RESEARCH ARTICLE OPEN ACCESS

Meetei Mayek Unicode Modeling Using Swarm Intelligence and Neural Networks

Wahengbam Kanan Kumar, Chetan Das Gupta, Jyoti Bordoloi, Yogesh Kumar Sharma
Maharishi Markandeshwar University, Ambala, India

ABSTRACT
The Different techniques have evolved for better optical character recognition for many scripts, yet very little literature has been found for Meetei Mayek script. The current paper exhibits a new approach to model and simulate handwritten Meetei mayek script by using advanced segmentation tools and recognition algorithms. Preprocessing of the acquired images is needed before segmentation and recognition steps; segmentation is done by using PSOFCM segmentation, while multilayer feed forward neural network with back propagation learning is used for the recognition purpose. It may be noted that PSOFCM segmentation proved useful for MRI image processing in our previous paper, the same technique is used for enhancing the characters. The detailed procedures along with the results are discussed in the sections shown below

***Keywords* –** Meetei Mayek script, recognition, Preprocessing, PSOFCM segmentation, Neural network

I. INTRODUCTION

The Meetei Mayek [5], [6],[12] is an abugida that is used for the Manipuri language; it is one of the official languages of the Indian state of Manipur and is spoken in the valley region of the state. It contains 18 original letters known as Eeyek Eepee, 9 additional letters known as Lom Eeyek, 8 letters with short ending known as Lonsum Eeyek, 8 vowel signs known as Cheitap Eeyek, 3 punctuation marks known as Khudam Eeyek and 10 numerical characters known as Cheishing Eeyek. The original figures or characters have been derived from human anatomy and named accordingly. A survey of this script can be found in the literature [7],[13],[14]. The problem definition being that previous works were based on recognizing only the ten Meetei mayek numerals using neural network while this paper uses neural network along with advance segmentation to recognize 34 handwritten Meetei Mayek characters.

Image processing [3] is an essential part of our day- to-day life, by using various techniques images which are affected either by noise or whose quality has been degraded can be processed so that the accurate original signal can be revived or can be enhanced to detect more features. Different methods such as enhancing, restoring, compressing, recognition, clustering techniques,[3] etc have been identified so far which can yield better results. Among this pattern recognition is one such approach where the input image is pre-processed properly and then by using certain techniques the machine is made to recognize certain patterns in the image. The application may encompass vehicle number plate recognition, bank cheque signature verification, etc. This paper implements the popular feed forward neural network [4],[7-11],[13] with back propagation learning mechanism for recognizing the segmented digits.

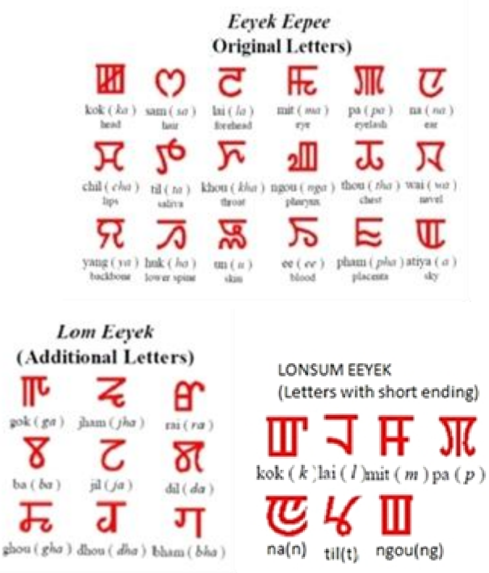


Fig. 1 The 34 Meetei Mayek characters used in this paper [12]

II. PSOFCM SEGMENTATION ALGORITHM

PSOFCM [1] stands for "Particle Swarm optimization incorporating Fuzzy C means algorithm. This method is the platform on which the whole idea of pattern recognition using neural network approach was hatched. Recently it was employed in segmenting MRI Brain images (Fig. 2) where the segmented images allowed easier diagnosis.

The requirement of this step is to classify each of the images into two clusters as it was done for MRI [1],[2]. Classifying an image into cluster improves the feature detection process.

Three samples of original MRI images were taken and performed PSOFCM segmentation whose results are shown in figure 2 for reference purposes. Keeping this idea in mind we thought of using it in this paper and check the results. Briefly speaking, this technique is the combination of Particle Swarm Optimization and Fuzzy C Means, the combination of these two processes seemed to overcome their individual demerits. It can also be noted that PSOFCM based segmentation was proven better than GAFCM based segmentation [2].

It comprises of the following steps:

The algorithm initializes a population (called a swarm) of candidate solutions also known as particles. They are shifted all over the search-space, their movements are guided by the individual best as well as the overall best positions encountered and then updated accordingly. These processes are repeated for a number of generations after which a good result may be hoped though not guaranteed.

Let us consider a N-dimensional search-space S □ R^N and as swarm consisting of I particles. The i-th particle is in effect an N-dimensional vector:

$P_i = (p_{i1}, p_{i2}, p_{iN})^T$ □ S, (in our program we initialized the number of particles in the swarm or swarm size as 20) The velocity of this is also a N-dimensional vector,

$Ve_i = (ve_{i1}, ve_{i2} \ldots ve_{iN})^T$ □ S

The best position of i-th particle (so far) in S is denoted by: $B_i = (B_{i1}, B_{i2} \ldots B_{iN})^T$ □ S

These particles' initial velocities and positions are randomly initialized. Let Y^* be the global best position amongst all the particles, and t be the program iterations. During each iteration, the velocity and position of each particle/swarm is updated using (1) and (2) as given below.

Further in this algorithm we are trying to minimize a function $B_i = f(P_i)$ using Fuzzy c means algorithm. The objective function of the fuzzy c means clustering algorithm is used for achieving the global minima of the clustering objective. In this way by minimizing the objective function value FCM keeps track of the best particle or best position ever.

$$Ve_i(t+1) = W * Ve_i(t) + c_1 r_1 (B_i(t) - P_i(t)) + c_2 r_2 (B^* - P_i(t)) \quad (1)$$

And, $P_i(t+1) = P_i(t) + Ve_i(t+1)$ (2)

Where

$B_i = B_i$, if $f(P_i) \geq f(B_i) = P_i$, if $f(P_i) < f(B_i)$ (3)

B^* □ $\{B_0, B_1 \ldots B_i\}$ such that

$f(B^*) = \min(f(B_0), f(B_1), \ldots.., f(B_i))$ (4)

r_1 and r_2 are elements from two uniform random sequences in the range (0,1).W is called inertia weight matrix representing weights of $Ve_i(t)$ as a contribution to $Ve_i(t+1)$.It is quite clear from (4) that B^* is the global best position amongst all the particles. The value of each component of Ve_i vector is clamped to range $[-ve_{max}, ve_{max}]$ to prevent PSOFCM from leaving the search space. c_1 and c_2 are acceleration co-efficient which controls the displacements of a particle in a single iteration. In this work we used W = 0.5 and $c_1 = c_2 = 2$

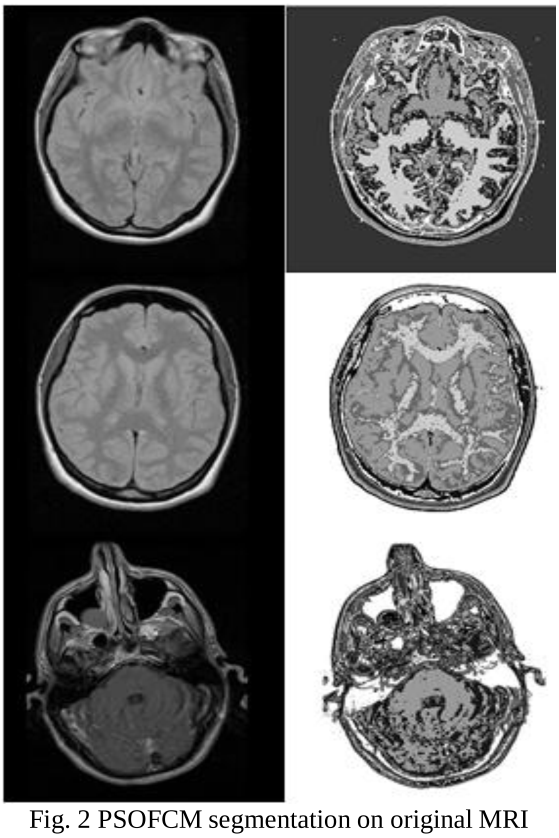


Fig. 2 PSOFCM segmentation on original MRI images.
Left column: Original images
Right Column: Segmented images

III. NEURAL NETWORK

Neural Networks [4] are computationally efficient models inspired by the working of neurons present in the human brain. As for instance, a human being can recognize another person that he has met before within a short time frame, here billions and billions of neuron process a large amount of data at a very high speed. Neural network employs similar techniques for pattern recognition; however there is a limitation to the amount of data it can process and at what speed because of limited number of neurons/resources as compared to the human brain. Although research are still going on to reduce these limitations, so that a proper system that has a very high level of accuracy can be designed.

A set of input neurons may be activated by the pixels of an input image symbolizing a letter or maybe a digit. These activations are passed, weighted and modified using some function according to the network designer or other neurons, upto the instant at which an output neuron is activated that determines which character was read. It consist of sets of adaptive weights that are tuned by a learning algorithm, it is also capable of approximating non-linear functions of their inputs. The adaptive weights are the connection strengths between neurons, which were activated during training and prediction stage. Back propagation is a systematic method of training multilayer artificial neural networks which is built on high mathematical concept and has very good application potential.

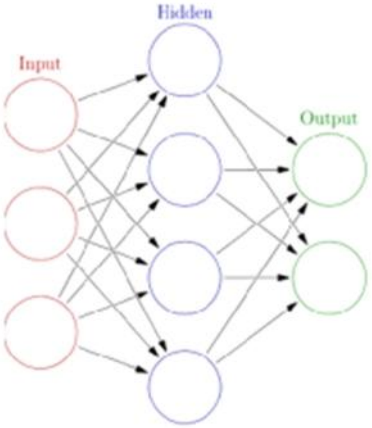
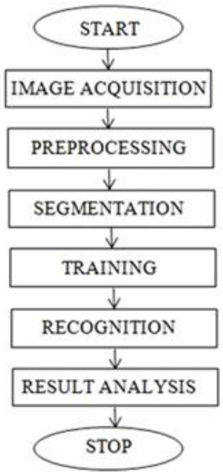


Fig. 3 Multilayer neural network structure

IV. THE PROPOSED RECOGNITION MODEL

A flowchart showing the basic implementation of the whole paper is given in the next column



A. Acquiring the image: A total of 3400 samples were collected from people belonging to different age groups to facilitate the machine to learn different patterns for each of the 34 characters. Out of the total 100 samples for each of the characters it was decided to use:

a) 50 samples * 34 characters = 1700 to train the neural network.
b) 50 samples*34 characters=1700 for testing the performance of the method.

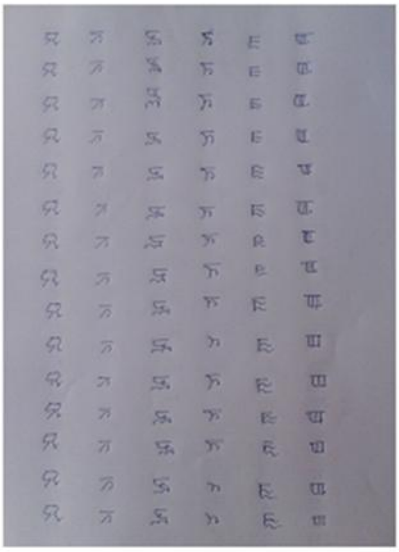


Fig. 4 Scanned image of a handwritten sample

***B. Pre-processing**:* The various methods that are being employed in this paper are listed below steps-by-step:

- Conversion of the input image into grayscale image.
- Thresholding.
- 2-D adaptive noise-removal filtration
- Line segmentation
- Character segmentation

- All the letters have 140X140 pixels because equal number of pixels is essential.

C. Segmentation*:* It is a simple process of partitioning an image into many segments so that the resulting image is much more easily comprehensible. To speak more precisely, pixels that share certain similar traits are brought under the same roof and labels are assigned to many such blocks. Particle swarm optimization incorporating fuzzy c means (PSOFCM) [1] is used to perform the segmentation of each of the characters. Section II of this paper discusses the steps of this algorithm in a nutshell.

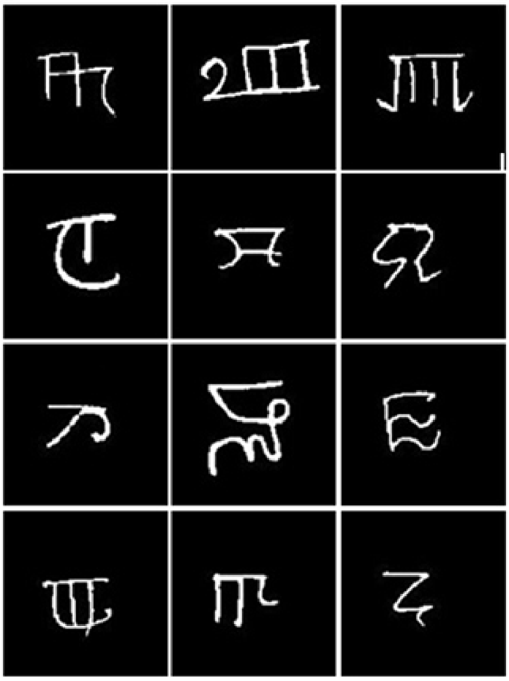
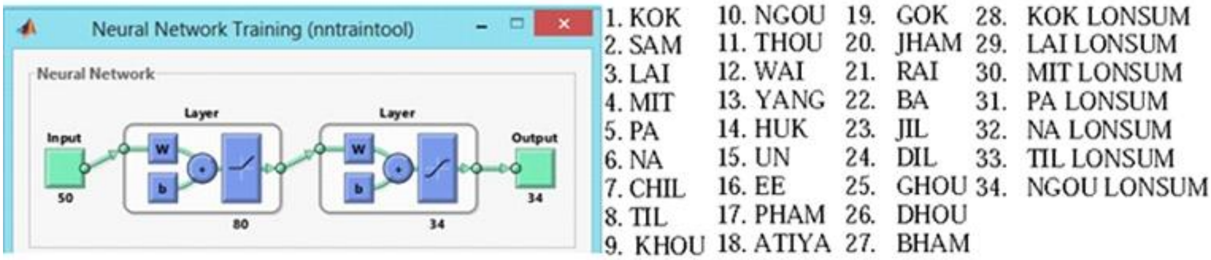


Fig. 5 some sample images of the PSOFCM segmented image

D. NEURAL NETWORK ARCHITECTURE: The feed forward neural network architecture with back propagation learning is used for recognition of the PSOFCM segmented images. Recognition of the 34 letters is done using by the three layers on the network. All the simulations, i.e. all the pre-processing steps, PSOFCM, and the neural network recognition process were done in MATLAB environment. Back propagation neural network with Gradient descent with momentum adaptive learning rate was used. The output vector is composed of 34 neurons as it has to hold 34 classes. Either 0 or 1 is used to indicate the place where the output is observed, 0 is changed to 1 when the particular digit has a high value. The training involves 2 passes: Forward pass in which the input signals propagate from the network input to the output; Reverse pass in which the calculated error signals propagates backward through the network and adjust weights. Traingdx function was used to train the neural network, weights and bias values are adjusted according to the gradient descent and learning rate. Performance goal is 1.00e-
09 and gradient is 1.00e-05 is kept at 0.9. Mean squared error is used to measure the performance.

V. SIMULATION RESULT

The experimental results shown below are obtained directly from the MATLAB environment. Since pronunciations of the letters are difficult we have denoted each of the letters by numbers as shown in figure 6. Table I shows the performance of the network, while Table II shows the recognition matrix after tabulating the result.



1. KOK
2. SAM
3. LAI
4. MIT
5. PA
6. NA
7. CHIL
8. TIL
9. KHOU
10. NGOU
11. THOU
12. WAI
13. YANG
14. HUK
15. UN
16. EE
17. PHAM
18. ATIYA
19. GOK
20. JHAM
21. RAI
22. BA
23. JIL
24. DIL
25. GHOU
26. DHOU
27. BHAM
28. KOK LONSUM
29. LAI LONSUM
30. MIT LONSUM
31. PA LONSUM
32. NA LONSUM
33. TIL LONSUM
34. NGOU LONSUM

Fig. 6 Neural network from the MATLAB environment; Notation of the characters in numeral form

TABLE I : PERFORMANCE EVALUATION

DIGITS	**NO. OF TEST SAMPLES**	**TRUE RECOGNITION**	**FALSE RECOGNITION**	**PRECISION (%)**
1	50	44	6	88
2	50	42	8	84
3	50	45	5	90
4	50	45	5	90
5	50	43	7	86
6	50	40	10	80
7	50	36	14	72
8	50	37	13	74
9	50	37	13	74
10	50	48	2	96
11	50	29	21	58
12	50	27	23	54
13	50	28	22	56
14	50	50	0	100
15	50	42	8	84
16	50	34	16	68
17	50	38	12	76
18	50	37	13	74
19	50	39	11	78
20	50	43	7	86
21	50	41	9	82
22	50	40	10	80
23	50	44	6	88
24	50	43	7	86
25	50	18	32	36
26	50	15	35	30
27	50	15	35	30
28	50	16	34	32
29	50	24	26	48
30	50	28	22	56
31	50	25	25	50
32	50	41	9	82
33	50	47	3	94
34	50	36	14	72

Where, True recognition = Number of times a character is identified correctly,
False recognition = Number of times a character is
Identified
Incorrectly,
Precision = Accuracy in percentage,
1,2,…,3=Meetei Mayek characters

TABLE II : RECOGNITION MATRIX FOR THE 34 CHARACTERS.

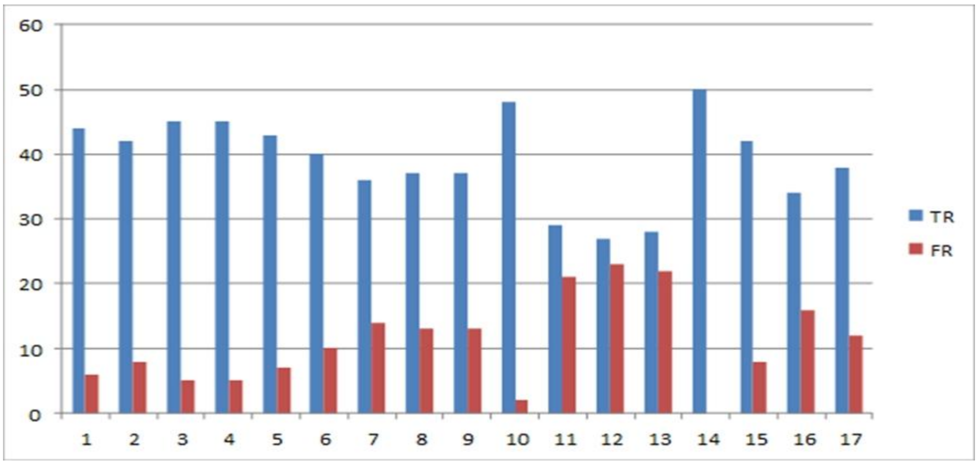
A*	**Characters recognized by the network (B*)**																																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
1	**44**	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0
2	0	**42**	0	1	1	0	0	0	1	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
3	0	0	**45**	0	0	2	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
4	1	0	0	**45**	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0
5	3	0	0	1	**43**	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	1	2	0	0	**40**	0	0	0	0	0	0	1	0	0	0	1	0	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0
7	1	0	1	1	2	0	**36**	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0	0	1	2	0	0	0
8	0	2	0	0	0	0	0	**37**	0	0	0	0	0	0	1	2	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0
9	0	0	0	3	0	0	0	0	**37**	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	2	0	2	0	1	0	3	0	0	0
10	1	0	0	0	1	0	0	0	0	**48**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	0	1	0	0	0	0	0	0	2	**29**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	1	13
12	0	0	0	2	1	0	1	4	0	0	0	**27**	0	0	0	0	0	0	0	0	0	0	0	0	1	4	7	0	0	0	3	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	**28**	0	4	16	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
14	0	0	0	0	0	0	0	0	0	0	0	0	0	**50**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**42**	3	2	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	**34**	7	1	0	4	0	2	0	0	0	1	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	4	**38**	1	0	0	1	0	0	2	2	0	0	0	0	0	0	0	0	0
18	2	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	4	**37**	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
19	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	5	0	2	**39**	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**43**	0	0	7	0	0	0	0	0	0	0	0	0	0	0
21	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**41**	1	0	1	0	0	0	0	0	2	0	0	0	2
22	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	1	1	0	**40**	0	2	0	0	0	0	0	0	0	0	1	1

23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	4	**44**	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	4	0	**43**	0	0	0	0	0	0	0	0	0	0
25	0	0	0	3	0	0	2	2	16	0	0	5	0	0	0	0	4	0	0	0	0	0	0	0	**18**	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	5	0	4	1	0	0	0	0	1	0	0	0	0	0	0	0	19	**15**	5	0	0	0	0	0	0	0
27	0	0	0	0	0	3	0	1	4	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	17	0	**15**	0	5	0	0	0	4	0
28	0	0	0	0	0	3	0	0	0	3	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	**16**	0	1	5	14	4	1
29	0	0	0	0	0	0	0	0	11	0	3	1	0	0	0	3	0	0	0	0	0	2	0	0	4	0	2	0	**24**	0	0	0	0	0
30	0	0	0	3	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	5	8	0	1	**28**	0	2	0	0
31	0	3	0	2	13	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2	1	2	0	0	0	0	0	0	0	0	**25**	0	0	0
32	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	2	0	0	0	0	0	1	1	**41**	1	1
33	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**47**	0
34	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	2	2	3	0	**36**

Where, A* = Characters according to the serial number indicated by Figure 6 which is input to the network.

B* = Output from the network after A* is applied to the network for comparison.

Here, in Table II we have chalked out a table showing the number of times a sample is correctly or wrongly identified as itself or as other characters. It is arranged in matrix format, where the numbers i.e. 1 to 34 arranged in the leftmost column is the sample taken at a particular instant of time, and the serial number i.e. 1 to 34 arranged in the topmost row shows the letter that a particular sample is recognized as by the neural network. This table is known as Recognition matrix.



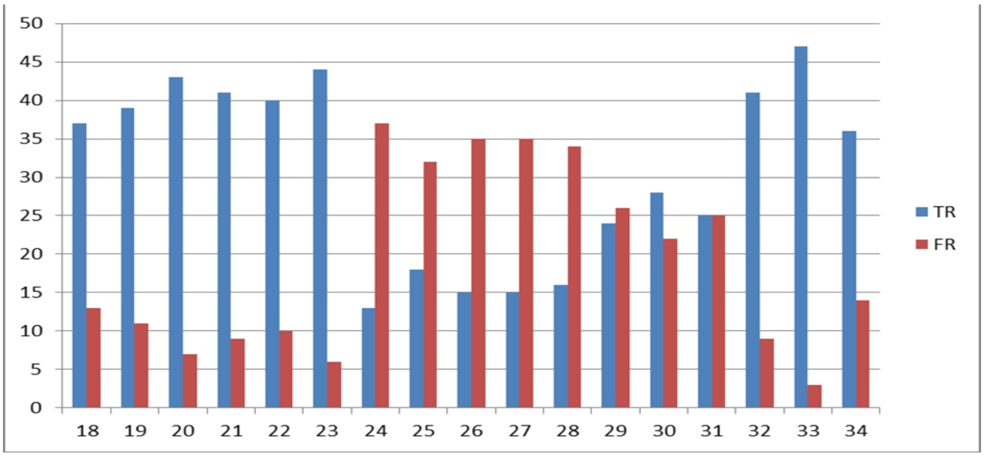


Fig. 7 True Recognition vs. False Recognition characteristics for easy analysis.

Where, along X-axis = Meetei Mayek Characters.

Along Y-axis = Number of times the letters are recognised out of the total 50 attempts for each character.

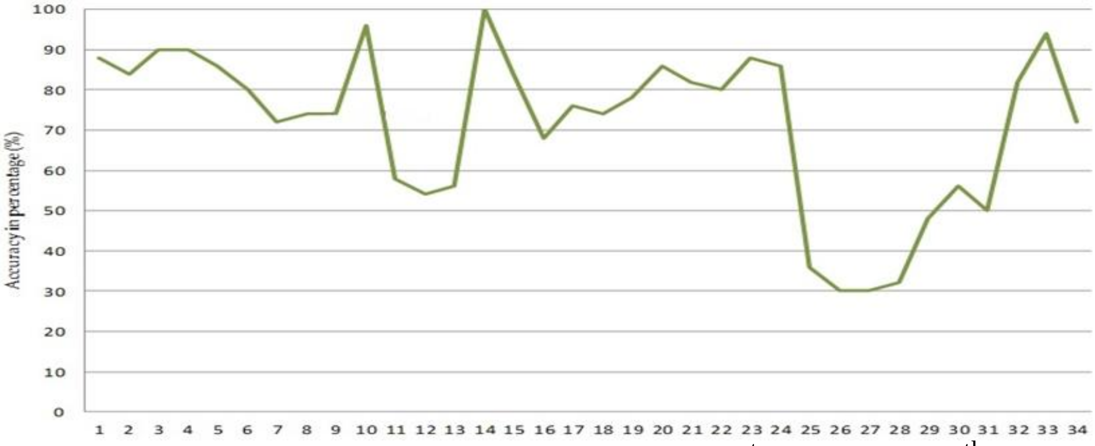


Fig. 8 Graph showing the accuracy in percentage startimg from the 1st character to the 34th character

VI. CONCLUSION

This paper reveals a new and yet powerful approach for classifying the Meetei Mayek characters by using swarm intelligence combined with fuzzy logic and artificial neural network. Although very little work have been done on the Meetei mayek letters, the present technique can be still compared with that of the previous two papers [13],[14] in which the accuracy were 85% and 95.16% respectively, in these papers only ten Meetei mayek numerals [12] were used for recognition.

On comparing this result with that of the previous papers the overall accuracy that we achieved is about 71.59%, which is good if we take into account the massive differences in the ways in which each of the characters are written and also the huge number of characters as compared to the ten numerals used earlier[13],[14]. It can be noted from table I that some characters have higher accuracy (as high as 100%, while the second highest and third highest accuracy value being 88% and 86%).

VII. SCOPE AND FUTURE WORK

The ideas presented through this paper can be implemented in the optical character recognition of other scripts as well. The future work includes implementing new algorithms to improve accuracy for some of the other samples that have accuracy as low as 30% (and also for those having accuracy in 70s, 50s and 60s)by using Neuro-fuzzy technique like ANFIS.

REFERENCE

[1] Romesh Laishram, Wahengbam Kanan Kumar,Anshuman Gupta, Khairnar Vinayak Prakash,"A novel MRI edge detection using PSOFCM segmentation and canny algorithm",*Proc. IEEE sponsored International conference on Electronic Systems, Signal Processing and ComputingTechnologies(ICESC),2014*,pp.398-401,2014,

[2] Romesh Laishram, W.Kanan Kumar Singh,

N.Ajit Kumar, Robndro K.,S. Jimriff "MRI Brain Edge Detection Using GAFCM Segmentation and Canny Algorithm", *International Journal of Advances in Electronics Engineering – IJAEE,volume 2 - Issue 3,* ISSN:- 2278-215X, pp. 168-171,December 8,2012

[3] Rafael C. Gonzalez, Richard E.Woods, "Digital Image Processing", Pearson Education, Second Edition, ISBN 81-7758-168-6, 2005

[4] Amit Konar, "Computational Intelligence Principles, Techniques Techniques and Applications", Springer edition , ISBN 3-540-20898-4 Springer Berlin Heidelberg, New York

[5] Wangkhemcha Chingtamlen , A short history of Kangleipak (Manipur) part-II, Kangleipak Historical & Cultural Research Centre, Sagolband Thangjam Leirak,Imphal,2007

[6] Ng. Kangjia Mangang, Revival of a closed account, a brief history of kanglei script and the Birth of phoon (zero)) in the world of arithmetic and astrology, Sanmahi Laining Amasung Punshiron Khupham (SalaiPunshipham), Lamshang,Imphal,2003

[7] U.Pal & B.B.Choudhuri,, "Indian script character recognition:a survey" *Pattern Recognition***,** 37,2004, pp 1887-1899

[8] Avani R.Vasant, G. R. Kulkarni, "Simulation and Modeling of Handwritten Gujarati Digits using Neural Network Approach,2011 IEEE International Conference on Computational Intelligence and computing

[9] Arindam Basu, "Small-Signal Neural Models and Their Applications", IEEE transactions on biomedical circuits and systems,vol. 6, no. 1, February 2012

[10] S N Sivanandam, S Sumathi, S N Deepa, "Introduction to Neural Networks using MATLAB 6.0 ",Tata McGraw-Hill, Eleventh reprint 2010, ISBN-13: 978-0-07-059112-7, ISBN-10: 0-07-059112-1

[11] Ankit Sharma, Dipti R Chaudhary "Character Recognition Using Neural Network", International Journal Of Engineering Trends And Technology (IJETTE). - Volume 4 Issue 4-April 2013.

[12] www.unicode.org/charts/PDF/UAB0.pdf

[13] Romesh Laishram, Angom Umakanta Singh, N.Chandrakumar Singh, A.Suresh Singh, and H.James, "Simulation and Modeling of Handwritten Meetei Mayek Digits using Neural Network Approach", Proc. of the Intl. Conf. on Advances in Electronics, Electrical and Computer Science Engineering-EEC 2012, PP. 355-358, 2012

[14] Chandan Jyoti Kumar and Sanjib Kumar Kalita, "Recognition of Handwritten Numerals of Manipuri Script", International Journal of Computer Applications, Volume 84, No.17, PP. 1-5, December, 2013