Optical Character Recognition using Ant Miner Algorithm: A Case Study on Oriya Character Recognition

Bhagirath Kumar
Department of CSE,
National Institute of Science and Technology, Berhampur, Odisha, India

Niraj Kumar
Acumen Consultancy Services Pvt. Ltd., Kolkata, India

Charulata Palai
Department of CSE,
National Institute of Science and Technology, Berhampur, Odisha, India

Pradeep Kumar Jena
Department of CSE,
National Institute of Science and Technology, Berhampur, Odisha, India

Subhagata Chattopadhyay
(corresponding author)
Department of CSE,
Camellia Institute of Engineering (CIE), Kolkata-700129, West Bengal, India

ABSTRACT

Optical Character Recognition (OCR) is one of the challenging areas in the domain of image processing, where the handwritten or printed characters are digitized by using an optical scanner. The image is then analyzed broadly by two methods – (i) matrix space analysis method and (ii) feature space analysis method. Matrix space analysis method takes more memory space and time, compared to feature space analysis. However, it works fine for the scripts in which the strokes are prominent, e.g. English numeric scripts. On the other hand, the feature analysis method is useful where the scripts are complex and having more similarity between the letters in its writing style. Hence, the feature analysis approach is more useful to many of the regional languages. In this paper, we have used the Ant-miner algorithm (AMA) for offline OCR of hand written *Oriya scripts*, popularly known as *Utkal lipi*. The AMA is a rule-based approach. The rules are incrementally tuned during the training. The Oriya language contains more than 50 distinct characters i.e. 12 Swara-varnas (i.e., vowels) and 38 Byanjan-varnas (i.e., consonants) and their composite characters. In this work, for the analysis, we define three types of 'block's as per the writing styles of the scripts. AMA is then tested with four characters from each 'block'. Finally, a character recognition tool has been developed using Matlab for observation and validation.

General terms*:* Ant miner algorithm, Optical character recognition (OCR), Oriya character pattern matching

Keywords*:* Off-line character recognition,Optcal character recgnition, image enhancemen, Oriya scripts, Ant-minor algorithm, Feature space analysis**.**

1. INTRODUCTION

Optical Character Recognition (OCR) is the technique which enables a machine to automatically recognize the characters or scripts written in the users' language. OCR is a process to translate the human readable hand-written characters to machine readable characters. To automate the process of learning the computer system is assisted with the technologies such as neural network based approach [1], where the pattern of the individual scripts are saved in the neural network weights [2]. The property of adaptation of the neural network helps to recognize the characters even in a noisy background [3][4][5]. The Zone centroid method [6][7][8], according to this approach the character image is divided into n number of equal zones and average distance from the zone centroid to each pixel present in the zone is computed. The curvelet transform is the other popular method as per its curvelet coefficients of an original image as well as its morphologically altered versions are used to train separate k–nearest neighbor classifiers [8]. The fuzzy logic-based feature analysis method is another interesting work [9], where the characters are segmented into '*n*' number of segments and the strokes as left curve, right curve, horizontal stroke, vertical stroke and slanted lines of the segments, which are used to represent the character.

Although many works have been done in the different national and regional languages such as Bangla [9][10], Devnagari [11][12], Arabic [13][14], Thai [15], Oriya [16][17], Hindi [18] and many other languages. The study shows that, OCR implementation in Oriya language is more challenging due to its writing style complexity. First, it is rounded in shape [19] hence the strokes may not be prominent. Second, few Oriya script are the mirror image or reverse of the image of others such as '**Dha**' vs '**Ma**' and '**Tha**' vs '**Ga**' hence they are equivalent in its feature space. Third, many scripts are alike in their pattern such as '**Ka**', '**La**', '**Ja**' and '**Dha**', '**Ta**'.

To handle the above difficulties the scripts has been categorized into three different 'block's as described in Table-1. These are (1) Open scripts, (2) Major-loop scripts, and (3) Minor-multi-loop scripts. It has been observed when the scripts are written casually or hurriedly, they violate the 'block' rules and as a result these are hardly or not been recognized. Fig-1 shows the two sample scripts '**Ka**', '**Dha**' and their casually written scripts with their global features. It is observed, although the scripts look alike in terms of global features, they are quite different locally. Hence a simple global feature analysis is not useful. To address this issue, we have considered a combination of global and local features of the scripts. The features are denoted by the 'block's (see table 1).

Table 1: Script 'Blocks' with descriptions

Block	*Name*	*Description*	*Example*
'block' 1	Open scripts	It does not have closed region	**'Da'** **'Dha'** **'Na'** **'Ma'**
'block' 2	Major loop script	Here at least 60% of the space is under one closed region	**'Gha'** **'Pa'** **'Sa'** **'Ya'**
'block' 3	Minor-multi loop script	It contain a one or more smaller closed region	**'Ka'** **'Chha'** **'Ga'** **'Tha'**

Ant Colony Optimization (ACO) algorithm [20] is an iterative process in which a population of simple agents repeatedly construct candidate solutions; This Optimization process has been proposed by *Colomi, Dorigo and Maniezzo* and is based on the foraging behavior of real ants. ACO incrementally generates solution-paths in the space of such components, adding new components to a state. The algorithm converges to an optimal-final solution, by accumulating the most effective sub-solutions [21]. The Ant-Miner algorithm (AMA) [22] is a modified version of the ACO, which is used for classification based on a set of rules. Initially the list of rules is empty and the training set consists of all the training cases. In each iteration of training, it discovers one classification rule. This rule is added to the list of discovered rules. After the training it is ensured that all the training cases are correctly covered by this set of rules.

Rest of the paper is organized as follows. *Section 2* describes the methodology adopted for this study. Results are shown and elaborated in *section 3*. Finally, *section 4* concludes the paper and directs the future extensions of this work.

2. METHODOLOGY

a. Image Preprocessing

The image preprocessing is the set of operations such as acquisition of grayscale image, which includes the elimination of the cover areas, digitization or binarization and thinning of the image. The image thinning is done using connected component analysis. It removes the thickness effect of the pen used for writing. Hence the generated binary image consists of sets of 1's and 0's that represents the ideal pattern for the hand-written character. Few original images and its preprocessed version is shown in fig-3.

b. Feature Extraction (Oriya Character)

As we have mentioned earlier that the Oriya scripts are rounded in shape and hence analyzing the strokes produced while writing is quite difficult. Again few scripts are mirror image and/or rotational image of others such as "**Ma**" is rotated by *180 degree* will be "**Dha**" and when the mirror image of "**Ga**" is rotated will be "**Tha**" so that they are equivalent in the feature space. These are few real challenges which we are addressing in this paper. To avoid the difficulties discussed above we segmented the image space into six equal zones i.e two rows and three columns (refer to fig.1) . The feature matrix consist of a well combination of the global and local (zone based) features. The **global features** are the number of (i) *loops*, number of (ii) *end points*, number of (iii) *horizontal strokes*, number of (iv) *vertical strokes* (v) *angular strokes* and (vi) *aspect ratio i.e. number of pixels on vs total number of pixels* in the image. The **local features** are the number of cross-point with (i) *three* and (ii) *four* connections. Here we also measure the Center of Gravity (CoG) of the cross-points and the end points.

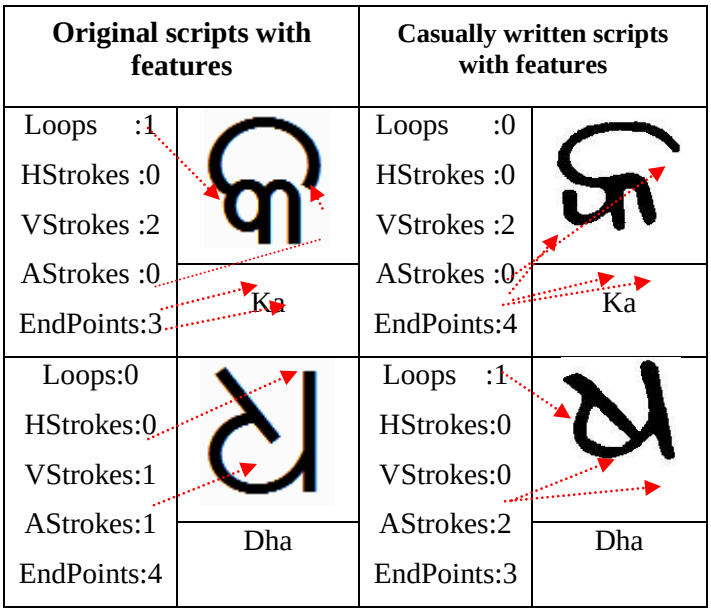

Fig.1: Global features of few common characters.

c. Ant-Miner Algorithm (AMA)

The AMA was proposed by Parpinelli and his Colleagues [23]. It applies the ant colony optimization heuristic for the classification which is used to discover an ordered list of classification rules. To discover classification rules of the form: *IF (Attribute$_1$= val$_1$ AND Attribute$_2$= val$_2$ AND...Attribute$_m$= val$_m$) THEN (predicted class)*. In our case all the different rules can have equal number of terms in their antecedent (the *IF* part). The consequent of a rule is a predicted class (the *THEN* part), i.e. the value that the rule predicts for the class attribute when an example satisfies the conjunction of terms in the rule antecedent. Classification rules have the advantage of representing knowledge at a high level of abstraction.

d. Pheromone Initialization

The pheromone values are initialized for each character as per the followings at time **t =0**.

$$\tau_{ij} = \frac{1}{\sum_{i=1}^{a} b_i} \qquad \ldots (1)$$

Where ***a*** is the total number of attributes, **i** is the index of an attribute, **j** is the index of a value in the domain of attribute **i**, and b_i is the number of values in the domain of attribute **i**.

e. Pheromone Updation

The pheromone levels are updated for all terms in a rule by an *Ant* and it is based upon the quality **Q** of that rule. The quality

value is measured as "**sensitivity × specificity**", which is defined as follows:

$$Q = \frac{TP}{TP + FN} \times \frac{TN}{FP + TN} \quad \ldots \quad (2)$$

Here TP denotes True Positive, FN means False Negative, TN determines True Negative, FP denotes False Positive and TP ÷ (TP + FN) is the **sensitivity**, TN ÷ (FP + TN) is the **specificity**. Once a rule has been accepted, the amount of pheromone increment to be done to each of the terms in that rule, which is determined by the following formula.

$$\tau_{ij}(t+1) = \tau_{ij}(t) + (\tau_{ij}(t).\delta) \quad \ldots (3)$$

Where, the δ is set based upon the rule quality **Q**, calculated as above. The $\mathbf{R_t}$ is the set of rules generated for a single script during training and $\mathbf{R_{Best}}$ stores the best rule for the script after the completion of the training. Since in here we deal with a static scenario, the pheromone evaporation step is merely skipped to reduce the complexity.

f. Character Recognition using Ant-Miner

A GUI based software tool has been developed using Matlab 2010a, ver.7.10.0.499 for the Optical Character Recognition, especially for handwritten Oriya scripts. The PC specification is as follows:

- MS Windows XP Professional version 2002 Service pack 3
- Pentium ® Dual-Core CPU with 2 GB RAM
- Processor speed 3.00 GHz.

The handwritten scripts are collected from ten users, 4 sample per scripts from each user. Hence 40 scripts are used for the training of each character. The system is trained with 4 characters from each 'block' (refer to table 1) of the scripts, as we mentioned there are 3 'block's, hence the system is trained with 12 Oriya characters.

Fig.2 shows the steps followed during training using AMA algorithm. The rule set $\boldsymbol{R_t}$ is initialized to null. For each Oriya character, 40 handwritten characters are used for training. The handwritten character images ($\boldsymbol{c_i}$) are collected using a 200 dpi digital scanner. Hence **N** = 40 i.e., the maximum number of characters used during training. For each character the ant system is initialized. Using the feature extraction process as discussed above, the features of a $\boldsymbol{c_i}$ is prepared. An ant uses the features of $\boldsymbol{c_i}$ to prepare its rule ***r*** used to recognize the character. If the rule set $\boldsymbol{R_t}$ is empty or the rule ***r*** is new rule then it is added to the $\boldsymbol{R_t}$ otherwise the pheromone value is updated. This process is carried out for **N** times and the rule base $\boldsymbol{R_t}$ is prepared. The $\boldsymbol{R_{Best}}$ is selected based upon the rule with highest pheromone value. The τ_{ij} is the database that contains the best rules for each trained $\boldsymbol{c_i}$. During training each time the character identified correctly is recoded as TP, the character identified wrongly is recoded as FP, each time the system fail to recognize a character is treated as FN, the TN value is set to *zero* since there is no invalid symbol in the training set.

During testing the $\boldsymbol{c_i}$ represents test character image. The features of $\boldsymbol{c_i}$ are calculated and the rule base is searched for a rule with the features of the $\boldsymbol{c_i}$. Each time an equivalent rule is found then it percentage of match is measured. If the percentage of match greater than the **threshold-value (*θ*)**, which is set by the user (85% in this case) and more than the previous matched value then the search index is updated. If no script is found with a matching value greater than the threshold the system shows unknown symbol.

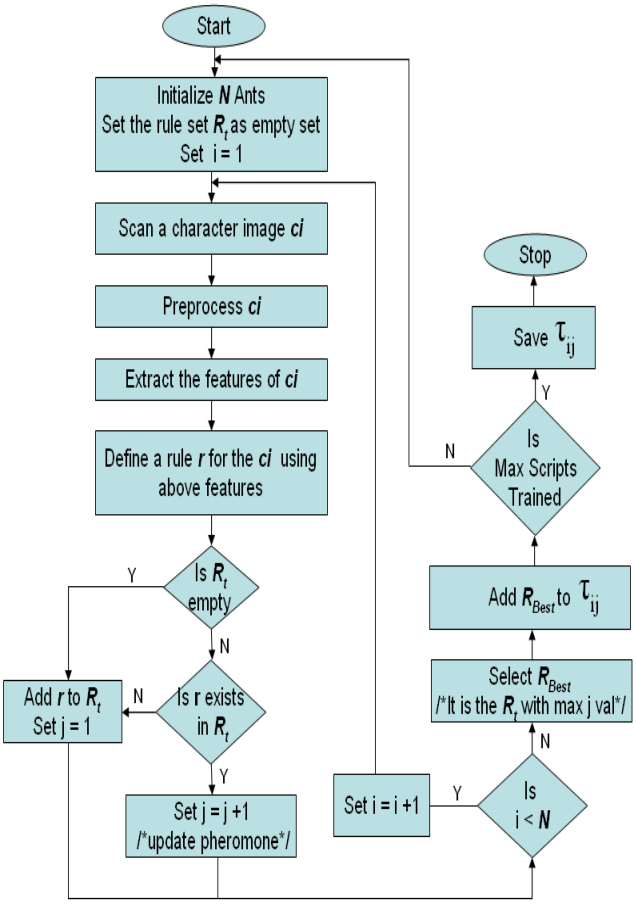

Fig.2 OCR Training using Ant-miner Algorithm.

3. RESULTS AND DISCUSSION

The contribution of this paper lies not only in the application of AMA in character recognition, but also showcasing the whole process in a software system, which is one important way to display the results on-line and therefore has been developed in several research fields [24-30]. Experimental results are shown in this section. Fig-3 shows the binary image equivalent of the handwritten character after eliminating the background noise [31]. It is useful to enhance the quality of the script irrespective of the thickness of the pen used for writing the script.

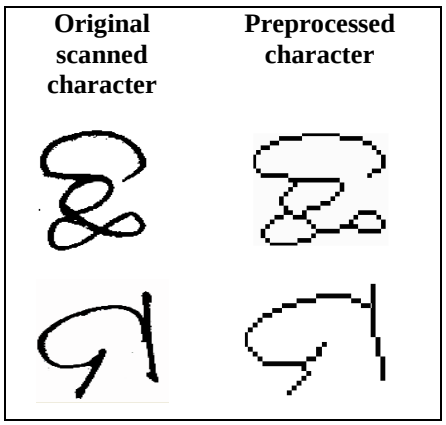

Fig-3: Shows hand written scripts & it's binaries images.

The Fig-4 shows the feature matrix of the trained characters. Each row represents the features of an individual character. The rule set is defined using the feature values. The rules are tuned by the ant system and best rule is selected as the system converges.

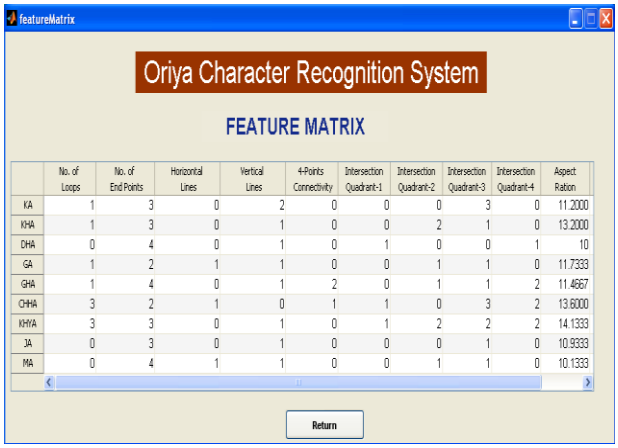

Fig- 4: Shows the feature value of the few characters.

The Fig-5 shows the hand-written character 'KA' with a match of 95.1333% and the Fig-6 shows the hand-written character 'GHA' with a match of 94.867%. It has been observed when the characters are written as per the 'block' rule discussed above, are well recognized by the system.

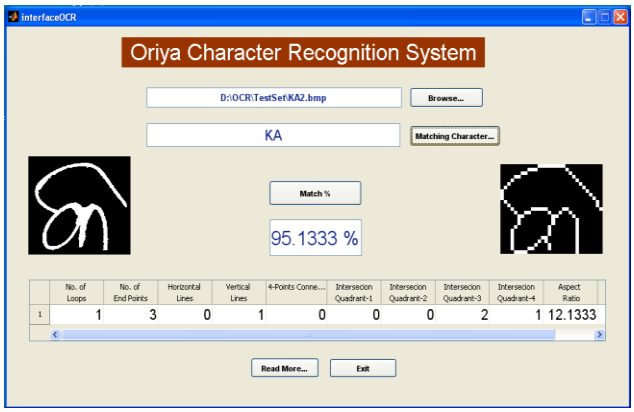

Fig- 5: Shows recognition of the Oriya character "Ka".

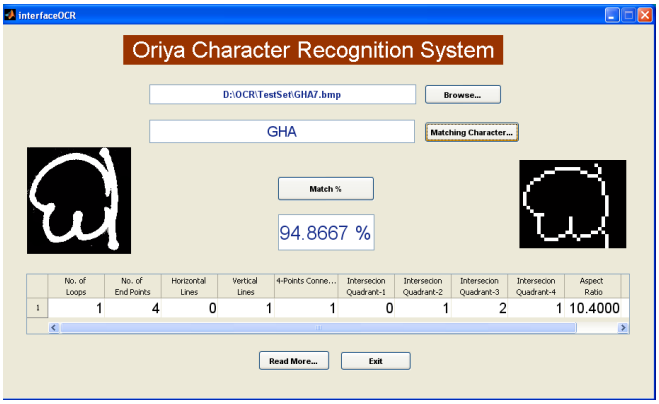

Fig- 6: Shows recognition of the Oriya character "Gha".

Table-2 represents the average-match and the best-match values of characters in percentile for each 'block' as proposed above. The algorithm is used to test 4 different characters from each 'block' and it has been observed that the average match percentile of the characters in the 'block'-2 and 'block'-3 are more precise in comparison to the 'block'-1 characters.

Table-2: 'block' of scripts with percentage of recognition

block Type	Name	Scripts	Average Recognition%	Best Recognition%
'block' 1	'Da'	ଦ	91.452	94.134
	'Dha'	ଧ	92.135	94.221
	'Na'	ନ	89.47	92.426
	'Ma'	ମ	90.442	93.165
'block' 2	'Gha'	ଘ	92.133	94.867
	'Pa'	ପ	94.256	96.221
	'Sa'	ସ	92.64	95.115
	'Ya'	ଯ	94.418	97.874
'block' 3	'Ka'	କ	92.614	95.133
	'Chha'	ଛ	92.172	95.533
	'Ga'	ଗ	94.46	97.629
	'Tha'	ଥ	94.134	96.142

4. CONCLUSIONS AND FUTURE WORK

In this paper we have presented an automated system for Oriya character recognition using Ant miner algorithm. Due to the fast convergence property of Ant miner algorithm, the rule set becomes stable within maximum 10 training characters; moreover it is a dependant of the quality of the scripts used in the training set. It is observed, if the scripts are written as per the 'block' rule defined above, then the average recognition rate is above 90%. When the scripts are written casually then either extra 'block's are created or the required 'block's are

missed still the characters are detected with less percentile of matching. The study shows the proposed 'block'-based rules are useful for the character recognition. Since the 'block'-1 scripts are recognized poorly. It concludes the variance in the writing style for the 'block'-1 character is more in comparison to other two 'block's.

The authors finally propose that the performance of the said system can be enhanced for the scripts written casually but readable by human being with the hybrid approaches. The system can be enhanced with a new segmentation approach to incorporate the complex scripts i.e. the scripts with additional vowels (*matras*). It will be helpful to resolve the issues of automating the reading of hand-written scripts for many regional scripts.

REFERENCES

[1] Tripathi R. C, Kumar V. "Character Recognition: A Neural Network Approach" Proceedings published in *International Journal of Computer Applications*, pp. 17-20, 2012.

[2] Krishna K., Goyal A., Chattopadhyay S. "Non-correlated Character Recognition using Hopfield Network: A Study". In the proceedings of *International Conference on Computer and Computational Intelligence (ICCCI-2011)* (Ed. Yi Xie) pp. 385-389 *Bangkok, Thailand* (2-4th December) 2011, ISBN: 978-0-7918-5992-6. DOI: http://dx.doi.org/10.1115/1.859926.paper62.

[3] Dash T., Nayak T., Chattopadhyay S. "Offline Handwritten Signature Verification using Associative Memory Net". *International Journal of Advanced Research in Computer Engineering & Technology* Vol. 1, Issue 4, pp. 370-374, 2012.

[4] Dash T., Chattopadhyay S., Nayak T., "Handwritten Signature Verification using Adaptive Resonance Theory Type-2 (ART-2) Net". *Journal of Global Research in Computer Science* Vol. 3 Issue 8, pp. 21-25, 2012.

[5] Dash T., Nayak T., Chattopadhyay S. "Offline Verification of Hand Written Signature Using Adaptive Resonance Theory Net (Type-1)". In the proceedings of the *4th International Conference on Electronic Computer Technology (ICECT-2012 Vol-2) Kanyakumari, India* (6-8 April'12). Editor: Yuan Li, pp. 205-210. ISBN: 978-1-4673-1849-5; DOI: 978-1-4673-1/12; IEEE catalog number: CFP1295F-PRT, IEEE Xplore.

[6] Dash T., Nayak T., Chattopadhyay S. "Handwritten Signature Verification (Offline) using Neural Network Approaches: A Comparative Study", *International Journal of Computer Applications*, 57(7): 33-41, 2012.

[7] Rajashekararadhya S.V, Ranjan P. V, "A Novel Zone Based Feature Extraction Algorithm for Handwritten Numeral Recognition of Four Indian Scripts", *Digital Technology Journal*, Vol. 2, pp. 41-51, 2009.

[8] Rajashekararadhya S.V., Ranjan P. V. "Efficient Zone Based Feature Extraction Algorithm For Hand Written Numeral Recognition of Four Popular South Indian Scripts", *Journal of Theoretical and Applied Information Technology*, *JATIT* 2008.

[9] Majumdar A. "Bangla Basic Character Recognition Using Digital Curvelet Transform", *Journal of Pattern Recognition Research* Vol.1, pp. 17-26, 2007.

[10] Nayak M.R., Nayak S., Manas Y., Bhanja Chaudhuri S., Chattopadhyay S. "Automatic Recognition of Handwritten Bangla Broken Characters: A Study on Simulating the Human Pattern Matching System", *International Journal of Computer Applications*, in press, 2012

[11] Mukherji P., Rege P.P. "Shape Feature and Fuzzy Logic Based Offline Devnagari Handwritten Optical Character Recognition" *Journal of Pattern Recognition Research*, Vol 5, No 1, pp. 52-68, 2010.

[12] Singh R., Yadav C. S., Verma P., Yadav V. "Optical Character Recognition (OCR) for Printed Devnagari Script Using Artificial Neural Network" *International Journal of Computer Science & Communication* vol. 1, no. 1, pp. 91-95, 2010.

[13] Lorigo L. M. and Govindaraju V, "Offline Arabic handwriting recognition: A survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 5, pp. 712-724, 2006.

[14] Cheung A., Bennamoun M., Bergmann N. W., Space Centre for Satellite Navigation, School of Electrical & Electronic System Engineering "An Arabic Optical Character Recognition System using Recognition-based Segmentation", *Pattern Recognition* vol. 34, pp 215-233, 2001.

[15] Phokharatkul P, Sankhuangaw K Somkuarnpanit, S, Phaiboon S, and Kimpan C "Off-line Hand Written Thai Character Recognition using Ant-Miner Algorithm", *World Academy of Science, Engineering and Technology 8*, pp. 768-773, 2005.

[16] Mishra S, Nanda D , Mohanty S "Oriya Character Recognition using Neural Networks", *IJCCT* Vol. 2 Spl. Issue 2, 2010 for International Conference [ICCT-2010],

[17] Chaudhuri B.B., Pal U., Mitra M. "Automatic Recognition of Printed Oriya Script", *Sadhana* vol. 27, Part 1, pp. 23-34, 2002.

[18] Manas Y., Nayak M. R., Chattopadhyay S. "Recognition and Classification of Broken Characters using Feed Forward Neural Network to Enhance an OCR Solution, *International Journal of Advanced Research in Computer Engineering & Technology* Vol. 1, No. 8, pp. 11-15, 2012.

[19] Arica N. and Fatos Yarman-Vural T., "An Overview of character recognition focused on off-line handwriting", *IEEE Transactions on System Man Cybernetics-Part C: Applications and Reviews*, vol. 31, no. 2, pp. 216-233, 2001.

[20] Baterina A V, Oppus C, "Image Edge Detection using Ant Colony Optimization, *WSEAS Transactions on Signal Processing*, Volume 6, Issue 2, pp. 58-67, 2010

[21] Nada M. A. Al Salami "Ant Colony Optimization Algorithm", *UbiCC Journal*, Volume 4, Number 3, pp. 823-826, 2009.

[22] Smaldon J., Freitas A. A. "A New Version of the Ant-Miner Algorithm Discovering Unordered Rule Sets", *GECCO'06*, Seattle, Washington, USA, July 8–12, 2006.

[23] Parpinelli R.S., Lopes H.S., Freitas A.A. "An ant colony algorithm for classification rule discovery". In Abbas H., Sarkar R., and Newton C. (Editors.). Data Mining: a Heuristic Approach, London: Idea Group Publishing, pp. 191-208.

[24] Chattopadhyay S., Banerjee S., Rabhi F.A, Acharya R. U. "A Case-based Reasoning System for Complex Medical Diagnoses". *Expert Systems: the Journal of Knowledge Engineering* (2012); published online on 13/6/2012; DOI: 10.1111/j.1468-0394.2012.00618.x (in press).

[25] Chattopadhyay S. "Psyconsultant I: A DSM-IV-Based Screening Tool for Adult Psychiatric Disorders in Indian Rural Health Center". *The Internet Journal of Medical Informatics* [Serial Online] vol. 3, no. 1, 2006.

[26] Chattopadhyay S. "A Prototype Depression Screening Tool for Rural Healthcare: A Step towards e-Health Informatics", *Journal of Medical Imaging and Health Informatics* vol. 2, issue 3, pp. 244-249, 2012.

[27] Chattopadhyay S., Sahu S.K. "A Predictive Stressor-integrated Model of Suicide Right from One's Birth: a Bayesian Approach", *Journal of Medical Imaging and Health Informatics* vol. 2, issue 2, pp. 125-131, 2012.

[28] Chattopadhyay S.. "Neurofuzzy Models to Automate the Grading of Old-age Depression". *Expert Systems: the Journal of Knowledge Engineering* (2012); DOI: 10.1111/exsy.12000 (in press).

[29] Satapathy S., Chattopadhyay S. "Observation-Prevention of Cardiac Risk Factors: an Indian Study", *Journal of Medical Imaging and Health Informatics* Vol. 2, No. 2, pp.102-113, 2012.

[30] Chattopadhyay S., Pratihar D. K, De Sarkar S. C.. "Statistical Modelling of Psychoses Data". *Computer Methods and Programs in Biomedicine* Vol. 100, No. 3, pp. 222-236, 2010.

[31] Manas Y., Jena P. K. "Enhanced Color Image Segmentation of Foreground Region using Particle Swarm Optimization" *International Journal of Computer Application* vol. 57, no. 8, pp. 18-23, 2012.