Journal of Computer Science 3 (9): 760-764, 2007
ISSN 1549-3636
© 2007 Science Publications

Performance Comparison of Different Image Sizes for Recognizing Unconstrained Handwritten Tamil Characters using SVM

[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 characters using Support Vector Machine (SVM). Data samples are collected from different writers on A4 sized documents. They are scanned using a flat bed scanner at a resolution of 300 dpi and stored as grey scale images. Various preprocessing operations are performed on the digitized image to enhance the quality of the image. Random sized preprocessed image is normalized to uniform sized image. Pixel densities are calculated for different zones of the image and these values are used as the features of a character. 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 3 different standard sizes of 32X32, 48X48 and 64X64. Pixel densities are calculated for various zones and also for overlapping zones of the 64X64 sized image. Best results are obtained for 64X64 sized normalized image with overlapping windows. The handwriting system is trained for 106 different characters and test results are given for 34 different Tamil characters. With a simple feature of pixel density, the system has achieved a very good recognition rate of 87.4% 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[5]. 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 [1][2]. Some applications of OCR are (1) reading aid for the blind, (2) automatic text entry into the computer for desktop publication, library cataloging, ledgering, etc. (3) automatic reading for sorting of postal mail, bank cheques and other documents, (4) language processing etc.

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[8].

In[3] the authors 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.

In[4] authors 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 relational constraints. Labeled graphs are used to describe the structural composition of characters in terms of the primitives and

Corresponding Author: N.Shanthi, Department of Information Technology, K.S.Rangasamy College of Technology, Tiruchengode – 637 215, Tamil Nadu, India. Tel: +919842013355

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.

In[8] the author 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.

In[9] a system is described to recognize handwritten Tamil characters using a two stage classification approach, for a subset of the Tamil alphabet. 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 paper 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 different standard size and the performance is compared. Generally, a character recognizer involves three tasks: preprocessing, feature extraction and classification. In this paper the first section deals an introduction to Tamil language and in the second section the preprocessing methods are described. The third and fourth section deals with feature extraction & recognition process. Finally the main results are presented and discussed.

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[9]. 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). Vowels and consonants are combined to form composite letters, making a total of 247 different characters and some Sanskrit characters. The complete Tamil alphabet and composite character formations are given in[4]. The advantage of having a separate symbol for each vowel in composite character formations, there is a possibility to reduce the number of symbols used by the alphabet.

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 [10]. 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[11].
- **Skeletonization:** reducing the patterns to thin line representation[12]; 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.
- **Normalization:** converting the random sized image into standard sized image; Bilinear interpolation technique is used to convert the random sized image into normalized image[7].

The input image size is random in nature. Image quality of the normalized image will vary depending on the input image size. So the random sized image is converted to different normalized size image of 32X32, 48X48 and 64X64 and the performance of the recognizer is compared.

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[6]. Each normalized image is divided into equal number of horizontal and vertical strips, producing a grid with square shaped zones. 64X64 sized image is divided into both nonoverlapping and overlapping zones. For each zone, the pixel density is calculated and therefore a vector created. Consider SXS is the size of the normalized image. In this work the normalized image is divided into different zone size of S/16 X S/16, S/8XS/8 and S/4XS/4. For example 64X64 sized image is divided into different zones of 4X4, 8X8 and 16X16.

When the zone size was small, it captured more detailed pixel variations. However, due to the varying nature of handwriting, there was high dissimilarity between the feature vectors of the same class. Large sized zones failed to capture the essential parts of characters, which make them distinct from others. The best results were produced by (S/8) X (S/8) pixel zones. Therefore, it is decided to use (S/8) X (S/8) zones for feature extraction. For 64X64 sized image, zone size of 8X8 produced best results. This produces 64 feature vector for nonoverlapping zones and 225 feature vectors for overlapping zones. Feature vector contains value between 0 and S/8XS/8 corresponding to the pixel density of each zone. Experimentation results show that 64X64 image with overlapping zones produced better results.

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 64X64 sized 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 offline 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…l where $x_i \in R^n$ and $y \in \{1,-1\}^i$, the support vector machines (SVM) require the solution of the following optimization problem:

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

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.
- Store the feature vectors of the characters and predefined class in a file.

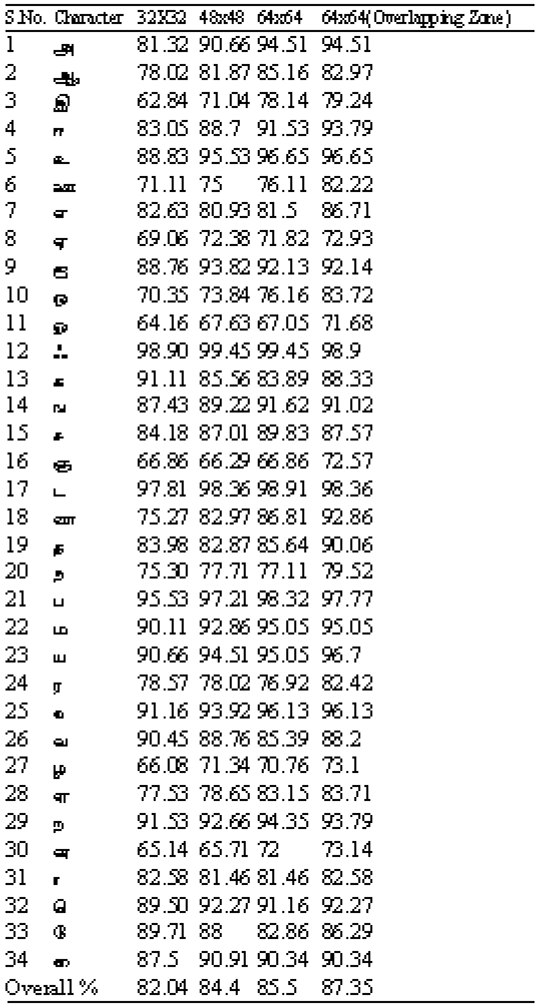
- The training module takes the input file and trains the network. The support vectors are stored in the target file.
- In the classification module, the features of the unknown character is calculated and given as input along with the support vectors.
- The classification module classifies the character and labels of the characters are given and stored in a file.

EXPERIMENTATION 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 Table1. 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 different sized 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 1. 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 1 show that the recognition rate for 32X32-image size is between 62.84 to 98.9%. Recognition rate for 48X48-image size is between 65.71 to 99.5%. For 64X64-image size the recognition rate is between 67 to 99.5% and for overlapping zone the recognition rate is between 71.7 to 98.9%. The experimentation result shows that the recognition rate increases when the normalized image size is large and the recognition rate reduces when the image size is small. This is because of the reduction of quality in the image when large sized image is reduced into very small size. Fig. 1 show that Support Vector Machine performs well for 64X64 sized normalized image with overlapping zones when compared to other sizes. For 27 characters, the overlapping zone of 64X64 sized image has higher recognition rate when compared to nonoverlapping zone. The overall recognition rate also increased from 85.5 to 87.35%.

Table 1: Recognition results for various handwritten Tamil characters



S.No.	Character	32X32	48x48	64x64	64x64(Overlapping Zone)
1	அ	81.32	90.66	94.51	94.51
2	ஆ	78.02	81.87	85.16	82.97
3	இ	62.84	71.04	78.14	79.24
4	ஈ	83.05	88.7	91.53	93.79
5	உ	88.83	95.53	96.65	96.65
6	ஊ	71.11	75	76.11	82.22
7	எ	82.63	80.93	81.5	86.71
8	ஏ	69.06	72.38	71.82	72.93
9	ஐ	88.76	93.82	92.13	92.14
10	ஒ	70.35	73.84	76.16	83.72
11	ஓ	64.16	67.63	67.05	71.68
12	ஃ	98.90	99.45	99.45	98.9
13	க	91.11	85.56	83.89	88.33
14	ங	87.43	89.22	91.62	91.02
15	ச	84.18	87.01	89.83	87.57
16	ஞ	66.86	66.29	66.86	72.57
17	ட	97.81	98.36	98.91	98.36
18	ண	75.27	82.97	86.81	92.86
19	த	83.98	82.87	85.64	90.06
20	ந	75.30	77.71	77.11	79.52
21	ப	95.53	97.21	98.32	97.77
22	ம	90.11	92.86	95.05	95.05
23	ய	90.66	94.51	95.05	96.7
24	ர	78.57	78.02	76.92	82.42
25	ல	91.16	93.92	96.13	96.13
26	வ	90.45	88.76	85.39	88.2
27	ழ	66.08	71.34	70.76	73.1
28	ள	77.53	78.65	83.15	83.71
29	ற	91.53	92.66	94.35	93.79
30	ன	65.14	65.71	72	73.14
31	ி	82.58	81.46	81.46	82.58
32	ெ	89.50	92.27	91.16	92.27
33	ே	89.71	88	82.86	86.29
34	ை	87.5	90.91	90.34	90.34
Overall %		82.04	84.4	85.5	87.35

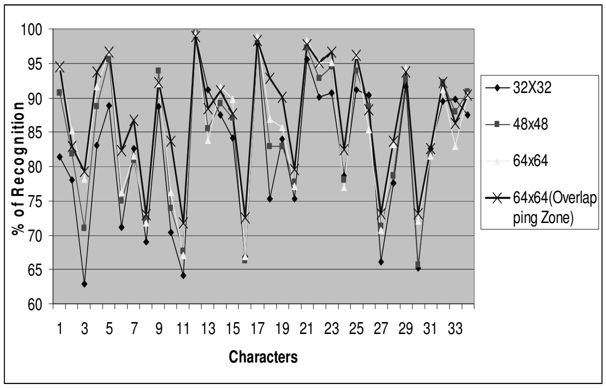


Fig. 1: Comparison of Recognition of Characters for different sized normal image

CONCLUSION

This paper 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 standard image sizes of 32X32, 48X48 and 64X64 with overlapping and nonoverlapping zones. The overall recognition rate varies from 82 to 87.4% for different image size. The experimentation result shows that the SVM based approach for 64X64 sized image with overlapping zones recognizes well when compared to other sizes. 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. 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

1. Mantas, J., 1986. An overview of character recognition methodologies, Pattern recognition, 19 (6): 425-430.
2. Govindan, V.K. and A.P. Shivaprasad, 1990. Character Recognition-A Review, Pattern Recognition, 23 (7): 671-683.
3. Siromoney *et al.*, 1978. Computer Recognition of Printed Tamil Character, Pattern Recognition 10: 243-247.
4. Chinnuswamy, P., and S.G. Krishnamoorthy, 1980. Recognition of Hand printed Tamil Characters, Pattern Recognition, 12: 141-152.
5. Pal, U., and B.B. Chaudhuri, 2004. Indian Script Character Recognition: a Survey, Pattern Recognition, 37: 1887-1899.
6. Trier *et al.*, 1996. Feature Extraction Methods for Character Recognition - A Survey, Pattern Recognition, 29 (4): 641-662.
7. Srihari *et al.*, 2000. On-line and Off-line Handwriting Recognition: A Comprehensive Survey, IEEE PAMI, 22 (1): 63-84.
8. Suresh *et al.*, 1999. Recognition of Hand printed Tamil Characters Using Classification Approach, ICAPRDT' 99, pp: 63-84.
9. Hewavitharana, S, and H.C. Fernando, 2002. A Two Stage Classification Approach to Tamil Handwriting Recognition, pp: 118-124, Tamil Internet 2002, California, USA.
10. Shanthi, N, and K. Duraiswamy, 2005. Preprocessing algorithms for the recognition of Tamil Handwritten Characters, pp: 77-82, Third International CALIBER 2005, Kochi.
11. Otsu, N, 1979. A Threshold Selection Method from Grey Level Histogram, IEEE Trans. System Man and Cyber., 9 (1): 62-66.
12. Lam, Lee, Suen, 1992. Thinning Methodologies-A Comprehensive Survey, IEEE PAMI, 14(9): 869-885.