MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

WORD BASED RECOGNITION OF CURSIVE SCRIPT

P.V.S. Rao

Tata Institute of Fundamental Research
Homi Bhabha Road Bombay 400005, India

ABSTRACT

We describe a system for on line or off line recognition of cursive script.

Our earlier work established that cursive script can be synthesised out of individual characters by using polynomial merging functions which satisfy boundary conditions of continuity of the displacement functions x(t) and y(t) for each character and their first and second derivatives. We showed that even individual characters could be synthesised out of more primitive elements by using the same merging functions. The elements we choose are straight lines: not the usual line segments but a much smaller number of directed lines which we call shape vectors, ranging from only three vectors for simple characters such as e, l and o and a maximum of seven for m.

We use slopes of the shape vectors and relative locations of points of maximum curvature (both highly quantised) as parameters for recognition. The system extracts parameters for individual characters from single specimens written in isolation and uses these to construct feature matrices for words in the vocabulary. During recognition, these are matched with the feature matrices of test words.

The system achieves recognition scores of 94% for vocabulary sizes in the range of 100 words.

INTRODUCTION

Computer based cursive script recognition has numerous applications in the industrial and service sectors. More over, the problem is intrinsically challenging. It is therefore not surprising that it has engaged the attention of researchers over the last several decades.

Among the more commonly used approaches for pattern recognition by computers are data abstraction, decomposition of complex patterns into simpler shapes and model based parameter extraction. The current paper presents a novel synthesis-based approach and uses segmentation and data abstraction to achieve good recognition. It is based on and is an extension of the author's recent work on cursive script synthesis from individual characters and even more primitive elements. In this sense, it is a synthesis-based approach.

EARLIER WORK

Earlier work by the author has established:

(a) that cursive script can be synthesised out of individual characters (see Fig.1) by using polynomial merging functions which satisfy boundary conditions of continuity of the displacement functions x(t) and y(t) for each character and their first and second derivatives[1]; and

(b) that the procedure lends itself to a Bezier curve
based formulation[2].

This approach was evolved keeping in view the fact that cursive writing avoids discontinuities (of shape) between individual characters as well as discontinuities in pen movement (stop, pen lift, move, pen down and start). It thus appears that the aim in connected writing is to achieve smooth tracing of the character sequences with minimal effort.

We have shown earlier [3] that even individual characters can be synthesised out of more primitive elements by using the same merging functions. The elements we choose for this purpose are straight lines: not the usual line segments but a much smaller number of directed lines which we call shape vectors (see Fig.2).

In this, we take advantage of the fact that the script characters in general have shapes which can be visualised as being composed of comparatively straight segments alternating with 'bends' or regions of relatively high curvature. For a character with n bends, we need only n+1 shape vectors. Thus, each script character needs only three to seven shape vectors, depending on the complexity of its shape; we need only three vectors for simple characters such as e, l and o and a maximum of seven for the most complex character, m. This synthesis is so good that the synthesised version is practically indistinguishable from the original, even in case of complex characters (See Fig.3).

General Approach: The 'character generator' shape vectors are derived from the original character by means of a simple procedure that identifies regions in the character which are comparatively straight. These are then approximated by straight lines by linear regression and positioned to be tangential to the original curve. The synthesised version of this character, obtained by 'merging' or concatenating these vectors, fits the original character so well that, when super imposed, the two are indistinguishable. Data reduction ratios in the range of 15 to 25 are possible in this system. This demonstrates that the shape vectors adequately characterise the identity and shape of the character. It is therefore obvious that they should provide a basis for script character recognition. In fact, a recognition accuracy of 94% has been achieved for a vocabulary size of 67 words.

In practice, we use slopes of the shape vectors and relative locations of points of maximum curvature (both highly quantised) as parameters for recognition. The system extracts parameters for individual characters from single specimens written in isolation and uses these to construct feature matrices for words in the vocabulary. These are used for matching with the feature matrices of test words during the recognition phase.

Segmentation is done by identifying points of maximum and minimum curvature or instantaneous pen velocity (called maxima and minima respectively). The slant angle of writing is established by using one of several simple methods:

(a) Plot a histogram of the instantaneous angle of slope at various points on the curve and determine the slope angle corresponding to the maximum point on the histogram.

(b) Join the top most and bottom most maxima in the character and measure the angle between them.

(c) Find the average of the slope angles subtended by the straight line approximations of the comparatively straight portions of the character.

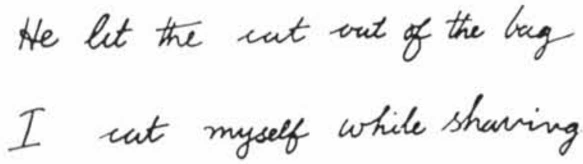
Methods (a) and (b) have been tried out and are equally satisfactory. Long letters such as f,g and l are used to determine slant, for obvious reasons. The slant is then eliminated by a coordinate transformation which makes the slant angle 90°.

Recognition Procedure: Parameters (see below for the specific parameters used in each case) are extracted from **single** samples of individual letters; these are the templates used for letter recognition. For recognition at the word level, a dictionary of word templates is generated from the same data for individual characters.

The parameter string extracted from the test word is compared with stored templates for the entire vocabulary and the word giving the best match is chosen. The current scheme incorporates a straight comparison; this extracts a heavy penalty for spurious (or missed) maxima and minima. More elaborate dictionary match methods will further improve the performance and upgrade the system for use with significantly larger vocabularies.

EXPERIMENTAL RESULTS

Experiment 1: The feasibility of the over all approach was tested in a preliminary experiment which used a minimal set of parameters: the sign of the curvature at the the maxima and the direction of movement at the minima (up, down, right and left). Even with such drastic data abstraction, it was possible to do character and word recognition. Understandably, similar letters such as e and l, u and a etc. get grouped together. Such grouping would in fact be advantageous in systems which use higher level information; e.g. see 'cut' and 'cat' in the following sentence pair.



Experiment 2: In a second experiment, the parameters used (see Fig.4) were:

(a) for the maxima: location (normalised and quantised) and direction of movement (clockwise or counter clockwise).

(b) for the minima: direction of movement (quantised: up, down, left and right).

Results are summarised in table 1.

Experiment 3: Preliminary experiments with neural networks (multilevel perceptron) using the same set of parameters[4] yield scores ranging between 99% (when the test and training sets are the same) and 80% (when other test sets - even from other subjects - are used).

CONCLUSIONS AND DISCUSSION

A major advantage of this approach is that it does not require training. It is very suitable for vocabularies in the range of up to a few hundreds of words.

The robustness of the approach is amply demonstrated by the fact that performance does not degrade even when the training and test sets are from different subjects.

Table 1 Recognition Scores for Experiment 2

	Run 1	Run 2
Vocabulary Size	24	67
Correct Words recognised by Exact Match	20	39
Correct Words recognised by Nearest Match	4	24
Incorrect	nil	4
Recognition Score	100%	94%

Fig.1. Connected script word generated by the character concatenation procedure.

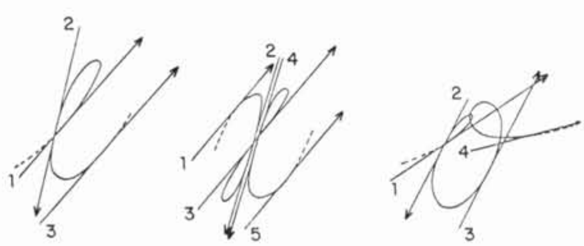


Fig.2. The synthesis of a character from its shape vectors.

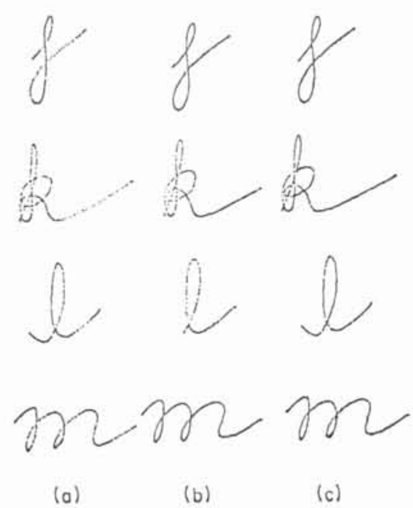


Fig.3. Original and resynthesised characters compared.
(a) original character.
(b) resynthesised character.
(c) (b) superimposed on (a).

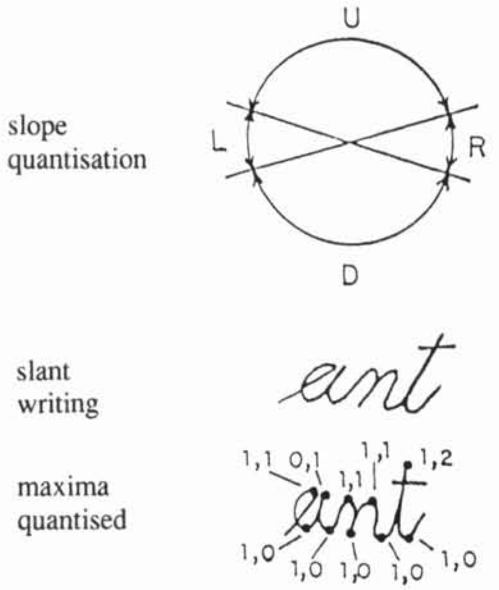


Fig.4. Parameters used for cursive script recognition.

REFERENCES

1. V. Ramasubramanian and P.V.S. Rao: Connected Script Synthesis by Character Concatenation - An overlap and Weighted Average Formulation; Computer Science and Informatics, Vol. 9, No. 1, pp 1-10, 1989
2. P.V.S. Rao and V. Ramasubramanian: Connected Script Synthesis by Character Concatenation - A Bezier Curve Formulation. J. Inst. ETE (Invited Paper - to appear)
3. P.V.S.Rao: Shape Vectors: An Efficient Parameteric Representation for the Synthesis and Recognition of Hand Script Characters. CSC Group Technical Report, TIFR, 1990.
4. P.V.S.Rao: Cursive Script Recognition Using Neural Nets, ICARCV '90, Singapore.