



0031-3203(93)E0002-3

## SEGMENTATION OF TOUCHING CHARACTERS IN PRINTED DOCUMENT RECOGNITION

SU LIANG,† M. SHRIDHAR‡ and M. AHMADI†

† Department of Electrical Engineering, University of Windsor, Windsor, Ontario, Canada N9B 3P4

‡ Department of Electrical and Computer Engineering, University of Michigan–Dearborn, Dearborn, MI 48128, U.S.A.

(Received 2 March 1993; in revised form 28 December 1993; received for publication 7 January 1994)

**Abstract**—A new discrimination function is presented for segmenting touching characters based on both pixel and profile projections. A dynamic recursive segmentation algorithm is developed for effectively segmenting touching characters. Contextual information and spell checking are used to correct errors caused by incorrect recognition and segmentation. Based on 12 real documents, a maximum 99.85% and a minimum 99.4% recognition accuracy is achieved.

Touching character segmentation    Dynamic recursive segmentation    Character contextual classes  
 Feature extraction    Spelling correction

### 1. INTRODUCTION

Document image processing has been the subject of extensive research since the early 1980s.<sup>(1,2)</sup> Today, character recognition systems dramatically facilitate the transfer of information into computer systems without intensive manual keying. Kahan *et al.* suggested that in practical applications, a document recognition system is required to read texts accurately with at least a 99.9% recognition rate.<sup>(3)</sup> It is not difficult to design a character recognition system that recognizes well-formed and well-spaced printed characters. However, it is a challenge to develop a system which can maintain such a high recognition rate, regardless of the quality of the input documents and the character fonts.<sup>(2-6)</sup> Presently, most recognition errors are due to character segmentation errors.<sup>(4-6)</sup> Very often, even in printed text, adjacent characters are touching, and may exist in an overlapped field. Therefore, it is essential to segment a given word correctly into its character components. Any failure or error in this segmentation step can lead to a critical loss of information from the document.

Several investigators have attempted to develop techniques for properly segmenting words into their character components. Kahan *et al.*<sup>(3)</sup> discussed the segmentation of touching characters. Tsujimoto and Asada<sup>(4)</sup> constructed a decision tree for resolving ambiguities in segmenting touching characters. Casey and Nagy<sup>(5)</sup> proposed a recursive segmentation algorithm for touching characters. Bose and Kuo<sup>(6)</sup> applied the Hidden Markov Model to the touched and degraded text recognition.

In this paper, we propose a dynamic recursive segmentation algorithm for segmentation of touching characters based on the method proposed by Casey

and Nagy.<sup>(5)</sup> In this process, several candidate cutting points are determined from the proposed discrimination functions; the algorithm then iteratively implements a forward segmentation or a backward merge procedure based on the outputs of a character classifier which operates on the components generated by segmentation algorithms. Using contextual information including a spell checker, further improvements in word recognition are achieved. Extensive studies with different text documents have attested to the feasibility of the proposed algorithm. The algorithm and test results are discussed in the following sections.

### 2. SEPARATION OF TOUCHING CHARACTERS

A typical text recognition system is shown in the flow chart in Fig. 1. Since the work described here deals with segmentation of touching characters, it is assumed that a line of text with the appropriately labelled connected components in the line have already been extracted using appropriate techniques.<sup>(7-9)</sup> Figure 2 shows the connected components and their labels for the word “with”. It is interesting to observe (Fig. 2(c)) that the letters “t” and “h” are touching.

#### 2.1. The modified discrimination function for touching characters

Among the many articles that have been published concerning the segmentation of touching characters,<sup>(3-6)</sup> we were especially interested in the algorithm proposed by Kahan *et al.*<sup>(3)</sup> They detected touching characters by using the ratio of the second-order difference of the vertical pixel projection to the value of the vertical projection as an objective function. The cutting points for touching characters were obtained in the

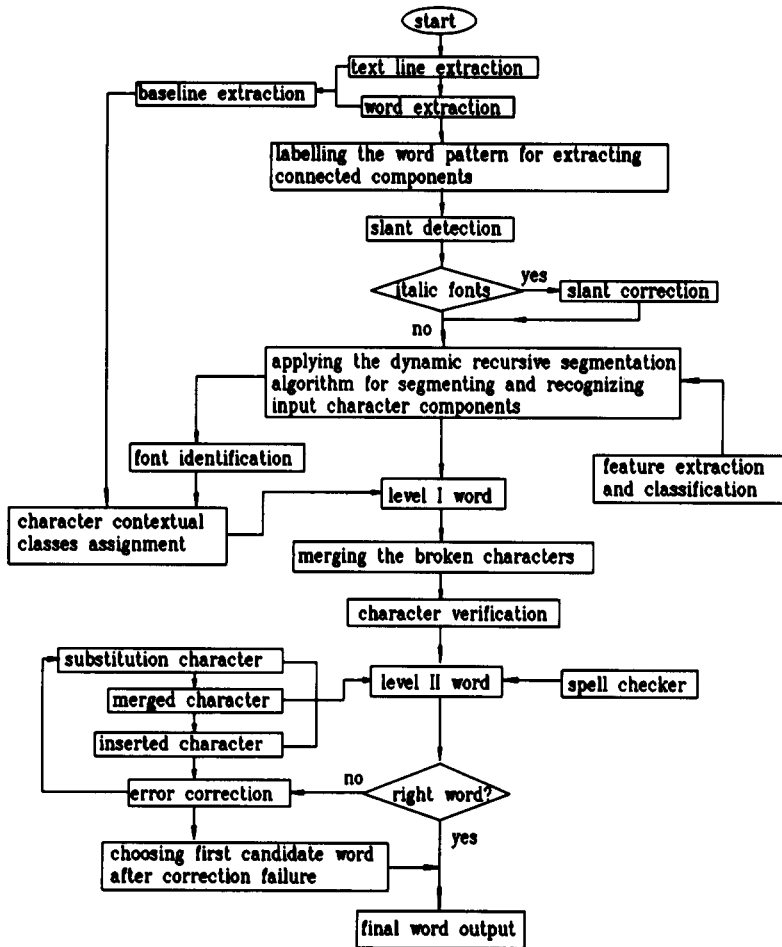


Fig. 1. Recognition system diagram.

horizontal positions where the segmenting objective function was maximized. This method is able to cut most lightly touching characters; however, it was unable to separate heavily touching characters, particularly such as “oo”, “oe” and “od”, etc., because the pixel projection waveforms for these kinds of touching characters vary gradually due to the lack of the vertical strokes near the touching points. Figure 3 shows an example of touching characters “oo”.

In order to improve the segmentation process, the authors propose two discrimination functions based on pixel and profile projections. The pixel projection and profile projection are described as follows:

(1) The pixel projection is defined as  $\{PXP(k), k = 1, 2, \dots, LT\}$ . It consists of the total number of black “1” pixels in each vertical column.  $LT$  is the length of the touching characters.

(2) The profile projection is defined as  $\{PFP(k) = TP(k) - BP(k), k = 1, 2, \dots, LT\}$ .  $TP(k)$  is the top profile of the external contour of the touching characters as seen from the top;  $BP(k)$  is the bottom profile of the external contour of the touching characters as seen from the bottom.

Figure 3 illustrates these projections for the component “oo”. Two segmentation discrimination functions based on the profile and pixel projections are then defined as follows:

$$F_1^\alpha(k) = \left[ \frac{PFP(k+L1) - 2PFP(k) + PFP(k-L1)}{PFP(k)} \right]^\alpha \quad (1)$$

$$F_2^\alpha(k) = \left[ \frac{PXP(k+L2) - 2PXP(k) + PXP(k-L2)}{PXP(k)} \right]^\alpha \quad (2)$$

where  $L1$  and  $L2$  in equations (1) and (2) denote the distances between the current column and the adjacent columns  $L1$  and  $L2$  apart.  $L1$  and  $L2$  are determined empirically based on the size of the characters in the documents. As an example for text with “times” font of size 11,  $L1$  and  $L2$  were chosen as 8 and 4, respectively.  $\alpha$  which represents the power of  $F1$  and  $F2$  is typically chosen to be an integer larger than 1. Figure 4 illustrates the effect of  $\alpha$  on the values of the discrimination functions. Figures 4(c) and (d) show the discrimination functions for  $\alpha = 1$  and 2, respectively. It is evident that

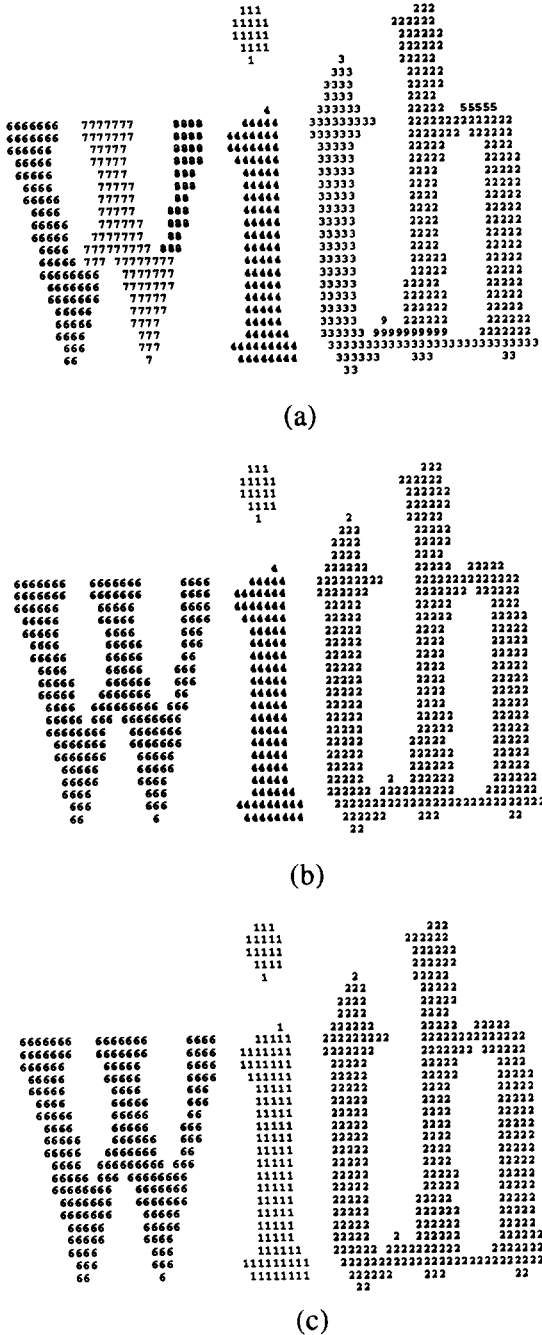


Fig. 2. (a) First scan of component labelling (“w” has 3 labels, ‘i’ has 2 labels and ‘th’ has 4 labels). (b) Second scan of component labelling (‘w’ has 1 label, ‘i’ has 2 labels and ‘th’ has 1 label). (c) Final result of the labelled “with” (‘w’, ‘i’ and ‘th’ have 1 label).

the values of the discrimination functions are increased significantly with  $\alpha = 2$ , especially when  $F_1$  and  $F_2$  are larger than 2. Using suitable threshold, candidate cutting points are derived for further processing. Figures 4(c) and (d) show four candidate cutting points for the component “oo” with a threshold of 11. It is noted that in this case, the correct cutting location is determined by the discrimination function  $F_1^2(k)$  based

on the profile projection. Figure 5 illustrates the process for the component “th”. In this case, it is clearly seen that the discrimination function  $F_2^2(k)$  based on the pixel projection yields candidate cutting points that contain the correct cutting location. We therefore define the discrimination function as follows:

$$F(k) = \begin{cases} F_2^2(k), & \text{if } F_1^2(k) > T \text{ \& } F_2^2(k) > T \\ F_1^2(k), & \text{if } F_1^2(k) > T \text{ \& } F_2^2(k) < T \\ F_2^2(k), & \text{if } F_1^2(k) < T \text{ \& } F_2^2(k) > T \\ 0, & \text{if } F_1^2(k) < T \text{ \& } F_2^2(k) < T. \end{cases} \quad (3)$$

Several cutting point candidates, including some false cutting points, were obtained at the locations where the values of the discrimination function were greater than a specific threshold. The optimal cutting points were found after several iterations by applying the dynamic recursive segmentation algorithm. Some broken characters caused by false cutting points can be merged using layout context information. This will be discussed in the following sections.

## 2.2. The dynamic recursive segmentation algorithm

The recursive segmentation algorithm with adaptive windows, developed by Casey and Nagy,<sup>(5)</sup> performed template matching after a suitable window was chosen. According to Casey and Nagy, if the classification failed, a different partitioning of the input pattern was tried. However, even for a specific font, proportional spacing characters make it difficult to choose a suitable window on the first trial. Tsujimoto and Asada used a *break cost* to segment touching characters.<sup>(4)</sup> They constructed a decision tree and a set of additional rules to obtain the character component sequences. However, their algorithm required intensive computation to build a decision tree and search for a correct path; the more touching characters in a word, the larger the decision tree would be. Also their algorithm tended to accept several paths in the tree structure concurrently.

In our approach, we proposed a dynamic recursive segmentation algorithm that implements a forward segmentation or a backward merge procedure based on the output of a character classifier operating on the components generated by the cutting points. This procedure terminates when the segments generated by specific points are recognized by the character classifier with high confidence. Figures 6 and 7 illustrate the algorithm for two types touching characters: the double-O “oo” and the serif touching pair “th”, and the triplet “thi”. Figure 6(a) shows the candidate cutting points determined by the discrimination function  $F(k)$ . The first cutting point yields segments P1 and P2 which are sent to the character classifier. The classifier recognizes P1 as a valid ‘c’ while P2 is rejected as an invalid character. P2 is then segmented by the second candidate cutting point yielding P3 and P5. P3 is rejected by the classifier, while P5 is recognized as a valid ‘o’. The segment P3 is then merged with segment P1 and the resulting segment P4 is correctly classified as a valid ‘o’. At this point both the left and

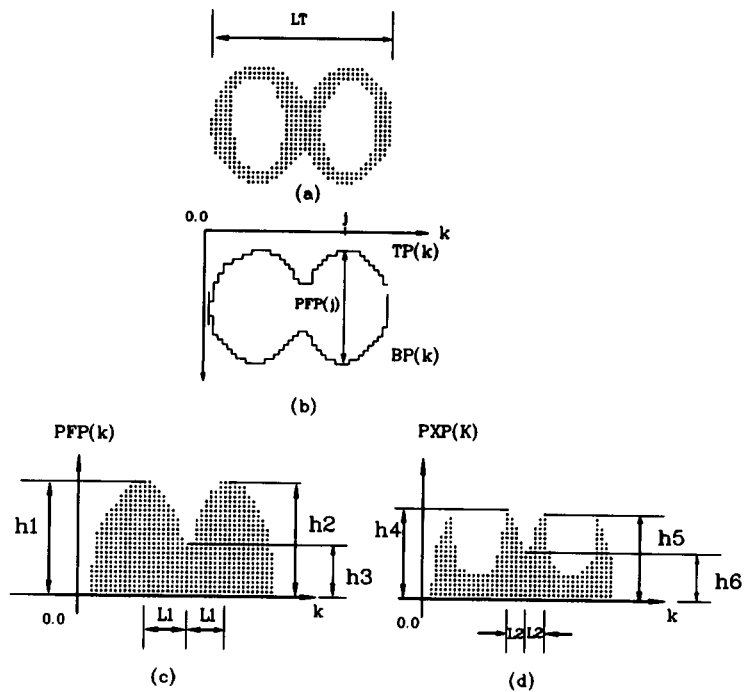


Fig. 3. (a) Touching component 'oo'. (b) Top and bottom profiles. (c) The profile projection  $PFP(k)$ . (d) The pixel projection  $PXP(k)$ .

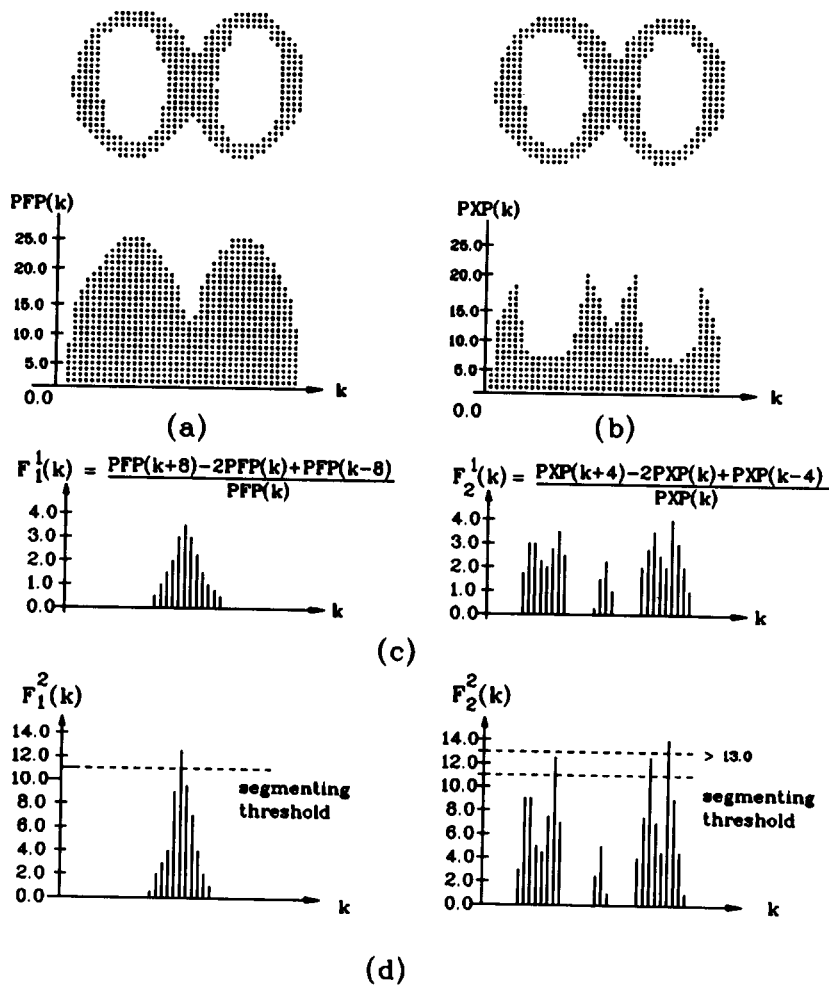


Fig. 4. (a) The double-o touching character and its profile projection. (b) The double-o touching character and its pixel projection. (c) The discrimination function values based on the profile projection and the pixel projection with  $\alpha = 1$ . (d) The discrimination function values based on the profile projection and the pixel projection with  $\alpha = 2$ .

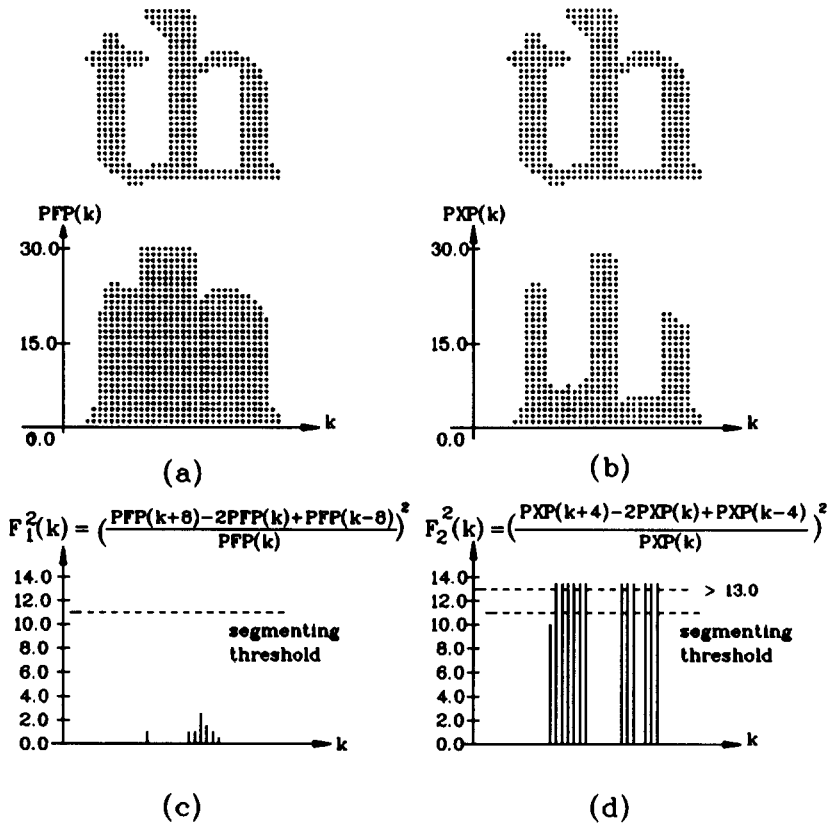


Fig. 5. The serif touching character 'th'. (a) The profile projection. (b) The pixel projection. (c) The discrimination function based on the profile projection. (d) The discrimination function based on the pixel projection.

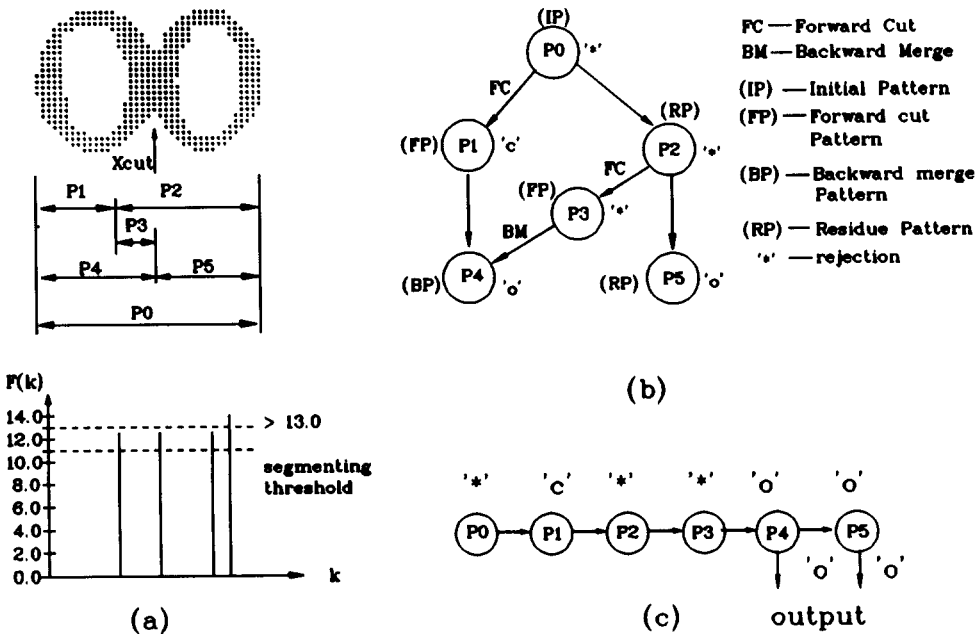


Fig. 6. (a) The segmented patterns. (b) Graphic representation. (c) Time sequence of segmentation.

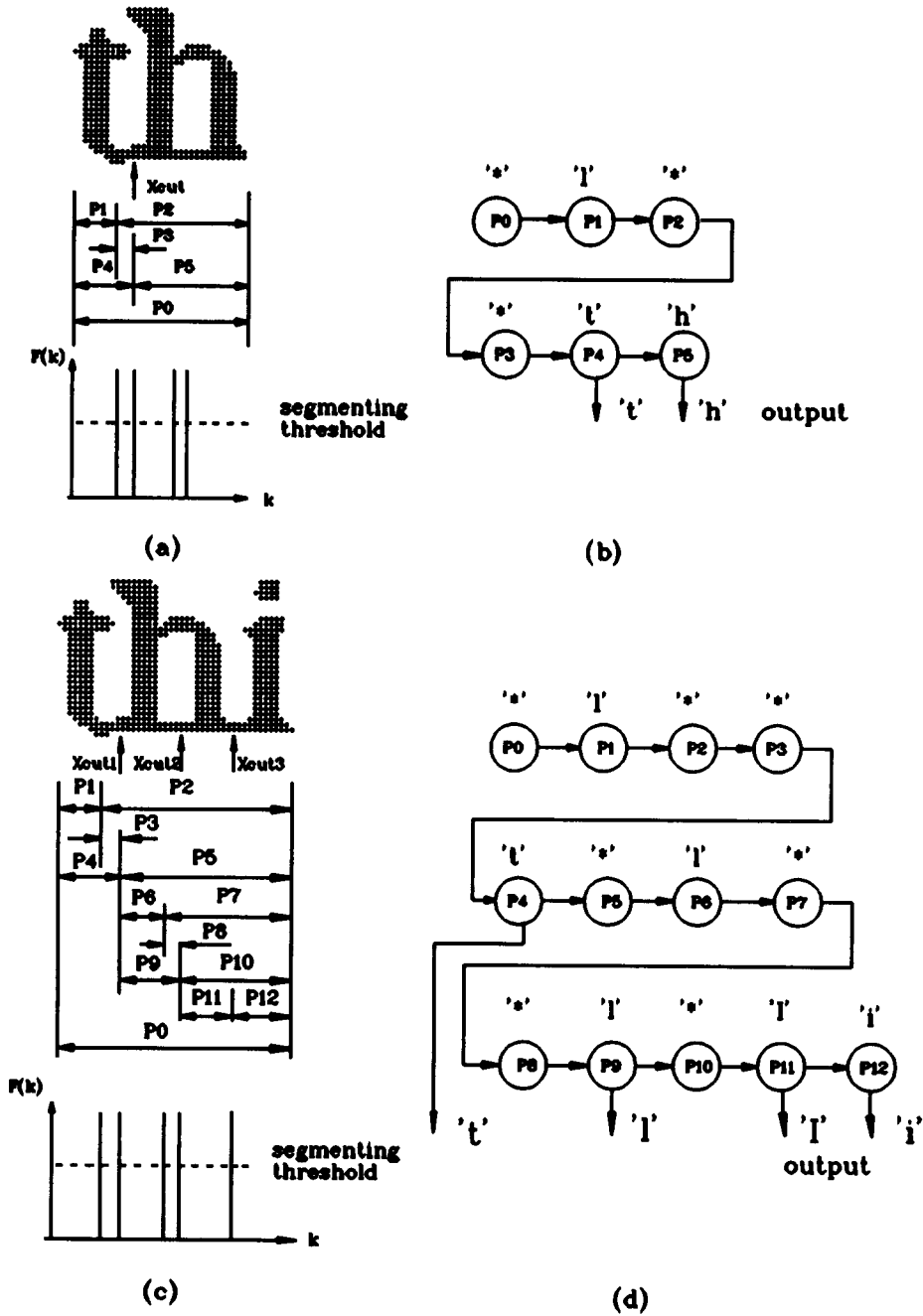


Fig. 7. (a) and (b) The serif touching character 'th' and its time sequence of the segmentation. (c) and (d) The serif touching character 'thi' and its time sequence of the segmentation.

right segments are recognized with high confidence as valid characters ("o" and "o" in this example). Figures 6(a)–(c) illustrate the various stages of this technique.

Figure 5 shows 12 candidate cutting points for splitting component "th" by using the discrimination function based on the pixel projection. In the segmentation process, even though cutting points sometimes deviate from the correct locations, the optimal cutting points  $X_{cut}$  are found after several forward and backward cutting iterations. Figure 7 illustrates the proposed algorithm for two sets of touching characters "th" and

"thi". The first case involving "th" with two possible cutting locations is very similar to the case of "oo" discussed earlier. The second case is interesting and brings out the effectiveness of the proposed algorithm. Due to the figure space limitation, only five candidate cutting points (Figs 7(c) and (d)) are generated by the discrimination function.

(1) Initially the triplet labelled P0 is segmented into two components P1 and P2 of which only P1 is accepted as a valid character "l" by the classifier.

(2) P2 is further segmented into P3 and P5, both of which are rejected by the classifier.

(3) P3 is merged with P1 yielding P4 which is accepted by the classifier as "t".

(4) P5 is further split into P6 and P7 of which only P6 is accepted by the classifier as "l" which P7 is rejected.

(5) P7 is split into two components P8 and P10, both of which are rejected by the classifier.

(6) P8 is merged with P6 yielding P9 that is accepted by the classifier as a valid "l".

(7) P10 is split into P11 and P12, both of which are accepted as valid characters "l" and "i".

At this stage we have a segmentation that yields four characters "t", "l", "l" and "i". It is obvious that this segmentation is erroneous as the correct segments should have been "t", "h" and "i". For touching characters that contain three or more characters, broken characters may appear in the sequence of the recognized characters. The error can only be resolved using contextual classes and other contextual information. This is discussed in later sections. Italic characters in a word were subject to slant correction prior to the application of recursive segmentation.<sup>(10)</sup>

### 3. FEATURE EXTRACTION AND CLASSIFICATION

Even in machine-printed documents, shape discrepancy among characters belonging to the same prototype is sometimes quite large because of the poor quality and low resolution of document images. Particularly, when touching characters are segmented, the

noise blobs near the cutting points overlap both sides of the characters, possibly resulting in a large dissimilarity between the input pattern and the corresponding sample class. Image processing techniques such as border tracing or component labelling are able to remove noise blobs that are not touching the character component; however, it is impossible to remove touching noise blobs by simple techniques. We have developed a new feature, the *Overlap-Neighbour-Direction-Feature*, that is largely based on the border chain code histogram proposed by Kimura and Shridhar.<sup>(11)</sup> They independently calculated a local histogram of the quantized chain codes (0 for "-", 1 for "/", 2 for "\", 4 for "l") in  $4 \times 4$  rectangular zones (Fig. 8) (see reference (11) for detail). The modified feature is then extracted as illustrated in Fig. 9, where there are six overlap regions across the interface of each sub-rectangle; three vertical strips with width  $m_2$ , and three horizontal strips with width  $m_1$ . When a local chain code histogram is calculated, the direction chain codes in the overlap regions contribute their direction values to the histograms of the related local sub-rectangles.

Kimura's feature extraction method of Kimura's created a large distance between the two "d"s shown in Figs 8(a) and (b) because the rightmost vertical stroke of two "d"s occupy the different sub-rectangles. However, the Euclidian distance between the feature vectors of the two input patterns in Figs 9(a) and (b) decreases with the modified feature by properly choosing two parameters,  $m_1$  and  $m_2$ . Our experiments indicate that with the *Overlap-Neighbour-Direction-Feature*, the Euclidian distance between the input patterns and the stored prototypes decreases by approx-

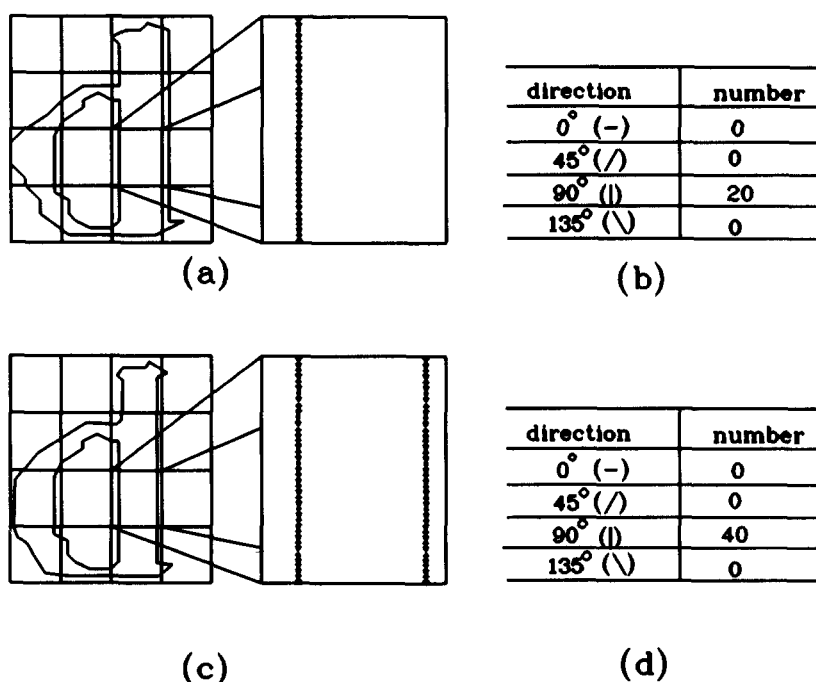


Fig. 8. The direction chain code feature. (a) and (c) Normalized character 'd's and their chain codes in a sub-rectangle. (b) and (d) Direction chain code histogram in a sub-rectangle.

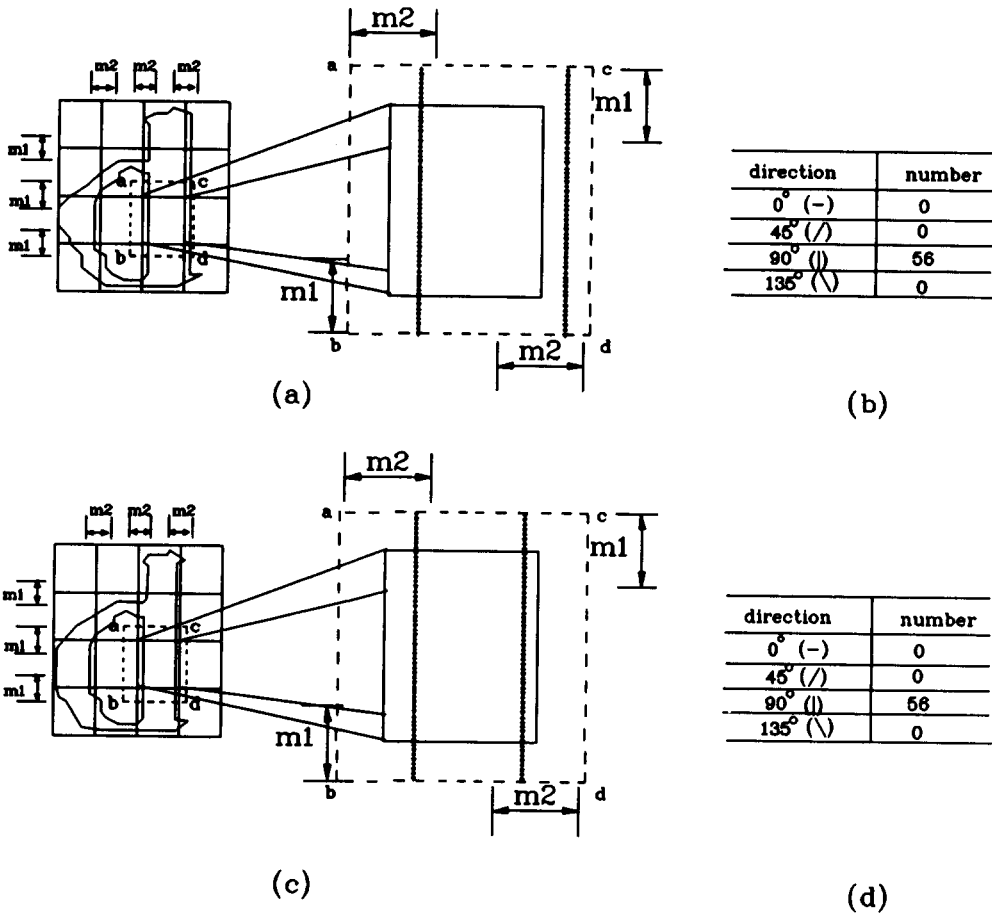


Fig. 9. The Overlap Neighbour Direction feature. (a) and (c) Normalized character 'd's and their chain codes in an extensive sub-rectangle. (b) and (d) The direction chain code histogram in an extensive sub-rectangle.

imately 23% on the average when  $m1 = 10$  and  $m2 = 8$  in the  $64W \times 80H$  pixel unit rectangular frame.

The minimum-distance classifier<sup>(12,13)</sup> is an effective technique for classification problems in which the pattern classes exhibit a reasonably limited degree of variability. For a specific and clear machine printed text, the pattern of each class tends to cluster tightly about a typical or representative pattern for that class. Under these conditions, a minimum-distance classifier can be a very effective approach to the classification problems.

Consider  $M$  pattern classes and assume that they are represented by the mean feature vectors of the training classes, i.e.  $\mu_1, \mu_2, \dots, \mu_m$ . The Euclidian distance between an input feature vector  $X$  and the  $i$ th class is given by

$$D_i = \|X - \mu_i\| = (X - \mu_i)^T (X - \mu_i). \quad (4)$$

In the experiments,  $X$  is a 64-dimensional feature vector, i.e.  $X = (x_1, x_2, \dots, x_{64})^T$ , where the superscript T denotes the transpose. Each component  $x_i$  of  $X$ , for  $i = 1, 2, \dots, 64$ , is composed of a local histogram of overlap-neighbour direction chain codes of a character shape boundary based on  $4 \times 4$  subsquares of a normalized binary pattern. A minimum-distance classifier

based on a feature extraction approach computes the distance from an input pattern to the mean vector of each class and assigns the pattern to the category of the closest mean, i.e.  $X$  is classified to the class  $i$  if  $D_i < D_j$ , for all  $j \neq i$ .

#### 4. WORD ANALYSIS BASED ON CONTEXTUAL KNOWLEDGE

##### 4.1. Character contextual classes

Using contextual information to analyze a recognized word can improve the performance of the text recognition system. Characters occurring in a text line are generally related to one of the classes based on their location with respect to the baselines.<sup>(14)</sup> In Table 1, the classification of character contextual classes is summarized for the fonts used in our application. To improve the flexibility and applicability of our document recognition system, we have made use of size and font information to achieve an accurate contextual class description of the character components in a particular text line. The "t" and "i" are aggregated into a specific contextual class, because they have a smaller number of pixels above the upper baseline than the other Ascender characters in a specific font. This



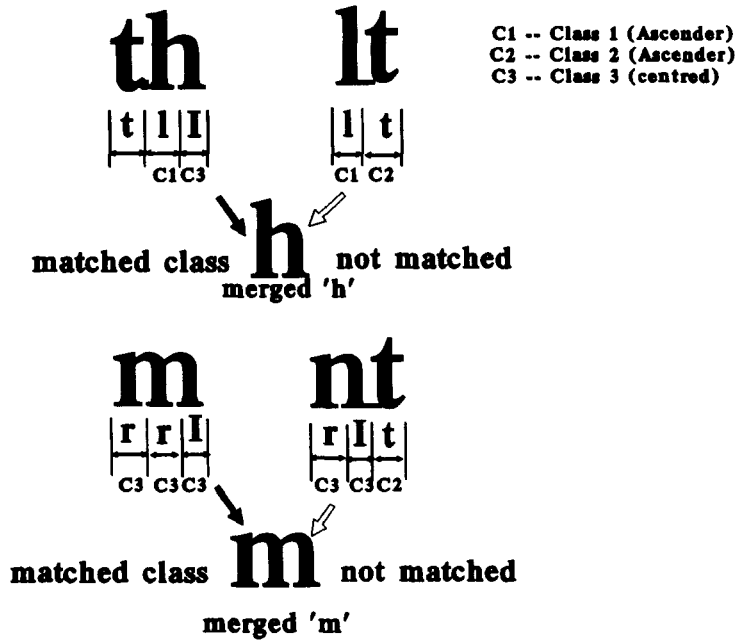


Fig. 10. Merging broken characters with the knowledge of the character contextual classes.

eliminates the ambiguities that easily occur between ("i" "I"), ("i" "I"), ("t" "f") etc. Character contextual classes define character sets as statistically disjoint from the results of classification. The incorrect segmentation achieved earlier for the connected component "thi" can be resolved by using contextual classes. As shown in Fig. 10, the combination "th" which can often be split into three valid characters "t", "l" and "I" can be corrected by recognizing that "l" and "I" do not belong the same contextual class and merging the two segments "l" and "I" into one segment that is accepted by the classifier as a valid "h". Thus using character contextual classes, the component "thi" is correctly split into "t" and "h" and "i".

To determine the contextual features of character components, the baselines of words are required. The most straightforward method to determine these baselines is to find the character locations with respect to

the maximal value of the first-order difference of the pixel projection of the text line image in the horizontal direction.<sup>(5)</sup> Unfortunately, naturally skewed text lines always appear in practical document images, resulting in incorrect character contextual class assignments. Projection of the segmented text lines can be used to relieve this problem (see Fig. 11).

We took advantage of a two-phase detection of baselines. The first phase was based on the baselines extracted from isolated words; failure of this phase prompted the activation of the second phase which extracted the baselines from segmented text lines. These baselines were used to assign a contextual class to an input character component. This two-phase strategy makes the process more accurate and effective.

4.2. Merging broken characters

The dynamic recursive segmentation algorithm described in Section 2 may decompose one character into two or three valid but incorrect character components. Here are some examples:

- an "m" is regarded as "r" and "n" or "r", "r" and "I";
- an "n" is replaced by "r" and "I";
- a "B" is segmented into "I" and "3";
- an "h" becomes "l" and "r";
- a "U" is separated into "I" and "J"; etc.

Therefore, a merge procedure based on the similarity calculation is performed to rejoin the broken characters in the following steps:

- (1) If the characters such as "l", "I", "J", "r", "n", "t", etc. appear adjacently in one word, they are considered to be candidate broken character components.

Table 1. Character contextual classes.

Class	Description	Example
1	Ascender	All capital letters, numerals, and b, d, f, h, k, l
2	Ascender	i, t (not for all fonts)
3	Centred	a, c, e, m, n, o, r, s, u, v, w, x, z
4	Descender	g, p, q, y
5	Full-height	j



Fig. 11. Baseline detection in a segmented text line.

(2) An adjustable window with a maximum size of three character units is moved from left to right across the input word to check if two or three pieces of the character components mentioned in step (1) appear adjacently and simultaneously in the window. If any such character components are detected, a classifier with a specific rejection threshold is applied to the image array corresponding to those detected character components. Any rejection indicates that the input array contains two or more distinct characters; otherwise, the composition of the character components probably is one character.

(3) The merge procedure is not applied if the character components to be merged are not in correct character contextual classes. For instances, the character "m" is only composed of Class 3 (Centred) broken components. The character "h" may contain a Class 1 (Ascender) broken component in the left side and a Class 3 (Centred) broken component in the right side. The touching character "It", even if accepted by the classifier, can be prevented from being merged to "h", because the touching character "It" is composed of one Class 1 (Ascender) component in the left side and one Class 2 (Ascender) component in the right side, violating the merge rule for the character "h" (Fig. 11).

#### 4.3. Character verification

If the output of the classifier do not match to the character contextual classes to which the characters belong, the characters may be corrected according to their contextual classes. For example, both the lower and upper cases of the characters c, o, p, s, u, v, w, x, y, and z can be distinguished by applying contextual class information. Some characters, such as "i" and "j" or "t" and "f", etc. may be misclassified after normalization because of their similarity in shape. The character verification procedure is applied to eliminate these ambiguities. If the classifier outputs for a given pattern does not match the contextual classes to which the pattern should belong, the second character candidate accepted by the classifier with matched contextual class will be considered as a possible substitution. For example, if the input component is recognized as "i" and its contextual class is 5 (i.e. the number of the pixels above the upper baseline and below the lower baseline is greater than a specific threshold), then the "i" will be replaced by an accepted candidate "j" which is a class 5 character (Table 1). Since the character verification process is not absolutely free of error due to baseline deviation, the final replacement is made only after spell checking.

### 5. SPELLING ERRORS AND SPELLING CORRECTION

The dictionary look-up method is the most reliable way to ascertain the character level context.<sup>(16-17)</sup> The main problem with dictionary look-up methods is the large size dictionary (at least 50,000 words) and consequent costs in memory size and searching time re-

quired to handle a realistic vocabulary.<sup>(16)</sup> Because speed is essential, the spelling correction program should be simple and efficient. Fortunately, the UNIX operating system provides a powerful spell command and a "system" or "popen" function in UNIX C that allows one to execute standard UNIX commands in C programs that makes "on line" spelling correction possible. We assumed that the dictionary in the UNIX system included most English words. To avoid making the correction procedure too complicated, we also assumed that spelling errors in a word came from one of three types of errors (substitution, inserted, and merged character errors), the total number of substitution and insert errors in one word did not exceed one, and the number of merged character errors did not exceed two. Generally, words with more than two different types of error cannot be corrected without human intervention. The procedure consists of the two following steps.

#### 5.1. Spelling errors

The words produced by the recognition system after the character verification phase are spell checked by the UNIX spell program. If the given word passes spelling checker test, it is assumed to be a correct word; otherwise, the error correction procedure is employed until a correct word is obtained. If the correction procedure fails to correct the error in the word, the system chooses the first candidate word as the final output. After merging broken characters and verifying characters with contextual class information, we found very few words to be rejected by the spell checker in our case. Only those words absent from the UNIX dictionary were referred to the error correction procedure.

#### 5.2. Spelling correction

*Correction of substitution character errors.* All characters in a word have corresponding sequential candidates. The number of character candidates corresponding to one character component is equal to the number of classes which have accepted Euclidian distances to the input pattern. These candidates are sorted in the order of their Euclidian distance values. The character candidate with the smallest Euclidian distance to the corresponding input pattern is placed at the top of the rating table. The algorithm outputs the first candidate word accepted by the spell checker.

*Correction for merged character errors.* Touching characters which have a similar shape to some isolated characters are frequently recognized as specific characters. These touching characters are even confusing to human beings if they do not appear in a word level. For example, touching "r" and "n" are regarded as "m" and touching "n" and "n" can be interpreted as "n" and "n" or "r" and "m". It is impossible to eliminate these ambiguities without spelling tools. The correction method proposed here is based on the knowledge of

those touching characters. If a word was again rejected by the spell checker after substitution error correction, we assumed that there was probably one or two merged character errors in the word. By using possible character compositions to substitute these merged characters, a series of word candidates were obtained. The first candidate to pass the spell checker became the output word.

*Correction of inserted character errors.* It is very difficult to delete insert errors without some contextual information. Insert errors are always caused by noise components which cannot be eliminated by a noise smoothing filter. Obviously, arbitrarily deleting

characters in word strings is impractical. Generally, inserted character errors occur in situations where the character contextual classes of those inserted components do not match the recognition result of these components. For instance, the pixel numbers of inserted characters are always below the specific threshold for a given character height. Hence, using context information is an important clue to delete the insert characters with a least amount of risk.

## 6. EXPERIMENTAL RESULTS

The documents used in our experiment were based on 12 pages of "NEWS LINE", a University of Windsor

### Teaching, research do mix, Smith says

"Giving professors the opportunity to improve their teaching does not obstruct a university's best researchers," Stuart Smith told a plenary session of the Congress of the Canadian Association of Physicists (CAP) here June 17.

Smith is the author of the 1991 report of the Commission of Inquiry on Canadian University Education. Last November, the university senate established committees to respond to that report.

The audience of physicists received Smith's commentary with some skepticism and Smith found himself defending his recommendations during the question period.

"Society believes the main function of a university is the dissemination of knowledge, but every piece of evidence suggests professors believe universities exist for professors to do research," said Smith, a former head of the Science Council of Canada. "But let's be frank. There is a lot of second-rate research being done by people who would rather be putting their energy into teaching."

He said society has bought into the idea of learning from scholars even though it would be less costly if colleges taught the first two years of university. However, over the past 20 years of shrinking funding for universities, the number of hours professors spent teaching actually went down.

Professors should have the flexibility to chose to be evaluated on their contributions to research or to teaching, according to Smith, who would replace a small percentage of research funding with grants for educational development.

He said Canadian universities have not suffered from funding shortfalls anymore than hospitals, highways and social services. "Universities do not have the dollars they think they should, but who does? We do not have a Harvard but the system is not in crisis."

Smith warned that if universities do not want to be judged arbitrarily by *Maclean's* magazine, they should conduct and publicize their own surveys of student, graduate and employer satisfaction.



*Behind the Scenes: Keeping tabs on progress at the Congress of the Canadian Association of Physicists (CAP) are, from left, Francine Brule, CAP administrator, Nigel Hedgecock, CAP education division head and Windsor physics professor and John Middleton, CAP president and researcher for Atomic Energy of Canada. (CR&P photo)*

### Windsor gives physicists positive charge

Over 400 physicists from across Canada attended the Congress of the Canadian Association of Physicists (CAP) hosted by Windsor's Department of Physics from June 10 to 14.

Nobel Prize-winning physicist Kenneth Wilson opened the congress with a public lecture in which he outlined his views for "jump starting" the entire education system.

"Education is like a car. When the car doesn't start, you know a part is not functioning. With education, I believe the part to repair is the classroom where teachers work in isolation."

The Ohio State University professor, who heads an education-reform initiative called Project Discovery, suggested a repair job involving continuous collaboration and retraining for active teachers and continuous reform and redesign for the education system.

"The developments we enjoy today in communications, transportation and other fields are the result of a continuous process of technical redevelopment. That principle should apply to the education system as well," says Wilson. "What's more, teaching should be collaborative, as it is in Japan."

Papers were presented on research developments, education and the transfer of technology from the laboratory to the workplace.

Department head Mordechai Schlessinger and Professor Gordon Drake, co-chairs of the organization committee, credit the success of the congress to the full participation of the entire department, including Jean Franklin, laboratory manager, who handled registrations.

### Teaching Awards

#### From page one

educational materials, course design and development of innovative teaching methods.

The awards were presented at the 50th Anniversary of the Class of '42 dinner. Also presented was the Alumni Award of Merit to Major-General Richard Rohmer, BA '48, and Justice Carl Zalev, BA '49.

Rohmer was chancellor of the university from 1978 to 1989. He is a prolific author and in 1978 was made a Commander of the Order of Military Merit.

Zalev was appointed senior judge of the District Court of the County of Essex in 1975, has been a member of the advisory board of the clinical law program and was the first Canadian judge to preside at the Annual Advocacy Institute of the Institute of Continuing Legal Education at University of Michigan.

Fig. 12. A sample document.

Teaching\* researcher  
do mix\* Smith says  
''Giving professor the opportunity to  
improve their teaching does not obstruct a  
university's best researchers.'' Stuart  
Smith told a plenary session of the  
Congress of the Canadian Association of  
Physicists (CAP) here June 17,  
Smith is the author of the 1991 report  
of the Commission of Inquiry on Canada  
University Education. Last November, the  
University senate established committees  
to respond to that report,  
The audience of physicists received  
Smith's commentary with some skepti-  
cism and Smith found himself defending  
his recommendations during the question  
period.  
''Society believes the main function of  
a university is the dissemination of  
knowledge, but every piece of evidence  
suggests professors believe universities  
exist for professors to do research.'' said  
Smith, a former head of the science  
Council of Canada. ''But let's be frank.  
There is a lot of second-rate research  
being done by people who would rather be  
putting their energy into teaching.''   
He said society has bought into the  
idea of learning from scholars even though  
it would be less costly if colleges taught  
the first two years of university. However,  
over the past 20 years of shrinking funding  
for universities, the number of hours  
professors spent teaching actually went  
down.

Fig. 13. Results after applying the dynamic recursive segmentation algorithm.

publication which contained 8 pages with 40% touching characters and 4 pages with about 60% touching characters. A typical document image is shown in Fig. 12. The document images were scanned with a 300 dpi scanner connected to an IBM 386. All algorithms in this paper were written in C, and executed on a SUN SPARC II work station. We collected and constructed the character sample classes in five fonts individually drawn from the practical document images. The number of samples in each class ranges from 6 samples per class to 100 samples per class, depending on the characters available in the training documents. The clustering technique "K-Means"<sup>(13)</sup> was used to specify and partition the given data sample sets. After clustering our raw sample data, each sample class set was subdivided into at least one subclass based on a selected threshold. Text, pictures, and graphic blocks were identified, and non-text parts were removed with the block segmentation and text discrimination approaches proposed by Wong *et al.*<sup>(8)</sup> It is conceivable that both segmentation and recognition may be considerably improved by using contextual information. In our case, layout contextual knowledge dealing with baseline information of text lines and the location of the character components with respect to their neigh-

bours was used to assign a specific class to all the character components in the text lines. We successfully merged the broken character components by using the character contextual classes.

A multiple-font system is realized by putting each of the font symbols into a separate class. To obtain high efficiency and minimal computation, font recognition is essential in the recognition system. Different prototypes for each class are constructed for different fonts by tagging corresponding font labels. There is little difficulty in classifying an unknown input pattern and assigning a font label to it at the same time. Since it is not guaranteed that every word contains only one font label, we used the font hypothesis process. The "frequency" of font labels is defined as the ratio of the number of the characters in a word belonging to font *i* to the total number of the characters in the word. It is used to determine mono-font states. The word pattern is recognized as being composed by font *i* if the "frequency" of font *i* in the word is greater than the "frequency" of any other fonts in the word. After the font is identified, the multiple font system is switched to the mono-font state. The mono-font state is not changed until the dissimilarity score, the average Euclidian distance of the characters in one word, is

Teaching\* researcher  
do mix\* Smith says  
''Giving professor the opportunity to  
improve their teaching does not obstruct a  
university's best researchers.'' {s}tuart  
Smith told a plenary session of the  
{c}ongress of the Canadian Association of  
Physicists ({c}AP) here June 17,  
smith is the author of the {l}991 report  
of the Commission of Inquiry on Canadian  
University Education. Last November, the  
university senate established committees  
to respond to that report,  
The audience of phys{l}cists received  
Smith's commentary with some skepti-  
cism and {s}mith found himself defending  
his recommendations during the question  
period.  
''Society believes the main function of  
a university is the dissemination of  
knowledge, but every piece of evidence  
suggests professors believe universi{l}ies  
exist for professors to do research.'' said  
Smith, a former head of the science  
Council of Canada.'' But let's be frank.  
There is a lot of second-rate research  
being done by people who would rather be  
putting their energy into teaching.''   
He said society has bought into the  
idea of lea{m}ing from scholars even though  
it would be less costly if colleges taught  
the f{l}rst two years of university. However,  
over the past 2{0} years of shrinking funding  
for universities, the number of hours  
professors spent teaching actually went  
down.

Fig. 14. Results after merging broken characters.

higher than a specific threshold. The font hypothesis process is then repeated. In this paper, we mainly focus on the segmentation of touching character. Therefore, only five fonts of the text are included in our system.

The approaches proposed for segmenting touching characters utilize multiple techniques. To make our interpretation more clear, we use Figs 13–16 to illustrate the application of various steps of each individual technique. Due to space limitation, only a part of results of the original document image Fig. 12 are illustrated. Figure 13 shows the results of the dynamic recursive segmentation algorithm based on Fig. 12. The curly braces were used to indicate the errors made by segmentation and recognition algorithms. Figure 14 shows the results after word contextual analysis including merging broken characters. Figure 15 shows

the results after character verification. Figure 16 shows the results after spelling correction.

Most touching characters, including the italicized fonts, were successfully separated. The recognition accuracy was 99.85% for the 8 page documents with 40% touching characters and 99.4% for the 4 pages containing about 60% touching characters. The recognition errors did not take into account small punctuations such as “,” and “.”, and those symbols that were not included in our sample sets.

Recognition errors in our system stemmed mainly from the following sources:

- (1) Some substitution errors, merged character errors were not corrected by spelling correction because the correct words were absent from the system dictionary, e.g. names of people or cities.

Table 2. Summary of segmentation and recognition results

Total number of characters	Total number of errors	Substitution errors (40%)	Broken errors (50%)	Merged errors (10%)	Inserted errors (10%)	Average recognition rate
54,000	185	74	92	19	0	99.65%

Teaching\* researcher  
do mix\* Smith says  
''Giving professor the oppourturnty to  
improve their teaching does not obstruct a  
university's best researchers.'' Stuart  
Smith told a plenary session of the  
Congress of the Canadian Association of  
Physicists (CAP) here June 17,  
smith is the author of the {1}991 report  
of the Commission of Inquiry on Canadian  
University Education. Last November, the  
university senate established committees  
to respond to that report,  
The audience of phys{i}cists received  
Smith's commentary with some skepti-  
cism and Smith found himself defending  
his recommendations during the question  
period.  
''Society believes the main function of  
a university is the dissemination of  
knowledge, but every piece of evidence  
suggests professors believe universi{t}ies  
exist for professors to do research.'' said  
Smith, a former head of the science  
Council of Canada. ''But Iet's be frank.  
There is a lot of second-rate research  
being done by people who would rather be  
putting their energy into teacliing.''   
He said society has bought into the  
idea of lea{m}ing from scholars even though  
it would be less costly if colleges taught  
the f{i}rst two years of university. However,  
over the past 2{0} years of shrinking funding  
for universities, the number of hours  
professors spent teaching actually went  
down.

Fig. 15. Result after character verification.

Teaching\* researcher  
do mix\* Smith says  
''Giving professor the opportunity to improve their teaching does not obstruct a university's best researchers.'' Stuart Smith told a plenary session of the Congress of the Canadian Association of Physicists (CAP) here June 17, smith is the author of the {1}991 report of the Commission of Inquiry on Canadian University Education. Last November, the university senate established committees to respond to that report, The audience of physicists received Smith's commentary with some skepticism and Smith found himself defending his recommendations during the question period.  
''Society believes the main function of a university is the dissemination of knowledge, but every piece of evidence suggests professors believe universities exist for professors to do research.'' said Smith, a former head of the science Council of Canada. ''But let's be frank. There is a lot of second-rate research being done by people who would rather be putting their energy into teaching.'' He said society has bought into the idea of learning from scholars even though it would be less costly if colleges taught the first two years of university. However, over the past 2{0} years of shrinking funding for universities, the number of hours professors spent teaching actually went down.

Fig. 16. Result after spelling correction.

(2) Some incorrect words were accepted by the spell checker before or while the correction procedure was applied.

(3) Incorrect words with more than one substitution or two merged character errors were not corrected.

In Table 2, we list the number of errors and the error type distribution occurred in our recognition system. About 30% of the errors were broken errors. They were caused by deviation of the baseline detection and the reject threshold set for merging the broken characters. About 60% of the errors were substitution errors between "l" and "1", "0" and "O", etc. About 10% of the errors came from merged errors that were not corrected by spelling correction because more than one type of error occur in a word.

This is a good result for texts with many touching characters. However, there are many factors that govern the recognition rates. These factors include the quality of the image, image resolution, character font types, and the degree of overlap and touching between two characters.

## 7. CONCLUSION

We have developed an effective segmentation strategy for segmenting a word into its character com-

ponents. With multiple segmentation technique, we have provided robustness to this strategy, and we feel that this technique can be introduced into an overall document recognition system. These segmentation techniques are very efficient and can be applied in real time if necessary.

## REFERENCES

1. Y. Y. Tang, C. Y. Suen, C. D. Yan and M. Cheriet, Document analysis and understanding: a brief survey, *1st Int. Conf. on Document Analysis and Recognition*, Saint-Malo, France, pp. 17-31, October (1991)
2. George Nagy, Teaching a computer to read, *11th IAPR Int. Conf. on Pattern Recognition*, Hague, Netherlands, pp. 225-229, 30 August-3 September (1992).
3. S. Kahan, T. Pavlidis and H. S. Baird, On the recognition of printed characters of any font and size, *IEEE Trans. Pattern Analysis Mach. Intell.* PAMI-9(2), 274-287 (1987).
4. S. Tsujimoto and H. Asada, Resolving ambiguity in segmenting touching characters, *1st Int. Conf. on Document Analysis and Recognition*, Saint-Malo, France, pp. 701-709, October (1991).
5. R. G. Casey and G. Nagy, Recursive segmentation and classification of composite character patterns, *Proc. 6th Int. Conf. Pattern Recognition*, Munich, Germany, pp. 1023-1026 (1982).
6. Chinmoy B. Bose and Shyh-shiaw Kuo, Connected and degraded text recognition using Hidden Markov Model,

- Hague, Netherlands, pp. 116–119, 30 August–3 September (1992).
7. Teruo Akiyama and Norihiro Hagita, Automated entry system for printed documents, *Pattern Recognition* **23**, 1141–1154 (1990).
  8. K. Y. Wong, R. G. Casey and F. M. Wahl, Document analysis system, *IBM J. Res. Dev.* **26**(6), 647–656 (1982).
  9. A. Rosenfeld and A. C. Kak, *Digital Image Processing*, 2nd Edn, pp. 347–349. Addison-Wesley, London (1982).
  10. S. Liang, M. Shridhar and M. Ahmadi, Efficient algorithms for segmentation and recognition of printed characters in document processing, *IEEE Pacific Rim Conf. on Communications, Computers and Signal Processing*, Victoria, British Columbia, Canada, Vol. 1, pp. 240–243, 19–21 May (1993).
  11. F. Kimura and M. Shridhar, Handwritten numerical recognition based on multiple algorithm, *Pattern Recognition* **24**, 969–983 (1991).
  12. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, pp. 26–27. Wiley, New York (1973).
  13. J. T. Tou and R. C. Gonzales, *Pattern Recognition Principles*. Addison-Wesley, London (1974).
  14. P. G. De Luca and A. Gisotti, Printed character preclassification based on word structure, *Pattern Recognition* **24**, 609–615 (1991).
  15. R. M. Bozinovic and S. N. Srihari, Off-line cursive script word recognition, *IEEE Trans. Pattern Analysis Mach. Intell.* **PAMI-11**(1), 68–83 (1989).
  16. C. J. Wells, L. J. Evett, P. E. Whitby and R. J. Whitrow, Fast dictionary look-up for contextual word recognition, *Pattern Recognition* **23**, 501–508 (1990).
  17. H. Takahashi, N. Itoh, T. Amano and A. Yamashita, A spelling correction method and its application to an OCR system, *Pattern Recognition* **23**, 363–377 (1990).

**About the Author**—SU LIANG received his B.Eng. in electrical engineering from Dalian Marine College, Dalian, China, and M.Sc. from the University of Science and Technology of China, Hefei, China, in 1983 and 1988, respectively. He is currently a Ph.D. candidate in electrical engineering at the University of Windsor, Canada. His research interests are in the area of document image processing and pattern recognition.

**About the Author**—M. SHRIDHAR received his M.Sc. in electrical engineering from Polytechnic Institute of Brooklyn, New York, in 1968, and Ph.D. in electrical engineering from the University of Aston in Birmingham, U.K., in 1970. From 1969 to 1985, Dr Shridhar was a faculty member in the Electrical Engineering Department of the University of Windsor, Canada. Since 1986, he has been with the University of Michigan–Dearborn where he is currently a Professor and Chairman of the Electrical and Computer Engineering Department. His research interests are in the area of image processing, pattern recognition and computer vision.

**About the Author**—M. AHMADI received the B.Sc.(EE) degree from Arya Mehr University in Tehran, Iran, and Ph.D. degree from Imperial College of London University, London, U.K., in 1971 and 1977, respectively. He has been with the Department of Electrical Engineering, University of Windsor, Canada, since 1980 and holds the rank of professor. His research interests include design, stability and realization of 2D and *N*-dimensional digital filters; image restoration; pattern recognition and computer vision. He has co-authored the book *Digital Filtering in 1-D and 2-Dimensions, Design and Applications*, published by Plenum, 1989. He has published over 170 articles in the above area. Dr Ahmadi is a Fellow of the Institution of Electrical Engineers in England (FIEE) and is a Senior of IEEE.