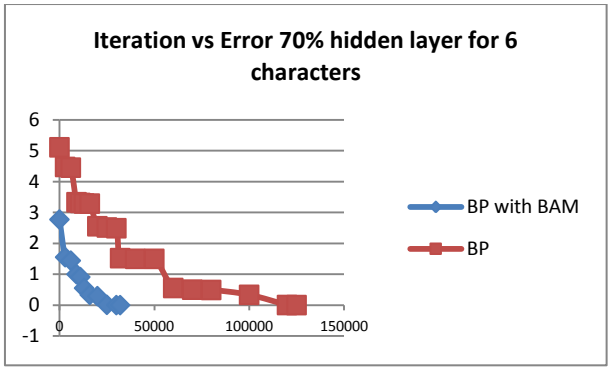
Graph 2. Iteration vs. Error (50% hidden layer and 2 character)



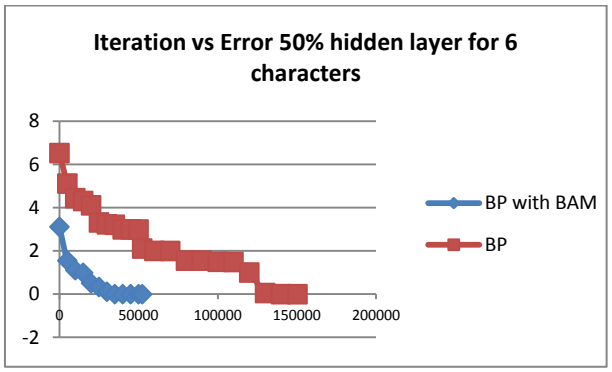
Graph 3. Iteration vs. Error (70% hidden layer and 4 character)



Graph 4. Iteration vs. Error (50% hidden layer and 4 character)



Graph 5. Iteration vs. Error (70% hidden layer and 6 character)



Graph 6. Iteration vs. Error (50% hidden layer and 6 character)

Comparing the performance of two networks as BPNN and Hybrid network decision is taken that the considered hybrid network takes less iteration than BPNN in completion of the training process.

5. CONCLUSION

This research addresses the recognition of Bangla characters using the BPNN and the hybrid (BAM+BPNN) network and make comparison in between this two network. This research work presented a fast and robust system for character recognition which is a combination of a local image sample representation and a hybrid network. The presented system is suitable one for a number of real-time applications. The system is capable of performing a classification in less than half a second for different classes of Bangla characters.

6. REFERENCES

[1] V.A. Kovalevsky, "*Character Readers and Pattern Recognition*", Washington D.C.; Spartan Books, 1968.

[2] Md. Abdus Sattar, Khaled Mahmud, Humayun Arafat and A F M Noor Uz Zaman, "*Segmenting Bangla Text for Optical Recognition*".

[3] S.M. Milky Mahmud, Nazib Shahrier, A.S.M Delowar Hossain, Md. Tareque Mohmud Chowdhury, Md.Abdus Sattar,", *An Efficient Segmentation Scheme for the Recognition of Printed Bangla characters*", Proceedings of ICCIT 2003, pp 283-286.

[4] Md. Al Mehedi Hasan, Md. Abdul Alim, Md. Wahedul Islam "*A New Approach to Bangla Text Extraction and Recognition From Textual Image*", Proceedings of ICCIT, 2005.

[5] F. Kimura, M. Shridhar, Z. Chen, "*Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words*", Proc. of 2nd ICDAR, 1993.

[6] R.G. Casey and G. Nagy, "*Recursive segmentation and classification of composite patterns*", Proc. 61h Intl. Conf. Pattern Recognition, 1982.

[7] About International Mother Language Day. http://en.wikipedia.org/wiki/

[9] Shivani Godara, Dr. Rajeev Gupta *"Comparison of Different Neural Networks for Iris Recognition: A review"* Network and Complex Systems ISSN 2224-610X (Paper) ISSN 2225-0603 (Online) Vol 2, No.4, 2012

[10] Md. Mahbub Alam and Dr. M. Abul Kashem *"A Complete Bangla OCR System for Printed Chracters"* 2010 JCIT, ISSN 2078-5828 (PRINT), ISSN 2218-5224 (ONLINE), VOLUME 01, ISSUE 01, MANUSCRIPT CODE: 100707

[11] Haithem Al-Ani, Ban N,Dhannoon, Huda M. Abass "*Printed Arabic Character Recognition using Neural Network"* Journal of Emerging Trends in Computing and Information SciencesVol. 5, No. 1 January 2014

[12] Md. Alamgir Badsha Md. Akkas Ali "*Handwritten Bangla Character Recognition Using Neura"l Network* International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 11, November 2012

[13] Md. Abul hasnat *"Research Report on Bangla OCR training and testing method"*Conference on Artificial Intelligence and Soft Computing, Marlbella Spain, ACTA press 2004.

[14] Farjana Yeasmin Omee, Shiam Shabbir Himel "*A Complete Workflow for Development of Bangla OCR"* International Journal of Computer Applications (0975 – 8887) Volume 21– No.9, May 2011

[15] Ravina Mithe, Supriya Indalkar, Nilam Divekar *"Optical Character Recognition"* International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-2, Issue-1, March 2013

[16] Ahmed Asif Chowdhury, Ejaj Ahmed, ShameemAhmed, Shohrab Hossain *"OpticalCharacter Recognition of Bangla Characters using neural network: A better approach"*

[17] Uzzal Kumar Prodhan, Md.Farukuzzaman Khan *"Implementation Of Back-Propagation Neural Network For Isolated Bangla Speech Recognition"* International Journal of Information Sciences and Techniques (IJIST) Vol.3, No.4, July 2013

[19] Shamim Ahmed, Mohammod Abul Kashem *"Enhancing the Character Segmentation Accuracy of Bangla OCR using BPNN"* International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064