Handwritten Tamil Character Recognition using Artificial Neural Networks

P. Banumathi[1] and Dr. G. M. Nasira [2]

Abstract--- This paper proposes an approach to recognize handwritten Tamil Character recognition. Handwritten Tamil character recognition refers to the process of conversion of handwritten Tamil character into printed Tamil character. It is difficult to process handwritten characters due to the great variations in writing styles, different size and orientation angle of the characters. In the proposed system the scanned image is preprocessed and segmented into paragraphs , paragraphs into lines, lines into words and words into character image glyph. Each character image glyph is subjected to feature extraction procedure, which extracts the features such as character height, width, number of horizontal and vertical lines, horizontally and vertically oriented curves, number of circles, number of slope lines, image centroid and special dots.

Keywords : Artificial Neural Network (ANN), Handwritten Tamil character recognition(HTCR) , Self Organizing Maps(SOM), Preprocessing , Segmentation, Feature extraction.

I. INTRODUCTION

The character recognition is the most challenging and tantalizing field, because the big research and development effort that has gone into it has not solved all commercially urgent and intellectually interesting problems[1]. Handwritten character recognition is the task of transforming a language represented in its own spatial form of graphical marks into a symbolic representation [3].

For more than thirty years researchers have been working on handwritten recognition. The ultimate goal of designing a handwritten recognition system with an accuracy rate of 100% is quite illusionary, because even human beings are not able to recognize every handwritten text without any doubt. Handwritten character recognition can be divided into two categories i.e. online handwritten character recognition and offline handwritten character recognition. Online handwritten character recognition deals with automatic conversion of characters, which are written on a special digitizer, tablet PC or Personal Digital Assistant (PDA) where a sensor picks up the pen tip movements as well as pen up / down switching. Handwritten Tamil character recognition deals with a data set, which is obtained from a scanned handwritten document.In this paper an Neural Network using Self Organizing Map is used to train and identify offline handwritten Tamil characters. The handwritten document is scanned using scanner and save it in TIF, JPG or GIF format. Pixel processing is first performed after the document is scanned. Pixel processing includes binarization, noise reduction and segmentation. Transforming gray scale images into black & white images is called binarization. Noise occurs from image transmission, photocopying, or degradation due to aging. Noise reduces by performing filtering on the image. Segmentation is performed on the scanned document. A Neural Network classifier finds the likelihood's which are used as input to a dynamic programming algorithm, which recognizes the entire character[1].

Tamil is an ancient language with a rich literary tradition. The alphabet set of Tamil splits into set of vowels, consonants, composite letters, and special letter. There are 12 vowels, 18 consonants, 216 composite letters and 1 special character (ak).

II. ARTIFICIAL NEURAL NETWORK

Artificial neural networks (ANN) have been developed as generalizations of mathematical models of biological nervous systems An ANN is a system based on the operation of biological neural networks. ANN has a good pattern recognition engines and robust classifiers. They have the ability to generalize by making decisions about imprecise input data. They also offer solutions to a variety of classification problems such as speech, signal and character recognition. The basic architecture consists of three types of neuron layers: input, hidden, and output layers. In feed-forward networks, the signal flow is from input to output units, strictly in a feed-forward direction. The data processing can extend over multiple units, but no feedback connections are present. Recurrent networks contain feedback connections[1].

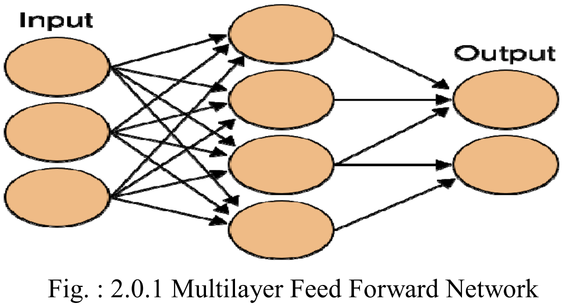


Fig. : 2.0.1 Multilayer Feed Forward Network

P. Banumathi is with the Department of Computer Science & Engineering in Kathir College of Engineering Neelambur Coimbatore (Dt) E-mail: banumathi_mohankumar@yahoo.com

Dr. G. M. Nasira is with the Government Arts College , salem. – 7. E-mail: nasiragm99@yahoo.com

978-1-61284-764-1/11/$26.00 ©2011 IEEE

III. The Proposed System Architecture

This handwritten Tamil Character Recognition system consists of various stages like Scanning, Preprocessing, Segmentation, Feature Extraction, Self Organizing Map classification and Recognition which is shown in the figure 3.0.1.

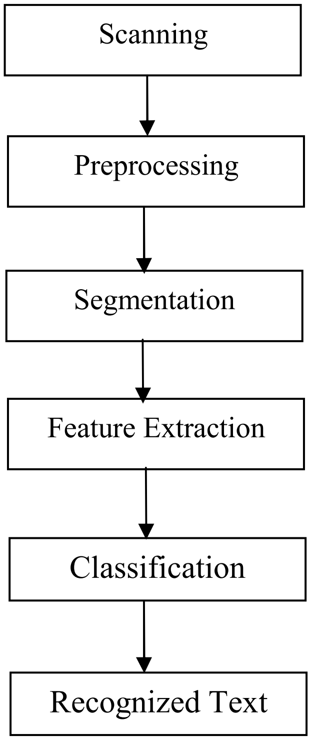


Fig. : 3.0.1 System Architecture

3.1. Scanning

Scanning is the process of obtaining a digitized image from a real world source. A handwritten document is chosen for scanning. Each step in the scanning process may introduce random changes into the values of pixels in the image. These changes are called *noise.* The document is sent to a program that saves in TIF, JPG or GIF format.

3.2 Preprocessing

Preprocessing is the first step in the processing of scanned image. The preprocessing consists of three main steps. They are Binariztion, Noise removal and Skew correction. There are two peak values available. A high peak corresponding to the white background and a smaller peak corresponding to the foreground. The binarized image is pre processed for noise removal. Noise may be due to the poor quality of the document or accumulated while scanning. This noise should be removed before further processing. After removing the noise the resultant image is checked for skewing. There are possibilities of image getting skewed with either left or right orientation. Here the image is first brightened and binarized. The function for skew detection checks for an angle of orientation between ±15 degrees and if detected then a simple image rotation is carried out till the lines match with the true horizontal axis [2], which produces a skew corrected image as given in figure 3.2.0.1.

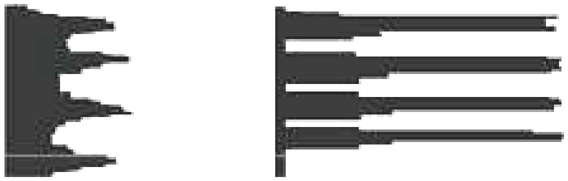
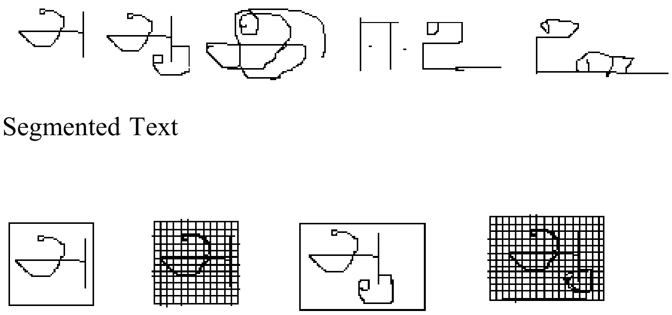


Fig. : 3.2.0.1 Histograms for skewed and skew corrected images

3.3 Segmentation

After pre-processing, the noise free image and skew corrected image is passed to the segmentation phase, where the image is decomposed into individual characters[2].

Original Text



Segmented Text

Fig. : 3.3.1 Original and Segmented Text

The goal of image segmentation is to cluster pixels into salient image regions. Segmentation extracts meaningful regions for analysis. A poor segmentation process produces mis-recognition or rejection. The binaries image is checked for inter line spaces. If inter line spaces are detected then the image is segmented into sets of paragraphs across the interline gap. The lines in the paragraphs are scanned for horizontal space intersection with respect to the background. Histogram of the image is used to detect the width of the horizontal lines. Then the lines are scanned vertically for vertical space Intersection. Page layout analysis and character separation are used to segment sub-words from the preprocessed image. The segmentation between lines of text is determined by scanning through the profile from the first row. If the difference of the number of black pixels between two rows is larger than a predefined threshold, a new line of text is indicated .The next large variation in the number of black pixels between another two rows indicates the bottom of the line. Sub-words are segmented from a line-segmented image in a similar method. Then the words are decomposed into characters using character width computation. Each character is scaled to fit into a 64x64 window and then thinning algorithm is applied to obtain the thinned / image, which is used in feature extraction. The figure 3.3.0.1 shows a character before thinning and after thinning.

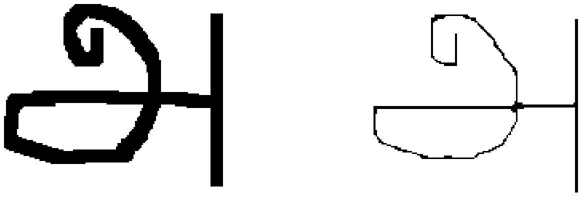


Fig. : 3.3.0.1 Character before and after Thinning

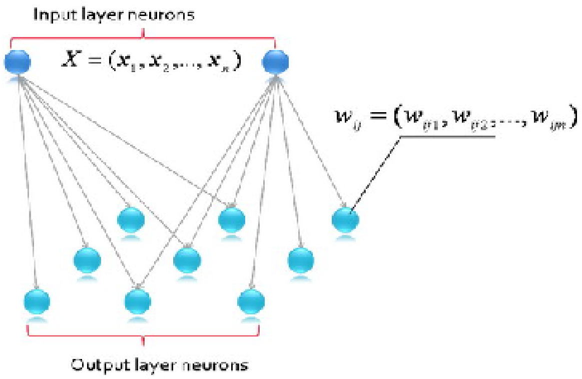
3.4 Feature Extraction

The next phase to segmentation is feature extraction where each character is represented as a feature vector, which becomes its identity. Feature extraction forms the backbone of the recognition process. The major goal of feature extraction is to extract a set of features, such as height of the character, width of the character, number of short and long horizontal lines present, number of short and long vertical lines present, number of circles present, number of horizontally and vertically oriented arcs, centroid of the image and pixels in the various regions of the character which maximizes the recognition rate.

3.5 Kohonen's Self Organizing Feature Map (SOFM)

The process of classification of documents was carried out in 3 phases. The first phase is document preprocessing. The second phase is the training process. The third phase is the test phase in which a document is classified and the weights of neighboring units are updated.

Kohonen's SOFMs are a type of unsupervised learning. The goal is to discover some underlying structure of the data. With this approach an input vector is presented to the network and the output is compared with the target vector. If they differ, the weights of the network are altered slightly to reduce the error in the output. This is repeated many times and with many sets of vector pairs until the network gives the desired output. The network is created from a 2D lattice of 'nodes', each of which is fully connected to the input layer. Fig.3.5.0.1 shows a very small Kohonen's network of 4 X 4 nodes connected to the input layer representing a two dimensional vector. All neurons in the output layer are well connected to adjacent neurons by a neighborhood relation depicting the structure of the map. Generally the output layer can be arranged in rectangular or hexagonal lattice.


Input layer neurons
$X = (x_1, x_2, \ldots, x_n)$
$w_{ij} = (w_{ij1}, w_{ij2}, \ldots, w_{ijm})$
Output layer neurons

Lots of activities in pre-processing stages helps to process this stage very easy. Self-organizing feature maps (SOFM) are unsupervised machine learning that learns by self-organizing and competition [20]. The main idea for this is to make it simple and acceptable for Kohonen SOM. It reduces a remarkable amount of time. SOM is clustering the input vector by calculating neuron weight vector according to some measure (e.g. Euclidean distance), thus weight vector that closet to input vector comes out as winning neuron. However, instead of updating only the winning neuron, all neurons within a certain neighborhood of the winning neuron are updated using the Kohonen rule [20]. The algorithm is described as follows, suppose the training set has sample vectors X, trains the SOM network has following steps:

1. Each node's weights are initialized.
2. A vector is chosen at random from the set of training data and presented to the lattice.
3. Every node is examined to calculate which one's weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU).
4. The radius of the neighborhood of the BMU is now calculated. This is a value that starts large, typically set to the 'radius' of the lattice but diminishes eah time-step. Any nodes found within this radius are deemed to be inside the BMU's neighborhood.
5. Each neighboring node's weights are adjusted to make them more like the input vector. The closer a node is to the BMU; the more its weights get altered.
6. Repeat step 2 for N iterations.

i) Firstly, all neuron nodes weights, defined as W j (1), j = 1…L, are initialized randomly. L is the number of neurons in the output layer.

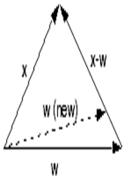
ii) K =Maximum (X,(k)), for iteration step k=1...K, get an input vector X(k) randomly or in order.

iii) Calculate Distance = X(k), *k* = 1…n , 1…n refers to neuron nodes.

iv) Select the winner output neuron j * with minimum distance.

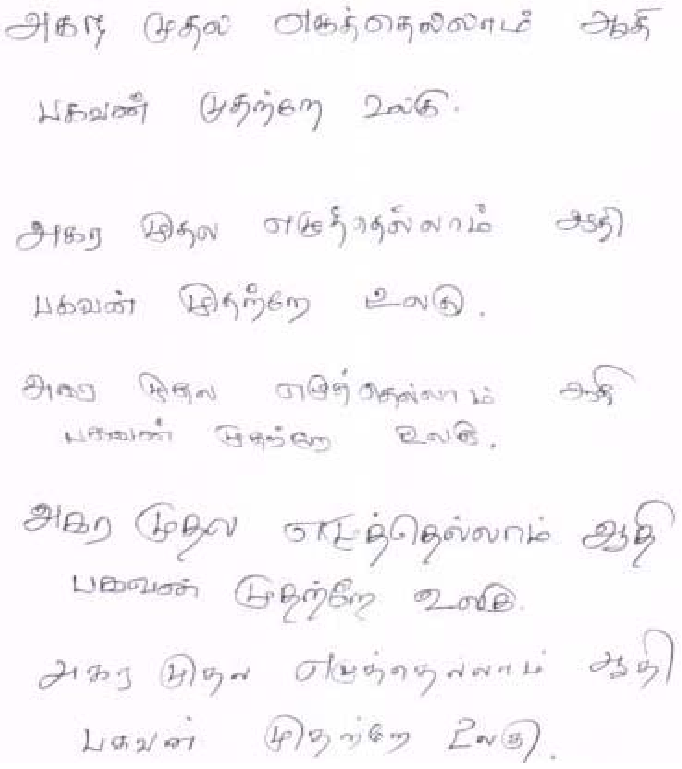
v) Update weightṣ W j(k+1) to neurons j * and its neighborhood
W j(k+1) = W j(k) + [(k+1) ∩ (j, j*(k+1),(k+1)][X(k+1) ⬍ W j (k)], j=1…..L

vi) If *k*=̣*K* go to step (ii).



IV. Findings Of The Proposed System

To test the effect handwriting style has on character recognition with this system, samples from the five persons were scanned and converted to a series of Matlab vectors.To test in an environment where 100% accuracy was obtainable, only the first 8 letters of each sample were used. This also reduced the amount of time and processing power needed to run the experiment. Each character image was converted to a Matlab vector three times, each time in a slightly different position. Letters from the sentence in the handwriting sample were used to create the test set to determine accuracy. In our first article this system was implemented for handwritten English character recognition(ie for word) using Hidden Markov Models.

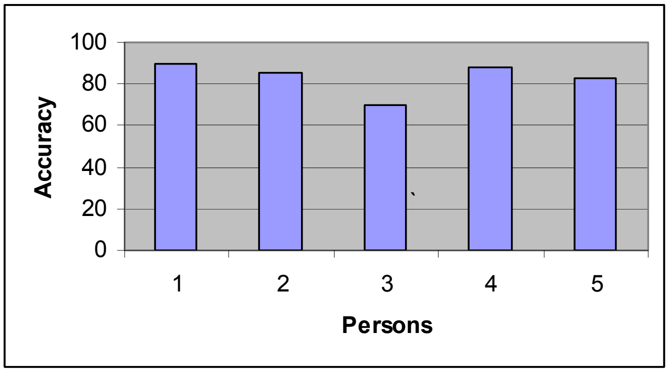

அகர முதல எழுத்தெல்லாம் ஆதி
பகவன் முதற்றே உலகு.

அகர முதல எழுத்தெல்லாம் ஆதி
பகவன் முதற்றே உலகு.

அகர முதல எழுத்தெல்லாம் ஆதி
பகவன் முதற்றே உலகு.

அகர முதல எழுத்தெல்லாம் ஆதி
பகவன் முதற்றே உலகு.

அகர முதல எழுத்தெல்லாம் ஆதி
பகவன் முதற்றே உலகு.



V. Conclusions

We investigated a new representation of Tamil Character Recognition, and used Kohonen SOM techniques efficiently classifies handwritten and also Printed Tamil characters. More effective and efficient feature detection techniques will make the system more powerful. There are still some more problems in recognition. They are, during letter segmentations and abnormally written characters (which misguide the system during recognition). Misrecognition could be avoided by using a word dictionary to look-up for possible character composition. The presence of contextual knowledge will help to eliminate the ambiguity. We show that, in practice, the proposed approach produces near optimal results besides outperforming the other methodologies in existence. Our future work in this regard will be analyzing the features of joined letters and incorporating better segmentation accuracy. Results indicate that the approach can be used for character recognition in other Indic scripts as well.

References

[1] Dr.Nasira G.M, Banumathi.P “Off-Line Handwritten Character Recognition with Hidden Markov Models” CiiT International Journal of Artificial Intelligent Systems and Machine Learning Vol.3, No 1, January 2011.

[2] Jagadeesh Kannan R, Prabhakar R “Off-Line Cursive Handwritten Tamil Character Recognition” Wseas Transactions on Signal Processing, Vol. 4, June 2008.

[3] Plamondon.R., and S.Srihari “Online and offline Handwriting Recognition : A Comprehensive Survey.” IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 22 No.1, 2000, pp 63-84..

[4] Bunke.H,Roth.M, and E.G. Schukat Talamazzini “Offline Cursive Handwriting Recognition using Hidden Markov Models”. Pattern Recognition, Vol.28 No. 9,1995, pp 1399-1413.

[5] Roongroj Nopsuwanchai , and Dan Povey , "Discriminative Training for HMM-Based Offline Handwritten Character Recognition". *IEEE in the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)*.

[6] Shashank Mathur, Vaibhav Aggarwal, Himanshu Joshi, Anil Ahlawat, "Off-Line Handwriting Recognition using Genetic Algorithm", International Book Series Information Science and Computing, June – July 2008.

[7] Wang, Patrick Shen-Pei, "Learning, Representation, Understanding and Recognition of Words – An Intelligent Approach", Fundamentals in Handwriting Recognition. Ed.Sebastiano Impedovo, New Yark, Springer Verlag, 1994.

[8] Skapura, David M, "Building Neural Networks", ACM Press, New York, pp. 29-33.

[9] Anil K. Jain, Jianchang Mao, K.M.Mohiuddin, Artificial Neural Networks, A Tutorial , Computer, Vol,29, n-3, p.31-44, March 1996.

BIOGRAPHY

P.Banumathi received BE, MCA, M.Phil and MBA in the year 1994, 2004, 2007 and 2008. She is having 12 Years of teaching experience and 5 years of Industrial experience. Her area of interest is Neural Networks. She has presented 15 technical papers in various Seminars / National Conferences. She has presented 2 technical papers in International Conference. She has published 2 articles in International Journal. She is a member of Indian Society for Technical Education (ISTE) and Computer Society of India (CSI).

Dr. G. M. Nasira received M.C.A. and M.Phil degree in the year 1995 and 2002 respectively and the Doctorate degree from Mother Teresa Women's University, Kodaikanal in the year 2008. She is having around 14 years of teaching experience in College. Her area of interest includes Artificial Neural Networks, Fuzzy Logic, Genetic Algorithm, Simulation and modeling. She has presented 38 technical papers in various Seminars / Conferences. She is a member of Indian Society for Technical Education (ISTE).