International Journal of Rough Sets and Data Analysis
Volume 6 • Issue 2 • April-June 2019

Recognition of Odia Handwritten Digits using Gradient based Feature Extraction Method and Clonal Selection Algorithm

Puspalata Pujari, Guru Ghasidas Vishwavidyalaya, Bilaspur, India
Babita Majhi, Guru Ghasidas Vishwavidyalaya, Bilaspur, India

ABSTRACT

This article aims to recognize Odia handwritten digits using gradient-based feature extraction techniques and Clonal Selection Algorithm-based (CSA) multilayer artificial neural network (MANN) classifier. For the extraction of features which contribute the most towards recognition from images, are extracted using gradient-based feature extraction techniques. Principal component analysis (PCA) is used for dimensionality reduction of extracted features. A MANN is used as a classifier for classification purposes. The weights of the MANN are adjusted using the CSA to get optimized set of weights. The proposed model is applied on Odia handwritten digits taken from the Indian Statistical Institution (ISI), Calcutta, which consists of four thousand samples. The results obtained from the experiment are compared with a genetic-based multi-layer artificial neural network (GA-MANN) model. The recognition accuracy of the CSA-MANN model is found to be 90.75%.

KEYWORDS

Artificial Neural Network, Classification, Clonal Selection Algorithm, Feature Extraction, Odia Digit Recognition, Preprocessing

INTRODUCTION

Hybrid approach is one of the main characteristics of the contribution of soft computing. Hybrid approach is the integration of already existing techniques. There are several population-based optimization algorithms inspired by biological evolution. These optimization algorithms can be integrated with other techniques to increase the performance of a model. CSA-MANN is an approach which incorporates clonal selection algorithm and artificial neural network (Majhi et al., 2011; Xuefeng et al., 2017). A Neural network is designed to adopt in a similar way the human nervous system works. It learns from a set of past data or pre-defined data with class label and develops a model that can be applied to unseen data without class label. They have the ability to adapt to circumstances and learn from past experience. Neural Network can model complex non-linear relationships and are appropriately suited for classification phenomenon into predetermined class. It is difficult to a get optimized set of weight by using MANN. On the other hand, a CSA is an evolutionary algorithm which mimics the behavior of living organism to adapt to the environment. In CSA initially a number of solutions are generated. Out of the possible solutions best possible solutions are chosen. Then some of the accepted solutions are generated using the factors of best choices and the process is repeated again and again. The above process is repeated until a desired solution is found. CSA is useful for finding the optimal solutions from a number of alternative solutions for a problem. CSA

DOI: 10.4018/IJRSDA.2019040102

can be effectively used to optimize the weights of a neural network which is very difficult task. In this proposed work CSA-MANN integrated model is used to determine the weights of a multilayer feed forward neural network using back propagation concept. The backpropagation algorithms are based on gradient descent-based learning method for the update of weights of the multi-layer artificial neural network. The network architecture of the proposed MANN is based on back propagation learning method and real coded chromosomes are used for implementing CSA as compared to binary coded methods. Optimization technique is used for the selection of best solution among a number of alternatives. A large number of optimization techniques have been applied by the authors for the identification of handwritten characters. In most of the cases the recognition accuracy of the system depends on the combination of feature extraction techniques and classifiers used for recognition. In Nawwaf et al. (2001) developed a system for recognition of online Arabic characters regardless of the orientation, position and size of the input pattern. Modified quadratic discriminant function (MQDF) for the classification of handwritten Chinese character (Xiaohua et al., 2017). Sparse coding is used to compact the parameters of MQDF. They have adopted maximum likelihood-based and the K-SVD methods to build two compact MQDF classifiers. In Rina et al. (2017) the authors integrated hidden Markov model and harmony search algorithm for recognition of writer independent online Kurdish character recognition. The system is applied on a dataset of 4500 words and a recognition rate of 93.52% is achieved with this model. Xuefeng Xiao et al. (2017) proposed convolutional neural network (CNN) model for recognition of handwritten Chinese character and adaptive drop-weight technique for pruning CNN parameter. In Ritesh et al. (2016) the authors have proposed region sampling methodologies based on a non-dominated sorting harmony search algorithm and a non-dominated sorting genetic algorithm for recognition of Bangla character. An axiomatic fuzzy set (AFS) based fuzzy logic is used for integrating the pareto-optimal solution from the multi-objective heuristics algorithm. Recognition accuracy of 86.64% and 98.23% is an obtained for handwritten Bangla character and digits respectively. A deep learning technique is presented in Roy et al., (2017) for recognition of compound character of Bangla script. The authors employed layer wise training to deep convolutional neural network augmented with RMSPROP algorithm for faster convergence. A new benchmark of recognition accuracy is highlighted on the CMATERdb 3.1.3.3 dataset. In Boufenar et al. (2018) the authors proposed deep convolutional neural network (DCNN) using transfer learning strategies on OIHACDB and AHCD off-line isolated handwritten Arabic character.

Recently clonal selection algorithm has been used in many applications like wind power forecasting, Brain MR image segmentation, short-term hydro thermal scheduling, power generators maintenance scheduling, automatic clustering, virus detection, for urban bus scheduling, construction site utilization planning and development of digital channel equalizer. In (Chitsaz et al., 2015) the authors have proposed Clonal selection algorithm-based wavelet neural network for optimization of free parameters of wavelet neural network for wind power forecasting. Tong Zhang et al. (2012) have incorporated CSA into hidden Markov random field (HMRF) model for brain image segmentation and produced more accurate result. R. K. Swain et al. (2011) proposed CSA based evolutionary approach for obtaining a solution to short – term hydro thermal scheduling problem. The proposed algorithm is compared with other evolutionary techniques and produced better result. In El-Sharkh (2013) the authors have introduced a novel method for power generators maintenance scheduling where CSA is used for obtaining optimum solution. A gene transposon-based clone selection algorithm (GTCSA) has been proposed in Liu et al. (2012) to find satisfied number of clusters automatically. The authors compared the results of the proposed model with other models and obtained good performance. A novel approach has been presented by the authors in Afaneh et al. (2013) for detection of computer virus using Clonal selection algorithm and achieved 94.4% accuracy in detection of virus. Xinguo Shui et al. (2015) has proposed a Clonal selection algorithm-based bus vehicle scheduling approach for finding an optimum solution to real world problem within a few seconds. In Wang et al., (2016) the authors have introduced Clonal selection algorithm for construction site utilization planning to support construction process and produced efficient results with the proposed system. In Nanda et al.,

(2008) the authors have proposed a novel digital channel equalizer using artificial immune system. The performance of the proposed model is found to be superior as compared to LMS and GA based models. In literature, MANN has been used in large number of applications for solving problems. Also, optimization algorithms have been applied to solve complex problems, but they are less explored in the field of pattern recognition. CSA based evolutionary technique has not been explored in literature for recognition Odia character. Development of novel methods for recognition of handwritten character is a very challenging issue as there are a large number of variations found in handwritten characters. Proper combination of feature extraction techniques and classification methods has significant effect on the development of an efficient recognition method for unconstrained handwritten characters. In this work a CSA algorithm is used for weight optimization of MANN classifier. All generic phases like preprocessing, feature extraction and recognition required for character recognition are applied to the images. The operations median filtering, canny edge detection and normalization of preprocessing phase are carried out in this paper. Gradient based feature extraction approach is used for extraction of features from the images of handwritten character. The extracted features are reduced further using principal component analysis (PCA). A combination of gradient based feature extraction approach and CSA based MANN model for recognition has not been attempted before. Experimental work shows that this combination scheme has the effect of increasing the performance of recognition system. Figure 1 shows overall representation of the proposed model.

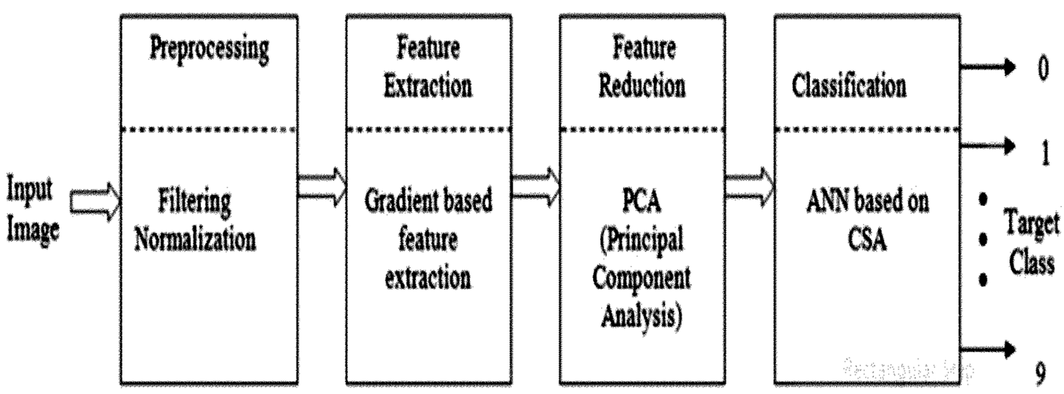
ORGANIZATION OF THE PAPER

The article is organized into ten sections. Second section describes preprocessing and feature extraction phase. Third section describes MANN with backpropagation. Fourth Section describes the clonal selection (CSA) algorithm. Section five describes CSA based MANN classifier. Weight optimization of MANN model using CSA algorithm is described in section six. Section seven describes basics of GA. Section 8 describes weight optimization of MANN model using CSA algorithm. Section nine highlights simulation carried out. Section ten describes the outcomes of the experiment carried out.

PREPROCESSING AND FEATURE EXTRACTION

The first phase in character recognition is preprocessing. This phase is carried out to improve the quality of image scanned and remove all type of abnormalities associated with the scanned input image. Various operations of preprocessing include normalization, thinning, filtering, skew correction and removal of noise. In this work the handwritten digits are considered to be unconstrained and

Figure 1. Clonal selection based multilayer artificial neural network model



isolated. The images of the digits are preprocessed first to remove noise and variability in images. A median filtering- based approach is used for filtering and the size of all images with different sizes are converted to a size of common standard 64 X 64 pixels. Canny edge detector is used for detection of edges of the images. A set of Odia handwritten digits (0-9) is shown in figure 2.

Feature Extraction using Gradient Technique

Extraction of feature has significant importance in character recognition. This step is highly necessary to extract significant features from the images. As, in an image all features are not significant, the important features which will contribute a lot to the recognition phase are extracted. In this work after preprocessing, gradient based approach is used for extraction of features from the enhanced image and feature vector is generated. The features of the images are further reduced using PCA. Basically, a gradient represents the variation in the direction of change of intensity or color of an image. The gradient from an image is calculated from the strength and directions of the pixels. After the generation of significant features, the dimensionality of the feature vector is further reduced from 2519 to 75 numbers by using PCA. The direction of image is shown in figure 3 and the strength of image is shown in figure 4. The direction and strength f (u, v) is calculated as follows

$$\left.\begin{array}{l} g_u = g\left(u+1, v+1\right) - g\left(u, v\right) \\ g_v = g\left(u+1, v\right) - g\left(u, v+1\right) \\ Direction: \theta\left(u, v\right) = tan^{-1}\left(\dfrac{g_u}{g_y}\right) \\ Strength: f\left(u, v\right) = \sqrt{\left(g_u\right)^2 + \left(g_v\right)^2} \end{array}\right\} \quad (1)$$

Calculation of Gradient from the images

The steps for calculation of gradient feature is as follows

1. Using the above equation (1) the direction of gradient is quantized to 32 levels with $\prod/16$ interval.
2. For each normalized image a block size of 9×9 is taken.
3. In each 32 direction the strength of the image is accumulated separately for each block.

As the images are circular in nature the gradient of the digit image is calculated taking into the circularity nature of the image.

Figure 2. shows a set of Odia handwritten digit (0-9)

Figure 3. Image showing direction of gradient

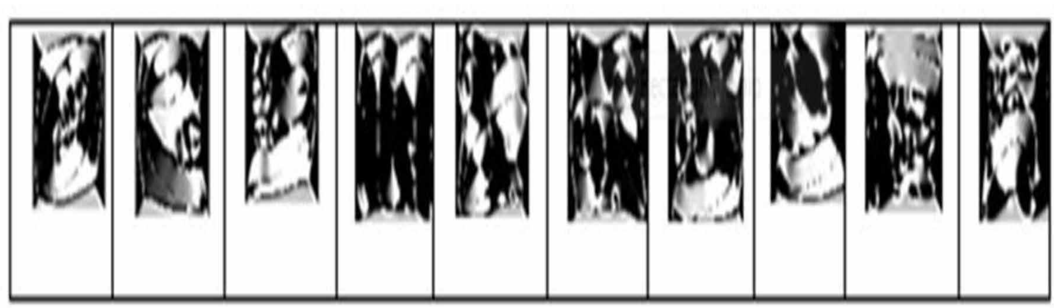
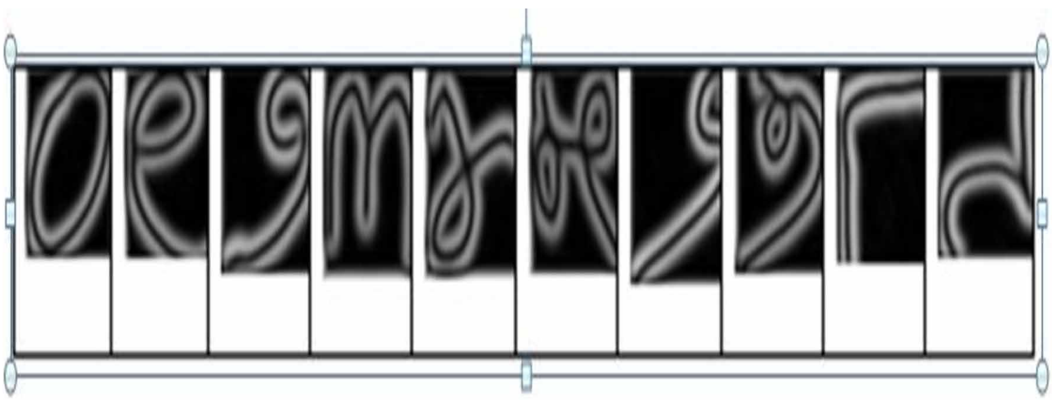


Figure 4. Image showing strength of gradient



MANN WITH BACKPROPAGATION LEARNING

Step involved for the development of ANN classifier with back propagation learning algorithm are as follows (Beale, 1996; Haykin, 1999).

Step 1: (Data collection) Collect the dataset to be used for the implementation of the proposed model.

Step 2: (Partitioning of data) divide available data set randomly into 90% and 10% of dataset.

Step 3: (Architecture) Select a suitable architecture for the MANN with input layer, hidden layer and output layer.

Step 4: (Parameter tuning and weight initialization) initialize the network weights and parameters. Often, the initial values are important in determining the effectiveness and length of training.

Step 5: (Data transformation) As the images are in the form of pixels, transform the images so that they can be used by the MANN without any complexity

Step 6: (Training) Multiply the inputs with the weights of the multilayer artificial neural network. Use activation function for obtaining outputs. Obtain final output from the network. Compare the obtained output with the target output to generate errors. Using the errors adjust the weights until a desired set of weights are obtained.

Step 7: (Testing) After development of model, test the model with unknown data with the finalized weights. Calculate the accuracy of the model using performance measures.

CLONAL SELECTION ALGORITHM (CSA)

Clonal selection algorithm is an optimization algorithm which falls under the area of artificial immune systems (Nanda et al., 2008). CSA is inspired by clonal selection theory of acquired immunity. Immunity is the condition of an organism to resist disease. CSA describes how the immune cells react to antigens. CSA learns the same way the antibodies of our immune system learn adaptively how to act against antigen and react upon it. CSA works taking into a problem function fun (z). The function fun(z) is optimized using CSA. At the initial step a number of possible solutions are generated. The affinity or fitness is calculated using the antibodies. Affinity determines out of number of antibodies the one which will be cloned with rest of the antibodies. After cloning the cloned antibodies are changed or mutated with mutation parameter. After mutation the fitness of each antibodies are calculated again and the antibodies are sorted according to their fitness value. The affinities are evaluated a number of times and finally affinity with the best outcome gives the final solution or optimum solution for the problem. The steps of Clonal Selection Algorithm are as follows

Step 1: Create a number of antibodies. The set of antibodies represents current candidate solutions of a problem.

Step 2: Calculate the fitness values of each antibodies.

Step 3: Sort the antibodies in ascending order according to their fitness obtained. In the present work lowest affinity or fitness indicates better fitness and highest affinity indicates lower fitness.

Step 4: Clone the rest of the antibodies with the lowest fitness antibodies with predefined ratio.

Step 5: With the mutation ratio mutate the antibodies. The antibodies with lowest affinity in the present work are mutated higher and the antibodies with higher affinity are mutated less to get an optimum solution.

Step 6: For each antibody evaluate the new affinity values.

Step 7: Till the desired solution is reached repeat the steps 3 to 6.

CSA BASED MANN CLASSIFIER

The schematic diagram of CSA-MANN model is shown in figure 5. The multilayer MANN has two hidden layers with an architecture 75-5-10. The output layer consists of ten outputs. Initially the input samples are applied to ANN and the weights are multiplied to the inputs to produce the intermediate outputs. The outputs produced from one layer are used as input for the next layer. Log Sigmoid function is used as the activation function. The final outputs of the MANN are compared with the actual output and error terms are generated. The error terms are used for obtaining fitness. Basing on the mean square error (MSE) the weights of the MANN are optimized using CSA.

WEIGHT UPDATION OF MANN MODEL USING CLONAL SELECTION ALGORITHM

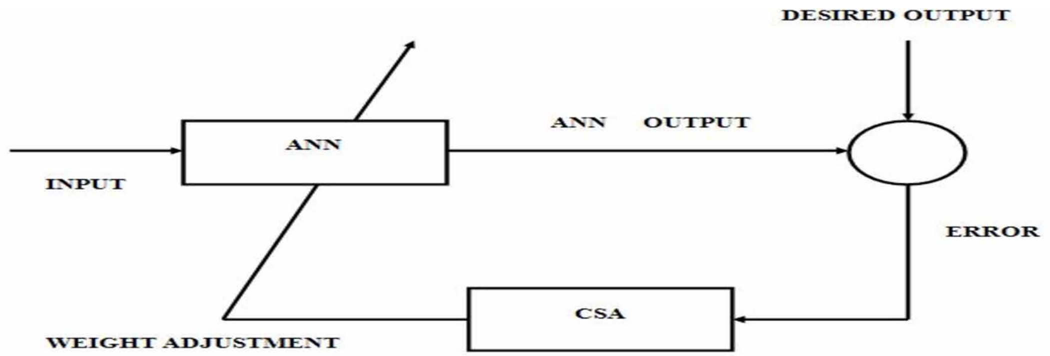
The steps for optimization of weights of MANN model using CSA are (Nanda et al., 2008)

Step1: Initialization: Randomly initialize the weights of the MANN model within the range 0 to 1. Each weight of the MANN model represents a cell and n number of such weight vectors is taken. Each weight vector represents a possible solution for the recognition problem.

Step2: Output calculation: Let 'k' be the total number of input samples taken. Pass the input samples to the MANN structure. Compute the output of the model using input samples and weight vector.

Error generation: The generated output $y^{'}\left(k,n\right)$ of the model for k^{th} sample and n^{th} vector is compared with the target output to generate error terms using (2)

Figure 5. Schematic diagram of CSA based MANN



$$e(k,n) = y(k,n) - y'(k,n) \tag{2}$$

Fitness Evaluation: For each weight vector mean square error (MSE) is obtained using (3)

$$MSE(n) = \frac{\sum_{k=1}^{K} e^2(k,n)}{K} \tag{3}$$

The fitness function is minimized by CSA to optimize the weights of MANN classifier.

Step 5: Selection: Corresponding to lowest MSE select the weights vector.
Step 6: Cloning: Duplicate all weight sets with the weight set having minimum MSE.
Step 7: Mutation: The cells are chosen randomly and mutated corresponding to mutation probability P_m
Step 8: Repeat the steps from 2 to 6 until desired solution is reached.

BASICS OF GENETIC ALOGORITHM

Genetic algorithm (GA) is based on the principle of theory of evolution given by Darwin (Rajasekaran et al., 2003). According to the principal of nature the high-fitted individual always dominates the low-fitted individual for resources. Unlike gradient based algorithm GA is a heuristic searching based adaptive evolutionary algorithm which is primary based on the principle of nature selection, crossover and mutation to reach towards its goal. The basic steps of genetic algorithm are as follows

Step 1: Initialization: Randomly initialize the population with an aim to include all possible solution to the given problem. Let N represents chromosomes of the population which are randomly initialized. Set the parameters of GA like crossover parameter P_{cross}, mutation parameter $P_{mutation}$, total size of the population, length of the chromosome and maximum number of iterations.

Step 2: Evaluation of fitness: Calculate the fitness of each chromosome using function $fun(Chm)$ using objective function i.e. $fun(Chm_1), fun(Chm_2) \ldots fun(Chm_N)$ where $Chm = Chm_1, Chm_2, \ldots, Chm_N$ for each solution Chm of the population.

Step 3: Selection: Select the chromosomes having best fitness value from the entire population so that they can be used for reproduction.

Step 4: Crossover: Generate new offspring using cross over operation with crossover probability P_{cross}.

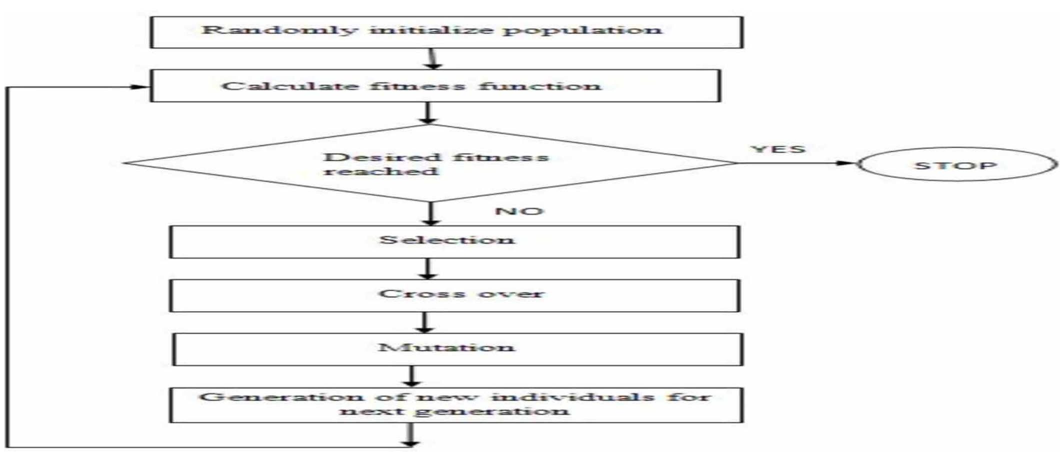
Step 5: Mutation: After cross over operation apply mutation operation with mutation probability $P_{mutation}$ in order to avoid the off springs to be same as the parents.
Step 6: Population for next generation: The chromosomes with lower fitness are replaced with the chromosomes with best fitness. Replace the earlier population with the new one.
Step 7: Until a desired solution to the problem is obtained go to step 2 for next generation.
Step 8: Optimum Solution: Obtain the optimum solution for the objective function under consideration. Figure 6 represents the flow diagram of genetic algorithm.

WEIGHT OPTIMIZATION USING GA-MANN MODEL

The final set of weights of MANN is obtained using an integrated model of GA and MANN. In GA based MANN, GA algorithm is used to optimize the weight of neural network (Rajesekaran et al., 2003). GA works with an initial population consisting of a number of chromosomes where each chromosome represents the weights of the ANN. Each chromosome is assigned a fitness value by applying the weights to N number of input samples in the training set. The fitness value gives the effectiveness of a solution towards solving a problem. The individual with worst fitness is replaced with individual with best fitness. This yield new individual strings as offspring. The individuals with best fitness values from the current generations are selected to generate new population of possible solution. The process is repeated many times and after certain generations chromosomes with best fitness are inherited to provide optimum solution to the problem under consideration. A BPN (Back propagation neural network) with 75-5-10 configuration is assumed in this paper.

The number of weights taken for the network is 425. Each weight is a real number and a random digit d = 5 is used for representing a weight value. The string S representing the length of the chromosome is 2125. Figure 7 represents sample chromosome, generated using (6) consists of 425 genes. During training phase N number of input samples of the dataset is multiplied with their corresponding weights and their outputs are calculated. The predicted output is compared with the desired output to produce the error e_i. At the completion of all samples N errors are produced. The desired output and the predicted output are used to calculate MSE (Mean Square Error). The mean square error (MSE) corresponding to i^{th} chromosome is determined by the following equation

Figure 6. Flow diagram of genetic algorithm



$$MSE\left(i\right)=\frac{\sum_{n=1}^{N}e^{2}\left(k\right)}{N} \tag{4}$$

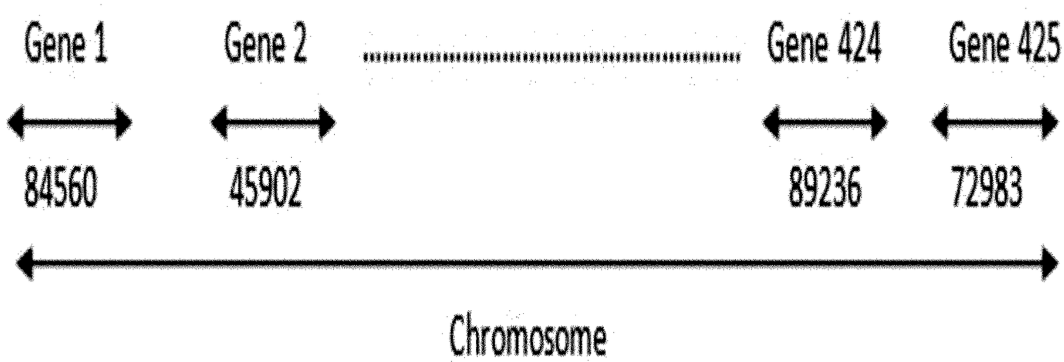
For each chromosome MSE is obtained and the average of all MSE is calculated to produce MMSE (Mean of mean square error).

The weight optimization process is repeated for a number of generations (t) until convergence. Equation for calculation of MMSE (Mean of mean square error) is as follows

$$MMSE\left(t\right)=mean\left(MSE\right) \tag{5}$$

To best distinguish among the target classes instead of binary coded GA, a real coded genetic algorithm (RCGA) is used in this paper. The RCGA is hybridized with MANN to get an optimized solution i.e. a suitable set of weights for MANN. The length of each gene in a chromosome is taken 5. So the total length of the chromosome is 2125. Take a population P_0 with size p. For the optimization of the weights of MANN, the proposed model uses algorithm RCGA-MANN-WT (). For evaluation of fitness function of individual chromosomes in the algorithm FITNESS () is used. Initial set of weights for the MANN are generated using RCGA-MANN-WT () algorithm. At first the inputs of ANN classifier is multiplied with the generated weights where each set of weights represents a chromosomes. All input samples are multiplied with same set of weights and the outputs are generated taking log sigmoid as activation function. Final output is obtained at the output layer. The error terms are generated comparing the evaluated output and target output. Using FITNESS () algorithm the fitness value of each chromosome is evaluated. The fitness values are used for adjusting the weights of MANN using evolutionary approach CSA. For cross over operation two-point crossover method is used with crossover parameter $p_{cross}=0.8$. In order to avoid the newly generated chromosomes to be same as the parents mutation operator is carries out with mutation parameter $p_{mutation}=0.1$. After mutation operation the newly generated off springs are combined with parent population. The chromosomes are sorted according to their fitness value from high fitness to low fitness. The chromosomes with high fitness values are selected out of the entire population p to generate new population. The new population is used in next generation. All generic steps of GA fitness calculation, selection operation, crossover operation, mutation operation and generation of new population for next generation are repeated a number of times until a desired solution or stopping criteria is reached.

Figure 7. Sample Chromosomes randomly generated for the BPN weights



Initial set of weights for MANN are generated using equation 6. Fitness of each individual is calculated using FITNESS () algorithm.

Let $chm_1, chm_2, \ldots, chm_d, \ldots, chm_L$ be a chromosome and $chm_{rd+1}, chm_{rd+2}, \ldots, chm_{(r+1)d}$ be the r[th] gene $\left(r \geq 0\right)$ in the chromosome. The weight, $weight_r$ for each chromosome is calculated as follows

$$weight_r = \begin{cases} + \dfrac{chm_{rd+2}10^{d-2} + chm_{rd+3}10^{d-3} + \ldots c_{hm(r+1)d}}{10^{d-2}}, & if\, 5 \leq chm_{rd+1} \leq 9 \\ - \dfrac{chm_{rd+2}10^{d-2} + chm_{rd+3}10^{d-3} + \ldots chm_{(r+1)d}}{10^{d-2}}, & if\, 0 \leq chm_{rd+1} < 5 \end{cases} \tag{6}$$

Weight Updation using RCGA-ANN-WT Model

```
{
initialize u = 0;
Using (6) generate initial population P_u with m number of real
coded chromosomes C_v^u . Each chromosome corresponds to a set of
weights for the MANN.
Repeat until the desired solution is not reached
{
Using FITNESS() algorithm evaluate the fitness value of  F_v^u value
of each chromosome, C_v^u ϵ P^u ;
Replace the chromosomes having low fitness value with the
chromosome having high fitness value and create the mating pool;
Apply two point cross over operation to the t population to
generate offspring;
Apply mutation operation using mutation operator;
Apply selection operation to generate new population;
u = u + 1;
Replace the old population P_u with new population;
}
Obtain suitable set of weights from P^u to be used finally by
MANN;
}
```

Calculation of Fitness Using FITNESS() Algorithm

```
Let Inp_u = (Inp_1u, Inp_2u, Inp_3u, ... , Inp_lu) represents the inputs and
Target_u = (Target_1u, Target_2u, Target_3u, ..., Target_nu) represents the outputs of a
MANN. Architecture of ANN used is 75-5-10 .
For each chromosome
{
Obtain initial set of weights weight_u from C_u  by using (6)
 Take N number of input samples and train MANN keeping Weight_u same
for all input sample.
For each input sample generate error using the following formula
Error_u = Σ_v (Target_vu − out_vu)^2  where out_u represents the output obtained
```

```
with MANN
Calculate the root mean square error  using (7)
```

$$Error = \sqrt{\frac{\sum_u Error_u}{N}}\ ,\ u = 1,2,\ldots,p; \tag{7}$$

```
Calculate the fitness F_u  for each chromosome as follows
```

$$F_u = \frac{1}{Error} \tag{8}$$

```
}
Output F_u for each chromosome
}
```

SIMULATION STUDY FOR CSA-MANN MODEL AND GA-MANN MODEL

The proposed CSA-MANN model is implemented using MATLAB. The handwritten Odia digit database collected from ISI Calcutta is used for the experimental work. The proposed system is trained with 3600 number of samples. The digits are categorized into at most ten categories. The classification model is tested with the rest of ten percent of samples consisting of 400 numbers of samples. The model is trained using the training set of data and the recognition accuracy is calculated for the test data. Fivefold cross validation is used for training the data. The images are normalized and median filtering is applied in preprocessing step. Canny edge detection approach is used for detection of edges. Gradient based approach is used to generate feature vector. The dimension of feature vector is further reduced using PCA. The features are then applied as inputs to the MANN model. The architecture of MANN is taken as 75-5-10 and sigmoid activation function is used for the outputs. The error terms are calculated based on the actual and calculated outputs. The weights of the CSA-MANN model are optimized using CSA algorithm. Mean square error (MSE) is computed using the error terms. The parameters of CSA are: 80 items, mutation parameter = 0.1, number of generation taken is 1000. For GA-MANN model the weights of the ANN are initialized using (6). The input sample of ANN is multiplied with the initial weights of ANN and outputs are produced. Log sigmoid activation function is used for obtaining the outputs. The final outputs generated from MANN are compared with the actual output for generating error terms. The weights of the MANN are adjusted using GA algorithm to minimize the error terms. The parameters used for GA-MANN are: 80 chromosomes, crossover parameter = 0.8, mutation parameter = 0.1, 1000 number of generation. Figure 8 shows the confusion matrix obtained during validation of CSA-MANN model with equal number of classes. This shows the effectiveness of the model in recognition of Odia handwritten digits. The recognition accuracy of CSA-MANN model is also compared with GA-MANN. Figure 9 shows the confusion matrix of GA-MANN hybrid model. Table 1 shows the individual accuracy of classes and overall accuracy for CSA-MANN and GA-MANN model. The results obtained show that the proposed CSA-MANN model yields higher recognition accuracy as compared to GA-MANN hybrid model.

CONCLUSION

In this article, an attempt has been made to recognize the Odia handwritten digits using CSA-MANN hybrid model. The weights of the MANN model are optimized with CSA. Gradient based

Figure 8. Confusion Matrix of CSA-MANN Model obtained with test data

Class	0	1	2	3	4	5	6	7	8	9
0	34	1	1	0	1	0	0	1	0	0
1	1	37	1	0	0	0	0	0	0	1
2	0	0	36	0	0	0	2	2	0	0
3	0	0	0	36	2	0	0	0	0	2
4	1	0	0	0	37	0	0	1	0	1
5	2	2	0	0	2	34	0	0	0	0
6	0	2	0	0	0	0	36	2	0	0
7	0	0	1	0	0	0	2	37	0	0
8	0	1	0	0	0	0	0	0	38	1
9	0	0	1	0	0	0	0	0	1	38

Figure 9. Confusion Matrix of GA-MANN Model obtained with test data

Class	0	1	2	3	4	5	6	7	8	9
0	32	3	3	0	0	0	0	2	0	0
1	2	35	2	0	0	0	0	1	0	0
2	2	0	36	0	0	0	0	2	0	0
3	0	0	0	32	0	0	3	0	2	3
4	2	1	0	0	37	0	0	0	0	0
5	0	1	0	0	4	33	0	2	0	0
6	0	3	0	0	0	0	33	4	0	0
7	0	2	0	0	0	0	2	36	0	0
8	2	0	0	0	1	0	0	0	35	2
9	3	0	0	0	2	0	0	0	3	32

approached is used to extract features from the digit images. Preprocessing of images is carried out before feature extraction to remove variations in images. The obtained features are reduced to a less number using PCA. The result obtained from the experiment is compared with GA-MANN model. From the experimental result it is exhibited that CSA-MANN model produced higher accuracy as compared to GA-MANN model in recognizing Odia handwritten digits with 90.75% of accuracy. The recognition accuracy is found to be between 90 to 95% for different digits. The performance of the proposed model can be further enhanced by using other suitable feature extraction techniques and optimization algorithms like bacterial foraging optimization (BFO), bee colony optimization (BCO), ant colony optimization (ACO), cat swarm optimization (CSO), bat swarm optimization (BSO), etc.

Table 1. Comparison of recognition accuracy of CSA-MANN model with GA-MANN model during validation

		CSA-MANN		GA-MANN	
Class	**No of samples in all cases**	**Number of Positive Cases**	**% of accuracy obtained**	**Number of Positive Cases**	**% of Accuracy obtained**
0	40	34	85	32	80
1	40	37	92.5	35	90
2	40	36	90	36	100
3	40	36	90	32	80
4	40	37	92.5	37	100
5	40	34	85	33	80
6	40	36	90	33	80
7	40	37	92.5	36	90
8	40	38	95	35	80
9	40	38	95	32	90
		Overall accuracy in % = 90.75		Overall accuracy in % = 85.25	

REFERENCES

Afaneh, S., Zitar, R. A., & Al-Hamami, A. (2013). Virus detection using clonal selection algorithm with Genetic Algorithm (VDC algorithm). *Applied Soft Computing*, *13*(1), 239–246.

Boufenar, C., Kerboua, A., & Batouche, M. (2018). Investigation on deep learning for off-line handwritten Arabic character recognition. *Cognitive Systems Research*, *50*, 180–195.

Chitsaz, H., Amjady, N., & Zareipour, H. (2015). Wind power forecast using wavelet neural network trained by improved Clonal selection algorithm. *Energy Conversion and Management*, *89*, 588–598.

Demuth, H. B., & Beale, M. H. (1996). *Neural Network Design*. Cengage Learning, India PVT Ltd.

El-Sharkh, M. Y. (2014, May). Clonal selection algorithm for power generators maintenance scheduling. *International Journal of Electrical Power & Energy Systems*, *57*, 73–78. doi:10.1016/j.ijepes.2013.11.051

Haykin, S. (1999). *Neural Networks A Comprehensive Foundation*. Pearson Prentice Hall.

Kharma, N. N., & Ward, R. K. (2001, November). A novel invariant mapping applied to hand written Arabic character recognition. *Pattern Recognition*, *34*(11), 2115–2120. doi:10.1016/S0031-3203(00)00140-0

Liu, R., Jiao, L., Zhang, X., & Li, Y. (2012, October). Gene transposon based clone selection algorithm for automatic clustering. *Information Sciences*, *204*, 1–22, 30. doi:10.1016/j.ins.2012.03.021

Majhi, B., Satpathy, J., & Rout, M. (2011). Efficient recognition of odia numerals using low complexity neural classifier. In *Proceedings of IEEE International Conference on Energy, Automation and Signal* (pp. 140-143). IEEE.

Nanda, S. J., Panda, G., & Majhi, B. (2008, December). Development of novel digital equalizers for noisy nonlinear channel using artificial immune system. In *Proceedings of the 2008 IEEE Region 10 and the Third international Conference on Industrial and Information Systems* (pp. 1-6). IEEE.

Rajasekaran, S., & Pai, G. V. (2003). *Neural networks, fuzzy logic and genetic algorithm: synthesis and applications*. PHI Learning Pvt. Ltd.

Roy, S., Das, N., Kundu, M., & Nasipuri, M. (2017, April). Handwritten isolated Bangla compound character recognition: A new benchmark using a novel deep learning approach. *Pattern Recognition Letters*, *90*, 15–21. doi:10.1016/j.patrec.2017.03.004

Sarkhel, R., Das, N., Saha, A. K., & Nasipuri, M. (2016, October). A multi-objective approach towards cost effective isolated handwritten Bangla character and digit recognition. *Pattern Recognition*, *58*, 172–189. doi:10.1016/j.patcog.2016.04.010

Shia, M., Fujisawab, Y., Wakabayashia, T., & Kimuraa, F. (2002). Handwritten numeral recognition using gradient and curvature of gray scale image. *Journal of Pattern Recognition*, *35*(10), 2051–2059. doi:10.1016/S0031-3203(01)00203-5

Shui, X., Zuo, X., Chen, C., & Smith, A. E. (2015, November). A clonal selection algorithm for urban bus vehicle scheduling. *Applied Soft Computing*, *36*, 36–44. doi:10.1016/j.asoc.2015.07.001

Swain, R. K., Barisal, A. K., Hota, P. K., & Chakrabarti, R. (2011, March). Short term hydro thermal scheduling using clonal selection algorithm. *International Journal of Electrical Power & Energy Systems*, *33*(3), 647–656. doi:10.1016/j.ijepes.2010.11.016

Wang, X., Deshpande, A. S., Dadi, G. B., & Salman, B. (2016). Application of clonal selection algorithm in construction site utilization planning optimization. *Procedia Engineering*, *145*, 267–273. doi:10.1016/j.proeng.2016.04.073

Wei, X., Lu, S., & Lu, Y. (2018). Compact MQDF classifiers using sparse coding for handwritten Chinese character recognition. *Pattern Recognition*, *76*, 679–690. doi:10.1016/j.patcog.2017.09.044

Xiao, X., Jin, L., Yang, Y., Yang, W., Sun, J., & Chang, T. (2017, December). Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition. *Pattern Recognition*, *72*, 72–81. doi:10.1016/j.patcog.2017.06.032

Zarro, R.D. & Anwer, M.A. (2017). Recognition based online Kurdish character recognition using hidden Markov model and harmony search. *Engineering Science and Technology, an International Journal, 20*(2), 783-794.

Zhang, T., Xia, Y., & Feng, D. D. (2012). An evolutionary HMRF approach to brain MR image segmentation using vlonal selection algorithm. *IFAC proceeding volumes, 45*(18), 6-11.

Puspalata Pujari is an assistant professor in CSIT department, GGV, Central University, Bilaspur, and Chhattisgarh, India. She received her M.C.A Degree from Berhampur University, Berhampur, Odisha, India in 1998. She is currently pursuing her Ph.D in the department of Computer Science and Information Technology, Guru Ghasidas Vishwavidyalaya, Central University, Bilaspur, Chhattisgarh, India. Her areas of interest include character recognition, soft computing, and data mining.

Babita Majhi, is presently working as an Assistant Professor in the department of CSIT, GGV, Central University, Bilaspur, India. She did her Ph.D. in 2009 from National Institute of Technology Rourkela and Post Doctoral research work at University of Sheffield, UK (Dec.2011-Dec. 2012) under Boyscast Fellowship of DST, Govt. of India. She has guided 4 Ph.D. and 8 M.Tech theses in the field of adaptive signal processing, computational finance and Soft-computing and has published 100 research papers in various referred International journals and conferences. Her research interests are adaptive signal processing, soft computing, evolutionary computing, computational finance, distributed signal processing, and data mining.