International Journal of Modern Trends in Engineering and Research
www.ijmter.com
e-ISSN No.:2349-9745, Date: 2-4 July, 2015

Devanagari Plane Script Recognition and Translation System Using TM-ST Approach

Bamb Kalpesh K.
Assistant Prof.- E&TC Dept., SNJB KBJ's College of Engineering, Chandwad, Nashik, India
Kalpeshkumar.bamb@gmail.com

Abstract- We envisioned serving simple technique with novel idea for the newcomers, working in the Devnagari Optical Character Recognition (DOCR) area. In this system, template matching technique and statistical technique is combinely used for the character recognition. Statistical information is considered as a feature of an image and here pixels values are considered as feature to achieve the recognition. The numbers of comparisons are drastically reduced compared to traditional template matching technique hence performance of the system is improved.
The correlation and related methods are used to find images that are similar to a template. The results are compared with results of template matching technique in terms of elapsed time that is time required to recognize the character here. The recognized devanagari characters then translated in to English through Google translator using Matlab which will provide human assistance in different applications. In English OCR results were of such type that's it can produce technology driven applications. The algorithm used is very effective and can serve as a basis for further research towards other similar Indian scripts.

Keywords- Correlation, Devanagari optical character recognition, segmentation, statistical technique, template matching

I. INTRODUCTION

Previously a lot of work done in DOCR area and still research is going on. Devnagari is the national language of India and generally spoken by 750 million people in India. Hence devanagari should be given more special consideration for analysis and document retrieval due to its popularity. DOCR is the system which can be classified based upon two major criteria: the data acquisition process (on-line or off-line) and the text type (machine-printed or hand-written) [2, 16].
No matter which class the problem belongs, in general there are five major stages in the any OCR system as Pre-processing, Segmentation, Feature Extraction, Character Classification, Post-processing . Off-line, on-line system has different approaches, but they share a lot of same problems, solutions [3, 7]. Devnagari consonant set is provided in Table 1.

Table 1. Consonants

क	ख	ग	घ	ङ	च	छ	ज	झ	ञ	ट
ठ	ड	ढ	ण	त	थ	द	ध	न	प	फ
ब	भ	म	य	र	ल	व	श	ष	स	ह

@IJMTER-2015, All rights Reserved

II. SYSTEM DESIGN

We develop a system prototype which reads the characters from the scanned or saved copy of printed Devanagari script by using combined approach (**method 2**) and compares the results with results of template matching technique (**method 1**) and then translate recognized character from devanagari to English through Google translator. As it is new idea and first attempt towards distinct combined approach, we consider the following constraints:

1) Characters should not be overlapping.
2) Inputs image should not contain ardhakshers (joint characters) and modifiers.

A good text recognizer system has many commercial and practical applications [15]. The system overview is shown in Fig 1.

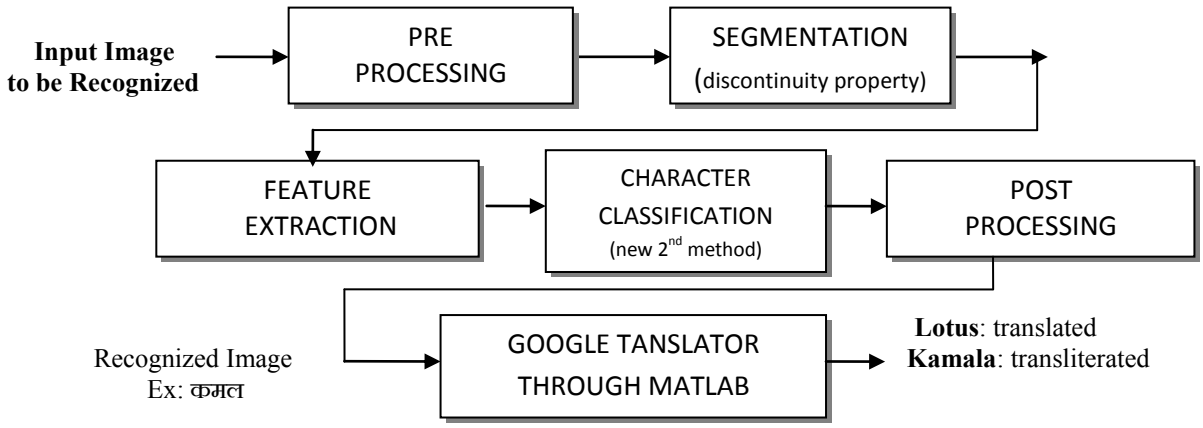

Figure 1. System Block diagram

2.1 Image Pre-processing

Input data usually stored in a file of picture elements, called pixels. These pixels have values: OFF (0) or ON (1) for binary images. This collected raw data must be further analyzed to get useful information. Such processing includes different steps named as pre-processing. Figure 2 below shows image after preprocessing.

Figure 2. Image after pre-processing

@IJMTER-2015, All rights Reserved

2.2 Segmentation

Segmentation algorithms based on either of two properties of intensity values namely discontinuity and similarity [1].

Here we used first approach is to partition of an image based on abrupt changes in intensity. Words can further be split in to individual character for classification by removing Shirorekha [9]. Fig. 3 shows the segmentation operation.

कमल हसन

a) Devanagari text image

क म ल ह स न

b) Segmented image

Figure 3. a) Text image b) Segmented image

2.3 Feature Extraction

Feature extraction is the process to retrieve the most important data from the raw data. The most important data means that's on the basis of that's the characters can be represented accurately [10]. In first method for character recognition template matching or matrix is used in which no features are actually extracted [6].

In second approach of recognition called statistical technique of recognition, 'statistical feature' method is used for feature extraction. In this method pixel values are used as feature set and how recognition is achieved is discussed in detail in further section.

2.4 Combined Approach for Character Recognition

We have implemented the Devanagari recognition system using two different methods or approaches:

1) Traditional Template matching approach (Method 1)

2) New Statistical technique approach combined with TM (Method 2)

Template matching or matrix matching is the simplest way of character recognition, based on comparing the stored prototypes as shown in Fig. 4 against the character or word to be recognized. The matching operation determines the degree of similarity between two vectors [11]. Correct segmentation of individual symbols decides the accuracy of character recognition technique. Cross correlation and normalized correlation method is used to compute the highest degree of matching [12].

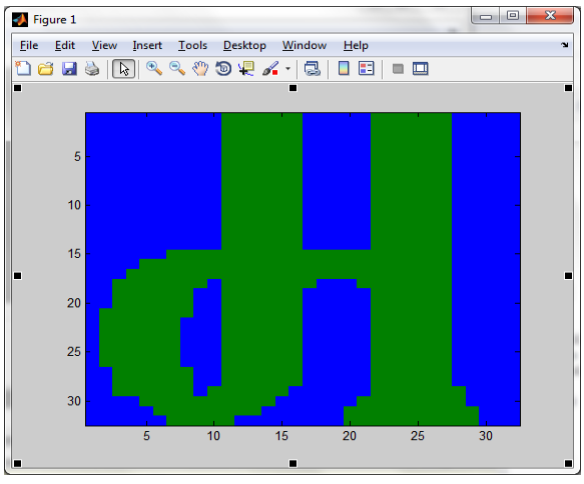

Figure 4. Template for 'म' 32x32 (.m file) (similar database is created for all characters)

@IJMTER-2015, All rights Reserved

In the combined approach pixel values are considered as feature to achieve the recognition. A pixel (picture element) is a dot or the most fundamental unit that makes up the image. Pixel is the term used most widely to denote the elements of digital image. Here pixels values are considered as feature to achieve the recognition [13, 14]. ***Matlab*** is the software that was used in developing the system prototype.
Idea behind the implemented approach is as follow:

1) Find out the number of non-zero elements that is 1's (pixel value) available in every template image of size 32 x 32 (total 1024 pixels). (Aparajita font is used as basic font to create a template)
2) Make four groups of consonants with specific range of number of 1's as shown in Table 2.
3) Calculate the number of 1's (pixel value) of input character.
4) Compare this input character only with one belonging group out of four.

In this method number of 1's (pixel value) or we can say that non-zero elements are find out of every template image. Similarly number of 1's (pixel value) or we can say that non-zero elements are find out of input image. Resulted output correlate with one of the group and after that comparison that is correlation operation is done. Hence time reduced to recognize the character as comparisons reduced. Hence comparison to recognize every character reduced to 9 from 33. Table 2 shows group of consonants as per their number of non-zero pixels in the related consonant.

Table 2. Group of consonants shows number of non-zero pixels in each consonant

GROUP	A	B	C	D
Non-zero Pixels (1's)	< 482	482 to 502	503 to 530	> 530
CONSONANTS	3	6	1	4
	5	10	2	15
	8	12	7	22
	11	16	9	23
	13	19	14	24
	18	20	17	25
	27	21	26	29
	32	28	30	31
			33	

Algorithm – steps to achieve the recognition:

1) Select the character image to be recognized.
2) Pre-processing (thresholding, inversion and normalization) done on input image.
3) If fused image that is a 'word' then segmentation operation is carried out.
4) Find out the number of pixels having value '1' in the input image of size 32 x32 that is non-zero pixels out of 1024 pixels.
5) If number of 1's is less than 482 then compare this image only with 1st group.
6) If it is between 482 to 502 then correlate it only with second group of templates.
7) If it is between 503 to 530 then correlate it with third group of templates.
8) Otherwise compare it with fourth group of templates. (more than 530)
9) Each character is compared with every template available in group of using CC.
10) Highest match character is stored as recognized character.

@IJMTER-2015, All rights Reserved

2.4.2 Matching with Correlation

Cross correlation is a standard method of estimating the degree to which two series are correlated. Correlation and related methods are used to find images or locations in an image that are similar to a template. There exists a template for all possible input characters. If *I(i ,j)* is the input character, *Tn(i, j)* is the template n, then the matching function *s(I ,Tn)* will return a value indicating how well template *n* matches the input character. Matching functions is based on the equation 1 shows normalized cross correlation (-1 to 1).

$$s(I,Tn) = \frac{\sum_{i=0}^{w}\sum_{j=0}^{h}(I(i,j)-|I|)(Tn(i,j)-|Tn|)}{\sqrt{\sum_{i=0}^{w}\sum_{j=0}^{h}(I(i,j)-|I|)^2}\ \sqrt{\sum_{i=0}^{w}\sum_{j=0}^{h}(Tn(i,j)-|Tn|)^2}} \quad \text{---1}$$

Where, |I| and |*Tn*| shows mean intensity of input image and template image respectively [1] [6].

III. RESULTS and DISCUSSION

Table 3. Performance analysis of two methods in terms of elapsed time

	ELAPSED TIME in SEC				ELAPSED TIME in SEC	
Consonants	Method 1	Method 2		Consonants	Method 1	Method 2
क	0.029	0.0036		ध	0.0231	0.0022
ख	0.0145	0.0019		न	0.0145	0.0011
ग	0.0202	0.0021		प	0.0933	0.0093
घ	0.0824	0.0089		फ	0.0863	0.0092
क्ष	0.0176	0.0014		ब	0.0143	0.0013
च	0.0236	0.0045		भ	0.0212	0.002
छ	0.0112	0.0054		म	0.0525	0.0056
ज	0.02	0.0099		य	0.0798	0.0077
झ	0.0987	0.0085		र	0.0201	0.0034
त्र	0.0264	0.0037		ल	0.0145	0.0019
ट	0.0213	0.0087		व	0.0112	0.001
ठ	0.0713	0.0062		श	0.0123	0.0034
ड	0.02	0.0039		ष	0.0747	0.0067
ढ	0.0112	0.0023		स	0.0256	0.0027
ण	0.0164	0.0016		ह	0.0202	0.002
त	0.0201	1.0021				
थ	0.0187	0.0096				
द	0.0533	0.0054				

@IJMTER-2015, All rights Reserved

3.1 Comparison Analysis of Method 1 and Method 2

1] **Recognition with method 1** – consider the example of word "KAMAL = कमल".

There are total 3 consonants to be recognized. As per the template matching method every input character will be compared with every template image available in database. There are total 33 templates are available. Hence number of comparison required to recognize the word KAMAL is 3 x 33 = 99.

2] **Recognition with method 2** – Non-zero elements of image (32x32) are considered as a feature of an image (refer table 2). Character 'क' contains 529 pixels with value 1 hence it belongs to group C. 'म' has 608 and 'ल' having 486 pixels (1's). Hence only 9+8+8 = 25 comparisons required. That is 74 (99-25) fewer comparisons required in second method. In second method only 9 comparisons will required instead 33 to recognize 1 character. Hence elapsed time that is time required to recognize the character is reduced drastically.

A result shows that 2nd method giving satisfactory results. We can observe that around 10 fold (10 times) less time is required to recognize the character in method 2 compared to method 1(refer table 3).

3.2 Translation

Proposed system is extended with useful application which provides human assistance in terms of translation. Now days there are many readymade applications of translation are available in the market. But as English is a global language and spoken worldwide most of the application are of English to other language translator. Recognized word as "KAMAL = कमल" gives the translated output as "LOTUS" in English as shown in Fig. 5.

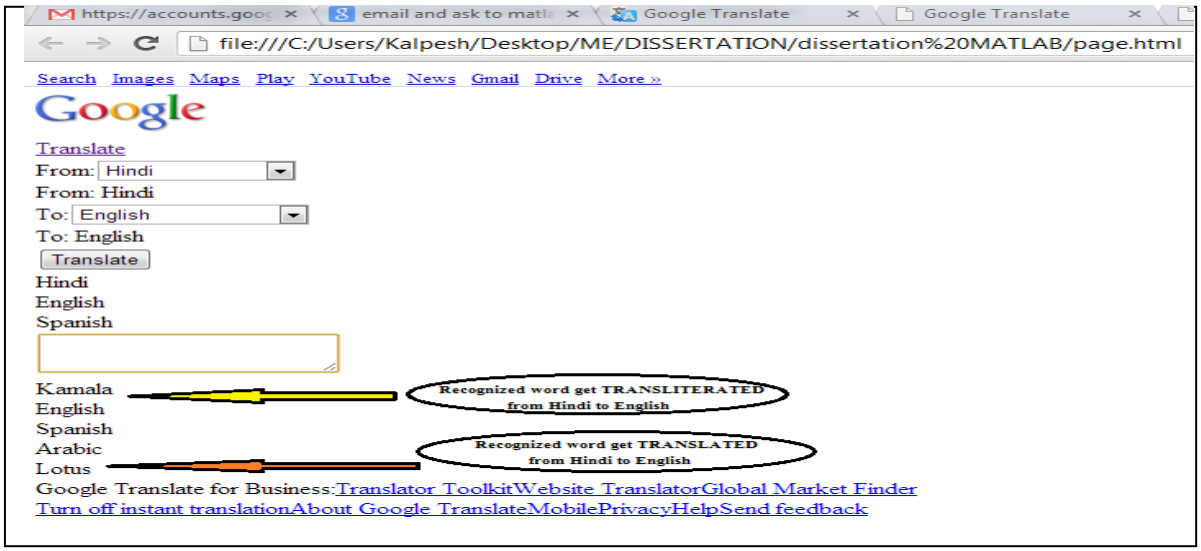

Figure 5. Hindi to English translation of Recognized Word 'Kamal' as 'Lotus

@IJMTER-2015, All rights Reserved

IV. CONCLUSION

Devanagari script recognition system using combined approach is implemented using software MATLAB. It is found that the numbers of comparisons are drastically reduced compared to traditional template matching technique. Also the elapsed time is reduced almost 10 times compare to template matching technique. Hence overall performance of the system is improved.

The proposed system gives satisfactory results and able to cope up with real time applications. The recognized devanagari characters or words are then translated to English to Google translator using Matlab which will provide human assistance in different applications. The algorithm used is very effective and can serve as a basis for further research towards other Indian scripts like Tamil, Bengali, Gujarati, and Oriya etc.

REFERENCES

[1] Rafel C. Gonzalez, Richard E. Woods, "Digital Image Processing", Pearson Prentice Hall, 3rd edition, 2009.
[2] Pratap, P.V.Shwetank, "A Review of Devnagari Character Recognition from Past to Future", IJCST, vol. no.3, pp. 77-82, March 2012.
[3] Vikas J Dongre, Vijay H Mankar, "A Review of Research on Devanagari Character Recognition", International Journal of Computer Applications (0975 –8887) Vol. 12 no.2, pp. 8-15, Nov. 2010.
[4] Hiromichi Fujisawa, Yasuaki Nakono, "Segmentation Methods for Character Recognition from Segmentation to Documentation Structure Analysis", IEEE, vol.80, no.7, pp. 1079-1091, July 1992.
[5] U. Pal, B.B. Chaudhari, "Indian Script Character Recognition: A Survey", IJERT, pp. 78-85, March 2004.
[6] Kailsh S. Sharma, A.R. Karwankar, Dr. A.S. Bhalchandra, "Devanagari Character Recognition using Self Organizing Maps", "ICCCCT 2010", IEEE, vol. 10, pp. 687-691, Aug. 2010.
[7] Mohmad Cheriet, Nawwaf Kharma, "Character Recognition Systems", John Wiley and Sons INC. Pub., 2007.
[8] R. Jayadevan, Satish R. Kolhe, "Offline Recognition of Devanagari Script: A Survey", IEEE, vol.41, no.6, pp. 782-796, Nov. 2011.
[9] S.K. Shah, A.Sharma, "Design and Implementation of Optical Character Recognition System to Recognize Gujrathi Script using Template Matching", IEIET, pp.44-49, June 2010.
[10] V. Ramana Murthy, M. Hanmundula,"Zoning Based Devnagari Character Recognition System", IJCA, vol.no. 27, pp. 45-49, Aug. 2011.
[11] Vikas J Dongre, Vijay H Mankar, "Devanagari Document Segmentation Using Histogram Approach", IJCSEIT, vol.1, no.3, pp. 46-53, August 2011.
[12] Mo Wening, Ding Zuchun, "A Digital Character Recognition Algorithm Based on the Template Weighted Match Degree" ICASP, vol.18, pp. 53-60, Jan. 2013.
[13] Mahesh Jangid, "Devanagari Isolated Recognition by using Statistical Features", IJCSE, vol.3, no.6, pp. 2400-2407, June2011.
[14] Chrinstin Pint, "Template Matching using Sum, Difference in Pixel Intensities", ICMV, pp. 978-988, Feb. 2011.
[15] Aarti Desai, Latesh Malik, Rashmi Velekar, "A new Methodology for Devanagari Character Recognition", JMIJMT, vol.1, pp. 56-60, Jan. 2011.
[16] Vedprakash Agnihotri, "Offline Devanagari Handwritten Script Recognition System", IJITCS, vol.8, pp. 37-42, July 2012.

@IJMTER-2015, All rights Reserved