Printed and Handwritten Character & Number Recognition of Devanagari Script using SVM and KNN

Anilkumar N. Holambe¹, Dr. Ravinder C. Thool²,
¹ Department of Information Technology
College Of Engineering
Osmanabad-413512 (M.S.) India
anholambe@yahoo.com
² Professor and Head
Department of Information Technology
Shri Guru Gobind Singhji Institute Of Engg & Technology,
Vishnupuri, Nanded-431606 (M.S.) India
rcthool@yahoo.com

Abstract—Recognition of Devanagari scripts is challenging problems. In Optical Character Recognition (OCR), a character or symbol to be recognized can be machine printed or handwritten characters/numerals. There are several approaches that deal with the problem of recognition of numerals/character. In this paper we have compared SVM and KNN on handwritten as well as on printed character and numerical database for this we have created four different database.

Index Terms—Devanagari, Optical Character Recognition, SVM, KNN, Database.

I. INTRODUCTION

Handwritten and printed character & numeral recognition has significant application potentials. The need for such a system has been increasingly felt even in a country like India. However, although significant development has already been achieved on this in scripts of developed nations, not much work has been reported on devanagari scripts. The development of a handwritten and printed character recognition engine for any script is always a challenging problem mainly because of the enormous variability of handwriting styles. The above factors provided the motivation for the proposed research work.

For this work, we have studied off-line recognition of handwritten Devanagari numerals. Devanagari is the script of a number of Indian languages, including Hindi and Marathi. Hindi is the third most popularly used language in the world after Chinese and English. The earliest available work on recognition of hand printed Devanagari characters is found in [1]. For recognition of handwritten Devanagari numerals, Ramakrishnan et al. [2] used independent component analysis technique for feature extraction from numeral images. Bajaj et al [3] considered a strategy combining decisions of multiple classifiers. In all these three studies, very small sets of samples were considered. In an attempt to develop a bilingual handwritten numeral recognition system, Lehal and Bhatt [4] used a set of global and local features derived from the right and left projection profiles of the numeral images for recognition of handwritten numerals of Devanagari and Roman scripts. There are also some studies on handwritten character recognition of other Indian scripts [5, 6].

The development of an efficient system for handwriting recognition, needs a large set of samples with ground truth. Generation of such a data set is always difficult since it is time consuming and labor intensive [7]. Such standard data sets for any Indian script did not exist. However, recently, a large database of handwritten Devanagari numeral and character images has been developed by authors. In this paper we have implemented SVM and KNN classifier. Both are used on printed and handwritten database.

II. EXTRACTION OF FEATURES

A. Database of printed and handwritten Devanagari numerals and characters.

In the present work we have developed printed and handwritten database. For printed we have used different ISM office fonts. For handwritten we have collect data from people of different age groups and from different profession. This data were scanned at 600 dpi using a HP flatbed scanner and stored as gray-level images. A few samples from this database are shown in Fig. 1.
The database is exclusively divided into training and test sets. The distribution of samples in these training and test sets over 10 digit classes for numerical data. For Devanagari (character script has about 11 vowels (‘svar’) and 33 consonants or (‘vyanjan’),and 11 modifiers so we organized data in 55 character classes. The handwritten database is collected from marathi peoples.

B. Preprocessing

The preprocessing steps performed in this work are steps for rectification of distorted images, improving the quality of images for ensuring better quality edges in the subsequent edge determination step and size normalization. In the scanning process, some distortion in images may be introduced due to pen quality, light hand handwriting, poor quality of the paper on which the numerals and characters are written etc. Further, many times edges show discontinuities leading to erroneous feature extraction. For cleaning of possible noise in the input image, it is first binarized by Otsu’s thresholding technique followed by its smoothing using median filter of window size 5. In this first we reduces a gray image into an intensity image and approximately segments the image by intensity thresholding. Then, it refines the segmentation using image edges. Here we have used segmented the character or numerical as vertical, horizontal and arc.

we have processed all printed characters and numerical similarly as shown in figure 3.

C. K-Nearest Neighbor Classification

The k-nearest neighbors classifier is used for classifying the Bengali alphabets. A detailed discussion on the K-NN classification technique can be found in dasarathy [8]. A short but formal definition of the K-NN Classification is as follows. Given a set of prototype vectors, \( T_{sb} = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \), the input vectors being \( x, \in R^n \) and corresponding. Targets being \( y, \in \{1,2, \ldots, c\} \), let \( R^s(x) = \{ x' : \|x - x'\| \leq r^2 \} \) be a ball centered in the vector x in which K prototype vectors \( x_i, i \in \{1,2, \ldots, l\} \), lie, i.e. \( \|x, x, \in R^s(x)\| = k \) the k-nearest neighbor classification rule \( q : X \rightarrow Y \) is defined as \( q(x) = \arg \max y (x,y) \),

Where v(x,y) is the number of prototype vectors \( x_i \) with targets \( y_i = y \) which lie in the ball \( x, \in R^s(x) \).

D. SVM Support Vector Machines Classification

Support vector Machines [10,9] are pair-wise discriminating classifiers with the ability to identify the decision boundary with maximal margin. Maximal margin results in better generalization – a highly desirable property for a classifier to perform well on a novel data set. From the statistical learning theory point of view, they are less complex (smaller VD dimension) and perform better (lower actual error) with limited training data. Classifiers like KNN provide excellent results (very low empirical error) on a data set on which
Identification of the optimal hyper plane for separation involves maximization of an appropriate objective function. The training process results in the identification of a set of important labeled support vectors \( x_i \) and a set of coefficients \( \alpha_i \). Support vectors are the samples near the decision boundary and most difficult to classify. They have class labels \( Y_i (i.e., \pm 1) \). The decision is made from:

\[
f(x) = \text{sgn}\left( \sum_{i=1}^{l} \alpha_i y_i K(x_i, x) \right)
\]

The function \( k \) in the previous equation is called the kernel function. It is defined as \( k(x,y) = \phi(x) \phi(y) \), where \( \phi : R^d \rightarrow H \) maps the data points in \( d \) dimensions to a higher dimensional (possibly infinite dimensional) space \( H \) for a linear SVM, \( k(x,y) = x.y \). We do not need to know the values of the images of the data points in \( H \) to solve the problem in \( H \). By finding specific cases of Kernel functions, we can arrive at neural networks or radial Basis functions.

### IV. RESULT AND OBSERVATION

Data used for the present work were collected from different individuals. We considered 15000 basic characters (vowels as well as consonants) and 10000 numerical samples of Devnagari for the experiment of the proposed work. We also formed printed database of ISM office fonts, in which we have used font size of 16 and different fonts.

**TABLE I. RECOGNITION RATE OF KNN CLASSIFIER**

| Numerical | vowels ('svar')(|%) | consonants ('vyanjan') without modifiers(|%) | consonants ('vyanjan') with modifiers(|%) |
|-----------|----------------|---------------------------------|---------------------------------|
| Handwritten | 79.08 | 76.02 | 73.07 |
| Printed | 94 | 91 | 89 |

**TABLE II. RECOGNITION RATE OF SVM CLASSIFIER**

| Numerical | vowels ('svar')(|%) | consonants ('vyanjan') without modifiers(|%) | consonants ('vyanjan') with modifiers(|%) |
|-----------|----------------|---------------------------------|---------------------------------|
| Handwritten | 85 | 83 | 80 |
| Printed | 93 | 95 | 94 |

From our experiment we have observe that recognition accuracy of SVM classifier is more than that of KNN classifier as shown in Table I & II, the accuracy is more in printed database than in handwritten database. The rejection rate is very high in handwritten database. Support Vector Machines provide better results than KNN classifiers. Computational complexity of KNN increases with more and more labeled samples.

**TABLE III. REJECTION VERSUS ERROR RATE COMPTUATION**

<table>
<thead>
<tr>
<th></th>
<th>Printed Database</th>
<th>Handwritten Database</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rejection (%)</td>
<td>Error (%)</td>
</tr>
<tr>
<td>Numerical</td>
<td>4.00</td>
<td>30.00</td>
</tr>
<tr>
<td>vowels ('svar')</td>
<td>5.00</td>
<td>20.00</td>
</tr>
<tr>
<td>consonants ('vyanjan') without modifiers</td>
<td>7.00</td>
<td>18.01</td>
</tr>
<tr>
<td>consonants ('vyanjan') with modifiers</td>
<td>15.00</td>
<td>11.00</td>
</tr>
</tbody>
</table>

**Error Rate**

![Error Rate](samples.png)

From our experiment we have observe that recognition accuracy of SVM classifier is more than that of KNN classifier as shown in Table I & II, the accuracy is more in printed database than in handwritten database. The rejection rate is very high in handwritten database. Support Vector Machines provide better results than KNN classifiers. Computational complexity of KNN increases with more and more labeled samples.
CONCLUSIONS

The performance of character recognition is dependent on the accuracy of stroke recognition. The results obtained for recognition of Devanagari show that reliable classification is possible using SVMs. The results also indicate the scope for further improvement, especially in the case of confusing character recognition. The advantage of SVM Classifier over other classifiers is that an Indian Language OCR Systems generally has a large number of classes and high dimensional feature vectors. Variability of characters is also very high at each occurrence. SVMs are well suited for such problems since they have excellent generalization capability.

REFERENCES