International Journal of Soft Computing 3 (2):139-143, 2008
ISSN: 1816-9503

Enhancing the Performance of Handwritten Tamil Character Recognition System by Slant Removal and Introducing Special Features

[1]N. Shanthi and [2]K. Duraiswamy
[1]Department of Information Technology, [2]Department of CSE,
K.S. Rangasamy College of Technology, Tiruchengode, India

Abstract: This study describes a system for recognizing offline handwritten Tamil character recognition system by removing slant and introducing special features like horizontal lines, vertical lines, slanting lines and holes. Data samples are collected from different writers on A4 sized documents. They are scanned using a flat bed scanner at a resolution of 300dpi and stored as grey scale images. Various preprocessing operations like thresholding, segmentation, skeletonization and slant removal are performed on the digitized image to enhance the quality of the image. The preprocessed image is normalized to an image of standard size 64×64. Pixel densities are calculated for different zones of the image and these values are used as the features of a character. Special features are also considered for 5 characters to improve their recognition rate. These features are used to train and test the support vector machine. The support vector machine is tested for the first time for recognizing handwritten Tamil characters. The recognition results are tested for 64×64 sized image with overlapping zones without considering slant, with considering slant and by introducing additional features for 5 characters. Best results are obtained by removing slant and by considering additional features. The handwriting system is trained for 106 different characters and test results are given for 34 different Tamil characters. The system has achieved a very good recognition rate of 91.25% on the totally unconstrained handwritten Tamil character database.

Key words: Tamil character recognition, support vector machine, preprocessing, feature extraction

INTRODUCTION

Handwriting has continued to persist as a means of communication and recording information in day-to-day life even with the introduction of new technologies. Recognition of characters is an important area in machine learning. Widespread acceptance of digital computers seemingly challenges the future of handwriting. However, in numerous situations, a pen together with paper or a small notepad is much more convenient than a keyboard. Optical Character Recognition (OCR) is a process of automatic recognition of characters by computers in optically scanned and digitized pages of text (Pal and Chaudhuri, 2004). OCR is one of the most fascinating and challenging areas of pattern recognition with various practical applications. It can contribute immensely to the advancement of an automation process and can improve the interface between man and machine in many applications (Mantas, 1986; Govindan and Shivaprasad, 1990). Some applications of OCR are reading aid for the blind, automatic text entry into the computer for desktop publication, library cataloging and ledgering. Automatic reading for sorting of postal mail, bank cheques and other documents and language processing.

Recognition of any handwritten characters with respect to any language is difficult, since, the handwritten characters differ not only from person to person but also according to the state of mood of the same person. Among different branches of character recognition it is easier to recognize English alphabets and numerals than Tamil characters (Suresh *et al*., 1999).

Siromoney *et al.* (1978) described a method for recognition of machine printed Tamil characters using an encoded character string dictionary. The scheme employs string features extracted by row- and column-wise scanning of character matrix. The features in each row (column) are encoded suitably depending upon the complexity of the script to be recognized.

Chinnuswamy and Krishnamoorthy (1980) proposed an approach for hand-printed Tamil character recognition. Here, the characters are assumed to be composed of line-like elements, called primitives, satisfying certain

Corresponding Author: N. Shanthi, Department of Information Technology, K.S.Rangasamy College of Technology, Tiruchengode-637 215, Tamil Nadu, India

relational constraints. Labeled graphs are used to describe the structural composition of characters in terms of the primitives and the relational constraints satisfied by them. The recognition procedure consists of converting the input image into a labeled graph representing the input character and computing correlation coefficients with the labeled graphs stored for a set of basic symbols.

Suresh *et al*. (1999) proposed an approach to use the fuzzy concept on handwritten Tamil characters to classify them as one among the prototype characters using a feature called distance from the frame and a suitable membership function. The unknown and prototype characters are preprocessed and considered for recognition.

A system is described to recognize handwritten Tamil characters using a two stage classification approach, for a subset of the Tamil alphabet by Hiwavitharana and Fernamd (2002). In the first stage, an unknown character is pre-classified into one of the three groups: core, ascending and descending characters. Then, in the second stage, members of the pre-classified group are further analyzed using a statistical classifier for final recognition.

This study presents a recognition system for offline unconstrained handwritten Tamil characters based on support vector machine. Recently Support Vector Machine (SVM) has received attention for character recognition. SVM is a new type of pattern classifier based on a novel statistical learning technique. Due to the difficulty in great variation among handwritten characters, the system is trained with 106 characters and tested for 34 selected Tamil characters. The characters are chosen such that the sample data set represents almost all the characters. The input size of an image is random in nature. They are converted into standard size. The performance is compared without considering slant and considering special features for selected characters. Generally, a character recognizer involves three tasks: preprocessing, feature extraction and classification.

TAMIL LANGUAGE

Tamil, which is a south Indian language, is one of the oldest languages in the world. It has been influenced by Sanskrit to a certain degree (Hiwavitharana and Fernamd, 2002). But Tamil is unrelated to the descendents of Sanskrit such as Hindi, Bengali and Gujarati. Most Tamil letters have circular shapes partially due to the fact that they were originally carved with needles on palm leaves, a technology that favored round shapes. Tamil script is used to write the Tamil language in Tamil Nadu, SriLanka, Singapore and parts of Malaysia, as well as to write minority languages such as Badaga. Tamil alphabet consists of 12 vowels, 18 consonants and one special character (AK). The vowels and consonants combine to form 216 compound characters, giving a total of 247 characters. In addition to the standard characters, 6 characters taken from the Grantha script, which was used in the Tamil region to write Sanskrit, are sometimes used to represent sounds not native to Tamil, that is, words borrowed from Sanskrit, Prakrit and other languages. The complete Tamil alphabet and composite character formations are given in Chinnuswamy and Krishnamoorthy (1980). The advantage of having a separate symbol in forming the composite character is that the number of symbols to be recognized can be reduced. The number of symbols to be recognized is reduced to 106.

PREPROCESSING

The raw input of the digitizer typically contains noise due to erratic hand movements and inaccuracies in digitization of the actual input. Original documents are often dirty due to smearing and smudging of text and aging (Shanty and Duraiswamy, 2005). In some cases, the documents are of very poor quality due to seeping of ink from the other side of the page and general degradation of the paper and ink. Preprocessing is concerned mainly with the reduction of these kinds of noise and variability in the input. The number and type of preprocessing algorithms employs on the scanned image depend on many factors such as paper quality, resolution of the scanned image, the amount of skew in the image and the layout of the text.

Preprocessing operations performed prior to recognition are:

- Thresholding, the task of converting a gray-scale image into a binary black-white image. Here Otsu's method of histogram-based global thresholding algorithm is used (Ostu, 1979).
- Skeletonization, reducing the patterns to thin line representation (Suen, 1992). Here Hilditch's algorithm is used for skeletonization.
- Line segmentation, the separation of individual lines of text. Horizontal histogram profile is used for segmenting the lines.
- Character segmentation, the isolation of individual characters. Vertical histogram profile is used for segmenting the characters.

- Slant removal, Removal of angle between the vertical direction and the direction of the strokes. The characters are rotated from -10° to +10° and they are used for training.
- Normalization, converting the random sized image into standard sized image. Bilinear interpolation technique is used to convert the random sized image into normalized image (Srihar *et al*., 2000). All the input images are converted to a standard size of 64×64.

FEATURE EXTRACTION

Feature extraction is the problem of extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class pattern variability (Trier *et al*., 1996). Each normalized image from the preprocessing stage is divided into equal number of horizontal and vertical strips, producing a grid with square shaped zones of size 8×8. Here the normalized image is divided into overlapping zones. Pixel 1-8 is taken as the first zone. Pixel 5-12 is taken as the second zone. For each zone, the pixel density is calculated and therefore a vector created. The pixel density varies from 0-64. For 64×64 sized image there will be 225 features for overlapping zones.

To improve the performance of the recognizer, 5 characters with low recognition rate are identified. The characters identified and their percentage of recognition is shown in Table 1.

Each of these characters classification result is individually analyzed. For example, when the classification result for ஞ is analyzed, the result shows that many of the character ஞ are recognized as ஞ. This is because of the similarity in handwriting between these 2 characters. A separate classifier is developed for recognizing ஞ and ஞ. So whenever any character is recognized as ஞ then it is given as input to this classifier. This classifier further classifies the character using the specific features like number of vertical lines.

Features specific to 5 particular characters shown in Table 1 are identified and calculated. The characters are chosen such that their recognition rates are low. Special features considered are number of vertical lines, number of horizontal lines, number of holes and number of slanting lines. This improves the recognition rate of the recognizer. Experimentation results show that 64×64 slant corrected image with specific features produced better results.

Table 1: Characters with low recognition rate

Characters with low %recognition rate	Similar characters	Recognition rate %
ஏ	எ,ர	78.45
ஜ	ஜ	72.80
ஞ	ஞ	77.70
ழ	டி	81.29
ண	ன,ள	74.85

RECOGNITION PROCESS

The next stage in the process of handwriting recognition is to recognize the features calculated from the normalized image extracted in the normalization stage. A variety of pattern recognition methods are available, and many have been used for handwriting recognition. Here, Support Vector Machine is used and the experimentation results show that the Support Vector Machine recognizes well for 64×64 sized slant corrected image with overlapping zones.

SUPPORT VECTOR MACHINES

Recently there has been an explosion on the topic of Support Vector Machines. SVMs are the most well known class of algorithms that use the idea of kernel substitution and referred as kernel methods. SVMs have achieved excellent recognition results in various pattern recognition applications. In off line handwritten character recognition they have been proved to be comparable or even superior to the standard techniques like Bayesian classifiers or multistage perceptrons.

Given a training set of instance-label pairs (x_i, y_i); $i = 1,\ldots, l$. where, $x_i \in R^n$ and $y \in \{1, -1\}^l$, the Support Vector Machines (SVM) require the solution of the following optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=0}^{l} \xi_i$$

$$\text{subject to } y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

Here, training vectors x_i are mapped into a higher (may be infinite) dimensional space by the function ϕ. Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $C > 0$ is the penalty parameter of the error term. Furthermore, $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is called the kernel function. In SVM a classification task usually involves training and testing data which consist of some data instances. Each instance in the training set contains one target value (class labels) and several attributes (features). The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given only the attributes. There are a number of kernels that can be used in support vector machine models. These include linear, polynomial, Radial Basis Function (RBF) and sigmoid. The RBF is the most popular choice of kernel type used in SVM and is used here.

SVM consists of a training module (svm_train) and a classification module (svm_predict). The proposed method works as follows:

- Collect data samples.
- Scan and store it as grey scale images.
- Preprocess the input image.
- Normalize the image and calculate the feature vectors in different angles from -10° to +10°.
- Store the feature vectors of the characters and predefined class in a file.
- The training module takes the input file and trains the network. The support vectors are stored in the target file.
- In the classification module, the features of the unknown character is calculated and given as input along with the support vectors.
- If the output is any of the selected character then again the image is given to another classifier which also considers specific features. This improves the recognition rate of the recognizer.
- The classification module classifies the character and labels of the characters are given and stored in a file.

RESULTS AND DISCUSSION

The system is trained with 35441 characters belonging to 106 different characters written by 117 different users. The testing data contained a separate set of 6048 characters belonging to 34 different characters. The characters chosen for testing is shown in Table 2. A portion of the training data is also used to test the system, to check how well the system responds to the data it has been trained on.

The system achieved 100% recognition rate for training data. The pixel densities are calculated for 64×64 normalized image for the unknown characters and the features are given to the SVM classification module. The characters are classified based on the highest match and the recognized characters are stored in a word file and the characters can be viewed using the available TAM (Tamil Monolingual) font. The recognition results for different sized normal image of test characters are shown in Table 2. The experimentation result shows that there are some misclassification results for test data and the recognition rate varies from different image size to image size.

Table 2: Recognition results for various handwritten Tamil characters

S.No.	Character	Recognition rate without considering slant	Recognition rate by removing slant features	Recognition rate by including specific
1	அ	94.51	96.15	96.15
2	ஆ	82.97	85.71	85.71
3	இ	79.24	82.51	82.51
4	ஈ	93.79	94.35	94.35
5	உ	96.65	96.65	96.65
6	ஊ	82.22	85.56	85.56
7	எ	86.71	88.44	88.44
8	ஏ	72.93	78.45	85.08
9		92.14	97.75	97.75
10	ஐ	83.72	87.21	87.21
11	ஒ	71.68	72.83	80.93
12	ஓ	98.90	99.45	99.45
13	ஃ	88.33	91.67	91.67
14	க	91.02	94.01	94.01
15	ங	87.57	93.79	93.79
16	ச	72.57	77.71	90.86
17	ஞ	98.36	98.91	98.91
18	ட	92.86	92.31	92.31
19	ண	90.06	92.82	92.82
20	த	79.52	82.53	82.53
21	ந	97.77	97.77	97.77
22	ப	95.05	96.70	96.70
23	ம	96.70	97.25	97.25
24	ய	82.42	87.36	87.36
25	ர	96.13	97.79	97.79
26	ல	88.20	89.89	89.89
27	வ	73.10	81.29	90.64
28	ழ	83.71	87.08	87.08
29	ள	93.79	94.92	94.92
30	ற	73.14	74.86	80.00
31	ன	82.58	87.08	87.08
32	ா	92.27	95.58	95.58
33	ெ	86.29	89.14	89.14
34	ே	90.34	93.75	93.75
Overall%	ை	87.35	90.01	91.25

Table 2 shows that the recognition rate for overlapping zone is between 71.7-98.9%. The recognition rate after removing slant is 72.83-99.45%. Similarly, the recognition rate increased for the 5 characters if special features are considered. The overall recognition rate increases from 87.35-91.25%. The experimentation result shows that the recognition rate increases when slant is removed and specific features are considered. This is because of the improvement in quality in the image when slope is reduced.

CONCLUSION

This study presents a system to recognize selected offline Tamil handwritten characters using SVM. The result shows that the algorithm works well for the selected set of 34 characters. The algorithm is tried for different features and also by removing slant. The overall recognition rate varies from 87.35-91.25%. The experimentation result shows that the SVM based approach for 64×64 sized slant removed image with overlapping zones and special features recognizes well with a recognition rate of 91.25%. The main recognition errors were due to abnormal writing and ambiguity among similar shaped characters. This recognition rate is achieved with a simple feature of pixel densities and 4 special features. The system can be extended to recognize complete set of Tamil alphabets. This requires splitting a composite character into basic recognizable symbols. Future work can also include extracting more robust features for the classifier to achieve better discrimination power.

REFERENCES

Chinnuswamy, P. and S.G. Krishnamoorthy, 1980. Recognition of hand printed Tamil characters. Pattern Recognition, 12: 141-152.

Govindan, V.K. and A.P. Shivaprasad, 1990. Character recognition: A Review. Pattern Recognition, 23: 671-683.

Hewavitharana, S. and H.C. Fernando, 2002. A two stage classification approach to Tamil handwriting recognition. Tamil Internet California, USA., pp: 118-124.

Mantas, J., 1986. An overview of character recognition methodologies. Pattern Recognition, 19: 425-430.

Otsu, N., 1979. A threshold selection method from grey level histogram. IEEE. Trans. Sys. Man and Cyber., 9: 62-66.

Pal, U. and B.B. Chaudhuri, 2004. Indian script character recognition: A Survey. Pattern Recognition, 37: 1887-1899.

Shanthi, N. and K. Duraiswamy, 2005. Preprocessing algorithms for the recognition of Tamil handwritten Characters. 3rd International CALIBER 2005, Kochi, pp: 77-82.

Siromoney *et al.*, 1978. Computer recognition of printed Tamil character. Pattern Recognition, 10: 243-247.

Srihari *et al.*, 2000. On-line and off-line handwriting recognition: A Comprehensive Survey. IEEE. PAMI, 22: 63-84.

Suen, L.L., 1992. Thinning Methodologies-A Comprehensive Survey. IEEE. PAMI., 14: 869-885.

Suresh *et al.*, 1999. Recognition of hand printed Tamil characters using classification approach. ICAPRDT., pp: 63-84.

Trier *et al.*, 1996. Feature extraction methods for character recognition: A Survey. Pattern Recognition, 29: 641-662.