Turkish Journal of Computer and Mathematics Education Vol.12 No.11 (2021), 5983-5591
Research Article

Prehistoric Stone Image Tamil Character Recognition using Optimized Deep Neural Network using Zernike Moments and Simplex Method

R.Prabavathi[1], J.Shiny Duela[2] V.Brindha Devi[3] S.SenthilPandi[4]
[1,4]Assistant Professor, [2,3] Associate Professor
[1,3] Department of Information Technology, Sri Sairam Institute of Technology, Chennai, India
[2] Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India
[4] Department of Information Technology, Mohamed Sathak A J College of Engineering, Chennai, India
E-mail:[1]prabavathi.raman@gmail.com

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 10 May 2021

Abstract— Prehistoric Tamil character recognition is an important field of research in pattern recognition and it is a technical challenge than other languages in respect to the similarity and complexity of characters. The main focus of this paper is to determine characters from any given text of Tamil consonants and vowels taken from the stone images. The research challenge in recognizing Tamil character is mainly because of the characters consisting of the number of holes, loops and curves. Even though there are various approaches provided by the researchers, challenges and issues still prevail. The proposed system overcomes the issues behind the Tamil character recognition and provides an improved approach. This system takes a challenge to recognize prehistoric Tamil characters using Deep Neural Network. Pre-processing is done by Binarization, De-noising, character segmentation and size normalization for the stone images and then goes for Feature extraction which forms the basic underlying part of recognizing each character. Characters are then classified by Back Propagation deep Neural Network. The optimization of neural networks is done using simplex method during back propagation with improved data set.

Keywords—Prehistoric stone images, Binarization, De-noising, Character Segmentation, Size Normalization Zernike moments, Deep Neural networks, Simplex method

I. INTRODUCTION

Most of the stone inscriptions that are found across different regions of the world reveal the details of lavishness, lifestyle, economic condition, culture, and also of the managerial regulations followed by various rulers and dynasties particular to those regions. The

Tamil script has 12 vowels and 18 consonants and one character that has been classified as vowel or consonants which together consist of 31 letters in them independent form and 216 letters in combinations summing up to 247. Lot of research works exists in the recognition of Tamil characters but accuracy is not obtained. Secondly only few character sets are considered. The main challenge in the recognization of Tamil character is because of the complexity in identifying curves, complex letter structure, variations in the letters, increased number of strokes and holes etc. The inscriptions considered for this work is taken from temple stones which represents the eight centuries. The inscriptions found depict various south Indian languages but this concentrates on Tamil characters. The system includes the following stages: image pre-processing, segmentation, feature extraction and classification. The process of ancient Tamil character recognition involves extraction of some defined characteristics called features to classify an unknown character into one of the known classes. Pre-processing is primarily used to reduce variations of handwritten characters. Segmentation is used for decomposition of the stone image into individual characters which makes the system to give more accurate results. A feature extractor is essential for efficient data representation and extracting meaningful features for later processing. The extracted features are then fed to the deep neural network for classification. Based on the classifications made the characters of the eighth centaury are recognized with the equivalent Tamil characters. The neural networks

are trained using backpropagation algorithm and simplex method is used for optimization of the delay caused during the training.

The rest of the paper is organized as follows; Section II describes the related works. The proposed work is explained in the other sections. Pre-Processing of the system is given section III. Section IV discusses about the features extracted. Classification and recognition are discussed in section V. In section VI performance of the system is discussed and finally VII concludes the work with future works.

II. RELATED WORK

Eighth century Tamil consonants are recognized through an artificial immune system [1]. This paper discusses about the smoothening, normalization and segmentation, the recognition of the system is based on artificial immune system. The results are obtained through stimulation results. [2] Explains the font reorganization on a single Chinese character. Naveen Kumar Talla and E.V Ramana proposed a system for enhancement of south Indian inscriptions using De-noising and character spotting techniques. The work of this system helps the epigraphers in minimizing the confusion created by manually interpreting the inscriptions. [3] S Venkata Krishna Kumar Poornima T V proposed a system to read the ancient Tamil characters belonging to various periods. The examined characters are taken from the script automatically and coordinate with the characters belonging to different periods using machine intelligence [7]. An enhancement on inscription images by Indu shridevi has discussed about the NGFICA based system which recognizes word and character improving the efficiency of OCR [5]. The proposed system recognizes ancient Tamil character from temple stone by using artificial neural network. The system goes for preprocessing of the stone images and then for segmentation, classification and recognition through the neural networks. The Experimental results show that the machine has successfully recognized the alphabets with the average accuracy which is significant and may be acceptable in some applications. Paper [6] proposes a system which uses Zernike moments. A Convolutional Neural Network model from the ground up is built, training it in offline mode with Tamil characters and achieving good recognition results on both the training and testing datasets is done [12]. Using deep learning techniques, this work attempts to set a benchmark for offline Handwritten Tamil Character Recognition. The twelve vowels in palm leaf manuscripts are recognized using B-spline curve recognition. Uniqueness and robustness are two advantages of the B-spline curve. In Tamil, each vowel has one or more curves of varying angles. The vowels are recognized by recognizing the combination of curves [13].

III. PRE-PROCESSING

A. Binarization

Binarization is the process of separation of pixel values into two colors as black and white (0 &1). [11] In image enhancement or analysis problem, it is necessary to identify the objects of interest from rest. It converts the gray scale image to binary image.

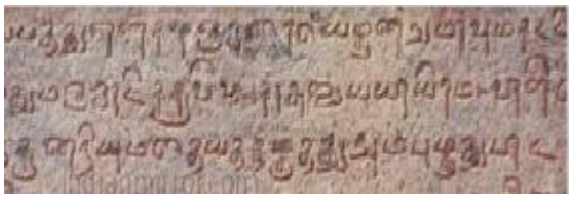

Fig 1: Eighth century stone inscription

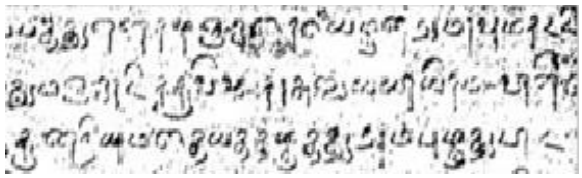

Fig 2: Binarized image

The algorithm uses Binarization technique that calculates an optimal threshold is for separating the two colors i.e. the two classes of pixels, (e.g. foreground and background) separating those two classes so that their combined spread (within-class variance) is minimal. If T is the global threshold of an image f(x,y) and g(x,y) is the threshold image then

$$G(x,y) = \begin{cases} 1, & if\ f(x,y) \geq T \\ 0, & otherwise \end{cases} \quad (1)$$

B. Denoising

Denoising is an important step for removal of noise from the stone images. Removal of noise based on filtering techniques is divided as spatial (mean, median and wiener filter) and frequency (Butterworth and Gaussian filter). This system uses wiener filter also known as minimum mean square error filter to remove noise. It the adaptive linear filter applied to an image locally, by taking into account the local image variance. When the variance in an image is large the Wiener filter results in light local smoothing, while when the variance is small, it gives an improved local smoothing.

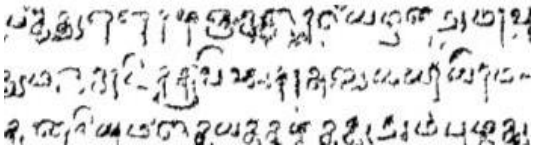

Fig 3: Denoising-Filtering

The minimum mean square error MSE is given by trace and covariance matrix where the estimated error vector is given by

$e = (\hat{x} - x)$ and its mean squared error is given as

$$MSE = tr\{E\{(\hat{x} - x)\ \widehat{(x - x)}^T\}\} \quad (2)$$

C. character segmentation

Character segmentation is considered one of the main steps in preprocessing especially in cursive scripts such as Arabic, Urdu and other scripts where characters are connected together. Therefore, there are many techniques developed for character segmentation and most of them are script specific and may not work with other scripts. Three types of segmentation are:

1. line segmentation
2. word segmentation
3. character segmentation

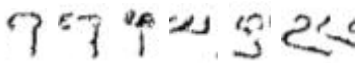

Fig 4: Segmented image for feature extraction

Since the system recognizes single characters from the image so here the image goes for separation between lines, Spacing between the words and characters.

D. Size Normalization

The result from the character segmentation stage provides isolated characters which are ready to be passed into the feature extraction stage; therefore, the isolated characters are normalized into a specific size, decided empirically or experimentally depending on the application and the feature extraction or classification techniques used.

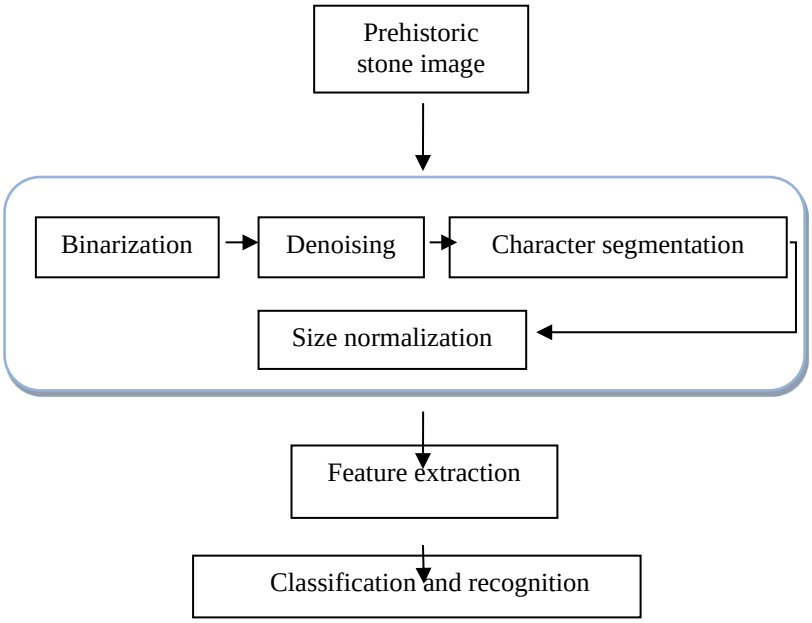

Fig 4: Overview of the proposed system

Then features are extracted from all characters with the same size in order to provide data uniformity.

The size normalization is

1. The aspect ratio of the image is taken to normalize the image to bigger size.
2. Size normalization is done using bilinear interpolation algorithm.

$$y(x_1,x_2) = \sum_{i=0}^{3}\sum_{j=0}^{3} C_{ij}\, t^i\, u^i \qquad (3)$$

$$y_1(x_1,x_2) = \sum_{i=0}^{3}\sum_{j=0}^{3} i\, C_{ij}\, t^{i-1}\, u^i\, (dt/dx_1) \qquad (4)$$

$$y_2(x_1,x_2) = \sum_{i=0}^{3}\sum_{j=0}^{3} j\, C_{ij}\, t^{i-1}\, u^i\, (du/dx_2) \qquad (5)$$

$$y_{12}(x_1,x_2) = \sum_{i=0}^{3}\sum_{j=0}^{3} i\, j^3 C_{ij}\, t^{i-1}\, u^{j-1} \qquad (6)$$

i=0 j=0

The algorithm reduces some of the visual distortion caused by resizing an image.

IV. FEATURE EXTRACTION

Feature extraction phase extracts the basic components of the Tamil character. The system uses Zernike moments to extract features. The character image is divided into 3x3 zones. From each zone features are extracted to form the feature vector. The goal of zoning is to obtain the local characteristics instead of global characteristics.

A. Zernike moments

Feature extraction is the process of extracting relevant features of the characters to form feature vectors which are used by classifiers for the recognition process. The Zernike moments are projections of the input image onto the space spanned by orthogonal function. A set of complex orthogonal polynomials are used. [10] Zernike moment is overcome the shortcomings of information redundancy present in the geometric moments. Zernike moments are rotation invariant. Moment functions are defined on images as the weighted sums of the image intensity function. Moment functions of order (p + q) are generally defined as

$$\Phi_{pq} = \int_x \int_y \psi_{pq}(x, y)\, f(x, y)\, dxdy \quad (7)$$

Zernike moments are constructed using a set of complex polynomials which form a complete orthogonal set on the unit disk with $(x^2+y^2) = 1$. Z_{mn} is given as follows:

$$Z_{mn} = m+1/\pi \int\int I(x, y)\, [V_{mn}(x, y)]\, dxdy \quad (8)$$

Where m and n define the order of moment and I(x, y) the gray level of a pixel of image I on which the moment is calculated. The Zernike polynomials $V_{mn}(x, y)$ are expressed in polar coordinates as follows:

$$V_{mn}(r, \Theta) = R_{mn}(r)\, e^{-jn\Theta}$$

Where $R_{mn}(r)$ is the orthogonal radial polynor (9)

$$R_{mn}(r) = \sum_{s=0}^{m-|n|} (-1)s \frac{(m-s)!}{s!\left[\frac{m-|n|-s}{2}\right]!\left[\frac{m-|n|-s}{2}\right]!}\, r^{m-2s}$$

(10)

Moments Z_{mn} are invariant under rotation and scale changes.

B. Invariance properties

Translation invariance of pseudo-Zernike moments can be achieved by translating the input image by its geometric moments, so that both m_{01} and m_{10} is zero on the resulting image. Similarly scale invariance is achieved by normalizing the image, so that the total area of foreground pixels is of a predetermined value β. Because pseudo-Zernike moments are defined on the unit disk, they are unaffected by rotation, as rotating the image only changes the phase angle of moments.

Rotation by an angle of ϕ can be written as:

$$Z_{pq} = Z_{pq} e^{-iq\phi} \quad (11)$$

Using the following transformation, both translation and scale invariance can be achieved, where β is an arbitrary constant value, specifying the overall area of the normalized image.

$$G(x,y) = f\left(\frac{\bar{x}}{a} + x, \frac{\bar{y}}{a} + y\right), \quad (12)$$

Where $\bar{x} = \frac{m_{10}}{m_{00}}$ and $\bar{y} = \frac{m_{10}}{m_{00}}$ $\quad a = \sqrt{\frac{\beta}{m_{00}}}$

The central moments can be calculated by the following formulas: $\mu_{00} = m_{00}$; $\mu_{10} = 0$; $\mu_{01} = 0$;

V. CLASSIFICATION AND RECOGNITION

A. Training the network:

Training your neural network to produce the correct outputs for the given inputs is an iterative process, in which you repeatedly present the network with an example, compare the output on this example (actual output) with the desired output (target output), and adjust the weights in the network to generate better output the next time (i.e., output that is closer to the correct answer). By training the network over and over with various images, and using the Backpropagation algorithm to adjust the weights, the network should learn to produce the correct answer.

B. Backpropagation Algorithm:

The general idea with the backpropagation algorithm is to use gradient descent to update the weights so as to minimize the squared error between the network output values and the target output values. The update rules are derived by taking the partial derivative of the error function with respect to the weights to determine each weight's contribution to the error. Then, each weight is adjusted, using gradient descent, according to its contribution to the error. Without going into the actual derivations here, the text is found in other sources. [9] This process occurs iteratively for each layer of the network, starting with the last set of weights, and working back towards the input layer.

Even though algorithms exist for the automatic configuration of the layer structure during training, in our case the number of input samples makes such approaches impossible, as training would take a very long time. Instead, the configuration had to be found using parameter tweaking, and partially training the network. As a result, a standard 3-layer back propagation network was used, having 16 × 16 input values, 256 neurons in the hidden layer and 1945 outputs. Another important design consideration for neural networks is how to represent the output values. In our case an obvious solution is to use a unique index identifying each of the input images. However, using a single value as the output value has several problems, first of all the network cannot generate values outside the [0, 1] or [−1, 1] range. Secondly, even if we map the indices into the said region, at most only two possible indices can be generated; those that are closest to the output value. To overcome this problem, we have to use a binary vector representing the appropriate indices. Each element of the output vector specifies whether the current index is the same as the input image.

C. Optimization using Simplex method:

Linear programming is used for optimizing a linear objective function which are subjected to constrains in form of linear inequation. It is used for solving industry related problems for example maximize the profit with the constrains of the resources. Our aim here is to maximize or minimize the objective function which is subjected to constrains. Objective function can be profit function which need to be maximized or cost function which need to be minimized. In linear programming problems there are two components one is objective function and second is a set in constrains in form in linear in-equations. Simplex algorithm is one of the algorithms which is used for solving this type of problem. In this paper we present an algorithm to train neural network using simplex method. This new algorithm does not resemble the backpropagation algorithm.

VI.PERFORMANCE OF THE SYSTEM

The comparison of the proposed system with other recognition system and the results produced by the proposed system is discussed here

A. Analysis with Other System

Table1. Shows the various works for recognition where each system takes different methods of producing the results. But still challenges prevail like more data sets are not taken and variants in character still exists. The proposed system takes the methods with produces accurate and efficient results.

Table1. Comparison with previous system

S.No	Recognition System	Results obtained
1	Character recognition user neural networks	85%
2	Recognition using improved feature extraction	Improved features provide optimum performance.
3.	Eighth century character recognition	85%
4.	Enhancement of south Indian inscriptions	Minimizes the confusion created by manually interpreting the inscriptions
5.	Zernike moments and neural networks for Arabic characters	98.27% with 116 images

B. Proposed System Results

The methods chosen for the proposed system is comparatively yields efficient and accurate results. The Prehistoric Tamil characters are recognized by extracting Zernike features and using deep neural network as classifier and optimization is done using simplex method.

Table2. Recognition and error rates The result obtained has produced 92.8% recognition
Rate and 1.32% of error rate are shown in table2.

METHODS USED	METHODS USED	Results + Error rates

Preprocessing	Binarization, Denoising, character segmentation, size normalization	
Feature extraction	Zernike moments	92.8% 1.32%
Classification	Multilayer feed forward neural networks with back propagation	

Linear programming is used for simplex method.

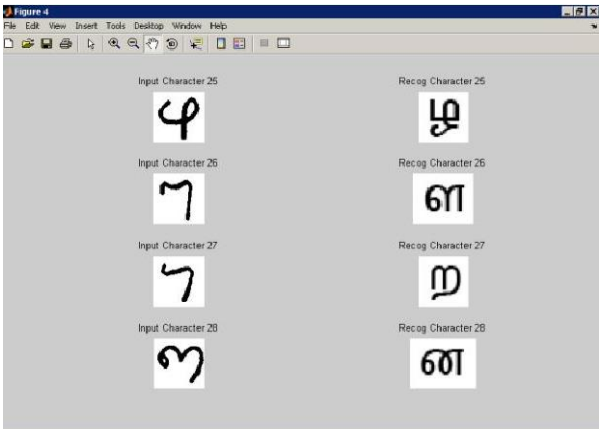

Fig 5: Recognition of Prehistoric Tamil characters

Fig 5 shows the Prehistoric Tamil characters are recognized by extracting Zernike features and using optimized deep neural network classifier. The result obtained has produced 92.8% recognition rate and 1.32% of error rate.

VII. CONCLUSION

The system recognizes prehistoric Tamil characters by extracting Zernike features with neural networks. The results obtained for recognition of pre-historic characters using Zernike moments shows that reliable classification is possible using deep neural networks. The proposed work uses simplex method for optimization of the deep neural networks in order to achieve accuracy. The challenges that prevail in the previous systems are reduced in this system by increased dataset and an optimized neural network method are chosen for recognition. Comparative study is made with the previous system and the proposed system with recognition rate and error rate is given. The result produced gives the efficiency and accuracy of the system.

The results also indicate the scope for further improvement, especially in the case of Tamil character recognition. Future work is directed towards incorporating a database of words for spell check at word level.

REFERENCES

[1] S. RajaKumar, V.Subbiah Bharathi "Eighth century Tamil consonants recognition from stone inscriptions" International conference on Recent Trends in Information Technology(ICRTIT), Pg 40-43, April 2012.

[2] Xiaoqing Ding, Tao Wu "Character Independent Font Recognition on a Single Chinese Character" IEEE Transactions on pattern analysis and machine intelligence, Vol.29, N0.2, pp-195-204, February 2007.

[3] Naveen Kumar Talla and E.V Ramana "Enhancement of South Indian Inscriptions Images Using De-noising and Character Spotting Techniques" International journal of Professional engineering studies, Vol 4, No. 2, pp-202-207, October 2014.

[4] M. V. Jeya Greeba, G.Bhuvaneswari "Recognition of Ancient Tamil character in Stone Inscription using Improved feature extraction" Vol 2, No.3, pp 38-41, February 2014.

[5] Indu, Sreedevi, Pandey, Rishi et al. "Enhancement of inscription images" National Conference on Communications (NCC), pp 1-5, February 2013.

[6] Seethalakshmi et.al, "Optical Character Recognition for printed Tamil text using Unicode", Journal of Zhejiang University SCIENCE, Vol 6A 11, pp-1297-1305, September 2005.

[7] S Venkata Krishna Kumar1 Poornima T V "An Efficient Period Prediction System for Tamil Epigraphical Scripts Using Transductive Support Vector Machine" International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 9, pp-7999-8002, September 2014.

[8] Mohana, H.S, Navya.K, Rajithkumar, B.K, et.al "Iterative Segmentation for character extraction in stone inscriptions" 2nd International Conference on Current Trends in Engineering and Technology (ICCTET), pp 321-327, July 2014.

[9] Lynne E. Parker, "Multilayer feedforward neural networks" , E. Alpaydin's book Introduction to Machine Learning, MIT Press, 2004; and T. Mitchell's book Machine Learning, McGraw Hill, 1997.

[10] Dhir. R, "Moment Based Invariant Feature Extraction Techniques for Bilingual Character Recognition" 2nd International Conference on Education Technology and Computer (ICETC), vol 4, pp 80-84, June 2010.

[11] T.S Suganya, Dr.S.Murugavalli "Binarization of Ancient Tamil Scripts from Stone Inscription" International global journal for research analysis, Vol 3, No. 11, pp 72-74, November 2014.

[12] B.R. Kavitha, C. Srimathi, "Benchmarking on offline Handwritten Tamil Character Recognition using convolutional neural networks",Journal of King Saud University - Computer and Information Sciences, June 2019.

[13] Suganya Athisayamani, A. Robert Singh, T. Athithan,Recognition of Ancient Tamil Palm Leaf Vowel Characters in Historical Documents using B-spline Curve Recognition,Procedia Computer Science, Volume 171, Pages 2302-2309, January 2020.