Structural Feature Based Classification of Printed Gujarati Characters

Mukesh Goswami[1] and Suman Mitra[2]

[1] Department of IT, Faculty of Technology, D.D. University
mgoswami.it@ddu.ac.in
[2] Dhirubhai Ambani Institute of Information and Communication Technology
suman_mitra@daiict.ac.in

Abstract. This paper presents a Structural feature based method for classification of printed Gujarati characters. The ability to provide incremental definition of characters in terms of its native components makes the proposal unique and versatile. It deals with varied sizes, font styles, and stoke widths. The features are validated on subset of machine printed Gujarati characters using a simple rule based classifier and the initial results are encouraging.

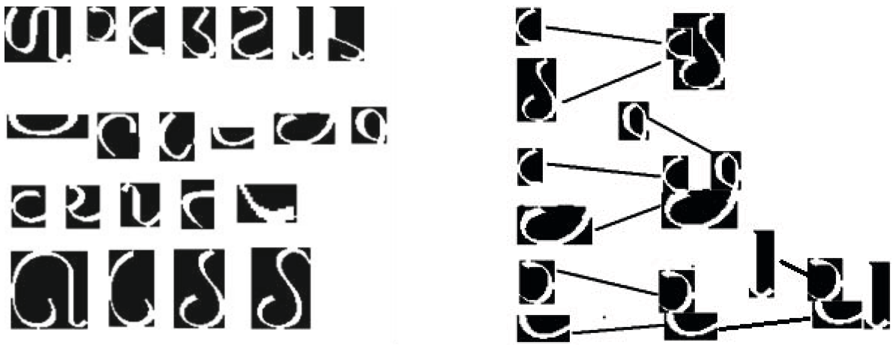
Keywords: Structural Features, Character Classification, Gujarati Characters.

1 Introduction

Development of Optical Character Recognition (OCR) technology for Indian text is more challenging then western text because of the complex character set as well as existence of modifiers and joint characters. Motivated by this many researchers have started working on the OCR for Indian text over a decade ago and it is still in moderate phase for many languages. The majority of the work found in the literature for classification of characters from Indian text can be divided into two major streams 1) Structural feature based approach and 2) Transform domain feature based approach. For many languages like Hindi, Bangla, Gurumukhi, Oriya etc., the structural features along with simple rule based classifier like decision trees have performed well[1–4]. On the other hand, the south Indian scripts like Tamil, Telugu, and Kannada, where it is difficult to identify characters from the general shape and structure, the transform domain features along with sophisticated classifiers like Neural Networks, Support Vector Machine etc. have done well[5–8]. Some work on classification of Gujarati text using transform domain features can be found in the literature[9–12]. Gujarati is derived from the ancient Devanagari script and having close resemblance with other north Indian script, primarily Hindi. The major difference between Gujarati and other north Indian script is the absence of “Shirorekha”, a head line running through all the characters forming the word. Even though Gujarati characters have well define shape and structure, but no structural features based method considered so far for the classification of Gujarati text. This paper attempts to propose a

P. Maji et al. (Eds.): PReMI 2013, LNCS 8251, pp. 82–87, 2013.

structural feature extraction method and its use in classification of subset of Gujarati characters. The rest of the paper is organized as follow Section 2 describes the brief of various steps used in structural feature extraction method and its use in classification, followed by simulation result and conclusion in Section 3 and 4 respectively.



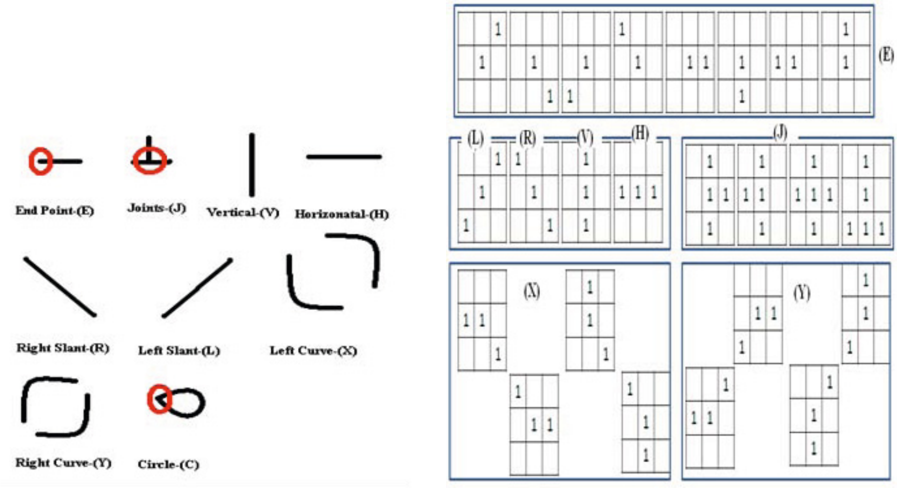
(a) (b)

Fig. 1. (a) Subset of components formulating majority of Gujarati character set (b) Formation of characters from the components

2 Proposed Methodology

The Fundamental idea for the formation of Gujarati characters is to consider every characters as combination of some components (also referred as strokes in some literature) connected in specific ordered. This break up of characters in components helps in exploiting the reusable components in similar looking characters. A total of 30 components are identified which formulates majority of characters from Gujarati character set. Some of these components are shown in Fig. 1a. An example of how characters are formulated by combining the components in a specific order is shown in Fig. 1b. Every component in turn can be defined as some sequence of primitive strokes. Set of primitive strokes that formulate a complex component is shown in Fig. 2a. All primitive strokes are represented by string symbol (as shown in Fig. 2a). Thus the component is described by sequence of string symbols representing primitive strokes and the character in turn can be defined as the sequence of components occurring in some specific order (as shown in Fig. 3). A complete system for identification of native components and formation of characters as ordered set of native components is described as follows.

Preprocessing and Applying 3x3 Pattern Mask. Input to the system is a binary character image which is passed through various preprocessing stages like noise removal, resizing and thinning. Thinning is one of the most important preprocessing steps that converts elongated character image into one pixel wide thinned image that preserves the original shape of the character without



(a) (b)

Fig. 2. (a) Set of primitive components (b) Set of 3x3 pattern mask to identify primitive components

losing the connectivity. Also it facilitates more elegant description of component shapes.Various 3x3 pattern masks (as shown in Fig. 2b) are applied to detect primitive strokes from the thinned image. Every detected primitive stroke is replaced by predefined numerical code. Thus, an MxN input image is converted into an MxN matrix of numerical codes.

Scanning, Component Separation and Symbol Array Generation. In order to preserve the component ordering, MxN matrix of numerical codes is scanned. Since Gujarati writing is left to right and top to bottom, zigzag scanning is used to find the start point. Modified contour tracing algorithm that selects the next component in clockwise direction is designed to scan components in nearly writing flow order starting from the start point. For example writing flow order of components present in the character image shown in Fig. 3 is $\{C_1C_2C_3C_4C_5\}$. During scanning, numerical code(s) are replaced with equivalent character symbols of primitive strokes to obtain symbol string representation. The components are separated from the symbol string by using end points symbol (E) and junction points symbol (J) as separators, thus every component is represented by string of symbol as shown in Fig. 3.

Noise Removal from the Symbol String and Identification of Component. A noise removal technique is designed to further fine tune the symbol string representation. Regular Expression (RE) matching based method is designed to identify every component, represented by a symbol string. It takes component string and RE file as input. The RE file defines set of regular expression for each component class. The method then generates target component string using all RE's present in the file and gives a matching score close to 1.0

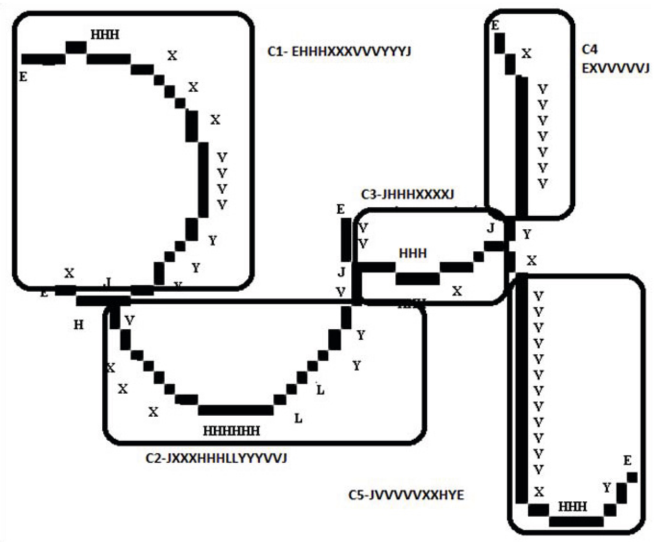


Fig. 3. Formation of Character as sequence of components described as sequence of primitive strokes represented by string symbols

if the string is fully generated by the given RE. Finally, the component represented by RE having maximum matching score is selected as target component and replaced by its id. Thus the character is finally represented by an ordered set of component ids'.

Classification. A simple rule based algorithm is designed to check the discriminating power of the proposed structural features. Every component is assigned some weight depending on how much it contributes in the recognition of a given character. Accumulated weights for all characters are maintained in the form of a vector. The algorithm proceeds as follows.

- Initialize the accumulated weight vector $W = \{w_{1,}w_{2,......}w_m\}$ to zero, for all m number of character classes.
- Get Input character image and pass it through the feature extraction stages to obtain the ordered set of components ids' $C = \{c_{1,}c_{2,} \ldots c_n\}$.
- Select the appropriate rule set $R(n)$ for different character classes depending on the number of component(n). Rule R_x for given character class x is a set of component value pair $< r_{xi}, v_{xi} >$ where v_{xi} is weight of component r_{xi} with respect to character class x.

```
1. For every rule $R_i$ corresponding to class $i$ in the selected rule set $R(n)$
2.     For every component value pair $< r_{ij}, v_{ij} >$ in $R_i$
3          if $r_{ij} \in C$ then increase the accumulated weight $w_i$ by $v_{ij}$
4.         else component $r_{ij}$ doe not contribute in defining character $i$.
```

- Select the Character class y with maximum of accumulated weights (w_y) as predicted class

3 Simulation Results

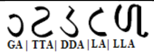
The proposed method was tested on moderate size database of 4000 machine printed character symbols from 20 different pages of 4 different machine printed books. The result of the experiment are shown in Table 1. It is evident from the results that even though the classifier being simple it gives very high accuracy for simple characters having 1, 2 or 3 components (around 95%) which constitute almost half the character set. The accuracy drops as the number of components increases in the character. When number of components are large, it becomes difficult to assign optimum weights to all character component manually. Thus a mechanism is needed to automatically find the optimum weights to components. Also the classifier does not consider ordering of component, which is crucial and may lead to better accuracy.

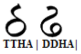
4 Conclusions and Future Work

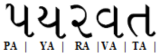
A structural feature extraction method is proposed for classification of printed Gujarati text and tested on machine printed character symbol data set of size 4000. The accuracy obtained is very high for simple character with small number of components, however method does require some improvement like finding the component weights automatically and exploiting the order of component. The salient feature of the proposal is its ability to provide incremental definition of

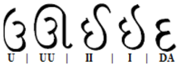
Table 1. Accuracy of various character class

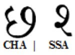
Sr.No.	No. of Components	Character Class	Accuracy

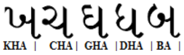
1	1		GA=97% TTA=98% DDA=97%
			LLA=91% LA=96%

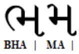
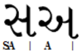
2	2		TTHA=91.66% DDHA=81.25%

3	3		PA=95% YA=90% RA=96%
			VA=88% TA=90%

4	3		U=87% UU=66.66% II=97%
			I=95% DA=95%

5	4		CHA=93% SSA=60%

6	5		KHA=83% CHA=47% GHA=86.66%
			DHA=98% BA=94%

7	6		BHA=85% MA=66%
8	7		SA=86% A=66%

characters in terms of components. As the scope of the work is to evaluate the proposed feature hence comparision with other methods is omited.

References

1. Sinha, R.M.K., Mahabala, H.N.: Machine Recognition of Devanagari Script. IEEE Transactions on Systems, Man, and Cybernetics 9, 435–441 (1979)
2. Chaudhuri, B.B., Pal, U.: A complete printed Bangla OCR system. Pattern Recognition 31, 531–549 (1998)
3. Pal, U., Chaudhuri, B.B.: Printed Devanagiri Script OCR System. Vivek 10, 12–24 (1997)
4. Lehal, G.S., Singh, C.: A Gurmukhi script recognition system. In: Proc. of the 15th International Conference on Pattern Recognition (ICPR), pp. 557–560 (2000)
5. Aparna, K.G., Ramakrishnan, A.G.: A complete Tamil optical character recognition system. In: Lopresti, D.P., Hu, J., Kashi, R.S. (eds.) DAS 2002. LNCS, vol. 2423, pp. 53–57. Springer, Heidelberg (2002)
6. Jawahar, C.V., Pavan Kumar, M.N.S.S.K., Ravi Kiran, S.S.: A bilingual OCR for Hindi-Telugu documents and its applications. In: Proceedings of Seventh International Conference on Document Analysis and Recognition, pp. 408–412. IEEE Computer. Soc. (2003)
7. Manjunath, V.N., Aradhyal, P.S., Kumar, G.H., Noushathl, S.: Fisher linear discriminant analysis based technique useful for efficient character recognition. In: Proc. of the 4th International Conference on Intelligent Sensing and Information Processing, pp. 49–52 (2006)
8. Ashwin, T., Sastry, P.: A font and size-independent ocr system for kannada documents using SVM. Sadhana 27 (2002)
9. Antani, S., Agnihotri, L.: Gujarati character recognition. In: Proceedings of the Fifth International Conference on Document Analysis and Recognition, ICDAR 1999 (Cat. No.PR00318), pp. 418–421. IEEE (1999)
10. Dholakia, J., Negi, A., Rama Mohan, S.: Progress in Gujarati Document Processing and Character Recognition. In: Govindaraju, V., Setlur, S. (eds.) Guide to OCR for Indic Scripts: Document Recognition and Retrieval, pp. 73–95. Springer Publishing Company (2009)
11. Hassan, E., Chaudhury, S., Gopal, M., Dholakia, J.: Use of MKL as symbol classifier for Gujarati character recognition. In: Proceedings of the 8th IAPR International Workshop on Document Analysis Systems DAS 2010, pp. 255–262. ACM Press, New York (2010)
12. Goswami, M. M., Prajapati H. B., Dabhi V. K.: Classification of printed Gujarati characters using SOM based k-Nearest Neighbor Classifier. In: IEEE International Conference on Image Information Processing, ICIIP 2012. pp. 1-5. IEEE (2013)