STRATEGIES IN CHARACTER SEGMENTATION: A SURVEY

Richard G. Casey and Eric Lecolinet

IBM Almaden Research Center and ENST Paris

Abstract

This paper provides a review of advances in character segmentation. Segmentation methods are listed under four main headings. The operation of attempting to decompose the image into classifiable units on the basis of general image features is called "dissection". The second class of methods avoids dissection, and segments the image either explicitly, by classification of specified windows, or implicitly by classification of subsets of spatial features collected from the image as a whole. The third strategy is a hybrid of the first two, employing dissection together with recombination rules to define potential segments, but using classification to select from the range of admissible segmentation possibilities offered by these subimages. Finally, holistic approaches that avoid segmentation by recognizing entire character strings as units are described.

1: Introduction

1.1: The role of segmentation in recognition processing

Character segmentation is an operation that seeks to decompose an image of a sequence of characters into subimages of individual symbols. It is one of the decision processes in a system for optical character recognition (OCR). Its decision, that a pattern isolated from the image is that of a character (or some other identifiable unit), can be right or wrong. It is wrong sufficiently often to make a major contribution to the error rate of the system.

This paper presents a survey whose focus is character segmentation. Other papers have surveyed segmentation as part of a larger work, e.g., cursive recognition [8][14][15][39][40][51], or document analysis [18][22].

1.2: Organization of methods

A review of available literature suggests three "pure" strategies for segmentation:

1. The classical approach: segments are identified based on "character-like" properties. This process of cutting up the image into meaningful components is given a special name, "dissection", in discussions below.
2. Recognition-based segmentation: the system searches the image for components that match predefined classes.
3. Holistic methods: the system seeks to recognize words as a whole, thus avoiding the need to segment into characters.

In strategy (1) the criterion for good segmentation is the agreement of general properties of the segments obtained with those expected for valid characters. Examples of such properties are height, width, separation from neighboring components, etc. In method (2) the criterion is recognition confidence, perhaps including syntactic or semantic correctness of the overall result. Holistic methods (3) in essence revert to the classical approach with words as the alphabet to be segmented.

In the following sections we discuss examples of each of these strategies, as well as a hybrid strategy in which a preliminary segmentation is implemented based on image features, but the best segmentation is chosen on the basis of classification of a set of hypotheses formed by combination of the initial segments. The discussion is primarily limited to off-line character recognition and to Western character sets.

2: Dissection techniques for segmentation

Methods discussed in this section are based on what will be termed "dissection" of an image. By dissection is meant the decomposition of the image into a sequence of subimages using general features. This is opposed to later methods that select subimages independent of content. Dissection is an intelligent process that analyzes an image without using specific class shape information.

2.1: White space and pitch

In machine printing, vertical whitespace often serves to separate successive characters. This property can be extended to handprint by providing separated boxes in which to print individual symbols. In applications such as billing, where document layout is specifically designed for OCR, additional spacing is built into the fonts used.

In machine print applications involving limited font sets each character may be known to occupy a block of fixed width. The pitch, or number of characters per unit of horizontal distance, provides a basis for estimating segmentation points. The sequence of segmentation points obtained for a given line of print should be approximately equally spaced at the distance corresponding to the pitch.

One well-documented early commercial machine that dealt with a relatively unconstrained environment was the reader installed at the U. S. Social Security Administration in 1965 [30]. This device read alphanumeric data typed by employers on forms submitted quarterly to the SSA. There

0-8186-7128-9/95 $4.00 © 1995 IEEE

was no way for SSA to impose constraints on the printing process. Typewriters might be of any age or condition, ribbons in any state of wear, and the font style might be one or more of approximately 200.

In the SSA reader segmentation was accomplished in two scans of a print line by a flying-spot scanner. On the initial scan, from left to right, the character pitch distance D was estimated by analog circuitry. On the return scan, right to left, the actual segmentation decisions were made using D. The principal rule applied was that a double white column triggered a segmentation boundary. If none was found within distance D, then segmentation was forced.

2.2: Projection analysis

The vertical projection (also called the "vertical histogram") of a print line, consists of a simple running count of the black pixels in each column. It can serve for detection of white space between successive letters. Moreover, it can indicate locations of vertical strokes in machine print, or regions of multiple lines in handprint. Thus analysis of the projection of a line of print has been used as a basis for segmentation. For example, in [44], in segmenting Kanji handprinted addresses, columns where the projection fell below a predefined threshold were candidates for splitting the image.

When printed characters touch, or overlap horizontally, the projection often contains a minimum at the proper segmentation column. In [1] the projection was first obtained, then the ratio of second derivative of this curve to its height was used as a criterion for choosing separating columns. This ratio tends to peak at minima of the projection, and avoids the problem of splitting at points along thin horizontal lines.

A prefiltering was implemented in [52] in order to improve the projection function. The filter ANDed adjacent columns prior to projection, producing a deeper valley at columns where only portions of the vertical edges of two adjacent characters are merged.

In order to deal with slanted characters [21] implemented projections at two-degree increments between -16 and +16 degrees from the vertical.

2.3: Connected component processing

Segmentation of handprint or kerned machine printing calls for a two-dimensional analysis, for even nontouching characters may not be separable along a single straight line. A common approach is based on determining connected black regions ("connected components", or "blobs"). Further processing may then be necessary to combine or split these components into character images.

There are two types of followup processing, one based on the "bounding box", i.e., the location and dimensions of each connected component, and the other based on detailed analysis of the the connected components.

The distribution of bounding boxes tells a great deal about the proper segmentation of an image consisting of non-cursive characters. By testing adjacency relationships to perform merging, or size and aspect ratios to trigger splitting mechanisms, much of the segmentation task can be accurately performed at a low cost in computation.

This approach has been applied, for example, in segmenting handwritten postcodes [9] using knowledge of the number of symbols in the code: six for the Canadian codes used in experiments. Connected components were joined or split according to rules based on height and width of their bounding boxes. The rather simple approach correctly classified 93% of 300 test codes, with only 2.7% incorrect segmentation and 4.3% rejection.

An experimental comparison of character segmentation by projection analysis vs. segmentation by connected components is reported in [55]. Both segmenters were tested on a large data base (272,870 handprinted digits) using the same follow-on classifier. Characters separated by connected component segmentation resulted in 97.5% recognition accuracy, while projection analysis (along a line of variable slope) yielded only 95.3% accuracy.

A number of dissection methods have been developed to detect special features of joined chaacters and to use them in splitting a character string image into subimages. Such methods often work as a follow-on to bounding box analysis. Only image components failing certain dimensional tests are subjected to detailed examination.

A priori knowledge may be used in the splitting process, e.g., an algorithm may assume that some but not all input characters can be connected. Characters in form data may be known to be digits or capital letters, placing a constraint on dimensional variations. Handwritten zip codes with more than three connected characters are rarely encountered, and the total number of symbols is known.

In [45] local vertical minima met in following the bottom contour of a connected component are used as "landmark points". The contour is followed from the leftmost minimum counter-clockwise until a turning point is found. This point is presumed to lie at the intersection of the two characters and a cut is performed vertically.

The algorithm proposed in [48] for segmenting digit strings not only detects likely segmentation points, but also computes an appropriate segmentation path. The first step sought a vertical cutting path. If none was found a special algorithm using a "Hit and Deflect Strategy" was called. This algorithm computes a curved segmentation path by iteratively moving a scanning point. The scanning point starts from the maximum peak in the bottom profile of the lower half of the image. It then moves upwards by means of simple rules which seek to avoid cutting the characters until further movement is impossible. [54] investigated extensions of the Hit and Deflect scheme.

Methods for defining splitting paths have been examined in a number of other studies as well. [12] performs background analysis to extract the face-up and face-down valleys, strokes and loop regions of component images. [24] applies a distance transform to the input image in order to compute the splitting path. A shortest-path method investigated in [53] produces an "optimum" segmentation path using dynamic programming.

2.4: Landmarks

In recognition of cursive writing it is common to analyze the image of a character string in order to define lower, middle and upper zones. This permits the ready detection of ascenders and descenders, features that can serve as "landmarks" for segmentation of the image. This technique was applied to on-line recognition in pioneering work by Frischkopf and Harmon [28].

2.5: Dissection with contextual postprocessing

Segmentation by dissection can later be subjected to evaluation using linguistic context, as shown in [4]. Here a Markov model represents splitting and merging as well as misclassification in a recognition process. The system seeks to correct such errors by minimizing an edit distance between output and words in a lexicon. Thus it does not directly define segmentation hypotheses, it merely tries to correct poorly made ones. The approach is influenced by earlier developments in speech recognition. A non-Markovian system reported in [8] uses a spell-checker to correct repeatedly-made merge and split errors in a complete text, rather than in single words as above.

Other methods described in this section dissect the input image into subimages that are not necessarily individual characters. These preliminary shapes, called "graphemes" or "pseudo-characters", are intended to fall into readily identifiable classes. A contextual mapping function from grapheme classes to symbols can then complete the recognition process. In doing so, the mapping function may combine or split grapheme classes, i.e., implement a many-to-one or one-to-many mapping. This amounts to a (implicit) resegmentation of the input.

An early use of this concept was [47], a system for off-line cursive script recognition. Here dissection into graphemes was first performed based on the detection of characteristic areas of the image. The classes recognized by the classifier did not correspond to letters, but to specific shapes that could be reliably segmented (typically combinations of letters, but also portions of letters). Only 17 non-exclusive classes were considered. Some dissection techniques for cursive script are based on the fact that lowercase characters may be linked by lower ligatures. Dissection techniques based on the principle of detecting ligatures were developed in [17][42][37]. The last study augmented this by the detection of possible pre-segmentation zones, and the use of a pre-recognition algorithm, whose aim was not to recognize characters, but to evaluate whether a subimage defined by the pre-segmenter was likely to constitute a valid character.

A similar presegmenter was presented in [31]. In this case analysis of the upper contour, and a set of rules based on contour direction, closure detection, and zone location were used. Upper contour analysis was also used in [34] for a pre-segmentation algorithm that served as part of the second stage of a hybrid recognition system. The first stage of this system also implemented a form of the hit and deflect strategy previously mentioned.

A technique for segmenting handwritten strings of variable length, was described in [20]. It employs upper and lower contour analysis and a splitting technique based on the hit and deflect strategy. Segmentation can also be based on the detection of minima of the lower contour as in [5]. A recent study which aims to locate "key letters" in cursive words employs background analysis to perform letter segmentation [13].

3: Recognition-based segmentation

3.1: Methods that search the image

Search methods can segment words into letters or other units without use of feature-based dissection algorithms. Rather, the image is divided systematically into many overlapping pieces without regard to content. These are classified as part of an attempt to find a coherent segmentation / recognition result. Systems using such a principle perform "recognition-based" segmentation: letter segmentation is a by-product of letter recognition. The approach is also called "segmentation-free" recognition. Conceptually, these methods originate from schemes in [35][7] for the recognition of machine-printed words. The basic principle is to use a mobile window of variable width to provide sequences of tentative segmentations which are confirmed (or not) by character recognition. Multiple sequences are obtained from the input image by varying the window placement and size. Each sequence is assessed as a whole based on recognition results.

In [35] the task was recognition of typewritten Cyrillic characters of poor quality. Although character spacing was fixed, Kovalevski's model assumed that the exact value of pitch and the location of the origin for the print line were known only approximately. He developed a solution under the assumption that segmentation occurred along columns. Correlation with prototype character images was used as a method of classification. He showed that the problem of segmenting to obtain a best overall match can be formulated as one of determining the path of maximum length in a graph, and that this path can be found by dynamic programming. This process was implemented in hardware to produce a working OCR system.

In [7] is reported a recursive splitting algorithm for machine-printed characters. This algorithm, also based on prototype matching, systematically tests all combinations of admissible separation boundaries until it either exhausts the set of cutpoints, or else finds an acceptable segmentation. An acceptable segmentation is one in which every segmented pattern matches a library prototype within a prespecified distance tolerance.

A technique combining dynamic programming and neural net recognition was proposed in [6]. This metod, called "Shortest Path Segmentation", selects the optimal consistent combination of cuts from a predefined set of windows. Given the set of candidate cuts, all possible "legal" segments are constructed by combination. A graph whose nodes represent acceptable segments is then created. Nodes are connected when they correspond to compatible neighbors. The paths of this graph represent all the legal segmentations of the word. Each node of the graph is then assigned a "distance" obtained by the neural net recognizer. The shortest path through the graph thus corresponds to the best recognition and segmentation of the word.

The method of "selective attention" [23] takes neural networks even further in the handling of segmentation problems. In this approach a neural net seeks recognizable patterns in an image input, but is inhibited automatically after recognition in order to ignore the region of the recognized character and search for new character images in neighboring regions.

3.2: Methods that segment a feature representation

3.2.1: Hidden Markov Models

The object of Hidden Markov Models is to model variations in printing or cursive writing as an underlying probabilistic structure which is not directly observable. This structure consists of a set of states plus transition probabilities between states. In addition, the observations that the system makes on an image are represented as random variables whose distribution depends on the state. These observations constitute a sequential feature representation of the input image. The survey [26] provides an introduction to its use in recognition applications.

Hidden Markov models can be applied in a number of ways in OCR. Here we are concerned with the case where the Markov model represents state-to-state transitions within a character. These transitions provide a sequence of observations on the character. Features are typically measured in the left-to-right direction. This facilitates the representation of a word as a concatenation of character models. In such a system segmentation is (implicitly) done in the course of matching the model against a given sequence of feature values gathered from a word image. That is, it decides where one character model leaves off and the next one begins, in the series of features analyzed.

Elementary HMMs describing letters can be combined to form either several model-discriminant word HMMs or a single path-discriminant model. In model-discriminant HMMs [10] a model is constructed for each word, while in the path discriminant HMM [36] one global model is constructed. Path discriminant HMMs can handle large vocabularies, but are generally less accurate than model-discriminant HMMs. They may incorporate a lexicon comparison module in order to ignore invalid letter sequences obtained by path optimization.

First order Markov models are employed in most applications; [36] is an example of a second order HMM. Models for cursive script ordinarily assume discrete feature values. However, continuous probability densities may also be used, as in [3].

3.2.2: Non-Markov approaches

A method stemming from concepts used in machine vision for recognition of occluded objects is reported in [11]. Here various features and their positions of occurrence are recorded for an image. Each feature contributes an amount of evidence for the existence of one or more characters at the position of occurrence. The positions are quantized into bins such that the evidence for each character indicated in a bin can be summed. to give a score for classification. These scores are subjected to contextual processing using a predefined lexicon in order to recognize words. The method is being applied to text printed in a known proportional font.

This family of recognition-based approaches has more often been aimed at cursive handwriting recognition. Probabilistic relaxation was used in [29] to read off-line handwritten words. The model was working on a hierarchical description of words derived from a skeletal representation. Relaxation was performed on the nodes of a stroke graph and of a letter graph where all possible segmentations were kept. Complexity was progressively reduced by keeping only the most likely solutions. N-gram statistics were also introduced to discard illegible combinations of letters.

A different approach uses the concept of regularities and singularities [49]. In this system, a stroke graph representing the word is obtained after skeletonization. The "singular parts", which are supposed to convey most of the information, were deduced by eliminating "regular part" of the word (the sinusoid-like path joining all cursive ligatures). The most robust features and characters (the "anchors") were then detected from a description chain derived from these singular parts and dynamic matching was used for analyzing the remaining parts.

A top-down directed word verification method called "backward matching" is proposed in [38].

4: Mixed strategies: "Oversegmenting"

Dissection and search methods can be combined in a hybrid approach to segmentation. A dissection algorithm is applied to the image, but the intent is to "oversegment", i.e., to cut the image in sufficiently many places that the correct segmentation boundaries are included among the cuts made. Once this is assured, the optimal segmentation is sought by evaluation of subsets of the cuts made. Each subset implies a segmentation hypothesis, and classification is brought to bear to evaluate the different hypotheses and choose the most promising segmentation.

The strategy in a simple form is illustrated in [22]. Here a great deal of effort was expended in analyzing the shapes of pairs of touching digits in the neighborhood of contact, leading to algorithms for determining likely separation boundaries. However, multiple separating points were tested, i.e., the touching character pair was oversegmented. Each candidate segmentation was tested separately by classification, and the split giving the highest recognition confidence was accepted. This approach reduced segmentation errors 100-fold compared with the previously used segmentation technique that did not employ recognition confidence.

Because touching was assumed limited to pairs, the above method could be implemented by splitting a single image along different cutting paths. Thus each segmentation hypothesis was generated in a single step. When the number of characters in the image to be dissected is not known a priori, or if there are many touching characters, e.g., cursive writing, then it is usual to generate the various hypotheses in two steps. In the first step a set of likely cutting paths is determined, and the input image is divided into elementary components by separating along each path. In the second step, segmentation hypotheses are generated by forming combinations of the components. All combinations meeting certain acceptability constraints (such as size, position, etc.) are produced and scored by classification confidence. An optimization algorithm, typically implemented on dynamic programming principles and possibly making use of contextual knowledge, does the actual selection.

A number of researchers began using this basic approach at about the same time, e.g., [33][19][50]. The latter two include lexical matching in the overall process.

It is also possible to carry out an oversegmenting procedure sequentially by evaluating trial separation boundaries [2]. In this work a neural net was trained to detect likely cutting columns for machine printed characters using neighborhood characteristics.

5: Holistic Strategies

Whole word recognition was introduced by Earnest in the beginning of the nineteen sixties [16]. Although it was designed for on-line recognition, his method followed an off-line methodology: data was gathered by means of a "photo-style" in a binary matrix and no temporal information was used. Recognition was based on the comparison of a collection of simple features extracted from the whole word against a lexicon of "codes" representing the "theoretical" shape of the possible words. Feature extraction was based on the determination of the middle zone of the words and ascenders and descenders were found by considering the part of the writing exceeding this zone. The lexicon of possible word codes was obtained by means of a transcoding table describing all the usual ways of writing letters.

Later holistic systems also use middle zone determination to detect the ascenders and descenders, and extract similar features: ascenders, descenders, directional strokes, cusps, diacritical marks, etc.

The main advances in recent techniques reside in the way comparison between hypotheses and references is performed. Recent comparison techniques are more flexible and better take into account the dramatic variability of handwriting. These techniques, originally introduced for speech recognition, generally rely on Dynamic Programming to satisfy optimization criteria based either on distance measurements or else on a probabilistic framework using Markov or Hidden Markov Chains.

Dynamic Programming was employed in [46] for check and city name recognition. Words were represented by a list of features indicating the presence of ascenders, descenders, directional strokes and closed loops. The "middle zone" was not delimited by straight lines, but by means of smooth curves following the central part of the word, even if slanted or irregular in size. Relative y-location was associated to every feature and uncertainty coefficients were introduced to make this representation more tolerant to distortion by avoiding binary decisions.

Hidden Markov Models are used in [43] for the recognition of literal digits and in [25] for off-line cheque recognition. Moreover, this second system also implement several Markov models at different recognition stages (word recognition and cheque amount recognition). Context is taken into account via prior probabilities of words and word trigrams. In [27] lines are extracted from binary images of words and accumulated in prototypes called "holographs". During the test phase, correlation is used to obtain a distance between an unknown word and each word prototype.

In machine printed text characters are regular, therefore feature representations are stable. Also, in a long document repetitions of the most common words occur with predictable frequency. In [32] these characteristics were combined to cluster the ten most common short words with good accuracy, as a precursor to word recognition. It was suggested that identification of the clusters could be done on the basis of unigram and bigram frequencies.

Use of holistic methods is ordinarily constrained to a specific lexicon. More general applications require a dynamic generation of holistic descriptions. This stage must be able to convert any word from its ASCII form to the holistic representation required by the recognition algorithm. Word representation must then be generated from generic information about letter and ligature representations using a reconstruction model. Such word reconstruction is required when dealing with a dynamically defined lexicon, as for instance in the postal application [41], where the list of possible city names is derived from zip code recognition.

6: Concluding remarks

Methods for treating the problem of segmentation in character recognition have developed remarkably in the last decade. A variety of techniques has emerged, influenced by developments in related fields such as speech and online recognition. In this paper we have proposed an organization of these methods under three basic strategies, with hybrid approaches also identified.

We have not attempted to compare the effectiveness of algorithms, or to discuss the crucial topic of evaluation. In truth, it would be very difficult to assess techniques separate from the systems for which they were developed. We believe that wise use of context and classifier confidence has led to improved accuracies, but there is little experimental data to permit an estimation of the amount of improvement to be ascribed to advanced techniques. Perhaps with the wider availability of standard databases, experimentation will be carried out to shed light on this issue.

References

[1] H.S. Baird, S. Kahan and T. Pavlidis, "Components of an omnifont page reader", Proc. 8th ICPR, Paris, pp. 344-348, 1986.

[2] T. Bayer, U. Kressel and M. Hammelsbeck, "Segmenting merged characters," Proc. 11th ICPR, vol. II. conf. B: Pattern Recognition Methodology and Systems, pp. 346-349, 1992.

[3] E.J. Bellegarda, J. R. Bellegarda, D. Nahamoo and K.S. Nathan, "A Probabilistic Framework for On-line Handwriting Recognition", Proc. IWFHR III, Buffalo, page 225, May 1993.

[4] R. Bozinovic, S.N. Srihari, "A String Correction Algorithm for Cursive Script Recognition", IEEE PAMI v4 n6 p655, 1982.

[5] R.M. Bozinovic and S.N. Srihari, "Off-Line Cursive Script

Recognition", IEEE PAMI vol. 11, no. 1, page 68, 1989.

[6] C.J.C. Burges, J.I. Be and C.R. Nohl, "Recognition of Handwritten Cursive Postal Words using Neural Networks", Proc. USPS 5th Adv. Technology. Conf., page A-117, Nov/Dec. 1992.

[7] R.G. Casey and G. Nagy, "Recursive Segmentation and Classification of Composite Patterns", 6th ICPR, page 1023, 1982.

[8] R.G. Casey, "Text OCR by solving a cryptogram," Proc. 8th ICPR, Paris, pp. 349-351, Oct. 1986.

[9] M. Cesar, R. Shinghal, "Algorithm for segmenting handwritten postal codes," Int. J. Man Mach Stud., v33 n1 p63, Jul. 1990.

[10] M. Chen, A.Kundu, "An Alternative to Variable Duration HMM in Handwritten Word Recognition", IWFHR III, Buffalo, p82, May 1993.

[11] C. Chen, J. DeCurtins, "Word recognition in a segmentation-free approach to OCR", ICDAR, Tsukuba, Japan, p573, Oct. 1993.

[12] M. Cheriet, Y.S. Huang and C.Y. Suen, "Background Region-Based Algorithm for the Segmentation of Connected Digits", Proc. 11th ICPR, vol. II, page 619, Sept 1992.

[13] M. Cheriet, "Reading Cursive Script by Parts", Pre-Proceedings IWFHR III, Buffalo, page 403, May 1993.

[14] G. Dimauro, S. Impedovo and G. Pirlo, "From Character to Cursive Script Recognition: Future Trends in Scientific Research", Proc. 11th ICPR, vol. II, page 516, Aug. 1992.

[15] C.E. Dunn, P.S.P. Wang, "Character Segmenting Techniques for Handwritten Text - A Survey", 11th ICPR, vII p577, August 1992.

[16] L.D. Earnest, "Machine Recognition of Cursive Writing", C. Cherry ed, *Information Processing*, Butterworth, London, 1962.

[17] R.W. Ehrich and K.J. Koehler, "Experiments in the Contextual Recognition of Cursive Script", IEEE Trans. Comp., v24 n2 p182, 1975.

[18] D.G. Elliman and I.T. Lancaster, "A Review of Segmentation and Contextual Analysis Techniques for Text Recognition", Pattern Recognition, vol. 23, no. 3/4, pp. 337-346, 1990.

[19] J.T. Favata and S.N. Srihari, "Recognition of General Handwritten Words using a Hypothesis Generation and Reduction Methodology", Proc. 5th USPS ATC, page 237, Nov/Dec. 1992.

[20] R. Fenrich, "Segmenting of Automatically located handwritten numeric strings", From Pixels to Features III, S. Impedovo and J.C. Simon (eds.), Elsevier, 1992, Chapter 1, page 47.

[21] P.D. Friday and C.G. Leedham, "A pre-segmenter for separating characters in unconstrained hand-printed text," Proc. Int. Conf. on Image Proc., Singapore, Sept. 1989.

[22] H. Fujisawa, Y. Nakano and K. Kurino, "Segmentation methods for character recognition: from segmentation to document structure analysis," Proc. IEEE, v 80, n 7 pp. 1079-1092, July 1992.

[23] K. Fukushima and T. Imagawa, "Recognition and segmentation of connected characters with selective attention", Neural Networks, vol. 6, no. 1, pp. 33-41, 1993.

[24] P. Gader, M. Magdi and J-H. Chiang, "Segmentation-Based Handwritten Word Recognition", Proc. USPS 5th ATC, Nov/Dec 1992.

[25] M. Gilloux, J.M. Bertille and M. Leroux, "Recognition of Handwritten Words in a Limited Dynamic Vocabulary", IWFHR III, Buffalo, page 417, 1993.

[26] M. Gilloux, "Hidden Markov Models in Handwriting Recognition", Fundamentals in Handwriting Recognition, S. Impedovo (Ed.), NATO ASI Series F, vol. 124, Springer Verlag, 1994.

[27] N. Gorsky, "Off-line Recognition of Bad Quality Handwritten Words Using Prototypes", Fundamentals in Handwriting Recognition, S. Impedovo (Ed.), NATO ASI Series F, vol. 124, Springer Verlag, 1994.

[28] L.D. Harmon, "Automatic Recognition of Print and Script", Proc. of the IEEE, vol. 60, no. 10, pp. 1165-1177, Oct. 72.

[29] K.C. Hayes, "Reading Handwritten Words Using Hierarchical Relaxation", Comp Graphics Image Proc, vol. 14, pp. 344-364, 1980.

[30] R.B. Hennis "The IBM 1975 Optical Page Reader: system design" IBM Journ. of Res. & Dev., pp. 346-353, Sept. 1968.

[31] M. Holt, M Beglou and S. Datta, "Slant-Independent Letter Segmentation for Off-line Cursive Script Recognition", **From Pixels to Features III**, S. Impedovo and J.C. Simon (eds.), Elsevier, 1992, p 41.

[32] J. Hull, S. Khoubyari and T.K. Ho, "Word image matching as a technique for degraded text recognition", Proc. 11th IAPR, The Hague, vol. II, conf. B, pp. 665-668, Sept 1992.

[33] F. Kimura, S. Tsuruoka, M. Shridhar and Z. Chen, "Context-directed handwritten word recognition for postal service applications", Proc. 5th US PS ATC, Washington DC, 1992.

[34] F. Kimura, M. Shridhar and N. Narasimhamurthi, "Lexicon Directed Segmentation-Recognition Procedure for Unconstrained Handwritten Words", Proc. IWFHR III, Buffalo, page 122, May 1993.

[35] V.A. Kovalevsky, "Character readers and Pattern Recognition", Spartan Books, Washington D.C., 1968.

[36] A. Kundu, Yang He and P. Bahl, "Recognition of Handwriten Words: First and Second Order Hidden Markov Model Based Approach", Pattern Recognition, vol. 22, no. 3, pp. 283, 1989.

[37] E. Lecolinet and J-P. Crettez, "A Grapheme-Based Segmentation Technique for Cursive Script Recognition", Proc ICDAR, St. Malo, France, page 740, Sept. 1991.

[38] E. Lecolinet, "A New Model for Context-Driven Word Recognition", Proc SDAIR, Las Vegas, page 135, April 1993.

[39] E. Lecolinet and O. Baret, "Cursive Word Recognition: Methods and Strategies", Fundamentals in Handwriting Recog, S. Impedovo (Ed.), NATO ASI Series F v 124, Springer Verlag, 1994.

[40] G. Lorette and Y. Lecourtier, "Is Recognition and Interpretation of Handwritten Text: a Scene Analysis Problem ?" Pre-Proceedings IWFHR III, Buffalo, page 184, May 1993.

[41] S. Madhvanath and V. Govindaraju, "Holisitic Lexicon Reduction", Proc. IWFHR III, Buffalo, page 71, May 1993.

[42] M. Maier, "Separating Characters in Scripted Documents", Proc. 8th ICPR, Paris, page 1056, 1986.

[43] R. Nag, K.H. Wong and F. Fallside, "Script Recognition Using Hidden Markov Models", IEEE ICASSP, Tokyo, p2071, 1986.

[44] K. Ohta, I. Kaneko, Y. Itamoto and Y. Nishijima, "Character segmentation of address reading/letter sorting machine" for the ministry of posts and telecommunications of Japan, NEC Research & Development, vol. 34, no. 2, pp. 248-256, Apr. 1993.

[45] S.K. Parui, B.B. Chaudhuri and D.D. Majumder, "Aprocedure for recognition of connected handwritten numerals", Int. J. Systems Sci., vol, 13, no. 9, pp. 1019-1029, 1982.

[46] B. Plessis, A. Sicsu. L. Heute, E. Lecolinet, O. Debon and J-V. Moreau, "A Multi-Classifier Strategy for the Recognition of Handwritten Cursive Words", Proc. Int. Conf. on Document Analysis and Recognition, Tsukuba City, Japan, pp. 642-645, Oct. 1993.

[47] K.M. Sayre, "Machine Recognition of Handwrittten Words: A Project Report", Pattern Recognition, vol. 5, pp. 213-228, 1973.

[48] M. Shridhar and A. Badreldin, "Recognition of Isolated and Simply Connected Handwritten Numerals", Patt Recog, v19 n1 p1, 1986.

[49] J.C. Simon, "Off-line Cursive Word Recognition", Proceedings of the IEEE, page 1150, July 1992.

[50] R.M.K Sinha et al, "Hybrid contextual text recognition with string matching," IEEE PAMI vol. 15, no. 9, pp. 915-925, Sept. 1993.

[51] C.C. Tappert, C.Y. Suen and T. Wakahara, "State of the Art in On-line Handwriting Recognition", IEEE PAMI, v12 n8 p787, 1990.

[52] S. Tsujimoto and H. Asada, "Major components of a complete text reading system", Proc. IEEE, v 80, n 7, p1133, July 1992.

[53] J. Wang and J. Jean, "Segmentation of merged characters by neural networks and shortest path", Patt Recog, v27 n5 p649, May 1994.

[54] J.M. Westall, M.S. Narasimha "Vertex directed segmentation of handwritten numerals", Patt Recog, v26 n10 p1473, Oct. 1993.

[55] R.A. Wilkinson, "Comparison of massively parallel segmenters," NIST Tech. Report, Gaithersburg, MD, Sept. 1992.