

Odia-Conjunct Character Recognition using Evolutionary Algorithm

Mamata Nayak^{1*} and Ajit Kumar Nayak²

¹ Dept. of Computer Science and Information Technology
Institute of Technical Education and Research,
S'O'A university
Bhubaneswar, India

² Dept. of Computer Science and Information Technology
Institute of Technical Education and Research,
S'O'A university
Bhubaneswar, India

*Corresponding author's email: mamatanayak [AT] soauniversity.ac.in

ABSTRACT— *Most efforts have been devoted to the recognition of isolated printed and handwritten Odia basic fonts; however recognition of complex conjunct characters is still an area of active research. This paper presents an effective text recognition scheme for Odia conjunct characters. The shape of conjunct characters is inherently complex and imposes great challenges to the researcher group of Odia Optical Character Recognition. The weightiness of this work lies on design and development of an optimized classification method by combining the back-propagation neural network with genetic algorithm to achieve both the training swiftness and accuracy for recognition. We compare the back-propagation neural network (BPNN) and genetic algorithm optimized in artificial neural network (GAONN) and found that GAONN produces more promising result then BPNN. Finally we show that the method yields high accuracy rate and depict the most promising research guidelines in this field.*

Keywords— Odia language; Optical Character Recognition (OCR); Artificial Neural Network (ANN); Genetic Algorithm (GA)

1. INTRODUCTION

Optical Character Recognition (OCR) is the process of detection of text and identification of characters from a printed document image. The electronic text can be edited, formatted for better display, efficient in search and replace of words as well as takes up less storage space. A substantial amount of research work has been carried out on OCR and during last few decades many articles have been published for different national and regional languages [1, 2, 3, 4, 5]. Now a day's several OCR systems for English or Roman language are available [6, 7]. Also there have been many attempts in recognizing printed documents in Indian scripts like Hindi [8, 9], Bangla [10, 11], Devanagari [12], Tamil [13], Telugu [14], Malayalam [15], Urdu [16], Kannada [17], Gurmukhi [18], Gujarati [19], Odia [22]. Odia language is the 10th popular language in India 33rd language of world and is used by around 33 million peoples of Odisha. However a few attempts have been made towards the recognition of conjunct characters of Odia language. In this paper we address the recognition of printed conjunct characters of Odia script.

2.1 Related Work

OCR systems are broadly categorised into four groups, off-line and on-line script recognition in addition to printed and hand written document recognition. Different approaches have been proposed by different researchers to design the OCR. A template matching approach was proposed in [20] where the recognition is made by mapping the already stored templates, multi-layer perception (MLP) and support vector machine (SVM) two different classifiers [21], has been proposed for the recognition of Bangla printed conjunct character.

So far the researcher proposed for Odia language mainly deals with recognition of basic Odia character. Chaudhuri et al. [22] suggested topological and stroke based feature for recognition using tree classifier. Also presence of similar shaped characters in Odia script which affect the recognition is shown by Mohanty and Behera [23]. Pal et al [24] made the character image into 49×49 blocks and extract gradient and curvature feature of those characters and recognition is performed by quadratic classifier. Meher and Basa [25] make the character of uniform size 20×14 and divide all into two groups with respect to their structural features and apply the artificial neural network classification. Also Nayak and

Nayak [26] trained the Tesseract OCR engine at character level but we observed it is of font type specific. Mishra et al. [27] analyze the Odia script numerals using Discrete Cosine Transformation (DCT) and Discrete Wavelet Transformation (DWT) to compute the feature vector and use Back propagation Neural Network for classification.

However none of the work address the recognition of conjunct where as discussed in the study are only recognition for basic characters or numerals of Odia script, so this is our first attempt towards the recognition of conjunct character of the language.

2.2 Motivation

The main challenges in design an OCR for Odia script is to recognise the conjunct (also referred as compound) character, those are made up of by combining two or more basic characters. We draw attention to some typical characteristics of these characters that make the problem difficult and challenging.

- Certain compound characters are very similar in shape and are referred to as confusing characters. Fig. 1 shows a representative set of pairs of confusing characters.

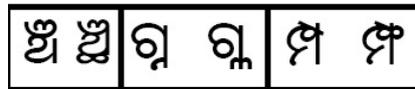


Fig 1: Confusing conjunct character pairs

- It is observed that, most of the symbols attached at the bottom of the base character are not separated physically but actually it is joined physically (touching) to the base character symbol. When a compound character is formed by joining three basic characters its shape becomes very complex, few symbols are shown in Fig.2



Fig 2: Compound characters using three basic characters

The rest of the paper is organized as follows. Proposed model of OCR is explained in part II with briefly describing each phases of it. Detail implementation, experimental analysis and comparison of the results of developed methods with other existing methods in part III. Finally part IV concludes the work with a future direction of research in this area.

2. OCR MODEL

The work here presents a methodology for development of Odia OCR using Neural network based procedure for classification with Genetic Algorithm based approach for optimal selection of weights of the network. The proposed hybrid multilayer neural network based OCR system for recognition of Odia characters is accomplished in two phases: training phase and testing phase. The steps followed to accomplish the task of conversion from image to text are as follows: Preprocessing, Segmentation, Feature Extraction, Classification, Recognition and Post Processing. Block diagram of the working procedure of it is shown in the Fig.3.

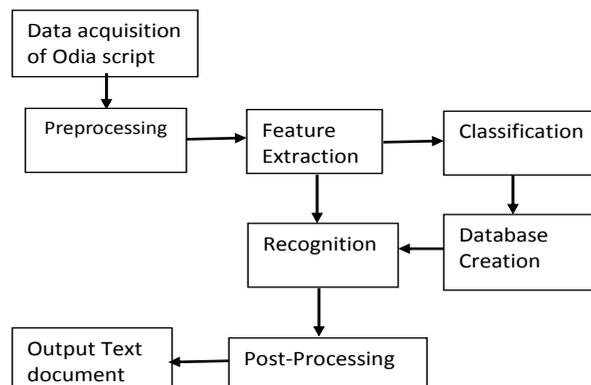


Fig 3: Block diagram of the proposed OCR system

2.1 Pre-processing

To remove the inconsistencies like noise, skew, variation in structure may be present in a document and for further processing of the image, it is required to convert the gray scale image into binary image in the pre-processing stage, referred as digitization.

An $m \times n$ gray scale image with p gray levels can be modeled to a 2D function $f(x, y)$ where $(x, y) \in \{0, 1, \dots, m - 1\} \times \{0, 1, \dots, n - 1\}$ and $f(x, y) \in \{0, 1, \dots, p - 1\}$ called as intensity of (x, y) . A corresponding binary value is obtained for $f(x, y)$ by considering a threshold value $\alpha \in \{0, 1, \dots, p - 1\}$.

$$f(x, y; \alpha) = \begin{cases} 1, & f(x, y) \geq \alpha \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

2.2 Feature extraction

The most significant aspect of OCR for any language is to analyse the image, find some special and distinct characteristics of each symbol to enable clear discrimination of one symbol from other. Feature extraction is the process of finding a set of parameters that uniquely defines and distinguish each symbol from other symbols.

Binary representation of the symbol as feature: In this method, the feature vector for each symbol is computed by considering shape of the binary image. This method is simple and robust as well as powerful technique in which each of the segmented binary symbol is resized (scaled) to a specific size of 15×15 .

It solves two purposes, firstly the segmentation process produces different sized images and this process scales every symbol to one size which is requirement that the feature vector extracted for each symbol should be of same size, secondly the thinning of the symbol is inherent to this process, which is a requirement in pre-processing stage. This value (15×15) is fixed after some hit-and-trial exercise. The obtained 15×15 matrix is then transformed into a column vector of size 225×1 with values considered in row-major order. Fig. 4(a) shows a segmented gray scale image ‘क’, Fig. 4(b) shows the matrix form of scaled binary image where 1’s are marked with gray colour. Fig. 5 is the column vector produced from the scaled binary image which serves as the feature vector of the respective symbol and is used as the input to the neural network as discussed in following sub-section.

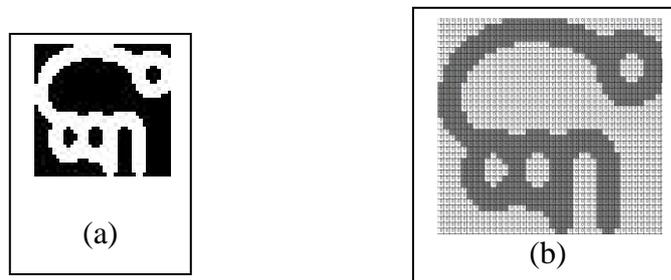


Fig 4(a): Image of symbol ‘kna’ (b) Binary representation after scaling.

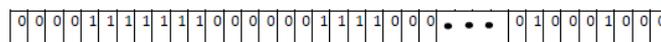


Fig 5: Feature vector obtained for the image shown in Fig.4

Similarly the feature vector of each symbol considered for training is extracted using above process and stored in a matrix. Each column of the final feature matrix contains the feature vector of one symbol.

2.3 Classification

Recognition of characters is an important task in pattern recognition. The complexity of the character recognition problem depends on the character set to be recognized. Multi layer back-propagation neural network [28, 29] is one of the most popular and widely used techniques for character recognition problems. Here we compare two classifiers for conjunct character recognition, first one is a BPNN and second one is an ANN with weight updating using GA (GAONN).

- I. **BPNN:** A four layer (1 Input, 2 hidden and 1 Output) back-propagation neural network is considered at first. Number of neurons used in the input layer is kept same as the number of elements in the feature vector of a symbol and number of output neurons chosen is equals to the total number of unique symbols used for classification (y_1, y_2, \dots, y_n) such that it classifies the character to class i if the value of output neurons are defined by the following

Eq(2). Block diagram of the network model is shown in the Fig.6.

$$y_j = \begin{cases} 1, & j = i \\ 0, & \forall y_{j \neq i} \end{cases} \quad (2)$$

where $i, j = 1 \dots n$

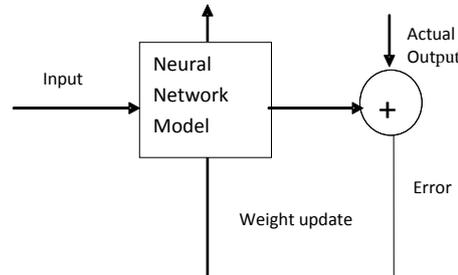


Fig 6: Block diagram of the Back propagation Neural Network

The numbers of neurons are chosen using the formula given in eq. 3 for the two hidden layers following the precondition that, these are never larger than twice the size of the input layer.

$$(N_f) / \left(\frac{r}{2}\right) \times (N_f) \quad (3)$$

where N_f : number of features and r : a random number

For each input, the weights are updated iteratively. The number of iterations is chosen from the number of iteration required to reach an error amount to half of the threshold and 500, whichever comes earlier. Again all training samples are trained repeatedly up to either error becomes less than threshold or it reaches maximum number of iterations. The sigmoid function given in eq. (4) is used as activation function.

$$g(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

The network is fully connected and its weights are updated using the gradient descent method as the adaptive learning function.

$$w^{new} = \alpha \times w^{old} - \eta \frac{\partial E}{\partial w} \quad (5)$$

$$E = \sum (y_i - y_{ti})^2 \quad (6)$$

Where α : Momentum Coefficient, y_i : obtained output for i^{th} input, η : Learning rate, y_{ti} : desired output for i^{th} input, E : Mean Square Error gradient

The parameters set for this architecture are listed as follows:

Momentum Coefficient (α) = 0.8

Learning rate (η) = 0.1800

Threshold = 0.01

Multilayer feed forward neural network is considered as a powerful classifier because of its high computation rate. But weakness of this method is the required time to obtain the best result is too long. To overcome the local minima problem of back propagation due to initial random weight parameters of the network a number of evolutionary algorithms has been tried by many researchers, which improve the performance of the classification at the cost of more execution time. Many evolutionary algorithms have been applied by several researchers to achieve better performance in character recognition for various scripts [30].

II. **GAONN:** Genetic algorithms (GAs) represent a class of the evolutionary algorithms family that has been successfully applied to various optimization processes. It can find the near global optimal solution in a large solution space quickly as of its parallel adaptive nature based on the mechanics of natural selection and natural genetic system. The steps followed for GAONN is shown as a block diagram in Fig.8.

Population Initialization: First step of genetic algorithm process is the Initialization of Population. Each chromosome in the population represents a candidate solution of weights required to train the multilayer neural network. Each gene of a chromosome is a real value and represents one weight for the neural network. Therefore, one chromosome consists of three parts. First part of the chromosome represents the weight matrix for input to first hidden layer, second part consists of weight matrix from first hidden layer to second hidden layer, and third part consists of weight matrix from second hidden layer to the output layer. By mathematical notation, the above assumptions can be expressed as follows:

Population size = Number of chromosomes = N

Number of genes in a chromosome = Number of weights required for multilayer neural network and the chromosome is

$$\begin{bmatrix} x_{i,j} \end{bmatrix}_{k \times l} \quad \begin{bmatrix} x_{i,j} \end{bmatrix}_{l \times m} \quad \begin{bmatrix} x_{i,j} \end{bmatrix}_{m \times n}$$

where k : number of input neurons

l : number of neurons in hidden layer1

m : number of neurons in hidden layer2

n : number of neurons in output layer

Number of input characters = I

Each x_i is a real value chosen from the range of (-5, +5) as follows

$$x_i = 10 * \text{rand}() - 5 \quad (7)$$

Crossover: Next, new individuals/chromosomes are created using crossover operation. If X_a and X_b are the chosen parents of which f and g are the fitness values form a child Y using the crossover operator. We have applied Blend alpha crossover (BLX-alpha) operator. It combine two parents X_a and X_b to generate the offspring by sampling a new value in the range $[\min_i - t, \max_i + t]$ at each position i , where as \min_i and \max_i are the smallest and largest value of both parents at the location i and $t = \max_i - \min_i$.

$$\gamma_i = (\min_i - t) + \alpha x (\text{abs}((\max_i + t) - (\min_i - t))) \quad (8)$$

where α is a weighting factor and the value of which is set to 0.25.

Fitness function: Fitness calculation needs to be done before enters into the selection phase. Fitness is a comparison value to determine which chromosome should be eliminated and replaced to improve the recognition rate or to minimize the training error. A block diagram of fitness evaluation is shown in Fig 7.

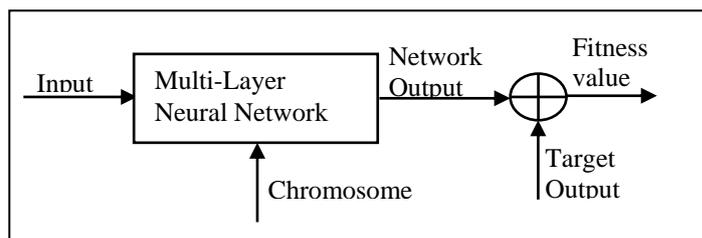


Fig 7: Fitness function evaluation

The error function defined as follows.

$$E_j = \sum_{i=1}^I (T_i - O_i)^2 \quad \text{for } j=1,2,\dots,N \quad (9)$$

where T_i is the target output of the i^{th} training vector and O_i is the output generated for the same.

The objective function used in this work is minimization of the error E for each training input symbol defined as follows.

$$\text{Objective function} = \min(E) \quad (10)$$

Elitism Selection: The idea in this system is to select distinct pair of chromosomes out of the existing chromosomes and newly created chromosomes. The chromosomes are arranged in increasing order according to their fitness values. The Elitism selection operation is applied with set of chromosomes in the arranged set containing parent and children. Block diagram of the frame work followed in this approach is shown in the Fig.8.

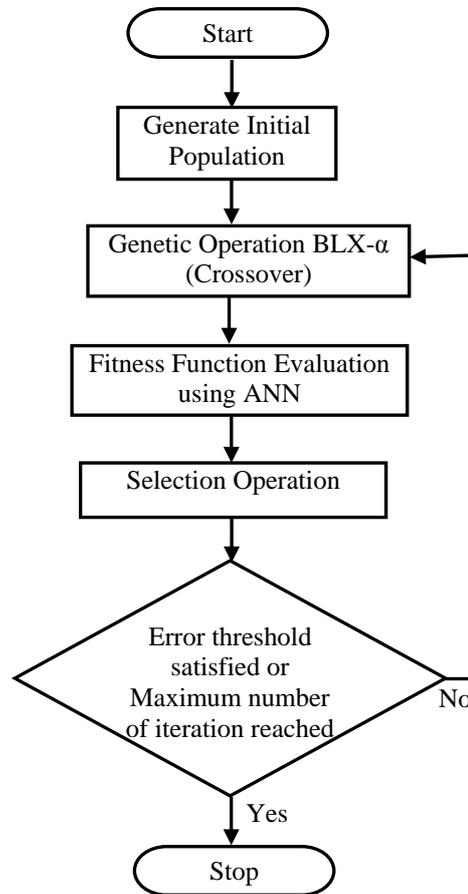


Fig 8: Flow diagram of the proposed approach

The parameters set for this architecture are listed as follows:

- population size = $N=200$,
- individual crossover probability = 0.8,
- BLX alpha operator $\alpha = 0.25$
- individual mutation probability = 0.9,

3. EXPERIMENTAL ANALYSIS

In this section, we present the experiments performed to evaluate the discussed methods. First we implement the Back Propagation Neural Network (BPNN), then Genetic Algorithm optimization Neural Network (GAONN) is implemented with the same set of Odia conjuncts as inputs. The character images in the .jpg format are collected as shown in Column1 of Tab1. To simplify the recognition process, preprocessing is performed to get into its binary conversion and each character image is resized to 15x15 pixels resolution. By reducing the size of image the number of input nodes to the neural network being reduced, that yields less synapses and less weight to be readjusted, so the training time will be quite reduced. The features are now extracted for each character and then a feature matrix for all the input images are created as discussed in Para II.B.

At first BPNN is trained for 1000 iteration. In each of the iteration error is computed as given in Eq. 9 and propagated back. The plot shown in the Fig. 7(a) represents the error rate conversion and it has been observed that the error rate is regular after 500 epochs. Fig. 7(b) represents the number of characters that could not be matched at each of the iteration. It can be observed that till end of the training three characters cannot be trained properly out of 20 character input and 17 characters are trained properly. However, the back-propagation network is not always performs superior and traps in local minima, thus for few symbols this network fails because of non differentiable characters.

Secondly the hybrid algorithm (GAONN) which is trained for all characters once at a time is considered. The

observation shows that it improves the training error convergences as shown in Fig. 7(c) and is stable after 250 epochs for all 20 characters. Fig. 7(d) represents the number of characters that could not be matched at the end of 1000 iterations. However it is shown for only 500 iterations, as after 250 iterations result obtained is constant. Moreover the proposed approach produce high classification accuracy in less training time but till the end of the training one character cannot be trained properly out of 20 training characters.

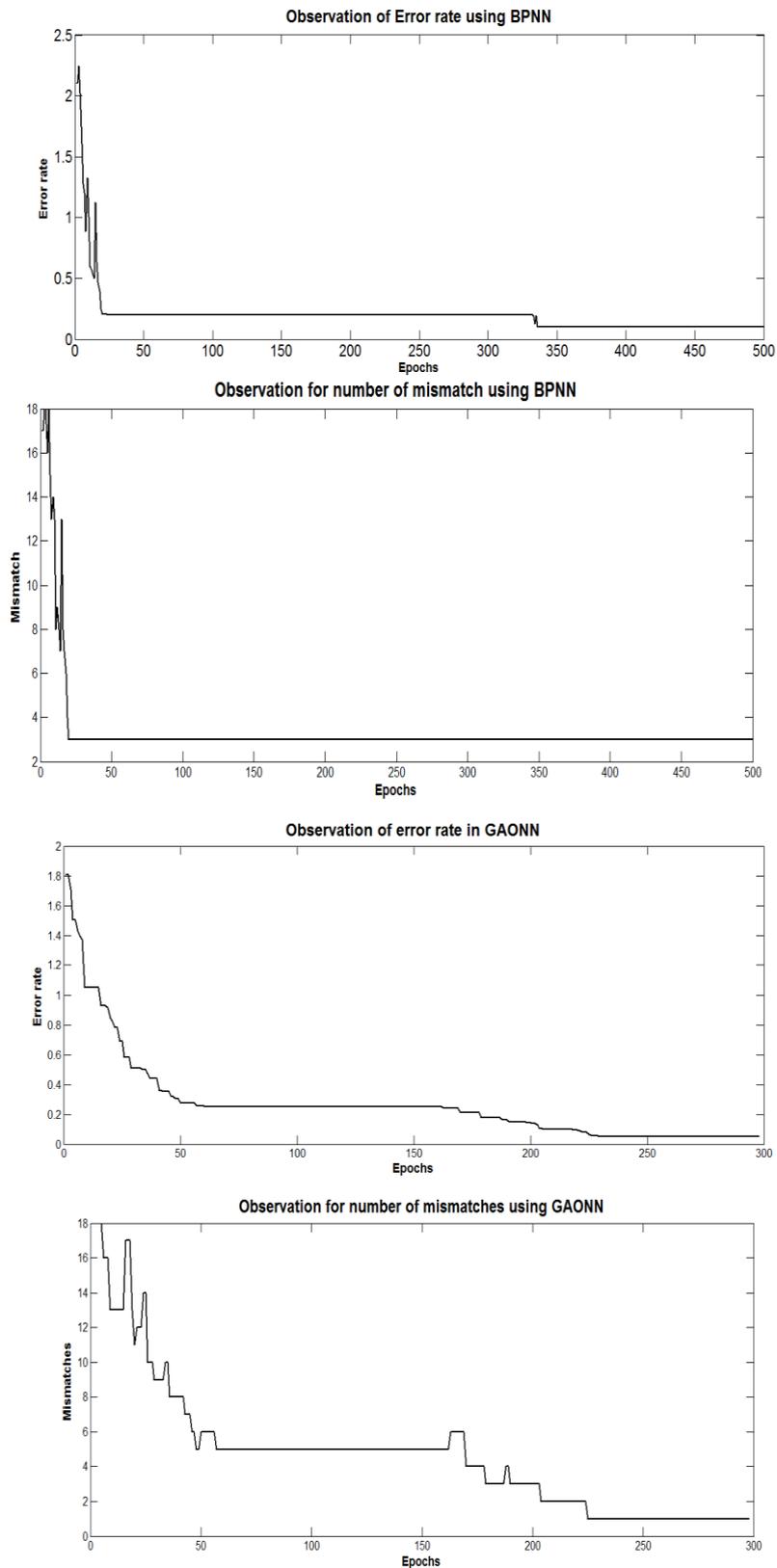


Fig 9: Training error rate curves and accuracy rate curves of BPNN and GAONN for all training samples (a), (b), (c) and (d) respectively

The final weight values obtained from both the approaches (BPNN and GAONN) were considered for evaluation of the performance over the conjunct Odia symbols respectively. In the training stage, we selected 20 training samples of conjunct characters listed in the Tab.1. Afterword, we collect 10 samples of each class from different sources such as italic type, bold type and scanned, printed documents. A comparison of character level rate of recognition for all of testing set symbols as well as the testing accuracy achieved during the experiment is shown in the Tab 1. We observed that the average recognition rate achieve using BPN and GAONN are 93.95% and 95.9% correspondingly.

Tab 1: Comparison of recognition

Sl. No.	Odia Conjunct character	Recognition Accuracy	
		BPNN	GAONN
1	କ (nka)	100	100
2	ଖ (nkha)	95	97
3	ଙ (nga)	100	100
4	ଢ (ngha)	98	100
5	ଞ (nca)	90	92
6	ଝ (ncha)	85	88
7	ଞ (nja)	95	95
8	ଝ (njha)	85	90
9	ଟ (nta)	98	98
10	ଠ (ntha)	95	95
11	ଡ (nda)	90	95
12	ଢ (ndha)	95	95
13	ଣ (nnta)	95	100
14	ଞ (nntha)	95	98
15	ତ (nnda)	100	100
16	ଥ (nndha)	98	100
17	ମ (mpa)	95	100
18	ମ (mpha)	85	87
19	ୟ (mba)	92	95
20	ୟ (mbha)	93	93

Therefore in term of accuracy the performance achieved using optimized neural network genetic algorithm (GAONN) is superior from back-propagation neural network (BPN). Besides its swiftness, the time required to train the network for recognition also improved for conjunct and complex character.

4. FUTURE DIRECTION AND CONCLUSION

As an extension to this work, further research can be done in two major directions as follows:

At first, the method (GA) can be experimented with various crossover operators and evaluated for every combination of crossover to select the best one. We expect that use of such kind of combination of operators may yields good results for the conjunct characters of Odia language. Next, adaptive changing of the parameters like crossover probability, mutation probability may helps further to improve the efficiency.

In the other direction it needs to develop an application that recognises a whole word or even a whole sentence at a time instead of a character. As compared to English language the character set of Odia language is larger enough. Therefore, segmentation techniques can also be further improved which reduce the number of training samples. Next, the OCR for Odia language may be developed to recognize characters efficiently and accurately for different font types and

font sizes.

The proposed binary feature extraction method used along with genetic algorithm optimised neural network classification technique has shown promising results for character level recognition of Odia conjuncts. So the proposed method may be applied successfully to recognize printed Odia characters.

5. REFERENCES

- [1] P. K Charles, V.Harish, M.Swathi, CH. Deepthi, “A Review on the Various Techniques used for Optical Character Recognition”, International Journal of Engineering Research and Applications (IJERA), No. 1, Vol. 2, pp.659-662, .2012
- [2] U. Pal, B.B. Chaudhuri, “Indian script character recognition: a survey”, The Journal of Pattern Recognition, No. 37, pp. 1887–1899, 2004.
- [3] D. Ghosh, T.a Dube, and A. P. Shivaprasad, “Script Recognition—A Review”, IEEE Transaction On Pattern Analysis And Machine Intelligence, No. 12, Vol. 2, 2010
- [4] A. K. Sadasivana, T.Senthilkumar, “Automatic Character Recognition in Complex Images”, International Conference on Communication Technology and System Design 2011, Elsevier Procedia Engineering
- [5] P. Xiu, H. S. Baird, “Whole-Book Recognition”, IEEE Transaction on Pattern Analysis and Machine Intelligence, No. 12, Vol. 34, 2012
- [6] M. Cheriet, M. E. Yacoubi, H. Fujisawa, D. Lopresti, G. Lorette, “Handwritten Recognition Research: twenty years of achievement and beyond”, Pattern Recognition 42 (12) (2009) 3131–3135.
- [7] G. Nagy, “Twenty Years of Document Image Analysis in PAMI,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp. 38-62, Jan. 2000.
- [8] B.Indira, M.Shalini, “Classification and Recognition of Printed Hindi Characters Using Artificial Neural Networks”, I.J. Image, Graphics and Signal Processing, No. 6, pp. 15-21, 2012
- [9] D. Yadav, S. Sánchez-Cuadrado and J. Morato, “Optical Character Recognition for Hindi Language Using a Neural-network Approach”, J Inf Process Syst, Vol.9, No.1, 2013.
- [10] S. Bag, G. Harit, P. Bhowmick, “Recognition of Bangla compound characters using structural decomposition”, Pattern Recognition, No.47, pp.1187–1201, 2014.
- [11] N. Das, R. Sarkar, S. Basu, P. K.Saha, M. Kundu, M. Nasipuri, “Handwritten Bangla character recognition using a softcomputing paradigm embedded in two pass approach”, PatternRecognition, No. 48, pp. 2054–2071, 2015.
- [12] N. Sankaran, A. Neelappa and C. V. Jawahar, “Devanagari Text Recognition: A Transcription Based Formulation”, 12th International Conference on Document Analysis and Recognition, pp.678-682, 2013.
- [13] R.J. Kannan, “A comparative study of optical character recognition for Tamil script”, European Journal of Scientific Research, No. 35 (4) , pp.570–582, 2009.
- [14] V. Lakshmi, C. Patvardhan, “A high accuracy OCR system for printed Telugu text”, Tencon Conference on Convergent technologies for the Asia-Pacific Region, Vol. 2, pp. 725 – 729.
- [15] M.A. Rahiman, M.S. Rajasree, “Printed Malayalam character recognition using back-propagation neural networks”, in: International Advance Computing Conference, 2009, pp. 197–201.
- [16] S. Naz, K. Hayat, M. I. Razzak, M.W. Anwar, S. A. Madani, S.U. Khan “The optical character recognition of Urdu-like cursive scripts”, Pattern Recognition, No.47, pp.1229–1248, 2014.
- [17] T.V. Ashwin, P.S. Sastry, “A font and size-independent OCR system for printed Kannada documents using support vector machines, Sādhanā 27 (1) (2002),35–58.
- [18] G.S.Lehal, “A Complete Machine-Printed Gurmukhi OCR System”, Springer Book Chapter, Guide to OCR for Indic Scripts, Advances in Pattern Recognition
- [19] A. A. Desai, “Gujarati handwritten numeral optical character reorganization through neural network”, Pattern Recognition Elsevier, Vol 43, 2010
- [20] U. Garain, B.B. Chaudhuri, “Compound character recognition by run-numberbased metric distance”, International Symposium on Electronic Imaging: Science and Technology, Vol. 3305, pp. 90–97, 1998,
- [21] N. Das, B. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, “Handwritten Bangla basic and compound character recognition using MLP and SVM classifier”, Journal of Computing, No. 2, Vol.2, pp.109–115, 2010.
- [22] B.B.Chaudhuri, U.Pal, M. Mitra, "Automatic Recognition of Printed Oriya Script", pp. 23-34, Sadhana, Vol.27, part 1,2002.
- [23] S. Mohanty, H. K. Behera, “A Novel Approach for Bilingual (English - Oriya) Script Identification and Recognition in a Printed Document”, International Journal of Image Processing (IJIP), CSC Journals, Kuala Lumpur, Malaysia, 175 – 191, No. 2, Vol.4.
- [24] U. Pal, T. Wakabayashi, and F. Kimura, “A System for Off-line Oriya Handwritten Character Recognition using Curvature Feature”, 10th International Conference on Information Technology, IEEE Society, pp.227-230.
- [25] S.D.Meher and D. Basa, “An Intelligent Scanner with Handwritten Odia Character Recognition Capability”, In: proceedings of IEEE Fifth International Sensing Technology(ICST), pp 53-59, 2011

- [26] M. Nayak, A. K. Nayak, "Odia Characters Recognition by Training Tesseract OCR Engine", *International Journal of Computer Applications* (0975 – 8887), pp.25-30, 2013
- [27] T.K.Mishra, B.Majhi, S.Panda, "A comparative analysis of image transformations for handwritten Odia numeral recognition", In : proceedings of IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 790-793, 2013.
- [28] V. Shrivastava and N. Sharma, "Artificial Neural Network Based Optical Character Recognition", *Signal & Image Processing : An International Journal (SIPIJ)*, Vol.3, No.5, 2012
- [29] M. D. Garris, C. L. Wilson, and J. L. Blue, "Neural Network-Based Systems for Handprint OCR Applications", *IEEE Transaction On Image Processing*, Vol. 7, No. 8, 1998
- [30] B. Xue, M. Zhang, and W. N. Browne, "Particle Swarm Optimization for Feature Selection in Classification: A Multi-Objective Approach", *IEEE Transaction On cybernetics*, Vol.34, No.6, pp-1656-1671, 2013.