Online Handwritten Character Recognition for Telugu Language Using Support Vector Machines

K. Vijay Kumar, R.Rajeshwara Rao

Abstract— A system for recognition of online handwritten telugu characters has been presented for Indian writing systems. A handwritten character is represented as a sequence of strokes whose features are extracted and classified. Support vector machines have been used for constructing the stroke recognition engine. The results have been presented after testing the system on Telugu scripts.

Keywords: Online Handwritten Character, Recognition, Stroke, feature extraction, Support Vector Machine.

I.INTRODUCTION

Handwritten Character Recognition(HCR) is the process of classifying written characters into appropriate classes based on the features extracted from each character. Character recognition is a part of a handwriting recognition problem.

HCR can be divided into two categories namely, online and off-line. On-line character recognition involves the identification of characters while they are written [1] and deals with time ordered sequences of data, pen up, and down movement and pressure sensitive pads that record the pen's pressure and velocity [2] (cite in [3]). On the other hand, off-line character recognition involves the recognition of already written character patterns in scanned digital image. The off-line character recognition is more complex and requires more research compared to an on-line character recognition

II.OBJECTIVES

The development of computer technology is widely used nowadays, expanded to an extent that computers become increasingly demanding into everyday life. Machine simulation of human functions [4] has been a very challenging research field since the advent of digital computers.

An automated character recognition system is a solution which will interpret characters automatically. The automatic recognition of character can be extremely useful where it is necessary to process a large volume of handwritten characters. HCR is a challenging problem since there is a variation of same character due to the change of fonts and sizes. The differences in font types and sizes make the recognition task difficult and resulting the recognition of character process become not good. Based on the problem statement above, this paper encompasses a set of objectives that is associated with milestones of the research process.

The objectives are:

• To review the issues and techniques for HCR which are :

Step 1: Preprocessing,

Step 2: Step Stroke Pre-Classification and

Step 3: Feature Extraction

Manuscript received December, 2013

K. Vijay Kumar, Asst. prof (CSE), Vivekananda Institute of Technology and Science SET, Karimnagar, AP, India

R.Rajeshwara Rao, Assoc.prof (CSE), JNTUK University College Of Engineering, Vizianagaram, AP, India

• To review support vector machine (SVM) as the classification method.

III. STEP 1: PREPROCESSING

The preprocessing algorithms are dealing with more specific problems such as binarization (thresholding), smoothing & noise removal, normalization, thinning, segmentation, skew detection and slant correction. Preprocessing stage involves all of the operations to produce a clean character image, so that it is can be used directly and efficiently by the feature extraction stage. Before extracting features from an image, a sequence of simple, common preprocessing is applied in order to standardize the data and make it feasible to the recognition algorithms and to reduce complexity [5].

3.1 Binarization (Thresholding):

Binarization is one of the most important techniques for preprocessing stage. Among many binarization techniques, the Otsu's method [6] is considered as the most commonly-used one in the survey papers in [7,8,9,10] (cite in [11]). If a pixelis greater than or equal to the threshold intensity, the resulting pixel is white ("0"). On the other hand, if a pixel in the image has intensity less than the threshold value, the resulting pixel is black ("1").

Binarization of image consists of that either global or local threshold.Global thresholding [5,12,13,14,15] has a good performance in the case that there is a good separation betweentheforeground and the background (cite in [16]). Otsu's global threshold method [5] finds the global threshold t that minimizes the intra-class variance of the resulting black and white pixels. Then the binarization is formed by the setting (cite in [17])

$$b_i = 1 \text{ if } x_i \geq t \text{ and } b_i = 0 \text{ if } x_i < t \qquad (1)$$

Unlike global approaches, local area information may guide the threshold value for each pixel in local (adaptive) thresholding techniques . A local algorithm is introduced in [18] that calculates a pixel-wise threshold by shifting a rectangular window across the image. The threshold T forthe center pixel of the window is computed using the mean m and the variance s of the gray values in the window:

$$T = m + ks \qquad (2)$$

where k is a constant set to -0.2. The value of k is used to determine how much of the total print of object boundary is taken as a part of the given object.

3.2 Smoothing and Noise Removal:

References The process of character scanning introduces noises.Smoothing is widely used in procedure for eliminating the noises introduced during the image capture. Smoothing and noise removal can be done by filtering.

There are two types of filtering which are linear and non-linear. The solution of noise can be done by comparing gray-level with those of neighbours. A pixel's neighoborhood is a set of pixels, defined by their locations relative to that pixel. If gray-level [19] is substantially larger or smaller than those of all or nearly all of it is neighbours, the point can be classified as a noise, and the graylevel of this noise can be replaced by the weighted average of the gray-level of it is neighbours.

3.3 Normalization:

Character can have different sizes, positions and orientation. The goal for character normalization is [20] to reduce the within-class variation of the shapes of the characters/digits in order to facilitate feature extraction process and also improve their classification accuracy. The current handwritten character recognition systems generally attempt to cope with pattern variations and distortions by linear or nonlinear pattern normalization [21].

3.4 Thinning :

Thinning is an important preprocessing step in optical character recognition. The purpose of thinning is to delete redundant information and at the same time retain the characteristic features of the image. In order to reduce the quantity of information minimally, a thinning algorithm play an important role in recognition of the telugu characters, figure, and drawing [22,23].

Thinning is usually involves removing points or layers of outline from a pattern until all the lines or curves are of unit width, or a single pixel wide [24,25] . The resulting set of lines or curves is called the skeleton of the object as shown in Figure 1.

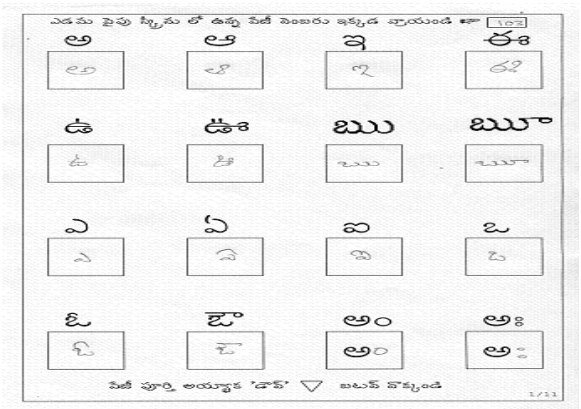

Figure 1: Skeleton produced by thinning process

More compact, you may use the solidus (/), the exp function, or appropriate exponents. Use parentheses to avoid ambiguities in denominators. Punctuate equations when they are part of a sentence,as in.

3.5 Segmentation:

OCR of cursive scripts presents a number of challenging problems in segmentation. The problem of the cursive handwriting is made complex by the fact that the writing is inherently ambiguous as the letter in a word are generally linked together, poorly written, and may even be missing. As a consequence, cursive script recognition requires sophisticated techniques, which uses a large amount of shape information and which compensates for the ambiguity by the use of contextual information [25].Segmentation is a crucial step of OCR systems as it extracts meaningful regions for analysis.

Segmentation is the process of detecting points and it is important for recognition because there is need to identify where a character starts and ends. Figure 2 shows the probability segmentation result for some of the segments in the word “eighteen”.

Figure 2: Segmentation result in the word ”eighteen”

3.6 Skew Detection:

Skew detection caused by document scanning and copying. Visually appears as a slope of the text lines with respect to the x-axis, and it mainly concerns the orientation of the text lines, see some examples of document skew in Figure 3 [26]. Many methods have been developed for the correction of skewed document images, basically they can be described in five categories:

- using projection profile.
- using Hough transform technique.
- Fourier method.
- by nearest-neighbor clustering; and
- correlation.

Figure 3: Symbol set for Telugu

3.7 Slant Correction:

Slant correction is an important factor in applications requiring word recognition, due to the fact that the slant corrected word is more easily segmented into characters (or subcharacters). It is possible to achieve more accurate word recognition, if slant correction techniques are based on local slant estimation, rather than the traditional average slant. Therefore, skew detection is one of the primary tasks to be solved in document image analysis systems, and it is necessary for aligning a document image before further processing .Figure 4 shows some samples of slanted handwritten numeral string.

If you wish, you may write in the first person singular or plural and use the active voice (“I observed that ...” or “We observed that ...” instead of “It was observed that ...”). Remember to check spelling. If your native language is not English, please get a native English-speaking colleague to proofread your paper.

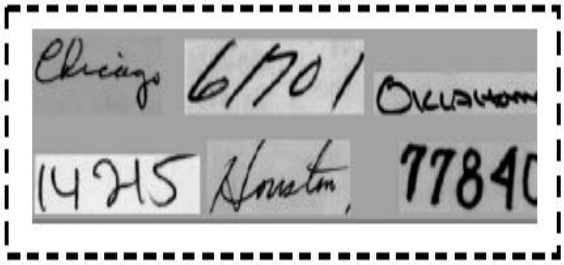

Figure 4: The examples of slanted handwritten numeral string.

IV. STROKE PRE-CLASSIFICATION

As mentioned above, a typical Telugu character can be divided vertically into three tiers:
top, middle and bottom. (Even four tiers are possible in extreme cases but such characters are rare.) All tiers may not be populated in a given character; the middle tier however is always present by default. Each character has a baseline stroke and usually one or more attached strokes at the top, bottom and side of the base stroke. Also there can be a lot of size variation in the strokes, with some of the strokes having very few constituent points for proper identification. Preclassification for Telugu seeks to classify the strokes in a character into four categories:

- main stroke,
- baseline auxiliary,
- top stroke and
- bottom stroke .

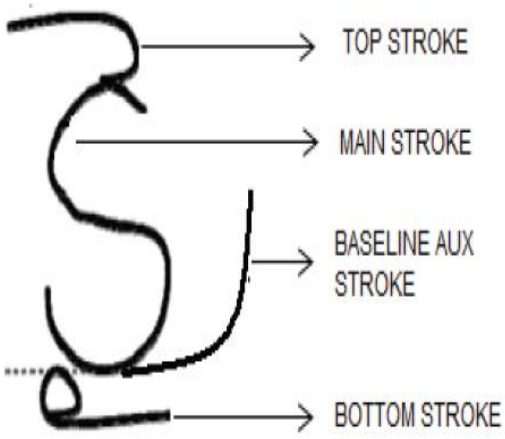

Figure 5. Tier-based Pre-Classification of a Telugu

Out of the total number of 235 strokes, there are 149 main strokes, 19 baseline strokes, 26 top strokes and 41 bottom strokes. The first stroke that is written in a character is assumed to be the main stroke, which is almost always true, but baseline auxiliary, bottom and top strokes can be written in any order. So to classify baseline auxiliary, bottom and top strokes, two methods are adopted.

Method 1:

The baseline auxiliary stroke is identified by a histogram of the y-co-ordinates of the stroke. The baseline stroke is the stroke closest to the interval with the least number of points. The top and bottom strokes are identified by considering their average vertical displacement from the horizontal line.

Method 2:

This method is based on a SVM. A feature vector is constructed by concatenating the main stroke and with the stroke that is being preclassified. Both strokes are resampled , so that either stroke has only 16 points. Such feature vectors are used to train a SVM-based pre-classifier.

V.FEATURE EXTRACTION

Once the strokes are pre-classified, feature vectors for stroke recognition are constructed. Different types of features are explored for best classification results. The following features are considered: 1) X and Y coordinate points, 2) Fourier transforms, (64 points for X, and 64 points for Y, yielding a 128 dimensional feature vector) 3) Hilbert transform logarithm of the spectral density, (64 points for X, and 64 points for Y, yielding a 128 dimensional feature vector) and 4) Wavelet features - the level is fixed to one, varied the families (and wavelet type (orthogonal and biorthogonal).Better results are obtained in Daubechies family and orthogonal wavelets, which is reported in this study (32 points for X, and 32 points for Y, yielding a 64 dimensional feature vector) .

5.1.Stroke Recognition using SVMs:

The features extracted from baseline auxiliary, bottom and top stroke preclasses are given to SVMs with Gaussian kernel to recognize the specific stroke. Since the main preclass is larger than the other 3, we handle this preclass differently.
This method uses 3 SVMs for classifying the main stroke. The feature vector extracted from the main stroke is passed to two different SVMs: Vowel classifier and Consonant classifier. Output vectors of the Vowel and Consonant SVM classifiers are concatenated (i.e. N classes of from Vowel classifier and M classes of Consonant classifier are combined to form N+M dimensional vector) and passed to third SVM for classifying the main stroke (fig. 6).

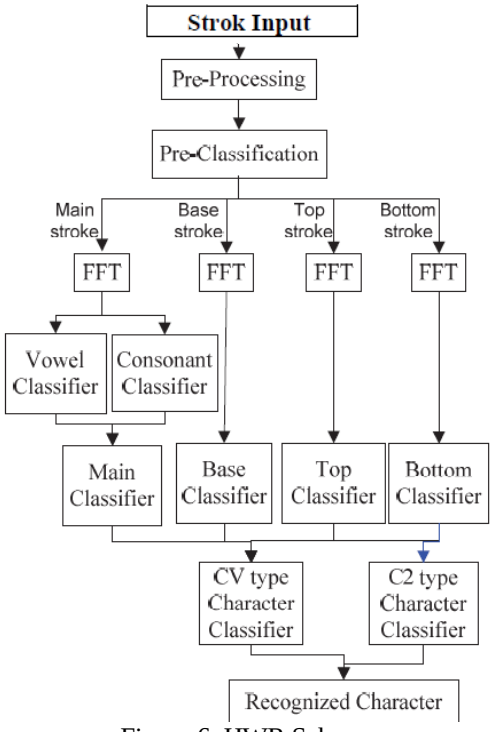

Figure 6. HWR Schema

5.2.Character Recognition:
Based on SVMs:

Information pertaining to C2 is always found in bottom strokes in Telugu characters. Two SVMs - a CV type and a C2 type classifier – are used to recognize the character. Main stroke, base stroke, top stroke and bottom stroke neglecting C2 type strokes (there are 215 of them) are taken as a feature vector to recognize consonant and vowel character.

This feature vector is a binary vector of dimension 215: a component is 1 (0) if the corresponding stroke is present (absent). In cases where there are multiple copies of the stroke, the component is set to the number of copies. If a C2 type bottom stroke (there are 18 of them) is present, then the corresponding C2 type bottom stroke-based feature vector is constructed and passed to the C2-type character classifier.

VI.CONCLUSION

We present a schemas for online recognition of Telugu characters. while this schema uses an SVM. In this schema , pre-classification is done using a SVM. The main stroke recognition in schema uses a more elaborate method and achieves higher performance. The proposed schemas are compared to similar systems developed for online Telugu recognition in the past. The system proposed by Jayaraman et al (2007), which is also based on tier-based preclassification of Telugu strokes, reports an overall stroke recognition accuracy of 86.9%.

Schema , with its more elaborate processing of main stroke, and use of FFT features, gives an overall stroke recognition accuracy of 96.69%. Our approach compares favorably with an even older SVM-based system for online Telugu stroke recognition which reports a best performance of 82.96% at stroke level (Swethalakshmi et al 2006). character-level performance is only slightly higher than stroke-level performance for two reasons: 1) rules corresponding

REFERENCES:

[1] G.E.M.D.C. Bandara, S.D. Pathirana, R.M. Ranawana,"Use of fuzzy feature descriptions to recognize handwritten alphanumeric characters," in 1st Conference on Fuzzy Systems and Knowledge Discovery, Singapore, 2002.

[2] Paul D. Gader, James M. Keller, Raghu Krishnapuram,Jung-Hsien Chiang, Magdi A. Mohamed, "Neural and fuzzy methods in handwriting recognition," Computer, vol.30, no. 2, pp. 79-86, Feb., 1997.

[3] Batuwita, K.B.M.R.; Bandara, G.E.M.D.C.; Fuzzy Recognition of Off-line Handwritten Numeric Characters Cybernetics and Intelligent Systems, 2006 IEEEConference on 7-9 June 2006 Page(s): 1 – 5.

[4] Arica, Nafiz; Yarman-Vural Fatos T; An Overview of Character Recognition Focused on Off-Line Handwriting;IEEE Transactions on Systems, Man, and Cybernetics- PartC: Applications and Reviews, Vol. 31, No. 2, may 2001.

[5] Ba-karait, N.O.S. Particle Swarm Optimization (PSO)-Based Clustering Algorithm for Handwritten Digits Recognition. MSc Thesis. Universiti Teknologi Malaysia, 2007.

[6] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Systems Man Cybernet. 9 (1) (1979) 62–66.

[7] B. Sankur and M. Sezgin, "A survey over image thresholding techniques and quantitative performance evaluation," Journal of Electronic Imaging, 2004.

[8] O.D. Trier, A.K. Jain, Goal-directed evaluation of binarization methods, IEEE Trans. Pattern Anal. Mach.Intell. 17 (12) (1995) 1191–1201.

[9] N. R. Pal and S. A Pal, "Review on image segmentation techniques," Pattern Recognition, Vol. 26, pp. 1277–1294,1993.

[10] P.K. Sahoo, S. Soltani, A.K.C. Wong, and Y.C. Chen, "A survey of thresholding techniques," Comput. Vis. Graph. Image Process., Vol. 41, pp. 233–260, 1988.

[11] Liju Dong; Ge Yu; An optimization-based approach to image binarization. Computer and Information Technology, 2004. CIT '04. The Fourth International Conference on 14- 16 Sept. 2004 Page(s):165 – 170.

[12] A. Rosenfeld, A.C. Kak, Digital Picture Processing, second ed.,Academic Press, New York, 1982.

[13] J. Kittler, J. Illingworth, On threshold selection using clustering criteria, IEEE Trans. Systems Man Cybernet. 15 (1985) 652–655.

[14] A.D. Brink, Thresholding of digital images using twodimensional entropies, Pattern Recognition 25 (8) (1992) 803–808.

[15] H. Yan, Unified formulation of a class of image thresholding techniques, Pattern Recognition 29 (12) (1996) 2025–2032.

[16] Gatos, B., Pratikakis, I. and Perantonis, S.J. (2005).Adaptive Degraded Document Image Binarization. Pattern Recognition. 39: 317 – 327.

[17] Gupta, M. R., Jacobson, N. P. and Garcia, E. K.(2006).OCR Binarization and Image Preprocessing for Searching Historical Documents. Pattern Recognition.40:389 – 397.

[18] J.R. Parker, C. Jennings, A.G. Salkauskas, Thresholding using an illumination model, ICDAR'93, 1993, pp. 270–273.

[19] Cheriet, M., Kharma, N., Liu, C-L., Suen. CY. (2007) Character Recognition Systems : a Guide for Student and Practitioners. John Wiley & Sons, Inc. United States of America.

[20] H.Yamada, K.Yamamoto, T.Saito: "A Non-linear Normalization Method for Handprinted Kanji Character Recognition – Line Density Equalization," PatternRecognition, vol.23, no.9, pp.1023-1029, (1990).

[21] Davies, E.R., and Plummer, A.P.N. Thinning algorithm, a critique and a new methodology. Pattern Recognition 14, 1 (1981), 53-63.

[22] Pfaltz, J.L., and Rosenfeld, A. Computer representation of planar regions by their skeletons. Commun. ACM 10, 2 (February 1967). 119-125.

[23] Paul Kwok. A thinning algorithm by contour generation.Source Communications of the ACM archive Volume 31,Issue 11 (November 1988). Pages: 1314 – 1324.

[24] N. Arica; Yarman-Vural, F.T.; Optical character recognition for cursive handwriting Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume 24,Issue 6, June 2002 Page(s):801 – 813.

[25] Changming Sun; Deyi Si; Skew and slant correction for document images using gradient direction. Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on Volume 1, 18-20 Aug. 1997 Page(s):142 - 146 vol.1.

[26] Yimei Ding; Ohyama, W.; Kimura, F.; Shridhar, M.; Local slant estimation for handwritten English words Frontiers in Handwriting Recognition, 2004. IWFHR-9 2004. Ninth International Workshop on 26-29 Oct. 2004 Page(s):328 -333.