International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-2 Number-2 Issue-4 June-2012

A Novel Technique to Read Small and Capital Handwritten Character

Ekta Tiwari[1], Maneesh Shreevastava[2]
M-Tech (IT), LNCT, Bhopal India[1], Prof (IT), LNCT Bhopal, India[2]

Abstract

A system has been developed for text writing systems using Support Vector Machines (SVM) is called Handwritten Character Recognition (HCR). The main challenge in handwritten character recognition for Small and Capital letter is to build a system that is able to distinguish between variation in writing the same stroke (when the same stroke is written by different writers or the same writer at different times) and minor variation in similar characters in the script. Other issues faced can be attributed to the large number of character and stroke classes. Some Indian scripts have character modifiers occurring in multiple non-overlapping horizontal units which are positioned on one or both sides of the main consonant. In such cases, we may also need to keep track of the sequence of horizontal units as they are written. The main problem in handwritten character recognition is recognition for Small and Capital letter is to build a system that is able to distinguish between variation in writing the same stroke and minor variation in similar characters in the script. Handwritten character recognition is not a new technology but it not gained public attention. The various features that are considered for classification are the character height, character width, the number of horizontal lines (long and short, image centroid and special dots. In this research paper extracted features were passed to a Support Vector Machine (SVM) where the characters are classified by Supervised Learning Algorithm. These classes are mapped onto for recognition. Then the text is reconstructed using fonts.

Keywords

OCR, Features, Support Vector Machine (SVM), Artificial Neural Networks, Handwritten Character Recognition, Stroke, Printed Characters.

1. Introduction

A number of methodologies have been proposed over the years for character recognition. Several techniques like OCR using correlation method and OCR using neural networks is reviewed in this research work. Optical Character Recognition is classified into two types, Offline recognition and online recognition. In offline recognition the source is either an image or a scanned form of the document whereas in online recognition the successive points are represented as a function of time and the order of strokes are also available. Documents are scanned using a scanner and are given to the OCR systems which recognizes the characters in the scanned documents and converts them into ASCII data. OCR has three processing steps, Document scanning process, Recognition process and Verifying process. In the document scanning step, a scanner is used to scan the handwritten or printed documents.

Handwritten character recognition for any Indian writing system is rendered complex [1] because of the presence of composite characters. it is a writing system where the vowels are written as diacritics on the consonants and a vowel is not explicitly written when it appears immediately after a consonant in a word. This combination of diacritics with consonants is called a composite character. A consonant can combine not only with each of the vowels of the writing system but also with other consonants to form ligatures.

A few models that have been applied for the HCR system include motor models, structure-based models, stochastic models and learning-based models. Learning-based models have received wide attention for pattern recognition problems. Neural network models have been reported to achieve better performance than other existing models in many recognition tasks. Support vector machines [2, 3] have also been observed to achieve reasonable generalization accuracy, especially in implementations of handwritten digit recognition and character recognition in Roman, Thai and Arabic scripts. The present work is on the development of systems for online HCR of Text using SVMs.

2. Literature Survey

In reviewing the literature, we find various algorithms for handwritten digit segmentation. A

direct comparison, though, is not a trivial task. On the contrary, it may not even be feasible.

In this research work [4], the algorithms for segmenting handwritten digits based on different concepts are compared by evaluating them under the same conditions of implementation. A robust experimental protocol based on a large synthetic database is used to assess each algorithm in terms of correct segmentation and computational time. Results on a real database are also presented. In addition to the overall performance of each algorithm, we show the performance for different types of connections, which provides an interesting categorization of each algorithm. Another contribution of this work concerns the complementarily of the algorithms.

We have observed that each method is able to segment samples that cannot be segmented by any other method, and do so independently of their individual performance. Based on this observation, we conclude that combining different segmentation algorithms may be an appropriate strategy for improving the correct segmentation rate.

Here they selected those algorithms that used to produce the segmentation cuts. Then, these algorithms were assessed for performance. The proposed evaluation criteria provided the global performance of each algorithm, as well as their performance on four different types of connections. The experimental results show that these algorithms achieve similar performances on both databases, which qualifies the synthetic dataset as a viable alternative for benchmarking segmentation algorithms.

During the evaluation, they observed that, independently of the overall performance, each method is able to segment some samples that cannot be segmented by any other method. It corroborates the argument that even a method with low overall performance can contribute to building a more reliable segmentation system.

As they have demonstrated, this kind of analysis also constitutes useful contribution to identifying complementarily among the segmentation algorithms, which can be used to develop more intelligent systems. The main challenge in building such an intelligent system lies in the correct identification of the connection types, which certainly is not a trivial task. In some cases, especially in real-time applications, this is a very important issue that can determine the success or failure of a handwriting recognition system.

3. Proposed Technique

Training and Recognition Techniques CASCR systems extensively use the methodologies of pattern recognition, which assigns an unknown sample into a predefined class. Numerous techniques for CASCR can be investigated in four general approaches of Pattern Recognition, as suggested in [5] the above approaches are neither necessarily independent nor disjoint from each other. Occasionally, a CASCR technique in one approach can also be considered to be a member of other approaches.

In all of the above approaches, CASCR techniques use either holistic or analytic strategies for the training and recognition stages: Holistic strategy employs top down approaches for recognizing the full word, eliminating the segmentation problem. The price for this computational saving is to constrain the problem of CASCR to limited vocabulary. Also, due to the complexity introduced by the representation of whole cursive word (compared to the complexity of a single character or stroke), the recognition accuracy is decreased. On the other hand, the analytic strategies employ bottom up approaches starting from stroke or character level and going towards producing a meaningful text. Explicit or implicit segmentation algorithms are required for this strategy, not only adding extra complexity to the problem, but also, introducing segmentation error to the system. However, with the cooperation of segmentation stage, the problem is reduced to the recognition of simple isolated characters or strokes, which can be handled for unlimited vocabulary with high recognition rates.

Template Matching

CASCR techniques vary widely according to the feature set selected from the long list of features, described in the previous section for image representation. Features can be as simple as the gray-level image frames with individual characters or words or as complicated as graph representation of character primitives. The simplest way of character recognition is based on matching the stored prototypes against the character or word to be recognized. Generally speaking, matching operation determines the degree of similarity between two vectors (group of pixels, shapes, curvature etc.) in the feature space. Matching techniques can be studied in three classes:

Direct Matching

A gray-level or binary input character is directly compared to a standard set of stored prototypes. According to a similarity measure a prototype matching is done for recognition. The matching techniques can be as simple as one-to-one comparison or as complex as decision tree analysis in which only selected pixels are tested. A template matcher can combine multiple information sources, including match strength and k-nearest neighbor measurements from different metrics [6], [7]. Although direct matching method is intuitive and has a solid mathematical background, the recognition rate of this method is very sensitive to noise.

Fig. 1: Gray Images

Fig. 2: binary input character

Deformable Templates and Elastic Matching

An alternative method is the use of deformable templates, where an image deformation is used to match an unknown image against a database of known images. In [8], two characters are matched by deforming the contour of one, to fit the edge strengths of the other. A dissimilarity measure is derived from the amount of deformation needed, the goodness of fit of the edges and the interior overlap between the deformed shapes (see fig 1).

The basic idea of elastic matching is to optimally match the unknown symbol against all possible elastic stretching and compression of each prototype. Once the feature space is formed, the unknown vector is matched using dynamic programming and a warping function [9], [10]. Since the curves obtained from the skeletonization of the characters could be distorted, elastic matching methods cannot deal with topological correlation between two patterns in the off-line CASCR. In order to avoid this difficulty, a self-organization matching approach is proposed in [11] for hand-printed character recognition, using thick strokes. Elastic matching is also popular in on-line recognition systems [12].

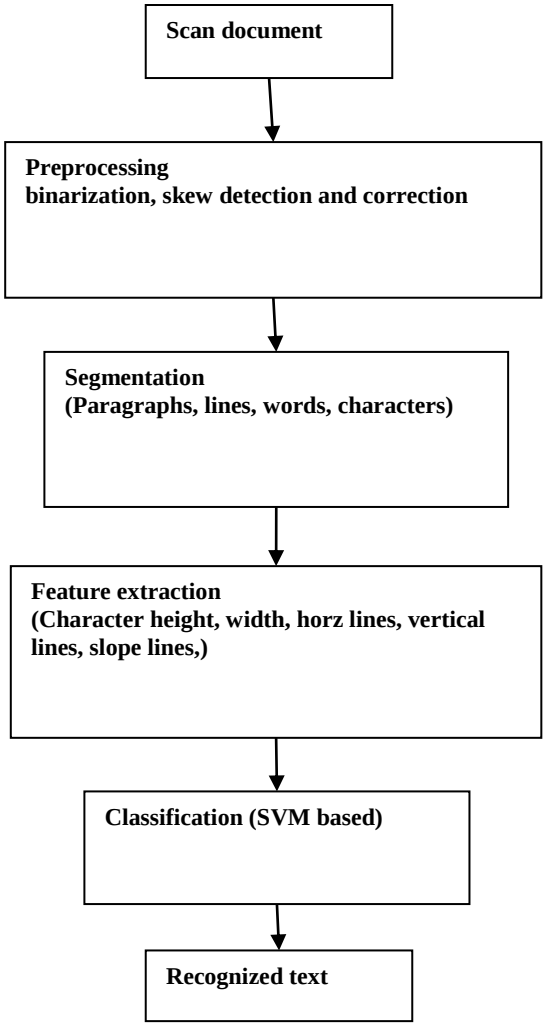

Fig 3: Block diagram of CASCR

Relaxation Matching

It is a symbolic level image matching technique that uses feature-based description for the character image. First, the matching regions are identified. Then, based on some well-defined ratings of the assignments, the image elements are compared to the model. This procedure requires a search technique in a multi-dimensional space, for finding the global maximum of some functions [13], [14]. Huang et al. proposed a multi font Chinese character recognition system in [15], where sampling points, including cross, branch and end points on the skeleton are taken as nodes of a graph. Each character class is represented by a constrained graph model, which captures the geometrical and topological invariance for the same class. Recognition is then made by a relaxation matching algorithm. In [16], Xie et al

proposed a handwritten Chinese character system, where small number of critical structural features, such as end points, hooks, T-shape, cross and corner are used. Recognition is done by computing the matching probabilities between two features by a relaxation method. The matching techniques mentioned above are sometimes used individually or combined in many ways as part of the CASCR schemes. The block diagram of CASCR consists of various states as shown in Fig.3. They are scanning phase, preprocessing, segmentation, feature extraction, classification (SVM, rule based, and ANN), and recognition and output verification

CASCR Functions
This phase includes the scanning state, preprocessing block, and segmentation and feature extraction.

Scanning the document
A properly printed document is chosen for scanning. It is placed over the scanner. A scanner software is invoked which scans the document. The document is sent to a program that saves it in preferably TIF, JPG or GIF format, so that the image of the document can be obtained when needed.

Preprocessing
This is the first step in the processing of scanned image. The scanned image is checked for skewing. There are possibilities of image getting skewed with either left or right orientation. The function for skew detection checks for an angle of orientation between ±15 degrees and if detected then a simple image rotation is carried out till the lines match with the true horizontal axis, Skew correction is done by rotating the image around an angle θ (−2.0).

Segmentation
After pre-processing, the noise free image is passed to the segmentation phase, where the image is decomposed into individual characters. the image and various steps in segmentation. Algorithm for segmentation:

Image is checked for inter line spaces.
If inter line spaces are detected then the image is segmented into sets of paragraphs across the interline gap. The lines in the paragraphs are scanned for horizontal space intersection with respect to the background.

Histogram of the image is used to detect the width of the horizontal lines. Then the lines are scanned vertically for vertical space intersection. Here histograms are used to detect the width of the words. Then the words are decomposed into characters using character width computation.

Classification
Classification is done using the features extracted in the previous step, which corresponds to each character glyph. These features are analyzed using the set of rules and labeled as belonging to different classes. This classification is generalized such that it works for all the fonts' types. The implicit regularization of the classifier's complexity avoids over fitting and mostly this leads to good generalizations. Some more properties are commonly seen as reasons for the success of SVMs in real-world problems. The optimality of the training result is guaranteed, fast training algorithms exist and little a-priori knowledge is required, i.e. only a labeled training set.

Classification using SVM
Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates a set of objects having different class memberships. A typical example is shown in Fig.2 where it is used to classify different types of character glyphs belonging to different fonts.

Single Recognition Engine Approach
Each stroke is represented using a 120-dimensional feature vector, where co-ordinates of the 60 points obtained after preprocessing are chosen as features. Gaussian kernel with a value of 30 for standard deviation has been found to give the best performance. A dot that occurs anywhere within the character is pre-classified based on the size of its bounding box and its relative position within the stroke. The results for the experiments carried out using the Gaussian kernel SVMs.

Feature extraction
This follows the segmentation phase of OCR where the individual image glyph is considered and extracted for features.
First a character glyph is defined by the following attributes:
Height of the character
Width of the character
Number of horizontal lines present short and long
Numbers of vertical lines present short and long
Numbers of circles present
Numbers of horizontally oriented arcs
Numbers of vertically oriented arcs

Centroid of the image
Position of the various features
Pixels in the various regions

[1 1 1 1 1 1 1 1]
[1 1 1 1 1 1 1 1]
[0 0 0 1 1 0 0 0]
[0 0 0 1 1 0 0 0]
[0 0 0 1 1 0 0 0]
[0 0 0 1 1 0 0 0]
[0 0 0 1 1 0 0 0]
[0 0 0 1 1 0 0 0]

For example:-
This matrix for T representation only

The various feature extraction algorithms are as follows:
Detection of character height and character width: This is detected by simply scanning the image glyph and finding the boundary of the glyph in the horizontal and vertical directions. Height=24 and Width=24

Horizontal line detection Here a mask is run over the entire image glyph and threshold which detects the horizontal line Vertical line detection Here a mask is applied over the entire range of pixels and threshold which detects the vertical line.

Slope lines detection the masks for the slope lines to detect slope lines these masks are applied over the image and then threshold suitably. Detection of circles and arcs here a new mask is derived and operated on each and every pixel in the image glyph and threshold which detects the circle. A 24×24 mask is taken. The arcs are detected using the circle detection algorithm and checked for the semi-circle and diameter the second phase of the OCR functions consists of classification and Unicode mapping and recognition strategies.

4. Test Results and Analysis

The CASCR is implemented in Matlab. Various experimental results are discussed below. Fig.7 of the segmentation phase output displays on the Image the original image and on the right side the segmented image. Fig.4 represents the feature extraction the feature extraction and rule based classification of consonants represent the typical scenarios in the recognition of text using the Matlab environment. Recognition stages and various charts are depicted for clarity. A comparative study on various classifiers was also conducted.
CASCR refers to the process of converting printed text documents into software translated Text. The printed documents available in the form of books, papers, magazines, etc. are scanned using standard scanners which produce an image of the scanned document. The preprocessed image is segmented using an algorithm which decomposes the scanned text into paragraphs using special space detection technique and then the paragraphs into lines using vertical histograms, and lines into words using horizontal histograms, and words into character image glyphs using horizontal histograms. Each image glyph is comprised of 24×42 pixels, 50×50 pixels, and 42×42 pixels,. Thus a database of character image glyphs is created out of the segmentation phase. Then all the image glyphs are considered for recognition using mapping. Each image glyph is passed through various routines which extract the features of the glyph.

The various features that are considered for classification are the character height, character width, the number of horizontal lines (long and short), the number of vertical lines (long and short), the horizontally oriented curves, the vertically oriented curves, the number of circles, number of slope lines, image centroid and special dots. The glyphs are now set ready for classification based on these features.

Fig 4: Input Image as jpg format Include capital and small letter

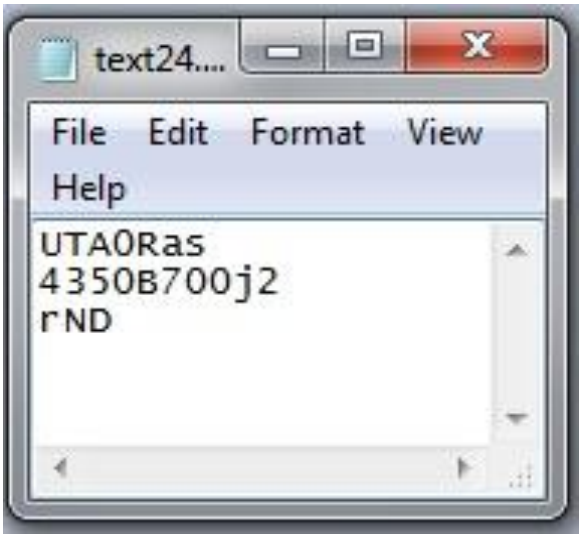

Fig 5: Result including input image and recognize latter in notepad

Table 1: SVM classify results

S. no	Number of fonts	Accuracy Including Small latter	Accuracy without Small latter
1	1	82%	97%
2	2	77%	90%
3	3	71%	87%
4	4	64%	80%
5	5	58%	76%

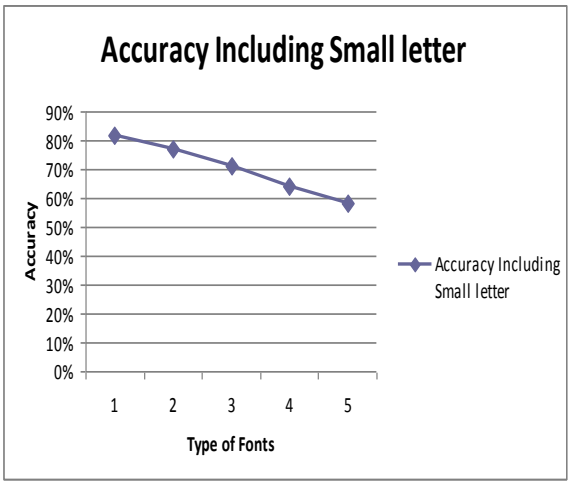

Fig 6: Chart for classification results

CASCR can also play a major role in the business environment. CASCR reduces cost and effort by eliminating manual data entry, etc. If CASCR is available, it becomes easier to extract and transform the data into business BASE and promote business without the need for large mobility (data, people). The increasing numbers of faxes and paper documents received by businesses often originate from the same suppliers or customers and have a format and layout that have not changed for some time. The data within these documents have to be manually interpreted and re-keyed into business applications as part of key business processes (e.g. Purchase Orders and Invoices into Accounting Systems for Accounts Receivables and Payables, students data etc.). The larger the volume of documents received, the greater the manual resource required entering the data into business applications. The scope for errors and delay to critical business processes also increases as volume increases, if it is handled manually. By scanning the documents to create TIFF image files and automatically routing electronic fax images to CASCR, the errors, cost and delay of manual data entry can be avoided, as CASCR can automatically extract data from the documents and format the data for onward delivery to other applications. Thus this research work discusses the various strategies and techniques involved in the recognition of different languages text.

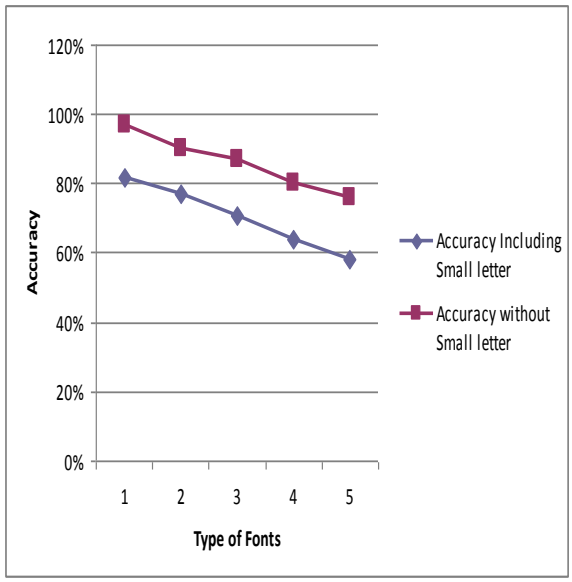

Fig 7: Chart for classification results

Table 2: Performance of the segmentation algorithms on the touching pairs extracted from NIST SD19

Algorithm	Correct segmentation (%)
Fujisawa et al. [27]	88.9
Shi and Govindaraju [28]	62.3
Fenrich and Krishnamoorthy [29]	96.9
Chen and Wang [30]	96.8
Pal et al. [31]	82.3
Elnagar and Alhajajj [32]	72.3
CASCR	97.0

The results of the correct segmentation are reported in Table 2. As we can see, the performance achieved in this dataset is similar to the performance reported in [11]. The slightly better performance achieved on the real dataset by most algorithms can be justified by the fact that it contains more data base, which are easier to segment. But this result show only single type of data but results of both small and capital letter data set. This research algorithm is best for both type of letter small and capital.

5. Conclusion

The performance of character recognition is dependent on the accurate recognition. The results obtained for recognition of characters show that reliable classification is possible using SVMs. The results also indicate the scope for further improvement. Future work is directed towards incorporating a database of words for spell-check at word level. The SVM-based methods described here for handwritten character recognition can be easily extended to other Indian scripts.

CASCR is aimed at recognizing printed document. The input document is read preprocessed, feature extracted and recognized and the recognized text is displayed in a picture box. Thus the CASCR is implemented using a Matlab. A complete tool bar is also provided for training, recognizing and editing options. CASCR eliminates the difficulty by making the data available in printed format. In a way CASCR provides a paperless. It can be accessed by people of varying category with ease and comfort. Still there are scholars who are available then processing and maintaining the students' records become easier. The students' forms can be directly. Scanned, extracted for details and directly transformed into a Student Database. Future work is directed towards incorporating a database of words for spell-check at word level. The SVM-based methods described here for Devanagari and Telugu handwritten character recognition can be easily extended to other Indian scripts.

References

[1] R.M.Suresh, "Fuzzy Technique Based Recognition of Handwritten Characters", LNAI 2955, 297-306, 2006.

[2] H. Bentounsi, M. Batouche, .Incremental support vectormachines for handwritten Arabic character recognition., Proceedings of the International Conference on Information and Communication Technologies, 2004, pp 1764-1767.

[3] Z. Bin, L. Yong, X. Shao-Wei, .Support vector machine and its application in handwritten numeral recognition., Proceedings of the 15th International Conference on Pattern Recognition, 2000, pp 720-723.

[4] F. C. Ribas • L. S. Oliveira • A. S. Britto Jr. "Handwritten digit segmentation: a comparative study", springer, mar 2012.

[5] A. K. Jain, R.P.W. Duin, J. Mao "Statistical Pattern Recognition: A Review ", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.22, no. 1, pp. 4-38, 2000.

[6] D. Tubbs, "A Note on Binary Template Matching", Pattern Recognition, vol.22, no.4, pp.359 - 365, 1989.

[7] P. D. Gader, B. Forester, M. Ganzberger, A. Gillies, B. Mitchell, M. Whalen, and T. Yocum, "Recognition of Handwritten Digits Using Template and Model Matching", Pattern Recognition, vol.24, no.5, pp.421-431, 1991.

[8] A. K. Jain, D. Zongker, "Representation and Recognition of Handwritten Digits Using Deformable Templates", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.19, no.12, pp.1386- 1391, 1997.

[9] J. Hu, T. Pavlidis, "A Hierarchical Approach to Efficient Curvilinear Object Searching", Computer Vision and Image Understanding, vol.63 (2), pp. 208-220, 1996 .

[10] C. C. Tappert, "Cursive Script Recognition by Elastic Matching", IBM Jour nal of Research and Development, vol.26, no.6, pp.765-771, 1982.

[11] C. Y. Liou, H.C. Yang, Handprinted Character Recognition Based on Spatial Topology Distance Measurements, ", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.18, no.9, pp941-945, 1996.

[12] M. Nakagawa, K. Akiyama, "A Linear-time Elastic Matching For Stroke Number Free Recognition of On-line Handwritten Characters", in Proc. Int. Workshop Frontiers in Handwriting Recognition, pp. 48-56, Taiwan, 1994.

[13] Keith E. Price, "Relaxation Matching Techniques Comparison", IEEE Trans. Pattern Analysis and Machine Intelligence, vol.7, no.5, pp. 617 - 623, 1985.

I am from Bhopal. My dob is 12/10/1986.I did my BE in IT from SATI Vidisha in 2007. Currently I am working as Lecturer In Dy Patil Institute of MCA Pune(MH). I published 1 paper in IJCABI vol: 02 no. 4 ISSN 0975-945X.