

Recognition of Words in Tamil Script Using Neural Network

M. Karthigaiselvi¹, T. Kathirvalavakumar²

Research Centre in Computer Science, V. H. N. Senthikumara Nadar College, Virudhunagar-626 001, Tamil Nadu, India

ABSTRACT

In this paper, word recognition using neural network is proposed. Recognition process is started with the partitioning of document image into lines, words, and characters and then capturing the local features of segmented characters. After classifying the characters, the word image is transferred into unique code based on character code. This code ideally describes any form of word including word with mixed styles and different sizes. Sequence of character codes of the word form input pattern and word code is a target value of the pattern. Neural network is used to train the patterns of the words. Trained network is tested with word patterns and is recognized or unrecognized based on the network error value. Experiments have been conducted with a local database to evaluate the performance of the word recognizing system and obtained good accuracy. This method can be applied for any language word recognition system as the training is based on only unique code of the characters and words belonging to the language.

Keywords: Segmentation, Feature extraction, Classification, Word recognition, Neural network, Back-propagation

I. INTRODUCTION

Artificial neural networks [12] have been extensively applied for document analysis and recognition. Most efforts have been devoted for the recognition of isolated handwritten and printed characters with widely recognized successful results.

Ho et al. [6] have proposed a method for word recognition in degraded images of machine-printed postal addresses on envelopes based on word shape analysis. Allen et al. [1] have demonstrated a Holistic off-line handwritten word case recognition using a multi layer perceptron consisting of all lowercase or all uppercase characters. Zhu and Hull [20] have presented an algorithm for word recognition in oriental language documents. This technique compared the feature vectors extracted from sequences of characters directly to the feature vectors for words. Lavrenko et al. [9] have presented a holistic word recognition approach for single-author historical documents. The recognition output can then be used to align lexicon terms and their respective location in the page image.

Yaeger et al. [18] have combined an artificial neural network (ANN) character classifier with context-driven search over character segmentation, word segmentation, and word recognition hypotheses to provide robust recognition of hand-printed English text in new models of Apple Computer's Newton Message Pad. Cho et al. [4] have presented a new method for modeling and recognizing cursive words with hidden Markov models (HMM). In the method,

sequences of thin fixed-width vertical frames are extracted from the image, capturing the local features of the handwriting. By quantizing the feature vectors of each frame, the input word image is represented as a Markov chain of discrete symbols.

Seni and Nasrabadi [16] have presented a system for writer independent large vocabulary recognition of on-line handwritten cursive words. The network recognizer avoids explicit segmentation of the input words by using a sliding window concept. Bharath and Madhvanath [2] have proposed a data-driven HMM-based online handwritten word recognition system for Tamil. Steinherz et al. [17] have reviewed the field of online cursive word recognition. They classify the field into three categories: segmentation-free methods, which compare a sequence of observations derived from a word image with similar references of words in the lexicon; segmentation-based methods, that look for the best match between consecutive sequences of primitive segments and letters of a possible word; and the perception-oriented approach, that relates to methods that perform a human-like reading technique, in which anchor features found all over the word are used to boot-strap a few candidates for a final evaluation phase.

Lecolinet et al. [10] have presented methods and strategies for cursive word recognition. Lu et al. [11] have proposed a new word shape coding scheme, which captures the document content through annotating each word

image by a word shape code including character ascenders/descenders, character holes, and character water reservoirs. Moghaddam et al. [13] have presented a system for preprocessing and word spotting of very old historical document images and is language independent. Document images are processed for extraction of salient information using a word spotting technique which does not need line and word segmentation. Frinken et al. [5] have presented a novel word spotting algorithm using BLSTM (Bidirectional Long Short-Term Memory) neural networks. Zhang and Tan [19] have presented a word image coding technique to extract features from each word image object and represent them using feature code strings for comparison. A novel word image annotation technique is presented which captures the document content by converting each word image into a word shape code. In particular, we convert word images by using a set of topological character shape features including character ascenders/ descenders, character holes, and character water reservoirs. Huang et al. [7] have proposed a word shape recognition method for

retrieving image-based documents. The method detects local extrema points in word segments to form so-called vertical bar patterns. These vertical bar patterns form the feature vector of a document.

Recognition of words in Tamil script using neural network is proposed in this paper. The recognition process is having more complexity because of segmentation problems. Due to these difficulties many of the previous approaches have failed to recognize the words correctly. But in this work, before recognize the words; problems in touching line segmentation and touching character segmentation are solved. So the result of recognition is promising. The rest of the paper is organized as follows: Section 2 describes the characteristics of Tamil script; Section 3 elaborates the preprocessing techniques that are performed to enhance the document image; Section 4 describes word segmentation procedure; Section 5 details the feature extraction and character classification; Section 6 describes word recognition; Section 7 describes the neural network training; Section 8 discusses the experimental results and Section 9 describes the conclusion.

II. CHARACTERISTICS OF TAMIL SCRIPT

Tamil is a widely spoken South Indian language with 247 characters (Fig. 1). It contains 12 vowels, 18 consonants, 1 special character and 216 compound characters. There are also symbols called modifiers (Table 1) those occupy specific positions around the base characters. When the

modifiers that get added on the left or right side of the base character it remains disjoint from the base character, but when those are added either at the top or bottom of the base characters those get connected and spread to the upper and lower zones respectively.

Vowels	Special Character
அ ஆ இ ஈ உ ஊ எ ஏ ஐ ஒ ஓ ஔ	ஃ
Consonants	
க ங ச ஞ ட ண த ந ப ம ய ர ல வ ழ ள ற ன்	
Vowel-Consonants	
க ங ச ஞ ட ண த ந ப ம ய ர ல வ ழ ள ற ன்	

Fig. 1: Vowels and consonants of Tamil script

a. Text line structure

A text line of Tamil script can be partitioned horizontally into three zones namely upper, middle and lower. Assumed four imaginary lines namely upper line, mean line, base line and lower line as shown in Fig. 2 are existing. The mean line is a horizontal line passes through maximum number of upper most points of the characters of a line and base line is the horizontal line passes through maximum number of lowermost points of the characters of a line. In the

text line, upper line joins the top of ascenders and lower line joins the bottom of descenders. The upper zone denotes the portion above the mean line, the middle zone covers the region below the mean line and above the base line, and the lower zone is the portion where modifiers can reside [3]. The upper zone is separated from the middle zone of a text line by mean line, and the middle zone is separated from the lower zone by base line.

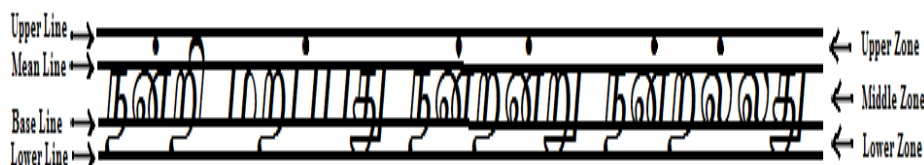


Fig. 2: Text line structure

III. PREPROCESSING

The input RGB image is converted into gray scale image. Smoothing is performed on the grayscale image to reduce the amount of high frequency noise using Wiener filter. The well-known adaptive thresholding Niblack's approach [14] is used for binarizing the image.

Gray scale pixel values are used in detecting skew angle. The upper left corner point is compared with upper right corner point to find out whether the document is left skewed or right skewed. A reference line is drawn from upper left corner point to lower left corner point. The skew of the document angle (θ°) is found by finding the orientation of the reference line. The document is rotated (θ°) in the anti-clockwise direction to correct the skew. Here point refers row, column of particular position. The proposed system can handle documents with skew angle between $+45^\circ$ and -45° . Slant removal technique proposed by Parvez and Mahmoud [15] is used to remove the slant on the segmented line when the characters are slanted to the right or left depending on the font style.

IV. WORD SEGMENTATION

To segment the text lines from the document image, the horizontal projection profile of the image is computed. Row with zero projection is used to segment the text lines. Sometimes lower zone characters of a line touches the upper zone characters of next line thus producing horizontally overlapping lines. The horizontally overlapping lines make the line segmentation more difficult. It becomes difficult to estimate the exact position of a row which segments a line from the next line. To segment this kind of overlapped lines Projection Based Lines Segmentation (PBLs) [8] is used. There is a possibility to have two or more overlapped lines when the strip has projection value greater than the one-third of average projection values. Observations reveals that rows with modifiers (Listed in Table 1) are with minimum projection values and hence rows with modifiers are not considered for segmentation but a row with minimum horizontal projection is identified from

the remaining rows and then the overlapped lines are separated into individual lines.

Vertical projection profile is used for word segmentation. In the first step the distance between adjacent characters in the text line image are computed. In the second step the computed distances are classified as either inter-word distances or inter-character distances using threshold value. The distances between words are always larger than distances between characters. Words can be segmented by comparing the distances with a suitable threshold. The threshold is defined as

$$\text{Threshold} = \frac{\text{Sum of the distances of adjacent characters}}{\text{Number of distances}}$$

When the distance value is greater than the threshold it is treated as a word gap otherwise it is a character gap. Words are segmented using the word gap.

V. FEATURE EXTRACTION AND CLASSIFICATION

Character segmentation is done after the individual words are segmented. Vertical white spaces serve to separate successive characters. Vertical projection method is used to split the words into sub images of individual characters. Sometimes body of one character touches the body of another character thus producing touching characters. To identify the touching characters char_threshold (CT) has been defined as 175% minimum width of the separated character. If the width of the separated character is greater than CT then the character have two or more characters. Touching characters are treated as a single character which leads to failure in character recognition. A column with minimum vertical projection is used to segment the touching characters. It has been identified from the experimental observations that minimum vertical projection values are appeared on the left most columns, right most columns and also appeared on the touching places. It makes difficult to estimate the exact position to segment the touching characters. But the touching characters are separated into individual characters by identifying a column with minimum vertical projection after ignoring the first and last column.

After separating the characters each character image is trimmed; vertical and horizontal projections are applied on the trimmed characters to identify feature vectors. Structural run based feature vector technique is used to extract structural features from printed Tamil characters. Identifying or recognizing Tamil characters need to identify modifier features existing on lower and upper part of the characters. Based on the structural properties of upper and lower modifiers, characters are divided into various categories such as upper, middle, lower and three zone characters respectively. The upper and lower zone may contain some modifiers. It has been identified that four modifiers are in the upper zone and twelve

modifiers are in the lower zone as shown in Table 1. The features namely number of loops, number of objects, number of runs at first row, number of runs at last row, number of vertical lines and number of horizontal lines, tetra bit features, height of the run at last column and width of the last row of the trimmed image are extracted to identify and distinguish every middle zone portion of the character. Feed forward neural network is chosen for feature classification. The extracted features of upper, middle and lower modifiers are used to train the network using standard back propagation algorithm.

Table 1: Categories of Upper & Lower zone modifiers

Zones	Modifier symbols	Sample character images with the modifier
Upper	• ̣ ̤ ̥	க், வீ, கி, ஓ
Lower	̦ ̧ ̨ ̩ ̪ ̫ ̬ ̭ ̮ ̯ ̰ ̱	ஞ, ர, த, இ, டி, ழ, ஶ, ஷ, ஹ, ள, ழ, ற, ள, ழ

VI. WORD RECOGNITION

Recognizing each word of printed Tamil document requires a collection of all possible valid words in a database. Unique number is assigned as a word code for each word in a database to recognize them. Each digit of a number is represented by its binary coded decimal (BCD) equivalent. Feedforward neural network system is designed to recognize individual word. It needs to receive individual characters of each word for processing. 124 different characters are identified in the Tamil script. Unique numbers is assigned to all possible characters of a Tamil script from 1 to

124 as a character code and are shown in Table 2 and its corresponding 7 bit binary value is used in the neural network. Feedforward neural network is trained to recognize the words in a database with a desired accuracy. In the trained network a word is said to be recognized when the network accuracy is lesser than or equal to the training accuracy otherwise it is termed as unrecognized word. This recognition system is font free and size free as the assigned code for character and word is font and size free.

Table 2: Character code

அ	1	வ்	26	சி	51	ம்	76	று	101
ஆ	2	ழ்	27	ளி	52	ய்	77	னு	102
இ	3	ள்	28	டி	53	ர்	78	கூ	103
ஈ	4	ற்	29	ணி	54	ல்	79	நூ	104
உ	5	ன்	30	தி	55	வ்	80	சூ	105
ஊ	6	க	31	நி	56	ழ்	81	னூ	106
எ	7	ங	32	பி	57	ள்	82	ஓ	107
ஏ	8	ச	33	மி	58	ற்	83	னூ	108
ஐ	9	ஞ	34	யி	59	ன்	84	தூ	109
ஔ	10	ட	35	ரி	60	கு	85	நூ	110
ஔ	11	ண	36	லி	61	ங	86	பூ	111
ஃ	12	த	37	வி	62	சு	87	மூ	112
க்	13	ந	38	ழி	63	னு	88	யூ	113
ங்	14	ப	39	ளி	64	டு	89	நூ	114
ச்	15	ம	40	றி	65	னூ	90	லூ	115
ஞ்	16	ய	41	னி	66	து	91	வூ	116

ட	17	ர	42	ஃ	67	நு	92	ழ	117
ண்	18	ல	43	ங்	68	பு	93	ளு	118
த்	19	வ	44	ஃ	69	மு	94	றா	119
ந்	20	ழ	45	ஃ	70	யு	95	னா	120
ப்	21	ள	46	ஃ	71	ரு	96	டு	121
ம்	22	ற	47	ண்	72	லு	97	ஃ	122
ய்	23	ன	48	ஃ	73	வு	98	ஃ	123
ர்	24	கி	49	ந்	74	மு	99	ஈ	124
ல்	25	நி	50	ப்	75	ளு	100		

VII. NEURAL NETWORK TRAINING

Recognizing all words in a database is the task of the neural network. Every word in the database is used for training. Trained and untrained words are identified by the trained network from the network error. Single hidden layer feed forward neural network is considered for this system. Number of neurons in the input layer is the product of length of the longest word in the data base and character code length. Number of neurons in the output layer is the product of number of digits in the size of the word database and length of a BCD form of a digit.

All input patterns to be represented in uniform length. For smaller length words, bit '0's are appended as a suffix in the pattern. For example if the word to be processed is 'அரிது' and maximum length word in a database is 10 then input pattern of a corresponding word 'அரிது' is 000001 0111100 1011011 0000000 0000000 0000000 0000000 0000000 0000000 0000000. Every word code is represented in uniform length. Required numbers of zeroes are prefixed with the code to keep the word length uniform. For example if a code of a word is 2 and size of the word database is 500 then the code 2 is represented as 002 and its BCD is 0000 0000 0010 which is the word code and is a target value of the word during network processing. If the code of a word is 459 then it is represented as 0100 0101 1001.

Standard backpropagation algorithm is used for network training. Mean square error is used as a measure for termination during training. A word is treated as recognized by the trained network if the network error is lesser than or equal to the termination condition error during training otherwise the word is treated as unrecognized.

VIII. EXPERIMENTAL RESULTS

Experiments are carried out with images of printed pages obtained from different Tamil literary periodicals collected from [21-26]. Documents are with single column text regions and with pdf format. These documents are initially converted into jpeg image format. 1000 different documents have been taken with different level of skew, slant as well as size. The proposed method is implemented in Matlab13.

Segmentation process is applied on the preprocessed document. In line segmentation, touching lines are identified and are segmented correctly. For example, the document specified in Fig. 3(a) has 5 lines but only 4 line breaks are recognized by projection profile technique as shown in Fig. 3(b). It concludes that the document is with overlapping lines. The method PBLs processed the document shown in Fig. 3(a) and segment it into 5 strips correctly as in Fig. 3(c).

சுழிகள் - சுழல்களுடனே விரைந்து

ஆற்றங்கரையில் ஆலமரங்கள் நெடுந்தூரத்துக்கு
நெடுந்தூரம் விழுதுகள் விட்டு விசாலமாய்ப்
படர்ந்திருந்தன. மரங்களின் பழைய இலைகள்
எல்லாம் உதிர்த்து புதிதாய்த் தளிர்விட்டிருந்த காலம்.

(a)

1 சுழிகள் - சுழல்களுடனே விரைந்து

2 ஆற்றங்கரையில் ஆலமரங்கள் நெடுந்தூரத்துக்கு
3 நெடுந்தூரம் விழுதுகள் விட்டு விசாலமாய்ப்
4 படர்ந்திருந்தன. மரங்களின் பழைய இலைகள்
5 எல்லாம் உதிர்த்து புதிதாய்த் தளிர்விட்டிருந்த காலம்.

(b)

1 சுழிகள் - சுழல்களுடனே விரைந்து
2 ஆற்றங்கரையில் ஆலமரங்கள் நெடுந்தூரத்துக்கு
3 நெடுந்தூரம் விழுதுகள் விட்டு விசாலமாய்ப்
4 படர்ந்திருந்தன. மரங்களின் பழைய இலைகள்
5 எல்லாம் உதிர்த்து புதிதாய்த் தளிர்விட்டிருந்த காலம்.

(c)

Fig 3: (a) Original Image of document15 (b) Strips obtained using horizontal projection (c) Identified Lines using PBLs method

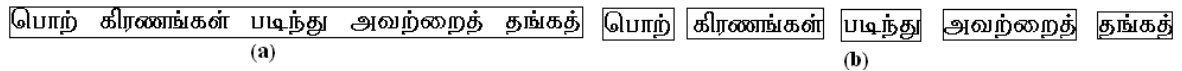


Fig. 4: (a) Original image (b) Segmented words

The space between the words is greater than the space between the characters. The line specified in Fig.4 (a) is segmented into 5 words successfully as shown in Fig.4 (b).

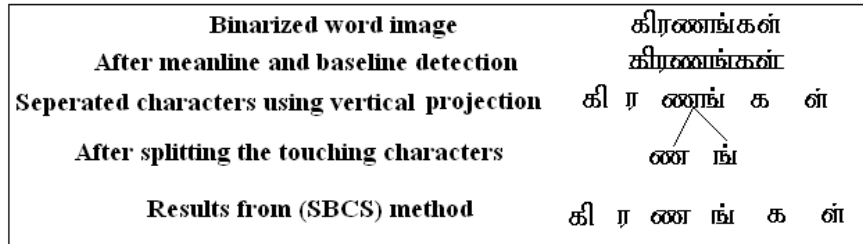


Fig. 5: Splitting of touching characters

Vertical projection segments the word கிரணங்கள் into 5 partitions கி(17) ர(10)ணங் (45), க்(13) and ள்(18) which is shown in Fig. 5 (value within the parentheses specifies the width of the character). CT of the word is computed as 21. The third partition (ணங்) has greater width value than CT. It implies that it is with touched characters and requires segmentation. As a result of splitting procedure the touched characters are split into individual characters ண and ங்.

After character segmentation, features are extracted from all characters in a word, which is then used in the neural network for classification. Features extracted from upper, middle, lower and three zone characters are illustrated in Table 3 with sample characters.

In order to test the performance of the proposed word recognition procedure, set of 500 word images with different fonts and sizes are collected from documents. The maximum word length of the dataset is 10. After character classification, each word image is encoded by concatenating binary value of consecutive character code of a word. The encoded words are input patterns of the network. Assign values 1 to 500 as word code for those words. Each word code is converted into BCD code of uniform length. The word “வண்ணங்கள்” during training is transformed into input pattern as follows. The first character ‘வ’ in the word is coded as 44, the second character ‘ண்’ is represented as 18, the third (ண) fourth (ங்), fifth (க்) and sixth (ள்) characters are coded as 36, 14, 31 and 28 respectively. The maximum length of the word in the dataset is 10, but the word has only 6 characters. To keep uniform word length remaining 4 character codes are set as zeroes. Each character code is converted into 7 bit binary values. Totally we get 70 bits to represent the word pattern and is shown in Fig.6. The target value of the word

is 65. Size of the database is 500 (three digit), so every word code needs 3 digit length. The word code 65 is represented as 065 and then it is converted into its BCD equivalent 0000 0110 0101.

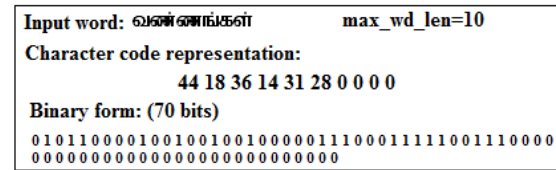


Fig. 6: Code representation for sample word

Single hidden layer feedforward neural network is trained with standard backpropagation algorithm for word recognition. Number of neurons in the input layer is 70. Number of neurons in Hidden Layer is set as 151 by trial and error. Number of output neuron is 12. The experiment is executed with the learning parameters $\lambda = 0.05$. This value is fixed by trial and error. Termination condition is fixed as 0.01 mean squared error (MSE). Among 500 words in the database 300 words are randomly chosen for training. Remaining 200 words and 100 words used in training are used for testing. Learning curve of the network training is shown in Fig 7. Testing of the trained network shows that the MSE of the non-trained patterns are greater than 0.01 and MSE of the trained patterns are lesser than 0.01 and are shown in Fig 8. Termination condition (0.01 MSE) used in the training phase is marked as error accuracy in Fig 8. The experiment is repeated with different number of randomly selected training and testing patterns with required learning parameter and the results obtained are shown in Table 4. In each experiment all the patterns used in testing phase are recognized as trained or untrained pattern based on the network error.

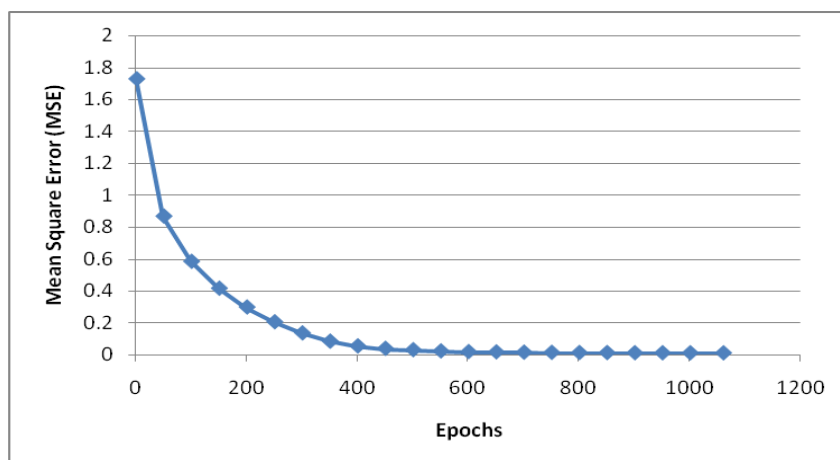


Fig.7: Learning Curve for word recognition

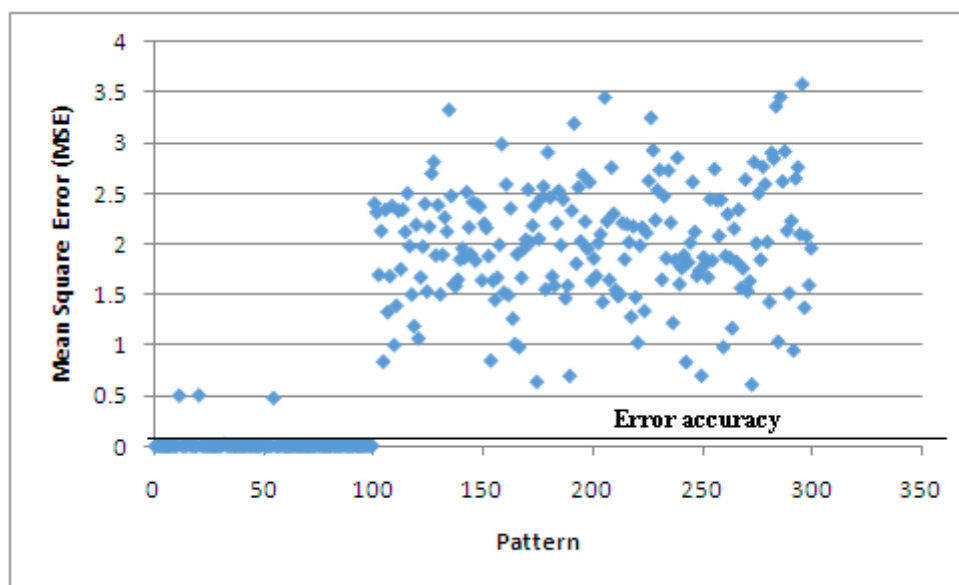


Fig. 8: network error of trained and untrained patterns during testing phase

Table 4. Word recognition results

S. No	# Training patterns	Network structure	Learning parameter	Termination condition (MSE)	# Epochs	Time (s)	Testing patterns		Test Accuracy (%)
							# Trained patterns	# Test patterns	
1	200	70-121-12	0.05	0.01	844	1662.412140	100	300	100%
2	300	70-151-12	0.05	0.01	1060	4009.372474	100	200	100%
3	400	70-171-12	0.07	0.01	1428	4612.960061	100	100	100%

IX. CONCLUSION

In this paper a single hidden layer neural network is used for recognizing printed Tamil script words. All Tamil characters are given unique code and similarly all words in the database are given unique code and are used in training. After

characters of the words are recognized, words are recognized as trained word or untrained word using network. The proposed word recognition process is simple and gives correct recognition. It is style and size (word length) independent. Experiment results show that any word is recognized by the trained

neural network as trained or untrained based on the network training accuracy. This leads to the result that if all valid semantically correct words in the dictionary of any language are used in the training phase of the network then the trained neural network identifies the semantically wrong words and correct words.

ACKNOWLEDGEMENT

The authors thank the university grants commission, Government of India for partially supporting this project (MRP: F.No. 42-144/2013(SR)).

REFERENCES

- [1]. Allen TJ, Sherkat N, Whitrow RJ (1999) Holistic Word Case Recognition using a Multi-Layer Perceptron Neural Network. IEE Colloquium on Document Image Processing and Multimedia.
- [2]. Bharath A, Madhvanath S (2007) Hidden Markov Models for Online Handwritten Tamil Word Recognition. IEEE ICDAR'2007, pp 23-26
- [3]. Chaudhuri BB, Pal U, Mitra M (2002) Automatic Recognition of Printed Oriya Script. *Sadhana* 27: 23-34.
- [4]. Cho W, Lee SW, Kim JH (1995) Modeling and Recognition of Cursive Words with Hidden Markov Models. *Pattern Recognition* 28(12): 1941-1953
- [5]. Frinken V, Fischer A, Bunke H (2010) A novel word spotting algorithm using bidirectional long short-term memory neural networks. In: Schwenker F, El Gayar N (eds) *Artificial neural networks in pattern recognition*. Springer, Berlin/Heidelberg, pp 185-196
- [6]. Ho TK, Hull JJ, Srihari SN (1992) A word shape analysis approach to lexicon based word recognition. *Pattern Recognition Letter* 13: 821-826
- [7]. Huang W, Tan CL, Sung SY, Xu Y (2001) Word Shape Recognition for Image-Based Document Retrieval. *IEEE International conference on Image processing*, pp1114-1117
- [8]. Kathirvalavakumar T, Karthigai Selvi M (2013) Efficient Touching Text Line Segmentation in Tamil Script Using Horizontal Projection. *International conference on Mining Intelligence and Knowledge Exploration, LNCS 8284*, pp 279-288
- [9]. Lavrenko V, Rath TM, Manmatha R (2004) Holistic Word Recognition for Handwritten Historical Documents. *IEEE International Workshop on Document Image Analysis for Libraries*, pp 278 - 287
- [10]. Lecolinet E, Baret O (1994) *Cursive Word Recognition: Methods and Strategies, Fundamentals in Handwriting Recognition*. Springer NATO ASI Series 124: 235-263
- [11]. Lu S, Li L, Tan CL (2008) Document image retrieval through word shape coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30: 1913-1918.
- [12]. Marinai S, Gori M, Soda G (2005) Artificial neural networks for document analysis and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27: 23-35
- [13]. Moghaddam RF, Cheriet M (2009) Application of multi-level classifiers and clustering for automatic word spotting in historical document images. In: *IEEE 10th international conference on document analysis and recognition*, Barcelona, pp 511-515.
- [14]. Niblack W (1986) *An Introduction to Digital Image Processing*. Prentice-Hall, Englewood Cliffs pp 115-116
- [15]. Parvez Mohammad Tanvir, Mahmoud Sabri A (2013) Arabic handwriting recognition using structural and syntactic pattern attributes. *Pattern Recognition* 46: 141-154
- [16]. Seni G, Nasrabadi NM (1994) An on-line cursive word recognition system. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp 404 - 410
- [17]. Steinerherz Tal, Rivlin Ehud, Intrator Nathan (1999) Online cursive script word recognition - a survey. *International Journal on Document Analysis and Recognition* 2: 90-110
- [18]. Yaeger L, Lyon R, Webb B (1997) Effective Training of a Neural Network Character Classifier for Word Recognition. In: *Advances in Neural Information Processing Systems*, pp 807-813.
- [19]. Zhang L, Tan CL (2005) A word image coding technique and its applications in information retrieval from imaged documents. In: *Proceedings of the international workshop on document analysis*, pp 69-92.
- [20]. Zhu J, Hull JJ (1994) Image-based Word Recognition in Oriental Language Document Images. *IEEE*, pp 300-304.
- [21]. <http://www.tamilagaasiriyar.com/p/tamil-e-books.html>
- [22]. <http://books.tamilcube.com/tamil/>
- [23]. <http://knowinyourself1.blogspot.in/2011/04/free-tamil-books-tamil-pdf-books.html>
- [24]. <http://www.projectmadurai.org/pmworks.html>
- [25]. <http://www.dinamalar.com>
- [26]. <http://kalvimalar.dinamalar.com/tamil/>