The State of the Art in On-Line Handwriting Recognition

CHARLES C. TAPPERT, MEMBER, IEEE, CHING Y. SUEN, FELLOW, IEEE, AND TORU WAKAHARA, MEMBER, IEEE

***Abstract*—This survey describes the state of the art of on-line handwriting recognition during a period of renewed activity in the field. It is based on an extensive review of the literature, including journal articles, conference proceedings, and patents. Shape recognition algorithms, preprocessing and postprocessing techniques, experimental systems, and commercial products are examined.**

***Index Terms*—Natural input to computers, on-line handwriting recognition, real-time character recognition, tablet digitizers.**

I. Introduction

ELECTRONIC tablets accurately capture the x–y coordinate data of pen-tip movement. Their advent in the late 1950's precipitated considerable activity in on-line handwriting recognition. This intense activity lasted through the 1960's, ebbed in the 1970's, and was renewed in the 1980's.

The renewed interest in on-line handwriting recognition stems from a number of factors. Compared to the 1960's, we now have more accurate electronic tablets, more compact and powerful computers, and better recognition algorithms. However, there are additional and perhaps more important reasons. First, the recent hardware advance of combining tablets and flat displays brings input and output into the same surface. This combination permits the use of electronic ink, providing immediate feedback to the writer of the digitized writing. *Electronic ink* is the instantaneous display of the trace of the motion of the stylus tip directly under the stylus. Second, efforts in automating office work have increased interest in more natural methods of entering data into machines. Third, we know more about user-interface design, particularly about issues of useability and user friendliness. Finally, we more clearly understand the applications appropriate for handwriting recognition.

For preparing a first draft and concentrating on content creation, pencil and paper are often favored over the keyboard. Handwriting recognition offers the same advantage. In contrast, for transcription (copying text into the machine), the keyboard is faster than handwriting for small-alphabet languages, like English. Therefore, handwriting recognition is not appropriate here. However, for large-alphabet languages, like Chinese, keyboards are cumbersome. Here, handwriting recognition for transcription is an alternative being intensely investigated, mainly by Japanese and Chinese scientists. Other important uses of handwriting recognition are editing, annotating, and other applications that are heavily interactive and that use direct pointing and manipulation. Tablets are also a powerful tool for input of sketches and drawings since they can accept both writing and graphics.

Other surveys have dealt with handwriting recognition. They described both off-line and on-line handwriting recognition [50], [233], cursive script recognition [162], and the recognition of machine-printed as well as handwritten characters, e.g., [100], [170], [231]. With the renewed activity in on-line handwriting recognition, we believe it is timely to devote a survey solely to this area.

II. On-Line Versus Off-Line Recognition

On-line handwriting recognition means that the machine recognizes the writing while the user writes. The term *real time* or *dynamic* has been used in place of on-line. Depending on the recognition technique and the speed of the computer, the recognition lags behind the writing to a greater or lesser extent. Most commercial character recognizers (see below) lag by only one or two characters. On-line recognition systems need only be fast enough to keep up with the writing. Average writing rates are 1.5–2.5 characters/s for English alphanumerics or 0.2–2.5 characters/s for Chinese characters. Peak rates for English can approach 5–10 characters/s, e.g., a sequence of 1's can be written quickly. For most recognition algorithms, this performance requirement can be met with current microprocessors.

On-line handwriting recognition requires a transducer that captures the writing as it is written. The most common of these devices is the electronic tablet or digitizer, which typically has a resolution of 200 points/in, a sampling rate of 100 points/s, and an indication of "inking" or *pen down*.

Off-line handwriting recognition, by contrast, is performed after the writing is completed. It can be performed

Manuscript received May 15, 1989; revised March 8, 1990. The work of C. Y. Suen was supported by the Natural Sciences and Engineering Research Council of Canada and the Department of Education of Quebec. A preliminary version of this paper was presented at the 9th International Conference on Pattern Recognition, November 1988.

C. C. Tappert is with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598.

C. Y. Suen is with the Concordia Center for Pattern Recognition and Machine Intelligence, Concordia University, Montreal, P.Q. H3G 1M8, Canada.

T. Wakahara is with the NTT Human Interface Laboratories, Kanagawa 238-03, Japan.

IEEE Log Number 9036251.

days, months, or years later. An optical scanner converts the image of the writing into a bit pattern. Scanners have x and y resolutions of typically 300–400 points/in. Off-line handwriting recognition is a subset of *optical character recognition* (OCR). Although most OCR work has been on machine-printed characters, there has been considerable effort on handwriting as well [1], [140], [178], [214], [277]. OCR systems typically process hundreds of characters a second.

Another distinction is between on-line and off-line capture of handwriting data. On-line data capture means that the machine data are being captured as a person writes. Off-line data capture means that the machine data are captured some time after the writing is created. Once they are captured, on-line or off-line handwriting data can be processed by the recognizer afterwards. Although on-line data are recognized immediately in most applications, there may be applications in which the recognition is done more appropriately at a later time.

An advantage of on-line devices is that they capture the temporal or dynamic information of the writing. This information consists of the *number* of strokes, the *order* of the strokes, the *direction* of the writing for each stroke, and the *speed* of the writing within each stroke. A *stroke* is the writing from pen down to pen up. Most on-line transducers capture the trace of the handwriting or line drawing as a sequence of coordinate points. By contrast, off-line conversion of scanner data to line drawings usually requires costly and imperfect preprocessing to extract contours and to thin or skeletonize them [231]. The temporal information provided by on-line entry improves recognition accuracy [7], [169]. In one experiment, on-line data were converted to the form of off-line data to show that on-line is superior to off-line recognition on the same underlying data [169]. Conversely, off-line data have been converted by line thinning to sequences of points similar to on-line data (but without the timing information), achieving reasonable recognition accuracy [79], [90].

The temporal information of on-line systems complicates recognition with variations that are not apparent in the static images. For example, the letter E can be written with one–four strokes (Fig. 1) or with various stroke orders or directions, and many variations can appear the same when completed. Nevertheless, these complications can be handled successfully, and the temporal information can be used to advantage.

Another advantage of on-line handwriting recognition is interactivity. In an editing application, for example, the writing of an editing symbol can cause the display to change appropriately. Also, recognition errors can be corrected immediately.

Yet another advantage is adaptation. When the user sees that some of his characters are not being accurately recognized, he can alter their drawing to improve recognition. Thus, the user adapts to the recognition system. On the other hand, some recognizers are capable of adapting to the writer, usually by storing samples of the writer's characters for subsequent recognition. In this fasion, there is adaptation of writer to machine and of machine to writer.

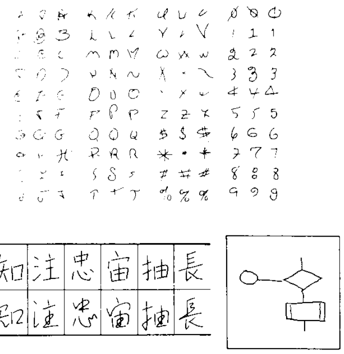


Fig. 1. Examples of handwritten symbols [182], [260], [262].

The main disadvantage of on-line handwriting recognition is that the writer is required to use special equipment. Unfortunately, current on-line equipment is not as comfortable and natural to use as pen and paper [245].

III. Digitizer Technology

Tablet digitizers have existed for three decades. The earliest we found documented was Dimond's "stylator" [54], [55]. The RAND table [49], [68], however, was clearly the most popular of the early digitizers and spurred initial activity in on-line handwriting recognition.

Digitizing tablets can be used for a variety of graphical interaction tasks. Six kinds of tasks have been listed for digitizers and other pointing devices: select, position, orient, path, quantify, and text input [73]. Here, we are concerned with the use of digitizers for the real-time capture of line drawings, such as handwriting, signatures, and flowcharts.

A number of technologies are available for tablet digitizers [89], [145], [265]. At this time, the two main ones are electromagnetic/electrostatic and pressure sensitive. The electromagnetic/electrostatic tablets [32], [33], [198], [205], [208] have x and y grids of conductors, spaced from 0.1 to 0.5 in apart, in the tablet and a loop of wire in the stylus tip. The position of the stylus tip is determined as follows. Either the grid or the loop is excited with an electromagnetic pulse, and the other detects the induced voltage or current in a sinusoidal signal. The tablet conductors are scanned to locate the pair closest to the loop, and interpolation is performed to determine the precise position between these two conductors.

Pressure-sensitive tablets [27], [84], [119], [167], [202], [254] have layers of conductive and resistive material with a mechanical spacing between the layers. An electrical potential is applied across one of the resistive layers, in either the x or y direction, to set up a voltage gradient that corresponds to position. Pressure from the stylus tip at a point results in the conductive layer picking off the voltage (and thus the position) from the resistive layer. The pressure-sensitive technology has the advantage of not requiring the use of a special stylus.

Other technologies include acoustic sensing in an air medium [25], [209], surface acoustic wave [110], [111], triangularization of reflected laser beams [157], and optical sensing of a light pen [72]. There are also special devices that code the presence of the stylus in specific areas or quantized directions of the pen [47], [54], [161].

The most interesting development in recent years has been the combination of input and output—digitizer and display—on the same surface. Transparent tablets have been reported as early as 1968 [249], and are now receiving renewed interest [142], [186], [215]. The latest development is to integrate the digitizer and display in a single hardware unit [238]. Usable digitizer/flat-display systems have been tested only recently [237], [245], [258]. Current tablet systems, opaque as well as transparent, are not easy to use [245], [265], and further improvements will be necessary for them to become acceptable.

The measuring precision of tablet digitizers is characterized by resolution, accuracy, and sampling rate. In order to capture the details of normal writing, the table requirements are stringent. The requirements are a resolution of at least 200 points/in and a sampling rate of at least 100 samples/s [138], [245], [265].

IV. Handwriting Properties and Recognition Problems

Handwriting Properties

A written language has an alphabet of characters (or letters), punctuation symbols, etc. The fundamental property of writing which makes communication possible is that differences between different characters are more significant than differences between different drawings of the same character. Some people argue that there are exceptions to this since *O* and *0* (or *I* and *1*) can be drawn identically, although the context usually provides the information necessary to distinguish letters from numbers. Nevertheless, written communication is not good without this fundamental property.

Handwriting consists of a time sequence of strokes, where a stroke is the writing from pen down to pen up. The characters of writing are usually formed in sequence, one character being completed before beginning the next, and the characters typically follow some spatial order, such as left to right. There are exceptions. In English cursive script, for example, crosses (for *t*'s and *x*'s) and dots (for *i*'s and *j*'s) tend to be delayed. First, the underlying portion of a word is drawn, and then the word is completed by drawing the crosses and dots.

Consider three written languages—English, Chinese, and Japanese. The English alphabet has 26 letters, and each letter has two forms, upper and lower case. English has two basic styles of writing, printing, and cursive script. English words consist of sequences of letters, five per word on the average. Upper case handprinted letters average about two strokes per letter, lower case about one stroke per letter, and cursive writing less than a stroke per letter.

In English, the position and size of the letters is important. Upper case letters sit on the baseline and are full sized. Lower case letters are smaller, and most are about half the height of upper case letters. Some lower case letters have an ascender, which extends upward to almost the height of the upper case letters, some have a descender, which extends down below the baseline, and some have both.

Chinese has a much larger set of characters (alphabet). A Chinese character can represent a word. There are about 50 000 characters, and a basic vocabulary consists of 3–5000 characters [232]. There are two basic styles of writing characters, block and cursive (Fig. 1, lower left). The block style is written carefully, with fairly strict adherence to proper stroke number and order. A character in the block style has an average of 8–10 strokes, the simplest character having one stroke, and the most complicated more than 30 [231]. Chinese characters consist of many strokes because there is a large number of them to be distinguished. The cursive style is written faster and with fewer strokes. This is accomplished by connecting some of the block style strokes and by using simpler radical (subcharacter) shapes.

The Japanese use Hiragana, Katakana, Kanji, and English alphanumerics. Hiragana and Katakana (called Kana) are phonetic alphabets, and each has 46 full-size characters. A small size of eight of the Kana characters together with additional markings indicate subtle phonetic differences. Kanji are Chinese characters, and a set of 6349 is the Japanese Industry Standard, although daily usage is limited to 2000. *Kanji* and *Chinese* characters have essentially the same meaning. We use Kanji when referring to Japanese studies and Chinese otherwise.

Handwritten Chinese characters are usually separated spatially, one from the other; in fact, they are often written in boxes. Handwritten English words are normally separated spatially and are often written on lined paper. Letters within a word, however, are not usually separated spatially. Handprinted letters often touch or overlap, even though written with different strokes, and several cursively written letters can be written with a single stroke.

All characters vary in both their static and dynamic properties. Static variation can occur, for example, in size or shape. Dynamic variation can occur in stroke number and order. English may have more variation in stroke direction than Chinese. English may also vary more in the presence or absence of retraces. A retrace is the overwriting of a stroke, usually done to avoid lifting the pen. The degree of variation depends on the style and speed of writing, with hasty writing usually showing greater variation. Handwriting variability has been studied in general [60], [269], [275], and also from the point of view of handwriting recognition [144], [264]. Handwriting education is also important, and a bibliography of references in this area is available [228].

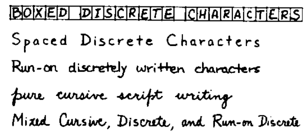


Fig. 2. Types of English writing [240].

Recognition Problems

There are many pattern recognition problems for handwriting and drawing on tablets. They include the recognition of language symbols, equations, line drawings, and gestural symbols, such as those used in editing (Fig. 1). The language symbol recognition problems include, for example, the large alphabet of Chinese characters, Japanese Hiragana and Katakana, Korean Hangul, Arabic, and the writing alphabets and styles of Western languages.

Fig. 2 illustrates the pattern recognition problems for the various writing styles of English. Those of other Western languages are similar. The writing toward the bottom of Fig. 2 is harder to recognize because the letters run together. Separating the letters is called "character segmentation." Discrete characters written in boxes require no character segmentation since the separation of the characters is provided by the writer. Spaced discrete characters require character segmentation. The problems in the lower part of Fig. 2 require advanced segmentation techniques, involving the interaction of character segmentation and recognition. Run-on discrete is easier to handle than cursive writing because a discrete character consists of one or more strokes, and segmentation can occur only after a stroke. For cursive writing, segmentation is necessary within strokes since several characters can be made with one stroke.

Shape discrimination between characters that look alike is difficult for machine recognition. Some characters have similar shapes, such as *U–V*, *C–L*, *a–d*, and *n–h*. Similar shapes also occur between certain characters and numbers, such as *O–0*, *I–1*, *l–1*, *Z–2*, *S–5*, *G–6*. Some of these pairs, such as *O–0* and *I–1*, can be written identically. They can only be distinguished by context. Also, many upper and lower case characters have similar shapes: *C–c*, *K–k*, *O–o*, etc. For most of these pairs, the distinguishing factor is the character size relative to the line spacing or to other character sizes. For others, such as *P–p* and *Y–y*, the distinction depends primarily on the position of the character relative to the baseline.

V. Preprocessing

Preprocessing of handwriting data is done prior to the application of shape recognition algorithms. Besides segmentation, this usually involves cleaning and smoothing strokes. Fig. 3 illustrates typical results of preprocessing.

External Segmentation

External segmentation is the isolation of various *writing units*, such as characters or words, prior to their recognition. Because several letters can be written with one

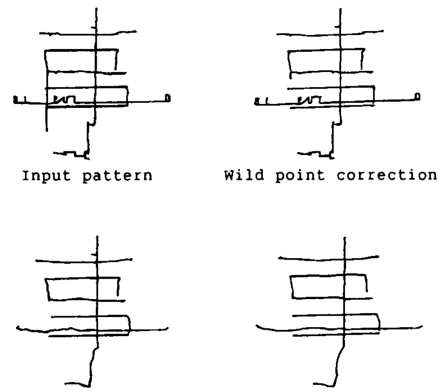


Fig. 3. Example of preprocessing.

stroke in a cursive word, some recognition is usually required for their isolation. Segmentation requiring recognition is called *internal* segmentation. Compared to relying on recognition to provide all segmentation decisions, external segmentation provides greater interactivity, savings of computation, and simplifies the job of the recognizer.

Perhaps the earliest means of segmentation was an explicit signal from the user [24], [93], [114]. Early spatial segmentation used only the x-coordinate information to separate the writing units by their projections on the x axis [82], [86].

Other early work used only temporal information to separate the writing units. When the time difference between the end of a stroke and the beginning of the next exceeds a time-out value, a character is assumed to be completed [51], [86], [94], [166]. For example, Casio markets a calculator watch on which the user can draw one character at a time with his finger, and a time out separates the characters.

For characters written in predefined boxes, the writer does most of the segmentation. Often, a time out allows characters to be replaced or corrected by rewriting. Also, spatial segmentation can handle a stroke that traverses (stretches) from one box to another. Commercial machines recognize boxed characters, and the segmentation is reliable. Segmentation algorithms for boxed discrete characters can be complex because of the opportunity to use the additional information provided by the boxes.

Different regions on a tablet can divide a complex character set into subsets. For example, the tablet area can be divided into boxes, and each box further divided into four subboxes. Then, say, a character written in the upper left subbox is assumed to be alphanumeric, one in the lower right to be Katakana, and one filling the large box to be Kanji [45].

Writing units spaced by the writer can be characters or words. Spacing of characters is not normal in Western languages; in English, many writers, even when asked to do so, have difficulty spacing characters consistently. In contrast, it is more natural for writers to space words when either printing or writing cursively. Recent spatial segmentation techniques check for a two-dimensional separation of the writing units [75], [149], [168]. One recent method combines spatial, temporal, and other information to achieve word segmentation [75]. Finally, knowing the number of characters in a string can enhance character segmentation [74], [224].

Noise Reduction

Many techniques, including algorithms from signal processing, can reduce noise in tablet data. The noise originates from the limiting accuracy of the tablet, the digitizing process, erratic hand motion, and the inaccuracies of pen-down indication.

Smoothing usually averages a point with its neighbors [6], [9], [15], [86], [114], [121], [124], [242]. A common technique averages a point with only previous points, permitting the computation to proceed as each point is received [6], [86].

Filtering, sometimes called thinning (not to be confused with off-line thinning of scanned images), eliminates duplicate data points and reduces the number of points. The form of filtering can depend on the recognition method. One filtering technique forces a minimum distance between consecutive points [6], [9], [15], [86], [94], [114], [121], [242], [282]. This produces points that tend to be equally spaced. When the writing is fast, however, the distance between successive data points may far exceed the minimum distance, and *interpolation* can help to obtain equally spaced points [22], [23].

Another filtering technique forces a minimum change in the direction of the tangent to the drawing for consecutive points [13]. This produces more points in regions of greater curvature.

Smoothing and thinning can be performed in one operation. An example of this is piecewise-linear curve fitting [93], [128].

Wild point correction can replace or eliminate an occasional spurious point, usually caused by a hardware problem. Since acceleration of hand motion is limited by the forces of muscular contraction and the masses of hand and pen [113], high accelerations [242] or velocities (changes in distance) [83], [94], [114], [194], [203] can detect wild points. As illustrated in Fig. 3, wild points (dots) appear as spurious lines in the input pattern since points within strokes are connected.

Dehooking algorithms eliminate hooks that can occur at the beginning, but more frequently at the end of strokes [166], [168], [242], [261]. Hooks are due to inaccuracies in pen-down detection and to rapid or erratic motion in placing the stylus on, or lifting it off, the tablet.

Dot reduction reduces dots to single points [242].

Stroke connection can eliminate extraneous pen lifts. One method connects strokes when the distance between a pen up and subsequent pen down is small relative to the character size [23].

Normalization

Deskewing algorithms correct character slant. Such algorithms can be applied to individual characters [29] or to whole words [22], [23].

Baseline drift correction orients the character or word relative to a baseline or horizontal [22], [23], [135].

Size normalization adjusts the character size to a standard [6], [22], [23], [29], [194], [195]. This process usually also normalizes for location by relocating the origin to the lower left corner or center of the character.

Stroke length normalization forces the number of points of a stroke to a specified number for easy alignment and subsequent classification [6], [58], [160].

VI. Shape Recognition

Shape recognition is the pattern recognition of shapes of writing units. In this section, we present and discuss shape recognition methods for characters, cursive script, words, gestures, equations, line drawings, and signatures.

Character Recognition

Many methods are available for on-line classification of characters. The main ones are described below. Some recognition systems use combinations of these methods.

Some shape-recognition methods rely on prior analysis of the characters of the alphabet. Features (ascenders, cusps, closures, etc.) can be alphabet specific. Sequences of coded zones can also be alphabet specific if the zones are chosen based on properties of the alphabet. Other methods, such as most of the signal processing ones, are essentially independent of the alphabet.

Feature Analysis: A set of features can represent a character. The features might be based on the static properties of the characters, the dynamic properties, or both. The features can be binary—for example, descender or no descender, dot or no dot. With binary features, the name assigned to an unknown character is often determined by a decision tree [78], [94], [95]. For example, for lower case English script letters, the presence of a descender reduces the choices to *f, g, j, p, q, y, z*. Then, if a dot is present, the only choice is *j* (Fig. 4). A disadvantage of this method is that it may not produce alternative character choices, which are usually desirable for postprocessing (see below). Recently, a binary decision tree used simple features to reduce the set of candidate characters to a small set for subsequent analysis by complex features [134].

The features can also be nonbinary. A fixed number of nonbinary features is common in pattern recognition, and many classification methods are available for dividing such a feature space into decision regions. For example, linear-discriminant functions can divide a feature space of Fourier coefficients [80].

Time Sequence of Zones, Directions, or Extremes: These methods rely primarily on dynamic information. A sequence of coded zones can represent a char-

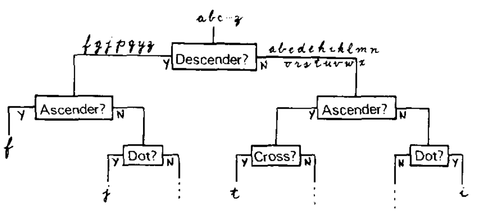


Fig. 4. Decision tree [78].

acter [24], [51], [56], [69], [81], [92], [95], [151], [203], [248]. The zones are specified by dividing up the rectangle that surrounds the written character. The character is superimposed on the rectangle, and the sequence of zones traversed by the pen tip is determined. This sequence, or a corresponding sequence of features, then assigns a name to the unknown character, often by exact match from a dictionary of zone sequences.

A similar method uses the sequence of directions of pen-tip motion during the writing of a character [34], [47], [87], [124], [204]. Using four primitive directions (up, down, left, right), one system coded the first four directions of the sequence and then classified the character by table lookup where the table had 256 ($4 \times 4 \times 4 \times 4$) entries [87]. As the number of directions and time intervals increases, table lookup becomes less practical, and the sequences are compared by curve matching.

Another method describes a character in terms of a sequence of points of local extrema (usually left, right, up, down). Such sequences are called chain codes [46], [47], [264].

Curve Matching: Curve matching is a popular signal-processing method. Curves from an unknown are matched against those of prototype characters, and the name of the prototype that best matches the unknown is assigned to the unknown. The curves matched are usually functions of time, like preprocessed x and y values, the direction angle of the tangent to the trajectory of the writing, or both [120], [121], [126], [127], [132], [193]–[196], [211], [283]. Using a code of eight directions, a character has been divided into ten time regions [160], [286] or six time regions [176]. Since Chinese characters consist mostly of straight strokes, approximating their strokes by a small number of fixed points (three–six) has been found successful [194].

An alternative to the matching of functions of time is the matching of Fourier coefficients obtained from the $x(t)$ and $y(t)$ curves [6], [80], [122], [123]. This method is appropriate when the characters can be represented by a reasonably small number of Fourier coefficients. Since straight-line strokes require high-order Fourier coefficients, this method has been found useful for characters consisting mostly of curved strokes, like the numerals, or of concatenations of many straight strokes, like Chinese characters [6].

Curve matching becomes equivalent to pattern matching in feature space when the number of points characterizing the curves is constant and in one-to-one correspondence [194]. This is a linear alignment of the points of the curve. However, due to nonlinearities, the best fit is usually not a linear matching or alignment. For many sequence comparison problems, *elastic matching* has been successful [143], [212]. Elastic matching has been applied to alphanumerics [58], [168], [240], [242], [246] and to Chinese characters [121], [213], [259], [283]. In one study, elastic matching halved the error rate of linear matching [242]. Fig. 5 illustrates elastic matching for a single-stroke character. Fig. 6 shows, for a three-stroke character, a typical computation region bounded by lines of slope $1/2$ and 2 that represent the limits of compression and expansion in the match. Because elastic matching is computationally intensive, often the prototypes are first pruned to reduce the number of matches [121], [148], [242].

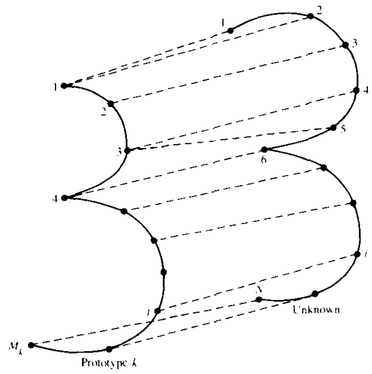


Fig. 5. Illustration of elastic matching [239].

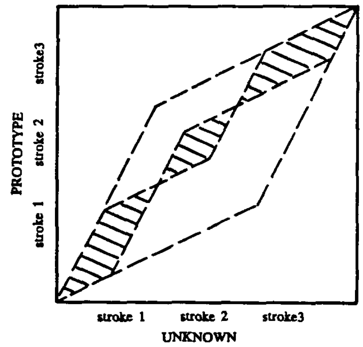


Fig. 6. Computation region for stroke-by-stroke elastic matching [242].

Application of a local affine transformation can enhance the shape discrimination of elastic matching. Using the point correspondence from elastic matching between input and reference patterns, a deformation vector field (DVF) is generated (Fig. 7). Then, DVF is approximated by means of iterative applications of a local affine transformation (LAT). Finally, further elastic matching between the input pattern and the deformed reference pattern superposed by low-order LAT components enhances shape discrimination, halving the error rate [256].

Stroke Codes: The stroke code method (Fig. 8) classifies subparts of a character and then identifies the char-

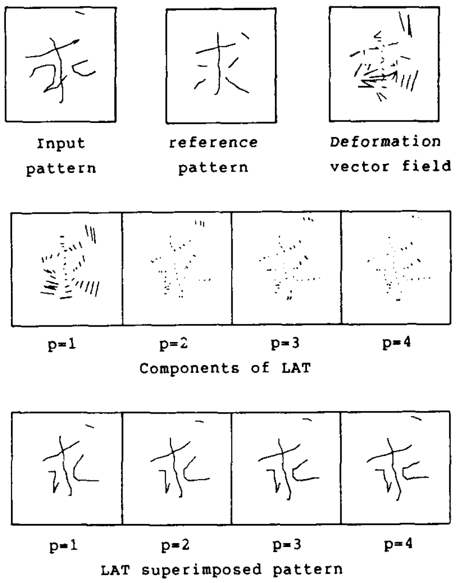


Fig. 7. Example of iterative local affine transformation [256].

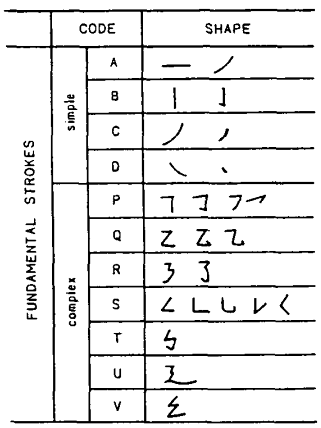


Fig. 8. Example of stroke codes [250].

acter from the sequence of classified subparts [85], [109], [147], [161], [165], [187], [217], [250], [282], [285]. One system uses 76 stroke codes of constituent shapes to specify and recognize more than 3000 Kanji characters [285]. Stroke classification uses the sequence of direction angles. Then, decision trees of stroke code sequences under relative positional constraints on strokes classify the radical or character.

Slope differences have been used to classify curve subparts of characters [192]. Also, direction angles were used as primitives in obtaining a descriptive, generative model [204].

Analysis-by-Synthesis: Yet another approach is analysis-by-synthesis (Fig. 9), sometimes called recognition-by-generation. Several studies concerned the modeling of handwriting generation [52], [53], [62]–[65], [112], [255], [279]. These models usually use strokes (stroke

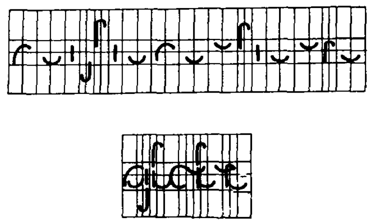


Fig. 9. Illustration of analysis-by-synthesis [63].

segments in our terminology) and rules for connecting them to build symbols. Symbols generated from the inventory of strokes constitute idealized standard representations of the symbols. An approximation to real handwritten symbols can be attained by specifying these strokes with mathematical models that describe the motion of the pen tip as a function of time. Then, a handwritten word can be divided into strokes, the strokes classified using the model parameters, and the letter sequences and words recognized [64], [173], [174], [284].

A similar approach used dynamic programming to match real and modeled strokes [210]. Berson used cross correlation of muscle controls estimated from the Van der Gon model to recognize an unknown against a set of prototypes [10]. Related to the analysis-by-synthesis studies is a theory of handwriting perception in which the dynamic information is inferred from the static form [76].

Pairwise Distinction: Perceptual studies have been instrumental in the development of pairwise distinction methods. Here, a special procedure separates each pair of characters that might be confused. For example, in English, the *C–O* distinction is one of closure, and the *V–Y* difference is one of line extension. Studying the way humans distinguish between such pairs led to a theory of characters based on *functional attributes* [16]–[18], [42], [43], [188], [219]–[221], [266], [267] (Fig. 10). Pair distinction by functional attributes has led to robust recognition methods, notably that in the commercial system by Pencept. Sometimes the same attribute differentiates more than a pair of characters. For example, line extension differentiates *D* and *P*, as well as *V* and *Y*. Pair distinction has been used in other systems [211], [234].

Other Methods: Another method represents a character by the number, order, and relative position of strokes; some strokes are divided into more parts, particularly those of characters with few strokes [132]. The *statistical* method of a Markov model is particularly suitable for dynamic information. For example, a first-order Markov model used eight states corresponding to eight pen-tip directions [71].

Recognition of Character Sequences

For some problems, character segmentation cannot be performed independently of character recognition. One such problem is that of English run-on printing (Fig. 2) where neighboring characters can touch or overlap one another. A similar problem occurs with Chinese characters where a segment may be a character or a portion of a

Functional Attribute	Character	Character Identity (state + +)	Character Identity (state + -)
Closure		O	C
Line Extension		Y	V
Line Addition		E	C
Symmetry of Intersection		A	R
Symmetry (vertical)		O	D
Smoothness		D	B
Verticality		N	Z

Fig. 10. Example of functional attributes [18].

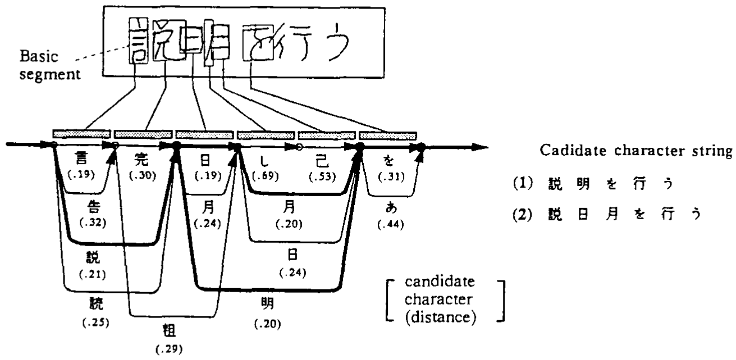


Fig. 11. Example of candidate lattice [181].

larger character. One method of solving this problem is to segment and match all writing units that can be characters, and then to rank the character sequences by their cumulative shape recognition scores [181], [241], [244]. This method is illustrated in Fig. 11. It has also been applied to recognize cursive script and line drawings (see below). Linguistic information can be used to reduce the choices [181], [243].

Recognition of Cursive Script

Cursive script is a common way of daily writing. Cursive script recognition is difficult because several characters can be written with a single stroke. Owing to the difficulty of this problem, there have not been many serious efforts toward obtaining a solution. Furthermore, these efforts have been restricted to lower case English. Early reviews of this work discuss problems, background, and accomplishments [11], [100], [162].

There have been two main philosophical approaches to this problem: a direct analytic approach, and an indirect analysis-by-synthesis approach. Most methods operate on word units. Most of these break a word into subparts. In contrast, the whole-word approach, described in the next section, leaves the words intact and avoids the segmentation problem entirely. The recognition accuracy results of several cursive script recognizers are presented in Section VIII.

A common approach is to analyze a word by stroke segments, where a *stroke segment* is a stroke or portion of a stroke. Then, sequences of stroke segments are used to identify letters. In many studies, a stroke segment is the trajectory resulting essentially from one muscular action. Stroke segments are used in both analytic and analysis-by-synthesis approaches. An early analytic approach used special features to locate stroke segments and a 100-word dictionary [78].

Stroke segmentation at points of minimum velocity was found useful in obtaining substrokes called upstrokes and downstrokes [173]. These strokes were used in an analysis-by-synthesis approach. First, all possible ordered sequences of the stroke segments were examined. Then, the partitioning yielding the largest number of letter identifications was chosen. In this study, practically unique letter specification was obtained from only the downstrokes of the writing. From this and other evidence, it appears that most of the information in cursive writing is in the downward moving portions of the writing, while the upward portions serve mainly as ligatures to join characters.

In another study, strokes were segmented in two stages, first at cusps, and then after the first "down" region of

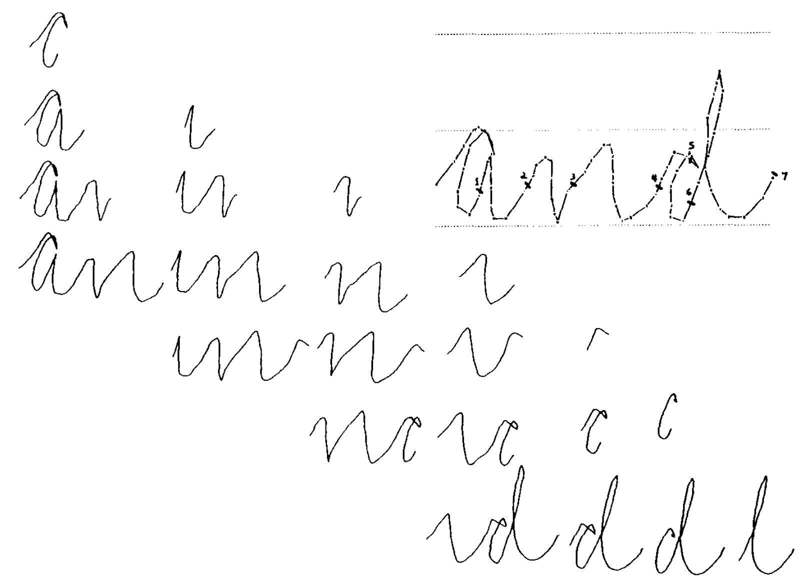


Fig. 12. Example of loose segmentation and units sent to recognizer [243].

an up–down–up–down sequence [177]. The analysis-by-synthesis approach appears most suitable for subletter segments corresponding to single motor actions. Also, see analysis-by-synthesis (above).

Words can also be analyzed on a letter-by-letter basis. In an analytic approach, segmentation was based on an estimate of letter width, and the resulting segments clumped features (cusp, closure, retrograde stroke, etc.) for letter identification by reference to stored features [78], [96], [98]. A syntactic approach employed user training and knowledge of letter formation [12]. Others have also explored the direct analytic approach to the segmentation of cursive letters [19], [105].

Elastic curve matching has also been applied to this problem. One study combined letter segmentation and recognition into one operation by, in essence, evaluating recognition (matching against stored letter prototypes) at all possible segmentations [239]. In another study, loose segmentation cut strokes of cursive writing into substrokes so that cursive characters consisted of one or more substrokes [243]. The cuts were made in regions that could be ligatures, that is, transitions from one character to the next (Fig. 12). An off-line study also used this type of loose segmentation [21]. Subletter prototypes have been used with elastic matching [273]. Elastic matching has also been used to compare an unknown word to hypothesized words, where each hypothesized word is formed by concatenating letter prototypes [35]. Off-line studies have also used hidden Markov models [146], [185], which are also applicable to on-line studies.

Recognition of Words

Although most shape-recognition procedures have been applied to individual characters, they have also been applied directly to whole words, usually cursive words of English [22], [61], [71], [77], [78], [97], [99]. Recognition procedures applied to whole words are usually identical or similar to those applied to characters. To ensure accurate recognition, the number of words can be small. One study used elastic matching with eight direction codes to recognize ten cursively written key words [71]. Such an approach may be useful in an application where the number of words to be recognized is severely restricted.

Recognition of Gestures

The term gesture here refers to hand markings, such as circles, brackets, and arrows, that function to indicate scope and commands. The more usual menu-oriented operations of pointing and selecting are also normally considered as gestures. However, they do not require special procedures for their recognition, and will not be dealt with here. A typical set of gestures is editing gestures.

Some gestures have properties that are different from those of handwritten characters. While most handwritten characters have regular heights and orientations, some gestures do not. For example, the standard proofreader's editing symbol for delete, which is a loop with a beginning and ending tail, can differ in size, rotation, and mirror image. Also, a circular or enclosing scoping gesture can be small for a small scope like a single letter or large for a large scope like a paragraph and can differ in shape. Therefore, new recognition methods are used for non-character-like gestures. One method uses higher order operations on 12 clock-like directions [136], [137]. Another system uses gestures for text and graphics editing, but the gesture recognition technique is not described [236].

Recognition of Equations and Line Drawings

The recognition of equations and line drawings introduces the complexity of operating in two dimensions [3],

[8], [28], [114], [115], [141], [182], [183]. Although this area is outside the primary scope of this survey, one system [182] for recognizing handsketched flowcharts will be described. All subfigures that can be symbols are extracted from the input sketch. Elastic matching distances are calculated between these candidate symbols and prototype symbols. Finally, the system simultaneously recognizes and segments the entie figure by choosing the candidate sequence that minimizes the total sum of distances along the sequence.

Signature Verification

Apart from recognizing the message content of handwriting, there have also been attempts to recognize individual characteristics of the writer. Most of these attempts have been directed toward automatic signature verification. Here, we refer to only a small number of studies on signature verification. Many studies have been based on acceleration or pressure or both.

We describe a typical signature verification system [102]–[104], [164], [274]. This system is based on the notion that signatures are produced by ballistic motions, that is, motions that do not require visual feedback. These motions are naturally produced and difficult to mimic. The forces that generate the ballistic motions can be captured from the accelerations of the pen tip. Segments of the signature are isolated for purposes of aligning the sample and reference signatures. Such alignment is required to account for discrepancies; for example, pen lifts can be present in one signature and not in the other. Pressure and acceleration correlations are computed. The acceleration correlation is made independent of rotation about the pen axis by using complex-pair algebra to combine the x and y acceleration signals. The correlation functions are weighted to control for a number of effects, the most important being a mechanism to penalize for portions of a signature not written. Although a theoretical study indicated that six accelerations were necessary to retain all of the ballistic information for a signature [158], [159], using two as above has yielded respectable signature verification accuracy.

Elastic matching has been applied to signature verification [26], [280]. Special hardware has also been developed [44], [48]. For example, a strain-gauge ballpoint pen has been developed to generate three signals, one corresponding to the downward force, and two to the x and y forces orthogonal in the plane of the paper [44]. Other approaches have been reported [150], and a recent survey on this subject has just appeared [201].

VII. Postprocessing

Postprocessing is processing of the output from shape recognition. Language information can increase the accuracy obtained by pure shape recognition. For handwriting input, some shape recognizers yield a single string of characters, while others yield a number of alternatives for each character, often with a measure of confidence for each alternative. A postprocessor can operate on this information to obtain estimates for larger linguistic units, such as words.

When the shape recognizer yields a single choice for each character, string correction algorithms are applicable [91]. A probabilistic model that allows insertions and deletions as well as substitutions was devised to operate on output from a cursive-script recognizer [20].

Alternate choices provide more information for postprocessing. For some problems, the number of characters in the word is known, as for spaced handwritten characters [59], [66], [155], [156]. For others, the number of characters in the word is part of the estimation problem [67], [241]. This can occur where the letters are not presegmented and several possibilities exist; for example, *d* could be *cl* or vice versa. Several methods produce a list of words in order of decreasing likelihood according to shape recognition scores. Subsequent dictionary lookup can then choose the dictionary entry with the best shape recognition score. Hypothesis generation and test is a common approach [21], [118]. Higher level linguistic rules such as syntax and semantics can also increase the recognition rate.

VIII. Recognition Results

In this section, we present and discuss the capabilities and recognition accuracies of experimental systems reported in the literature. On-line recognition results of handprinted characters up to 1980 were summarized in an earlier survey [233].

It is difficult, if not impossible, to compare the results of the various experimental studies. This is due to the many uncontrolled and incommensurate variables in equipment, procedures, data, writers, and evaluation. Also, these variables are not adequately described by some authors. Hence, the reader should bear this in mind when looking at the results presented in the next section.

Human readers have significant error rates when recognizing characters out of context. For isolated, handprinted block characters of English, error rates of 4% [189] and 1.25% [230] have been reported. For manuscript writing (lower case printing), an error rate of 2.4% was reported [230]. For cursive writing, an error rate of 4.4% was reported [230].

Recognition Results on Characters

Table I compares a number of experimental systems for handprinted characters reported since 1980. Most of the systems were designed for Japanese and Chinese writing [187], [213], [217], [260], [281]–[283], [285]. The other two were designed for Western writing, one for German [168] and one for English [216], [242].

As described earlier, there are basically two styles of writing Chinese characters, block and cursive. For efficiency, it is desirable for the recognizer to be free of constraints on stroke number and order, particularly for the cursive style where significant variation in stroke number and order occurs. Stroke direction, in contrast, tends to be consistent. Two of the systems listed were designed to

TABLE I
EXPERIMENTAL SYSTEMS FOR HANDPRINTED CHARACTERS

Author	Alphabet Size	Alphabet	Stroke number & order freedom	Segmentation	Recognition Method	Rate (percent)	Writer Dependent
Yhap '81 (IBM, USA)	2260	Chinese (square)		Boxes	Stroke code sequence 72 stroke codes	80 96 (with 3 tries)	No
Nakagawa '82 (academia, Japan)	1162	Kana, Kanji (square)		Boxes	Stroke code sequence 29 stroke codes	89 (Kanji) 90 (Katakana)	No
Yoshida '82 (NEC, Japan)	2100	Kana, Kanji (square) Alphanumerics	Number free	Boxes	Elastic match directions	98.3	No
Wakahara '84 (NTT, Japan)	1945	Kanji (cursive)	Number and order free	Boxes	Elastic match x/y coordinates	95.2	No
Ye '84 (acad., Switzerland)	500	Chinese (square)	Order free	Boxes	Stroke code sequence	92 (order free) 99 (constrained)	No
Murase '85 (NTT, Japan)	1991	Hiragana Kanji (square)	Order free	Structure analysis	Elastic match x/y, directions	96.3	No
Sato '85 (academia, Japan)	559	Kana Kanji (cursive)	Number free	Boxes	Elastic match x/y coordinates	97.1	Yes
Yurugi '85 (OKI, Japan)	3280	Kana, Kanji (square) Alphanum., Special	Order free	Boxes	Stroke code sequence 76 stroke codes	95.5	No
Mandler '85 '87 (AEG, Germany)	39	Alphanumerics (German alphabet)		Boxes	Elastic match, dir. Features	96.2	Yes
Tappert '87 (IBM, USA)	44	Alphanumerics Special		Boxes	Elastic match x/y, directions	97.2	Yes
Shiau '88 (ERSO, Taiwan)	5400	Chinese (square)		Boxes	21 stroke codes Features	97.4(no features) 99.4(features)	No

recognize cursively written Kanji, and both use the method of elastic matching [213], [260]. Only one of these is free of constraints on both stroke number and order. For small alphabets, like the alphanumerics, it is not so important for the recognizer to be free of these constraints since the number of occurring variations can be handled reasonably with additional prototypes.

In all but one of the systems, the user performs most of the segmentation by writing the characters in boxes. The other system segments run-on Kanji characters internally by the structural analysis method [184]. First, spatial information decomposes an entire character string into a sequence of elementary patterns, which are similar to or smaller than radicals. Then, candidate characters are generated by combining elementary patterns under structural constraint, and they are matched against prototype characters. Finally, the optimal candidate character sequence is obtained by minimizing the total sum of distances. The obtained sequence is both a recognition and segmentation result.

The recognition methods were stroke codes and elastic matching. Yhap used 72 stroke codes, Nakagawa 29, Yurugi 76, and Shiau 21. The parameters used in elastic matching were the normalized x and y coordinates, the directions, or both. In other studies with stroke codes, Chen *et al.* [36] used 26 stroke codes and Hsu *et al.* [116] used 21 represented by 8 direction codes.

Most of the recognition accuracy rates reported in these studies are over 95%. One might expect the recognition rates on the two small-alphabet systems to be significantly higher than those on the large-alphabet systems since the small alphabets are essentially small subsets of the large alphabets. That this is not the case is likely due to less careful writing used to test those systems. For example, some of the writers used to test the IBM system did not consistently write *O* and *0* or *I* and *1* differently [216].

Tuning the system to the writer can enhance the accuracy of elastic matching. Of the two systems designed to recognize cursive Kanji, this may account for the higher recognition rate for the writer-dependent system. Training the system on the user's writing requires an enrollment period, the length of which is proportional to the size of the alphabet. Therefore, writer-dependent systems are more readily justified for small alphabets because training is faster.

Other systems recognize a wide variety of character sets. These sets range from Fortran and special symbols [9], [14], [87], [120], [225], to Pitman's shorthand [153], [154], and characters of other alphabets [2], [117], [122], [139], [248], such as Arabic, Greek, Korean, and Russian. Many on-line systems have the capability of adding easily to the character set, a desirable feature in some applications.

Recognition Results on Cursive Script

Table II compares several experimental systems for cursive writing. To reduce the difficulty of the problem, only the lower case script letters are recognized by these systems.

The first significant system is that of Harmon [78], [96]. Key features and average character width segment the writing into individual letters for subsequent recognition by feature analysis. Thus, even though several letters are often written by a single stroke, an external segmentation procedure is employed.

The system of Burr [29] constrains the user to write each character with a single stroke. In order to do this, the dots of the *i* and *j* are omitted, and the crosses of *t* and *x* are written with the same stroke used to write the initial portion of the character. Thus, this system does not operate on connected writing, but uses the writer constraint to segment the letters. This study illustrates the power of using a dictionary, increasing character recognition accuracy from 90 to essentially 100%.

The system of Tappert [239] combines character segmentation and recognition into one operation. This is done by using elastic matching and permitting transitions from

TABLE II
EXPERIMENTAL SYSTEMS FOR CURSIVE SCRIPT

Author	Segmentation	Method	Rate (percent)	Writer Dependent
Harmon '61 '62 (BTL, USA)	External	features	60-90	No
Burr '80 (BTL, USA)	Special	directions elastic match dictionary	90 100 (dictionary)	Yes
Tappert '82 (IBM, USA)	Internal	directions, heights elastic match letter digrams	97	Yes
Higgins & Whitrow '84'87 (academia, UK)	Internal	features letter quadgrams dictionary		Yes

The alphabet for these systems was the 26 lower-case script characters.

one character model to another. Thus, segmentation is internal to the recognition process.

The system of Higgins and Whitrow [107], [268] uses hierarchical processing. Beginning with initial estimates of segmentation points, features are extracted for the classification of characters, and then letter quadgrams and a dictionary are used to obtain the word choices.

The full-word recognition approaches cited earlier are also worthy of mention, but are not included in the table. While this approach can be useful for small vocabularies, current thinking is that it is not viable for the general problem.

IX. APPLICATIONS SYSTEMS

Commercial Systems

Tables III and IV describe several commercially available handwriting recognizers. Table III compares systems using opaque tablets, and Table IV compares those using integrated tablet/LCD devices.

Most systems come with a tablet. One reason for this is that the recognition algorithms are usually optimized for a particular tablet. Pressure-sensitive tablets are often used because they are relatively inexpensive and can be used with an ordinary writing instrument. Two systems use transparent tablets to combine input and output (tablet and display) on the same surface. In many systems, recognition is performed by a microprocessor built into the tablet. These systems usually have a small display of one or several lines for displaying the output of the recognizer to the user. In other systems, a recognition board fits into a standard PC, and the screen of the PC is used for displaying the output.

The systems are designed for English or, alternatively, for Japanese or Chinese writing. The alphabets vary greatly. For Chinese, a basic vocabulary consists of 2–4000 characters. The Japanese use Kana, Chinese, and English characters. For English writing systems, the set of special characters, such as punctuation, varies considerably from one system to another. The English alphabet characters are usually upper case. Only the Linus system recognizes lower case.

The newer systems use an integrated tablet/LCD (Table IV). The most interesting of these are Linus and Cannon. While most systems require that characters be written in boxes to provide segmentation prior to recognition, these systems use internal segmentation; that is, character separation relies on recognition. The Linus system is used in hospitals to make handwritten entries on electronic medical charts. The Cannon system uses handwriting and direct pointing to support a calculator, calendar and scheduler, world clock for 130 cities (by pointing on a world map), simple word processor for Japanese, simple business charts, and a handwriting notebook.

The recognition methods vary widely. The dominant methods are feature analysis and stroke code sequences, used by seven of the systems. Since all systems use features, the term feature analysis is not very descriptive, and manufacturers that describe their system in this way are giving away minimal information. Many of the feature-analysis systems likely use stroke code sequences. Three systems use chain codes, three use template matching, and two use elastic matching. The chain codes are of extreme points: left, right, top, and bottom. Template matching simply means matching an unknown character against stored character templates. These template-matching systems match sequences of x/y coordinates, probably in a linear manner. Elastic matching is a form of nonlinear template matching and was described above.

The reported recognition time for these systems varies from 0.1 to over 1 s/character and usually depends on the character to be recognized. Alphanumerics take about 0.5 s/character to write in boxes, and Chinese characters, particularly those with many strokes, take a second or more. Most of these systems are sufficiently fast to keep up with the writing. The NEC system is somewhat slower than the others because of the greater computation required for elastic matching. Recognition accuracy rates, especially those reported in advertising literature, usually exceed 95%. For most systems, these rates can only be achieved with careful writing by cooperative users. Table I contains the rates for the NEC and OKI systems. All these systems are writer independent, but the Panasonic system also trains to a significantly higher recognition rate in a writer-dependent mode.

The purchase price of these recognizers varies from under \$1000 to over \$3000. These prices are subject to change due to market conditions and fluctuations in currency exchange rates.

Present handwriting recognizers provide an alternative interface to existing applications. The main application is that of filling out forms, for example, claim forms in an insurance company. In another application, Pencept replaced the keyboard in a standard computer-aided design (CAD) system with a handwriting-recognition interface [101], eliminating the need to shift attention between tablet and keyboard. In these applications, the handwriting recognizer merely replaces the keyboard by emulation [190], and the applications software does not need modification.

Prototype Systems

Table V describes some recent university and industrial prototype systems. The NTT system [182], [183] recog-

TABLE III
COMMERCIAL HANDWRITING SYSTEMS ON OPAQUE TABLETS

Company Model	Tablet	Alphabet Size	Alphabet	Segmentation	Recognition Method	Reco time (sec/char)	Cost[1]
CIC (USA) The Handwriter	Electromagnetic	3200	Kana, Kanji Alphanum., Special	Boxes 7 x 7mm	Chain codes	0.5-1.0	$3320
Data Entry Systems(USA) Scriptwriter	Pressure sensitive	50	Alphanumerics Special	Boxes 5 x 6mm (min.)	Feature analysis	0.5	$1595
Inforite (USA)	Pressure sensitive	50	Alphanumerics Special	Boxes 4 x 6mm (min.)	Feature analysis	0.5	$1950
NEC (Japan) CR-100	Electromagnetic	2100	Kana, Kanji Alphanumerics	Boxes 10 x 10mm	Elastic match 16 direction codes	1.0-2.0	490,000Y
Nestor (USA)	Electromagnetic	50	Alphanumerics Special	Boxes	Neural network	0.1	$1595
OKI (Japan) SR230	Pressure sensitive	3280	Kana, Kanji Alphanum., Special	Boxes 10 x 10mm	Stroke code sequence 76 stroke codes	0.5	480,000Y
Pencept (USA) Penpad 310, 320	Electromagnetic	110	Alphanum., Special Writable ASCII	Boxes 5 x 5mm (min.)	Chain codes Functional attrib.	0.1	$1495
Personal Writer (USA) PW15S PW15SL	Electromagnetic	150	Alphanumerics Writable ASCII Lower case	Boxes 8 x 12mm (PS15S) Internal (PS15SL)	Chain codes Features	0.1-0.2	$895 $1795
Quest (UK) MPC1031	Pressure sensitive	62	Alphanumerics Special	Boxes 6 x 9mm	Feature analysis	0.5	
Skylight Software (USA)[2] Handprints		45	Alphanumerics Special	Performed by application	Feature analysis	0.2	$800
Tateishi (Japan) AZ-1000	Electromagnetic	3100	Kana, Kanji Alphanum., Special	Boxes 12 x 12mm	Template match x/y and directions	0.2	400,000Y
Toshiba (Japan) PA7704/7704H	Electromagnetic	2500	Kana, Kanji Alphanum., Special	Boxes 5 x 5mm (min.)	Template match x/y coordinates	1.0	400,000Y
Sharp (Japan) CO-8331	Electromagnetic	2300	Kana, Kanji Alphanumerics	Boxes 10 x 10mm	Template match x/y and directions	0.5-1.0	500,000Y
Sharp (Japan) WD-01TG	Presure sensitive	240	Kana, Alphanum., Special	One box 30 x 30mm	Template match x/y and directions	0.2-0.5	34,800Y
Sanyo (Japan) SWP-OLC1	Electromagnetic	2592	Kana, Kanji Alphanum., Special	Boxes 20 x 20mm	Template match x/y coordinates	0.4	360,000Y
Plus (Japan) Word Runner	Pressure sensitive	110	Kana, Alphanumerics	One box 25 x 25mm	Template match x/y coordinates	0.2-0.5	19,800Y

[1] $ = $US, Y = Japanese Yen.
[2] C-language software tool, Mitsubishi GRAFNET-01 tablet ($400 additional), requires IBM PC (not included).

TABLE IV
COMMERCIAL HANDWRITING SYSTEMS ON TABLET/LCD INTEGRATED DEVICE

Company Model	Tablet	Alphabet Size	Alphabet	Segmentation	Recognition Method	Reco time (sec/char)	Cost[1]
Linus (USA)	Transparent electro. over LCD	75	Alphanumerics Lower case, Special	Internal	Elastic match Feature analysis	0.2-0.3	$2995
Panasonic (Japan) Panaword RL-W450	Transparent electro. over LCD	2800	Kana, Kanji Alphanum., Special	One box 45 x 55mm	Stroke code sequence 28 stroke, 8 dir codes	0.4	200,000Y
Cannon (Japan) IN-3000	Transparent pressure sensitive over LCD	123	Katakana, Alphanum., Special	Internal	Stroke code sequence 8 direction codes	0.2-0.5	65,000Y
Seiko Epson (Japan) WORDBank-Pen	Transparent pressure sensitive over LCD	3242	Kana, Kanji, Alphanum., Special	One box 25 x 30mm	Template match x/y coordinates	1.0	24,800Y
Hitachi (Japan) OP511	Transparent electro. over LCD	2400	Kana, Kanji, Alphanumerics	Boxes 20 x 20mm	Template match x/y and directions	0.2-1.0	*

[1] $ = $US, Y = Japanese Yen.
* only available as part of integrated workstation

TABLE V
EXPERIMENTAL APPLICATIONS SYSTEMS FOR HANDWRITING AND DIRECT MANIPULATION

Organization	Application	Alphabet Size	Alphabet
NTT, Japan Murase & Wakahara '83 '86	Recognizing hand-drawn flowcharts	28	Special
NTT, Japan Kojima & Toida '88	Recognizing hand-drawn charts and editing text/figure by handwriting	73	Editing symbols Special
Kagawa & Osaka Universities, Japan Yamasaki, et al. '84	Handwriting trainer for Chinese characters (stroke order, shape)	1000	Chinese
University of Toronto, Canada Lee '83	Direct manipulation interface for an office information system		Alphanumerics Special
AEG, Germany Doster '85, Miletzki, et al. '87	Paper interface project	50	Alphanumerics Special
Ecole Polytechnique de Montreal, Canada Plamondon & Baron '86	Fortran coding by stepwise refinement	53	Alphanumerics Special
IBM, USA Rhyne '86, Wolf '87	Spreadsheet creation and editing by direct manipulation and gesture recognition	50	Alphanumerics Special
Helsinki Univ. of Technology, Finland Kankaanpaa '88	Text editor using standard proof correction symbols	12	Editing symbols

nizes handdrawn symbols used in flowcharts and block diagrams. The system uses 28 symbols, such as those for magnetic drum, disk, and tape. The system elastically matches input and reference symbols, and it requires no indication of symbol segmentation and no restriction on stroke order within a symbol.

The Kagawa and Osaka University system [278] teaches people how to write Chinese characters properly. It gives instructions about the stroke order, outline of the character, and subcharacter shape. Interstroke distances determine the stroke order of the input character relative to that of the reference. Displayed outlines (convex hulls) of the input and reference characters contrast their gross character shapes. The relationship of subpatterns is taught by displaying three figures: the input character, the input character with subpatterns moved to best fit those of the reference character, and the reference character. Stroke shape is taught by displaying the same three characters, but with the poorly drawn strokes reshaped (in bold lines) in the center figure.

Other systems have been designed to teach students how to write Chinese characters [37], [38], [40], [129], [179], [253]. Another system [191] is a trainer for writing cursive English letters in the Palmer method of penmanship, a popular handwriting system taught in elementary schools throughout North America. Fairhurst described the general problem of teaching two-dimensional drawings on a digitizing tablet using pattern matching [70]. In his system, a student tries to copy the drawing of a character made by the machine, and receives a pass or fail grade, depending on how closely the drawn character matches the prototype. An aid for dictionary lookup has also been implemented for Chinese characters [88].

The University of Toronto system [152] explores a direct-manipulation and editing-gesture interface in an office information system. It electronically models a typical paper office, integrating facilities such as electronic mail, file and retrieval, and text processing into a common environment to facilitate access and control of the information.

The AEG system [57], [175] is similar to the Toronto system, but also uses on-line handwriting recognition and OCR to transfer information from paper to machine representation. Printers provide the reverse operation from machine to paper representation. Users can perform paper-like editing of the machine representation. The system also handles mixed-mode documents and includes color. The AEG work is part of the Paper Interface project; other participants include Plessey (Great Britain), Olivetti (Italy), and Philips (Germany).

The system at Ecole Polytechnique de Montreal [199], [200] uses on-line character recognition to enter and edit Fortran. All input to the system is through a graphic tablet. This interface allows the user to focus on the creation of the Fortran program. Apart from the alphanumeric characters, the system uses special symbols to delete, insert, and control loops. Groner's system [87] was perhaps the earliest to recognize handwritten programming code. A similar application is the recognition of Pitman's shorthand into machine-readable text [154].

The IBM system [206], [207], [272] uses on-line character and gesture recognition for creating and editing spreadsheets. A transparent tablet and LCD display provide input and output on the same surface. Electronic inking echos the path traced by the stylus and aids the user while writing. After the user writes a character, the recognized character replaces the inking. After writing a gesture, the system performs the required action, e.g., erase, copy, move, insert, sum a row of numbers. This gestural interface was found significantly faster than the conventional CRT [271]. Related studies have focused on behavioral issues of the user, such as consistency in the use of handdrawn gestures [270], [272].

The Helsinki University of Technology system [130] recognizes editing symbols for text editing. A flat electroluminescent panel and a resistive touch-sensitive tablet provide input and output on the same surface. Documents to be edited can contain both text and pictures.

Other prototype systems perform editing [211], text processing [236], sketch editing [31], [197], computer-aided design [86], [114], analysis of handwritten scenes [171], and editing of musical scores [226].

Advanced uses of handwriting recognition and direct manipulation can permit more efficient operations. This can require either significant changes to, or entirely new, application programs. An example of this is handdrawn gestural interfaces, an area of substantial recent interest. Such interfaces can enhance productivity, for example, by the ability to operate directly on graphical objects by touching them, in contrast to using a command language [223]. The use of handdrawn proofreading symbols for text editing appeared 20 years ago [41]. The recent interest has been in the use of gestural interfaces to highly interactive, visually oriented applications [30]. These interfaces support editing or incremental change and include both text- and image-oriented applications.

These applications indicate that we better understand how to use handwriting recognition in an interactive application [262], [263]. For preparation of a first draft or of programming code, handwriting input via appropriate interface to the machine should permit the user to concentrate on the content creation process. For pure entry of text or data, handwriting recognition may be an alternative to cumbersome keyboards for large-alphabet languages like Chinese. Other appropriate uses for handwriting recognition might be for editing, annotating, filling of forms, working with spreadsheets, and for other applications that rely heavily on interactivity and that use direct pointing and manipulation.

X. Discussion

Understanding Handwriting

Advances in handwriting recognition and our understanding of handwriting go hand in hand. A fundamental step toward the understanding of handwriting is the gathering and analyzing of statistics related to variation in character shape, slant, and stroke number, order, and direction. Insight for analyzing such statistics may come from modeling the process of handwriting generation or from studying the perceptual aspects of handwriting.

Modeling, the handwriting generation process led to recognition methods using analysis-by-synthesis, and perceptual studies led to some pairwise distinction methods (see above). A bibliography contains references to techniques for describing characters from the point of view of cognitive psychology [218].

In recent years, a series of three international conferences on handwriting brought together scientists from such diverse disciplines as computer science, experimental psychology, bioengineering, neurology, and education. The first was the International Workshop on the Motor Aspects of Handwriting, held at the Department of Experimental Psychology, University of Nijmegen, The Netherlands, in 1982 [227], [252]. Next came the Second International Symposium on the Neural and Motor Aspects of Handwriting, held at the Department of Psychology, University of Hong Kong, in 1985 [131]. Finally, the Third International Symposium on Handwriting and Computer Applications was held at Ecole Polytechnique de Montreal in 1987 [125]. Although most are not cited individually here, the papers from these conferences serve as an introduction to a wide range of handwriting studies.

Computers and digitizing equipment are also being used more effectively to study the handwriting process [251].

Robust Algorithms

For on-line as opposed to off-line recognition, there is less emphasis on the development of robust recognition algorithms. There are a number of reasons for this. First, the *quality* of data obtained from most on-line systems is poorer in terms of accuracy and resolution than that of off-line data. Second, feedback to and from the user is possible with on-line systems. Thus, the user can often adapt to the system and achieve higher recognition accuracy as he or she becomes accustomed to it. Furthermore, many recognizers adapt to the user over time. As a result, 95% may be an acceptable initial rate for on-line systems, whereas higher accuracy is indispensable for off-line systems. Third, the temporal information available on-line makes it easier to achieve acceptable recognition accuracy for many applications. Often, simpler recognition methods will suffice. Fourth, on-line recognition is usually implemented on small computers. This is due primarily to the nature of the applications, such as keyboard replacement, where microprocessors are appropriate. Implementation on small systems is also made possible because simple recognition methods suffice (mentioned above) and because the speed of recognition need only keep up with the writing speed. Finally, the *quantity* of data available for the development of on-line recognizers is small relative to that for off-line recognizers. This is because data are collected in real time and require the writer to use special writing equipment.

For on-line recognition systems to become more usable and for commercial products to be successful, we believe that the development of robust recognition algorithms is required. The establishment of databases for on-line handwriting will encourage this development. Databases have long been available for off-line handprinting [108], [180]. On the positive side, on-line systems have gained more sophistication in recent years by exploiting the dynamic information, enabling research on understanding the handwriting process and on the analysis of character shape deformation.

For careful writing on tablets, rather simple recognition methods have been adequate due to the availability of temporal information. Even if the temporal information is unstable for the difficult handwriting problems, such as cursive Chinese characters, on-line recognition will still maintain a distinct edge over off-line methods. However, more robust recognition methods are necessary for less careful writing and for the difficult problems. Since humans do reasonably well in these situations, it may be essential to devise shape-recognition methods without the temporal information. This may lead to the efficient definition of stable shape primitives that describe a variety of handwriting deformation. Also, robust measures of shape similarity and dissimilarity might be developed to improve discrimination ability. Moreover, it is desirable to develop prediction methods of feature or primitive deformation. One such method might be based on the statistical estimation of feature deformation. A deterministic method might be based on a handwriting generation model.

Flexibility of Handwriting Recognizers

Flexibility is an important property of handwriting recognition systems. Writing on a tablet with no constraints can be as varied as writing with pen and paper. At the other extreme, the writing can be limited to characters printed carefully, one to a box. Further, the writing of each character can be constrained to a specific number, order, and direction of strokes. Generally speaking, flexibility is concerned with where and how characters are written. For example, they may be written in boxes to control both vertical and horizontal spacing, on lined paper to control vertical spacing, or on blank paper. They may be written carefully, completely constrained with respect to variation in number, order, and direction of strokes, or there may be no such constraints. Thus, the writing can be highly constrained or as varied as unconstrained writing on blank paper.

We have seen that most commercial systems only accept characters handprinted in boxes. Constraints such as writing in boxes reduce the difficulty of the recognition problem for the machine. However, such constraints can slow the writing speed and result in machines difficult to use, possibly introducing such restrictive constraints that humans reject the process [229], [245].

In order to make handprinting more regular, there has been some work on constrained fonts and possible standards [4], [5], [276]. Some attempt has also been made to measure the quality of a set of handprinted characters [235] and to select a character set suitable for machine recognition [222], [234]. Although most of this work was for handprint OCR methods, standards for on-line drawing of characters might be appropriate for some applications. This work is important for two reasons. First, a high recognition rate is easier to attain with a standard for handprinting. This will facilitate widespread use of on-line recognition systems, especially in areas where accurate handprinting is required, such as for official application forms. Second, degrees of handwriting deformation are more easily defined. This aids a quantitative analysis and understanding of handwriting.

Flexibility with regard to stroke number, order, and direction is important. For alphanumerics, differences in stroke number, order, and direction can be handled by adding prototypes in some systems [240], [241]. For Chinese characters, simply adding prototypes does not seem feasible because of the greater number of possibilities involved. Stroke-order differences have been handled more efficiently by several methods. One method reorders the strokes [106]. Another method calculates all stroke distances, and then chooses the stroke correspondence yielding the smallest sum of the stroke distances [196], [259]. Differences in stroke number have been handled successfully by a stroke linkage method [259].

Learning

Whereas most off-line handprint recognition systems are based on trainable algorithms, earlier on-line systems were not [233]. Today, most on-line systems do use trainable algorithms in order to achieve more robust recognition. For on-line systems, this learning can be either off line or on line. Off-line learning simply means that the recognition system is trained on earlier obtained data that are labeled with the character names. On-line systems also have the capability of learning by taking advantage of feedback from the user to obtain, for example, the name of a character when an error or reject occurs. On-line learning is particularly important when the recognition approach is to tune the system to each individual user. Here, it is usually desirable to have the learning or enrollment period short, with possible additional learning occurring during actual use of the system.

Evaluation of Recognition Methods

It is difficult to assess the value of the many recognition methods [163]. There are perhaps three main categories of work. First, university studies, such as a thesis, are typically small, but deep studies. Second, industrial projects or studies often lead to a prototype system. Third, some industrial projects lead to a commercial product. Although research at any one of these levels may have sound ideas, robust techniques are most likely to be found in successful commercial systems, and next likely in industrial prototype systems. Unfortunately, information about techniques used in commercial systems is often proprietary. Nevertheless, whenever possible, we have described these techniques, and have provided in the reference section a large number of patents granted in this field.

It is critical to establish databases of on-line handwriting. In Japan, off-line databases of handwritten Kanji characters were provided by the Ministry of International Trade and Industry. These databases made it possible to assess many recognition methods, and they contributed substantially toward the development of robust recognition methods.

Anticipated Advances

Future developments in hardware will improve tablet and display technology. Unified tablet/display systems will likely be jointly manufactured by interspersing tablet and display elements. This should essentially eliminate the parallax present with current technology. Large-sized units should also become available.

Advances in computers, such as parallel processors and VLSI architectures, will enable the use of more powerful algorithms. For example, a study has indicated that a VLSI architecture could greatly speed the computation necessary for elastic matching [39].

In the area of software development, more sophisticated recognition algorithms are possible for reasonably natural writing, including cursive script. The development of more sophisticated handwriting analysis procedures will lead to a greater understanding of the handwriting process [257]. For example, with anticipated machine capability, it should be possible to gather statistics relating to such factors as character shape, slant, and stroke variation.

Using a single input device, the stylus, for entering data and commands on an integrated tablet display, is a more direct means for handwriting and gesture input. Such a user interface realizes productivity gains over conventional alternatives for a number of applications. A dynamic medium the size of a notebook was first envisioned by Kay [133], and recently described by Mel *et al.* [172]. Such user-friendly devices that work as ordinary pen, eraser, and paper might be indispensable.

Finally, future interfaces might integrate a variety of input/output technologies. For example, we might interact with machines by voice, handwriting, body movements, and facial expressions. On-line recognition technology will certainly play a major role in handwriting interactions. Here, handwriting will mean characters, shorthand, line figures, and many kinds of gestures.

XI. Glossary of Terms

This section contains an alphabetical list of the definitions of terms used in this paper. We believe that some standardization of the terminology in this area would be helpful.

Electronic ink is the instantaneous display of the trace of the motion of the stylus tip directly under the stylus. In a table-display system, it is the electronic equivalent of normal ink.

Off-line handwriting recognition is performed after the writing is completed. The writing is usually captured by an optical scanning device.

On-line (real-time, dynamic) handwriting recognition is machine recognition of writing as it is being written on a table digitizer.

Optical character recognition (OCR) is the recognition of characters from optical data. The characters may be machine printed or handwritten.

Pen down, for the normal situation of writing with pen on paper, is simply the state in which the pen is "inking," while *pen up* is the noninking state. For electronic tablets, pen down is the electronic equivalent of the state in which inking occurs. For many tablets, a microswitch in the pen tip closes when the pen is in contact with the tablet surface to indicate pen down.

Postprocessing is processing of the output from shape recognition.

Preprocessing is processing of the handwriting data prior to shape recognition.

Segmentation is the machine separation of *writing units* from each other. *External* segmentation does not require recognition, while *internal* does.

Shape recognition is the pattern recognition of shapes of writing units.

A *stroke* consists of the writing from pen down to pen up.

A *stroke segment* is a stroke or portion of a stroke.

Writing units are clearly defined units of writing, such as strokes, characters, and words. Stroke segments also qualify if clearly defined.

Acknowledgment

C. C. Tappert wishes to thank T. Fujisaki and H. A. Ellozy for their support, J. R. Ward for his extensive bibliography, and J. Kim, J. S. Lipscomb, J. R. Ward, and others, who have read the manuscript and offered helpful suggestions. T. Wakahara would like to thank K. Odaka for many fruitful discussions regarding this work.

References

[1] P. Ahmed and C. Y. Suen, "Computer recognition of totally unconstrained handwritten Zip codes," *Int. J. Pattern Recognition Artificial Intell.*, vol. 1, pp.1-15, 1987.

[2] A. Amin, A. Kaced, J. Haton, and R. Mohr, "Hand written Arabic character recognition by the I.R.A.C. sytem," in *Proc. 5th Int. Conf. Pattern Recognition*, 1980, pp. 729-731.

[3] R. H. Anderson, "Syntax-directed recognition of handprinted two-dimensional mathematics," in M. Klerer and J. Reinfelds, Eds., *Interactive Systems for Experimental Applied Mathematics.* New York: Academic, 1968, pp. 436-459.

[4] ANSI, "Character set for hand-printing," Amer. Nat. Standards Inst., Inc., 1974.

[5] R. S. Apsey, "Human factors for constrained handprint for OCR," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 292-296, Apr. 1978.

[6] H. Arakawa, K. Odaka, and I. Masuda, "On-line recognition of handwritten characters—Alphanumerics, Hiragana, Katakana, Kanji," in *Proc. 4th Int. Joint Conf. Pattern Recognition*, Nov. 1978, pp. 810-812.

[7] P. W. Becker and K. A. Nielsen, "Pattern recognition using dynamic pictorial information," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-2, pp. 434-437, July 1972.

[8] A. Belaid and J. P. Haton, "A syntactic approach for handwritten mathematical formula recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 105-111, 1984.

[9] M. I. Bernstein, "A method for recognizing handprinted characters in real-time," in L. N. Kanal, Ed., *Pattern Recognition.* Washington: Thompson, 1968, pp. 109-114.

[10] T. A. Berson, "Dynamic handwriting recognition by computer," Ph.D. dissertation, Univ. London, London, England, 1977.

[11] M. Berthod, "On-line analysis of cursive writing," in C. Y. Suen and R. De Mori, Eds., *Computer Analysis and Perception: Vol. 1—Visual Signals.* Boca Raton, FL: CRC Press, 1982, pp. 55-81.

[12] M. Berthod and S. Ahyan, "On-line cursive script recognition: A structural approach with learning," in *Proc. 5th Int. Conf. Pattern Recognition*, Dec. 1980, pp. 723-725.

[13] M. Berthod and J. P. Maroy, "Morphological features and sequential information in real-time hand-printing recognition," in *Proc. 2nd Int. Joint Conf. Pattern Recognition*, Aug. 1974, pp. 358-362.

[14] —, "Learning in syntactic recognition of symbols drawn on a graphic tablet," *Comput. Graphics Image Processing*, vol. 9, pp. 166-182, 1979.

[15] B. Blesser, "Multistage digital filtering utilizing several criteria," U.S. Patent 4 375 081, Feb. 1983.

[16] B. A. Blesser, T. T. Kuklinski, and R. J. Shillman, "Empirical tests for feature selection based on a psychological theory of character recognition," *Pattern Recognition*, vol. 8, pp. 77-85, 1976.

[17] B. Blesser, R. Shillman, C. Cox, T. Kuklinski, J. Ventura, and M. Eden, "Character recognition based on phenomenological attributes," *Visible Language*, vol. 7, pp. 209-223, 1973.

[18] B. Blesser, R. Shillman, T. Kuklinski, C. Cox, M. Eden, and J. Ventura, "A theoretical approach for character recognition based on phenomenological attributes," in *Proc. 1st Int. Joint Conf. Pattern Recognition*, Oct. 1973, pp. 33-40.

[19] R. E. Bonner, "Segmentation method and apparatus," U.S. Patent 3 344 399, Sept. 1967.

[20] R. Bozinovic and S. N. Shrihari, "A string correction algorithm for cursive script recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 655-663, Nov. 1982.

[21] —, "Off-line cursive script word recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 68-83, Jan. 1989.

[22] M. K. Brown and S. Ganapathy, "Cursive script recognition," in *Proc. Int. Conf. Cybern. and Soc.*, 1980, pp. 47-51.

[23] —, "Preprocessing techniques for cursive script word recognition," *Pattern Recognition*, vol. 16, pp. 447-458, 1983.

[24] R. M. Brown, "On-line computer recognition of handprinted characters," *IEEE Trans. Electron. Comput.*, vol. EC-13, pp. 750-752, Dec. 1964.

[25] P. de Bruyne, "Compact large-area graphic digitizer for personal computers," *IEEE Comput. Graphics Appl.*, pp. 49-53, Dec. 1986.

[26] P. de Bruyne and R. Forre, "Signature verification with elastic image matching," in *Proc. 1986 Int. Carnahan Conf. Security Technol.*, Aug. 1986, pp. 113-118.

[27] D. Buckle and T. D. Strand, "Processing of information," U.S. Patent 4 262 281, Apr. 1981.

[28] D. J. Burr, "Elastic matching of line drawings," in *Proc. 5th Int. Conf. Pattern Recognition*, 1980, pp.223-228.

[29] —, "Designing a handwriting reader," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, pp. 554-559, Sept. 1983.

[30] W. Buxton, "Chunking and phrasing and the design of human-computer dialogs," in *Proc. IFIP World Comput. Congr.*, Sept. 1986.

[31] W. Buxton, E. Fiume, R. Hill, A. Lee, and C. Woo, "Continuous hand-gesture driven input," in *Proc. Graphical Interface'83*, 1983, pp. 191-195.

[32] F. P. Carau and M. A. Tremblay, "Travelling wave digitizer," U.S. Patent 4 255 617, Mar. 1981.

[33] P. C. Carvey, "Electrographic system," U.S. Patent 3 975 592, Aug. 1976.

[34] S. K. Chang and D. H. Lo, "An experimental system for the recognition of handwritten Chinese characters," in *Proc. 1st Int. Symp. Comput. and Chinese Input-Output Syst.*, 1973, pp. 257-267.

[35] C. Chen, G. Lorette, and M. Gaudaire, "Hierarchical primitives matching: A structural method for cursive script recognition," in *Proc. 4th Scandinavian Conf. Image Anal.*, June 1985, pp.59-65.

[36] K.-J. Chen, K.-C. Li, and Y.-L. Chang, "A system for on-line recognition of Chinese characters," *Comput. Processing of Chinese and Oriental Languages*, vol. 3, pp. 309-318, Mar. 1988.

[37] S. Chen, "Computer aided learning of the writing of Chinese characters," Ph.D. dissertation, Washington Univ., St. Louis, MO, 1973.

[38] C. C. Cheng, "Computer-based Chinese teaching program at Illinois," *J. Chinese Language Teachers Ass.*, vol. 8, pp. 75-79, 1973.

[39] H.-D. Cheng and K.-S. Fu, "VLSI architecture for dynamic time-warp recognition of handwritten symbols," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 603-613, June 1986.

[40] H. Y. H. Chuang and S. C. Chen, "Computer aided instruction in Chinese characters," in *Proc. 1st Int. Symp. Comput. and Chinese Input-Output Syst.*, 1973, pp. 599-616.

[41] M. L. Coleman, "Text editing on a graphic display device using hand-drawn proofreader's symbols," in M. Faiman and J. Nievergelt, Ed., *Pertinent Concepts in Computer Graphics.* Urbana, IL: Univ. Illinois Press, 1969, pp. 282-290.

[42] C. H. Cox, III, B. A. Blesser, and M. Eden, "The graphical context of printed characters," *Visible Language*, vol. 12, pp. 428-447, 1978.

[43] C. H. Cox, III, P. Coueignoux, B. Blesser, and M. Eden, "Skeletons: A link between theoretical and physical letter descriptions," *Pattern Recognition*, vol. 15, pp. 11-22, 1982.

[44] H. D. Crane and J. S. Ostrem, "Automatic signature verification using a three-axis force-sensitive pen," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, pp. 329-337, May 1983.

[45] H. D. Crane, J. S. Ostrem, and P. K. Edberg, "Method for distinguishing between complex character sets," U.S. Patent 4 531 231, July 1985.

[46] H. D. Crane and R. E. Savoie, "Handwriting system," U.S. Patent 3 930 229, Dec. 1975.

[47] —, "An on-line data entry system for handprinted characters," *Computer*, vol. 10, pp. 43-50, Mar. 1977.

[48] H. D. Crane, D. E. Wolf, and J. S. Ostrem, "The SRI pen system for automatic signature verification," in *Proc. Symp. NBS Trends and Appl. 1977*, May 1977, pp. 32-39.

[49] M. R. Davis and T. O. Ellis, "The Rand tablet: A man-machine graphical communication device," in *Proc. FJCC*, 1964, pp. 325-331.

[50] R. H. Davis and J. Lyall, "Recognition of handwritten characters—A review," *Image Vision Comput.*, pp. 208-218, Nov. 1986.

[51] A. M. Day, J. R. Parks, and P. J. Pobgee, "One-line written input to computers," in *Machine Perception of Pictures and Patterns.* London, England: Inst. Phys., 1972, pp. 233-240.

[52] J. J. Denier van der Gon, J. P. Thuring, and J. Strackee, "A handwriting simulator," *Phys. Med. Biol.*, vol. 6, pp. 407-414, 1962.

[53] J. J. Denier van der Gon and J. P. Thuring, "The guiding of human writing movements," *Kybernetik*, vol. 2, pp. 145-148, Feb. 1965.

[54] T. L. Dimond, "Devices for reading handwritten characters," in *Proc. Eastern Joint Comput. Conf.*, Dec. 1957, pp. 232-237.

[55] —, "Machine reading of handwritten characters," U.S. Patent 3 108 254, Oct. 1963.

[56] A. V. Donahey, "Character recognition system and method," U.S. Patent 3 996 557, Dec. 1976.

[57] W. Doster, "Pattern recognition techniques as a stimulus for advanced text processing," in *Proc. 2nd Int. Conf. Text Processing Syst.*, 1985, pp. 155-161.

[58] W. Doster and R. Oed, "Word processing with on-line script recognition," *IEEE MICRO*, vol. 4, pp. 36-43, Oct. 1984.

[59] R. O. Duda and P. E. Hart, "Experiments in the recognition of hand-printed text: Part II—Context analysis," in *Proc. FJCC*, 1968, pp. 1139-1149.

[60] J. Duvernoy and D. Charraut, "Stability and stationarity of cursive handwriting," *Pattern Recognition*, vol. 11, pp. 145-154, 1979.

[61] L. D. Earnest, "Machine recognition of cursive writing," in C. Cherry, Ed., *Information Processing.* London, England: Butterworths, 1962, pp. 462-466.

[62] M. Eden, "On the formalization of handwriting," in C. Cherry, Ed., *Information Theory (4th London Symp.).* London, England: Butterworths, 1961.

[63] —, "Handwriting and pattern recognition," *IRE Trans. Inform. Theory*, vol. IT-8, 1962.

[64] —, "Handwriting generation and recognition," in P. A. Kolers and M. Eden, Ed., *Recognizing Patterns.* Cambridge, MA: M.I.T. Press, 1968, pp. 138-154.

[65] M. Eden and M. Halle, "The characterization of cursive writing," in C. Cherry, Ed., *Information Theory (4th London Symp.).* London, England: Butterworths, 1961, pp. 287-299.

[66] R. W. Ehrich, "A contextual post-processor for cursive script recognition—Summary," in *Proc. 1st Int. Joint Conf. Pattern Recognition*, Oct. 1973, pp. 169-171.

[67] R. W. Ehrich and K. J. Koehler, "Experiments in the contextual recognition of cursive script," *IEEE Trans. Comput.*, vol. C-24, pp. 182-194, Feb. 1975.

[68] T. O. Ellis, P. V. Estates, and M. R. Davis, "Digital computer and graphic input system," U.S. Patent 3 399 401, Aug. 1968.

[69] J. Engdahl, "Data entry and decoding system for scripted data," U.S. Patent 4 005 400, Jan. 1977.

[70] M. C. Fairhurst, "Image characteristics as assessment criteria for an electronic writing aid," in *Proc. Int. Conf. Man-Machine Syst.*, July 1982, pp. 191-195.

[71] R. F. Farag, "Word-level recognition of cursive script," *IEEE Trans. Comput.*, vol. C-28, pp. 172-175, Feb. 1979.

[72] O. H. Fernald, "Optical graphic data tablet," U.S. Patent 3 761 877, Sept. 1973.

[73] J. D. Foley, V. L. Wallace, and P. Chan, "The human factors of computer graphics interaction techniques," *IEEE Comput. Graphics Appl.*, vol. 4, pp. 13-48, Nov. 1984.

[74] A. S. Fox, J. Kim, and C. C. Tappert, "Segmenter for known number of characters," *IBM Tech. Disclosure Bull.*, vol. 27, pp. 3691-3693, Dec. 1984.

[75] A. S. Fox and C. C. Tappert, "On-line external word segmentation for handwriting recognition," in *Proc. 3rd Int. Symp. Handwriting Comput. Appl.*, July 1987.

[76] J. J. Freyd, "Representing the dynamics of a static form," *Memory and Cognition*, vol. 11, pp. 342-346, 1983.

[77] L. S. Frishkopf, "Automatic recognition of handwriting," U.S. Patent 3 133 266, May 1964.

[78] L. S. Frishkopf and L. D. Harmon, "Machine reading of cursive script," in C. Cherry, Ed., *Information Theory (4th London Symp.).* London, England: Butterworths, 1961, pp. 300-316.

[79] Y. Fujimoto, S. Kadota, S. Hayashi, M. Yamamoto, S. Yajima, and M. Yasuda, "Recognition of handprinted characters by nonlinear elastic matching," in *Proc. 3rd Int. Joint Conf. Pattern Recognition*, Nov. 1976, pp. 113-118.

[80] H. Fujisaki, S. Nagai, and N. Hidaka, "On-line recognition of handwritten numerals," *Annu. Rep.*, Eng. Res. Inst., Faculty Eng., Univ. Tokyo, Japan, vol. 30, pp. 103-110, Aug. 1971.

[81] H. L. Funk and S. F. Kambic, "Handwritten character recognition apparatus," U.S. Patent 3 500 323, Mar. 1970.

[82] G. Gaillat, "An on-line recognizer with learning capabilities," in *Proc. 2nd Int. Joint Conf. Pattern Recognition*, Aug. 1974, pp. 305-306.

[83] B. R. Gaines, I. D. McKellar, W. P. Dinger, S. R. Fast, B. J. Fowles, M. A. Fraccaro, G. C. Jolivet, and A. B. Maludzinski, "Some experience in the real-time processing of handwriting," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 630-632.

[84] W. Gibson and J. Talmage, "Nonplanar transparent electrographic sensor," U.S. Patent 4 220 815, Sept. 1980.

[85] E. C. Greanias and E. F. Yhap, "Chinese/Kanji on-line recognition system," U.S. Patent 4 365 235, Dec. 1982.

[86] G. F. Groner, "Real-time recognition of handprinted text," in *Proc. FJCC*, 1966, pp. 591-601.

[87] —, "Real-time recognition of handprinted symbols," in L. N. Kanal, Ed., *Pattern Recognition.* Washington, DC: Thompson, 1968, pp. 103-108.

[88] G. F. Groner, J. F. Heafner, and T. W. Robinson, "On-line computer classification of handprinted Chinese characters as a translation aid," *IEEE Trans. Electron. Comput.*, vol. EC-16, pp. 856-860, Dec. 1967.

[89] D. J. Grover, "Graphics tablets—A review," *Displays* (IPC Business Press, England), pp. 83-93, July 1979.

[90] S. A. Guberman and V. V. Rozentsveig, "Algorithm for the recognition of handwritten text," *Avtomatika i Telemekanika*, vol. 5, pp. 122-129, May 1976.

[91] P. A. V. Hall and G. R. Dowling, "Approximate string matching," *Comput. Surveys*, vol. 12, pp. 381-402, 1980.

[92] R. E. Hall and L. N. Hulbert, "Machine recognition of symbols," U.S. Patent 3 676 848, July 1972.

[93] S. Hanaki, K. Asai, and K. Kiji, "Real time recognition of handprinted characters—Japanese character 'Katakana' recognition," in *Proc.2nd Hawaii Int. Conf. Syst. Sci.*, Jan. 1969, pp. 345-348.

[94] S. Hanaki, T. Temma, and H. Yoshida, "An on-line character recognition aimed at a substitution for a billing machine keyboard," *Pattern Recognition*, vol. 8, pp. 63-71, 1976.

[95] S. Hanaki and T. Yamazaki, "On-line recognition of handprinted Kanji characters," *Pattern Recognition*, vol. 12, pp. 421-429, 1980.

[96] L. D. Harmon, "Automatic reading of cursive script," in G. L. Fischer Jr., D. K. Pollock, B. Raddack, and M. E. Stevens, Ed., *Optical Character Recognition.* Washington, DC: Spartan, 1962, pp. 151-152(A).

[97] —, "Handwriting reader recognizes whole words," *Electronics*, vol. 35, pp. 29-31, Aug. 1962.

[98] —, "Method and apparatus for reading cursive script," U.S. Patent 3 111 646, Nov. 1963.

[99] —, "Automatic reading of cursive script," U.S. Patent 3 127 588, Mar. 1964.

[100] —, "Automatic recognition of print and script," *Proc. IEEE*, vol. 60, pp. 1165-1176, Oct. 1972.

[101] G. Hart, "Pencept Penpad with Pencad," *PC Mag.*, vol. 5, pp. 152-153, Mar. 1986.
[102] N. M. Herbst and C. N. Liu, "Automatic signature verification based on accelerometry," *IBM J. Res. Develop.*, vol. 21, pp. 245-253, May 1977.
[103] —, "Automatic verification of signatures by means of acceleration patterns," in *Proc. IEEE Comput. Soc. Conf. Pattern Recognition Image Processing*, June 1977, pp. 331-336.
[104] —, "Automatic signature verification," in C. Y. Suen and R. De Mori, Eds., *Computer Analysis and Perception: Vol. 1—Visual Signals.* Boca Raton, FL: CRC Press, 1982, pp. 83-105.
[105] N. M. Herbst and J. H. Morrissey, "Segmentation mechanism for cursive script character recognition systems," Patent 4 024 500, May 1977.
[106] Y. Hidai, K. Ooi, and Y. Nakamura, "Stroke re-ordering algorithm for on-line hand-written character recognition," in *Proc. 8th Int. Conf. Pattern Recognition*, Oct. 1986, pp. 934-936.
[107] C. A. Higgins and R. Whitrow, "On-line cursive script recognition," in *Proc. Interact'84, 1st IFIP Conf. Human-Comput. Interaction*, Sept. 1984, pp. 140-144.
[108] W. H. Highleyman, "Data for character recognition studies," *IEEE Trans. Electron. Comput.*, vol. EC-12, pp. 135-136, Apr. 1963.
[109] R. Hing-Hua, "A practical recognition system for inputting handwritten Chinese characters on-line," in *Proc. 1988 Int. Conf. Comput. Processing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 62-64.
[110] A. M. Hlady, "A touch sensitive *xy* position encoder for computer input," in *Proc. FJCC*, vol. 35, 1969, pp. 545-551.
[111] —, "Touch sensitive position encoder using a layered sheet," U.S. Patent 3 916 099, Oct. 1975.
[112] J. M. Hollerbach, "A study of motor control through analysis and synthesis of handwriting," *Visible Language*, vol. 13, pp. 252-264, 1979.
[113] —, "Understanding manipulator control by synthesizing human handwriting," in P. H. Winston, Ed., *Artificial Intelligence: An MIT Perspective*, vol. 2, pp. 311-332, 1979.
[114] M. Hosaka and F. Kimura, "An interactive geometrical design system with handwriting input," in *Proc. 6th IFIPS Congr.*, Aug. 1977, pp. 167-172.
[115] —, "Use of handwriting action in construction of models," in H. Inose, Ed., *Scientific Information Systems in Japan.* Amsterdam, The Netherlands: North-Holland, 1981, pp. 83-90.
[116] C.-C. Hsu, G. Y. Tang, and Y. T. Tsern, "A syntactic-semantic approach to recognize handwritten Chinese characters by using a digitizing tablet as the input device," *Comput. Processing of Chinese and Oriental Languages*, vol. 2, pp. 198-215, Oct. 1986.
[117] Y. K. Huh and H. L. Beus, "On-line recognition of hand-printed Korean characters," *Pattern Recognition*, vol. 15, pp. 445-453, 1982.
[118] J. J. Hull and S. N. Srihari, "A computational approach to word shape recognition: Hypothesis generation and testing," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition*, June 1986, pp. 156-161.
[119] G. S. Hurst and W. C. Colwell, Jr., "Discriminating contact sensor," U.S. Patent 3 911 215, Oct. 1975.
[120] T. Ichikawa and J. Yoshida, "On-line recognition of handprinted characters with associative read-out of patterns in a memory," in *Proc. 2nd Int. Joint Conf. Pattern Recognition*, Aug. 1974, pp. 206-207.
[121] K. Ikeda, T. Yamamura, Y. Mitamura, S. Fujiwara, Y. Tominaga, and T. Kiyono, "On-line recognition of hand-written characters utilizing positional and stroke vector sequences," in *Proc. 4th Int. Joint Conf. Pattern Recognition*, Nov. 1978, pp. 813-815.
[122] S. Impedovo, "Plane curve classification through Fourier descriptors: 'An application to Arabic hand-written numeral recognition,'" in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1069-1072.
[123] S. Impedovo, B. Marangelli, and A. M. Fanelli, "A Fourier descriptor set for recognizing nonstylized numerals," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 640-645, Aug. 1978.
[124] S. Impedovo, B. Marangelli, and V. L. Plantamura, "Real-time recognition of handwritten numerals," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, pp. 145-148, Feb. 1976.
[125] ISHCA, *Proc. 3rd Int. Symp. Handwriting and Comput. Appl.*, Ecole Polytechnique de Montreal, Montreal, P.Q., Canada, 1987.
[126] K. Ishigaki and T. Morishita, "A top-down online handwritten character recognition method via the denotation of variation," in *Proc. 1988 Int. Conf. Comput. Procesing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 141-145.
[127] Y. Ishii, "Stroke order free online handwritten Kanji character recognition method by means of stroke representation points," *Trans. Inst. Electron. Commun. Eng. Japan*, pp. 1069-1072, 1986.
[128] M. R. Ito and T. L. Chui, "On-line computer recognition of proposed standard ANSI (USASI) handprinted characters," *Pattern Recognition*, vol. 10, pp. 341-349, 1978.
[129] H. Kamada, Y. Yamaguchi, Y. Kijima, and T. Fujita, "A CAL system based on Kanji recognition for learning Japanese," in *Proc. 3rd Int. Symp. Handwriting Comput. Appl.*, July 1987, pp. 188-190.
[130] A. Kankaanpaa, "FIDS—A flat-panel interactive display system," *IEEE Comput. Graphics Appl.*, pp. 71-82, Mar. 1988.
[131] H. S. R. Kao, G. P. Van Galen, and R. Hoosain, Eds., *Graphonomics: Contemporary Research in Handwriting.* Amsterdam: North-Holland, 1986.
[132] O. Kato, T. Fujita, M. Niwa, T. Morishita, N. Fujii, and J. Tanahashi, "A handwriting input system for Japanese," in *Proc. IFIP*, 1978, pp. 689-694.
[133] A. Kay and A. Goldberg, "Personal dynamic media," *Computer*, pp. 31-41, Mar. 1977.
[134] D. D. Kerrick and A. C. Bovik, "Microprocessor-based recognition of handprinted characters from a tablet input," *Pattern Recognition*, vol. 21, pp. 525-537, 1988.
[135] J. Kim, "Baseline drift correction for handwritten text," *IBM Tech. Disclosure Bull.*, vol. 25, p. 511, Mar. 1983.
[136] —, "Gesture recognition by feature analysis," IBM Res. Rep. RC 12472, Dec. 1986.
[137] —, "On-line gesture recognition by feature analysis," in *Proc. Vision Interface'88*, June 1988, pp. 51-55.
[138] J. Kim and C. C. Tappert, "Handwriting recognition accuracy versus tablet resolution and sampling rate," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 917-918.
[139] T. Kim, T. Agui, and M. Nakajima, "On-line recognition of Korean characters by multistage translation of directional codes," (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J65-D, pp. 177-184, Feb. 1985.
[140] A. L. Knoll, "Experiments with characteristic loci for recognition of handprinted characters," *IEEE Trans. Comput.*, vol. C-18, pp. 366-372, Apr. 1969.
[141] H. Kojima and T. Toida, "On-line hand-drawn line-figure recognition and its application," in *Proc. 9th Int. Conf. Pattern Recognition*, Nov. 1988, pp. 1138-1142.
[142] S. Komada, M. Mori, and T. Watanabe, "An electronic transparent tablet of handprinted characters," in *1985 SID Int. Symp. Dig. Tech. Papers*, May 1985, pp. 28-31.
[143] J. B. Kruskal, "An overview of sequence comparison: Time warps, string edits, and macromolecules," *SIAM Rev.*, vol. 25, pp. 201-237, Apr. 1983.
[144] T. T. Kuklinski, "Components of handprint style variability," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 924-926.
[145] —, "A case for digitizer tablets," *Comput. Graphics World*, pp. 45-52, May 1985.
[146] A. Kundu and P. Bahl, "Recognition of handwritten script: A hidden Markov model based approach," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1988, pp. 928-931.
[147] H.-W. Kuo, J.-C. Lee, and T.-C. Kao, "IOLCR, An on-line Chinese character recognition system," in *Proc. 1988 Int. Conf. Comput. Processing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 108-112.
[148] J. M. Kurtzberg, "Feature analysis for symbol recognition by elastic matching," *IBM. J. Res. Develop.*, vol. 31, pp. 91-95, Jan. 1987.
[149] J. M. Kurtzberg and C. C. Tappert, "Segmentation procedure for handwritten symbols and words," *IBM Tech. Disclosure Bull.*, vol. 25, pp. 3848-3852, Dec. 1982.
[150] F. Lamarche and R. Plamondon, "Segmentation and feature extraction of handwritten signature patterns," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 756-759.
[151] R. S. Ledeen, "The Ledeen character recognizer," in W. M. Newman and R. F. Sproull, Eds., *Principles of Interactive Computer Graphics.* New York: McGraw-Hill, 1973, pp. 575-582.
[152] A. Lee and F. H. Lochovsky, "Enhancing the usability of an office information system through direct manipulation," *Proc. Comput. Human Interaction*, 1983, pp. 130-134.
[153] C. G. Leedham and A. C. Downton, "Automatic recognition and transcription of Pitman's handwritten shorthand—An approach to shortforms," *Pattern Recognition*, vol. 20, pp. 341-348, 1987.
[154] C. G. Leedham, A. C. Downton, C. P. Brooks, and A. F. Newell, "On-line acquisition of Pitman's handwritten shorthand as a means

of rapid data entry," in *Proc. Interact'84, 1st IFIP Conf. Human-Comput. Interaction*, Sept. 1985, pp. 86-91.

[155] C. Lettera, M. Maier, L. Masera, and M. T. Pareschi, "Handwritten documents recognition with dictionary information support," *CSELT Tech. Rep. (Italy)*, vol. 13, pp. 419-422, 1985.

[156] C. Lettera, L. Masera, C. Paoli, and R. Porinelli, "Use of a dictionary in conjunction with a handwritten texts recognizer," in *Proc. 8th Int. Conf. Pattern Recognition*, Oct. 1986, pp. 699-701.

[157] J. L. Levine, R. L. Garwin, and M. A. Schappert, "An electronic podium for the classroom," in *1987 SID Int. Symp. Dig. Tech. Papers*, 1987, pp. 258-260.

[158] J. S. Lew, "Kinematic theory of signature verification measurements," *Math. Biosci.*, vol. 48, pp. 25-51, 1980.

[159] —, "Optimal accelerometer layouts for data recovery in signature verification," *IBM J. Res. Develop.*, vol. 24, pp. 496-511, July 1980.

[160] C. C. Li, T. L. Teng, M. J. Zobrak, and T. W. Sze, "On recognition of handwriting Chinese characters," in *Proc. 1st Princeton Conf. Inform. Sci. Syst.*, 1967, pp. 235-239.

[161] M.-Y. Lin and W.-H. Tsai, "A new approach to on-line Chinese character recognition by sentence contextual information using the relaxation technique," in *Proc. 1988 Int. Conf. Comput. Processing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 131-134.

[162] N. Lindgren, "Machine recognition of human language, Part III—Cursive script recognition," *IEEE Spectrum*, pp. 104-116, May 1965.

[163] Y. Litvin, "Principles of evaluation for handwritten and cursive text recognition methods," GTE Lab. Res. TN-401.1, Apr. 1982.

[164] C. N. Liu, N. M. Herbst, and N. J. Anthony, "Automatic signature verification: System description and field test results," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, pp. 35-38, Jan. 1979.

[165] S.-C. Loh, C.-W. Chan, and S.-C. Chan, "On-line recognition of hand-written Chinese characters," in *Proc. 1988 Int. Conf. Comput. Processing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 58-61.

[166] W. W. Loy and I. D. Landau, "An on-line procedure for recognition of handprinted alphanumeric characters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 422-427, July 1982.

[167] L. J. Lukis and G. P. Duhig, "Character recognition device," U.S. Patent 4 493 104, Jan. 1985.

[168] E. Mandler, "Advanced preprocessing technique for on-line script recognition of nonconnected symbols," in *Proc. 3rd Int. Symp. Handwriting Comput. Appl.*, July 1987, pp. 64-66.

[169] E. Mandler, R. Oed, and W. Doster, "Experiments in on-line script recognition," in *Proc. 4th Scandinavian Conf. Image Anal.*, June 1985, pp. 75-86.

[170] J. Mantas, "An overview of character recognition methodologies," *Pattern Recognition*, pp. 425-430, 1986.

[171] T. Marill, A. K. Hartley, T. G. Evans, B. H. Bloom, D. M. R. Park, T. P. Hart, and D. L. Darley, "Cyclops 1: A second generation recognition system," in *Proc. JFCC*, Nov. 1963, pp. 27-33.

[172] B. W. Mel, S. M. Omohundro, A. D. Robinson, S. S. Skiena, K. H. Thearling, L. T. Young, and S. Wolfram, "Tablet: Personal computer in the year 2000," *Commun. ACM*, pp. 639-646, June 1988.

[173] P. Mermelstein and M. Eden, "Experiments on computer recognition of connected handwritten words," *Inform. Contr.*, vol. 7, pp. 255-270, June 1964.

[174] —, "A system for automatic recognition of handwritten words," in *Proc. FJCC*, 1964, pp. 333-342.

[175] U. Miletzki, C. Steigner, W. Doster, G. Foraroli, H. Lobl, and P. Moulds, "Paper interfaces for office systems," in *ESPRIT'86: Results and Achievements.* Amsterdam: North-Holland, 1987, pp. 373-386.

[176] G. M. Miller, "On-line recognition of hand-generated symbols," in *Proc. FJCC*, vol. 34, 1969, pp. 399-412.

[177] —, "Real-time classification of handwritten script words," in *Proc. IFIP Congr., Inform. Processing 1971.* Amsterdam: North Holland, 1972, pp. 218-223.

[178] S. Mori, K. Yamamoto, and M. Yasuda, "Research on machine recognition of handprinted characters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 386-405, July 1984.

[179] T. Morishita, M. Ooura, and Y. Ishii, "A Kanji recognition method which detects writing errors," *Comput. Processing of Chinese and Oriental Languages*, vol. 3, pp. 351-365, Mar. 1988.

[180] J. H. Munson, "Experiments in the recognition of hand-printed text: Part I—Character recognition," in *Proc. FJCC*, Dec. 1968, pp. 1125-1138.

[181] H. Murase, "Online recognition of free-format Japanese handwritings," in *Proc. 9th Int. Conf. Pattern Recognition*, Nov. 1988, pp. 1143-1147.

[182] H. Murase and T. Wakahara, "Online hand-sketched figure recognition," *Pattern Recognition*, vol. 19, pp. 147-160, 1986.

[183] H. Murase, T. Wakahara, and M. Umeda, "On-line recognition algorithm for hand-sketched flowchart by candidate lattice method," *Syst. Comput. Contr.*, vol. 14, pp. 37-46, 1983.

[184] —, "Online writing-box free character string recognition by candidate character lattice method" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J68-D, pp. 765-772, Apr. 1985.

[185] R. Nag, K. H. Wong, and F. Fallside, "Script recognition using hidden Markov models," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1986, pp. 2071-2074.

[186] T. Nagayama, J. Shibuya, and T. Kawakita, "Pen-touch-type-electro-magnetic transparent touch panel," in *1985 SID Int. Symp. Dig. Tech. Papers*, May 1985, pp. 32-35.

[187] M. Nakagawa, K. Aoki, T. Manabe, S. Kimura, and N. Takahashi, "On-line recognition of handwritten Japanese characters in JOLIS-1," in *Proc. 6th Int. Conf. Pattern Recognition*, 1982, pp. 776-779.

[188] M. J. Naus and R. J. Shillman, "Why a Y is not a V: A new look at the distinctive features of letters," *J. Exp. Psychol.: Human Perception and Performance*, vol. 2, pp. 394-400, 1976.

[189] U. Neisser and P. Weene, "A note on human recognition of handprint characters," *Inform. Contr.*, vol. 3, pp. 191-196, 1960.

[190] A. Nilssen and J. R. Ward, "Apparatus and method for emulating computer keyboard input with a handprint terminal," U.S. Patent 4 562 304, Dec. 1985.

[191] W. R. Nugent, "The on-line recognition of cursive writing using geometric-topological invariants of stroke succession," in *Proc. 1st Annu. IEEE Comput. Conf.*, 1967, pp. 145-148.

[192] J. F. O'Callaghan, "Problems in on-line character recognition," in S. Kaneff, Ed., *Picture Language Machines.* New York: Academic, 1970.

[193] K. Odaka, "On-line pattern recognition system for hand-written characters," U.S. Patent 4 284 975, Aug. 1981.

[194] K. Odaka, H. Arakawa, and I. Masuda, "On-line recognition of handwritten characters by approximating each stroke with several points," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-12, pp. 898-903, 1982.

[195] K. Odaka and I. Masuda, "Pattern recognition system for hand-written characters operating on an on-line real-time basis," U.S. Patent 4 317 109, Feb. 1982.

[196] K. Odaka, T. Wakahara, and I. Masuda, "Stroke-order-independent on-line character recognition algorithm and its application," *Rev. Elec. Commun. Lab.*, vol. 34, pp. 79-85, 1986.

[197] J. Pavlidis and C. J. Van Wyk, "An automatic beautifier for drawings and illustrations," *ACM Comput. Graphics*, vol. 19, pp. 225-234, July 1985.

[198] W. Pepper, "Human-machine interface apparatus," U.S. Patent 4 071 691, Jan. 1978.

[199] R. Plamondon and R. Baron, "On-line recognition of handprint schematic pseudocode for automatic Fortran code generation," in *Proc. 8th Int. Conf. Pattern Recognition*, Oct. 1986, pp. 741-744.

[200] —, "A dedicated microcomputer for handwritten interaction with a software tool: system prototyping," *J. Microcomput. Appl.*, vol. 9, pp. 51-61, 1986.

[201] R. Plamondon and G. Lorette, "Automatic signature verification and writer identification—The state of the art," *Pattern Recognition*, vol. 22, pp. 107-131, 1989.

[202] P. J. Pobgee, A. M. Day, and J. R. Parks, "Online handwritten input to computers," *Mach. Perception of Patterns and Pictures*, Inst. Phys., Apr. 1972.

[203] P. J. Pobgee and J. R. Parks, "Application of a low cost graphical input tablet," *Inform. Processing*, vol. 71, pp. 738-741, 1972.

[204] V. M. Powers, "Pen direction sequences in character recognition," *Pattern Recognition*, vol. 5, pp. 291-302, Mar. 1973.

[205] R. W. Prugh and B. J. Fadden, "Graphic digitizer," U.S. Patent 4 206 314, June 1980.

[206] J. R. Rhyne, "Dialog management for gestural interfaces," IBM Res. Rep. RC12244, Sept. 1986.

[207] J. R. Rhyne and C. G. Wolf, "Gestural interfaces for information processing applications," IBM Res. Rep. RC12179, Sept. 1986.

[208] R. T. Rocheleau, "Coarse position digitizer," U.S. Patent 4 242 843, Jan. 1981.

[209] J. J. Romein, "Acoustic writing combination, comprising a stylus with an associated writing tablet," U.S. Patent 4 246 439, Jan. 1981.

[210] V. I. Rybak and G. I. Fursin, "Recognition of handwritten symbols in the process of writing," *Kibernetika*, vol. 2, pp. 104-112, Mar. 1972.

[211] T. Sakai, K. Odaka, and T. Toida, "Several approaches to development of on-line handwritten character input equipment," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1052-1054.

[212] D. Sankoff and J. B. Kruskal, *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison.* London: Addison-Wesley, 1983.

[213] Y. Sato and H. Adachi, "Online recognition of cursive writings" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J68-D, pp. 2116-2122, Dec. 1985.

[214] K. M. Sayre, "Machine recognition of handwritten words: A project report," *Pattern Recognition*, vol. 5, pp. 213-228, 1973.

[215] P. Schlosser and R. Kable, "A new high-performance digitizer for computer graphics and display applications," in *1985 SID Int. Symp. Dig. Tech. Papers*, May 1985, pp. 42-43.

[216] J. W. Schoonard, J. D. Gould, M. Bieber, and A. Fusca, "A behavioral study of a computer hand print recognition system," IBM Res. Rep. RC 12494, Feb. 1987.

[217] S.-L. Shiau, J.-W. Chen, A.-J. Hsieh, and S.-J. Kung, "On-line handwritten Chinese character recognition by string matching," in *Proc. 1988 Int. Conf. Comput. Processing of Chinese and Oriental Languages*, Aug.-Sept. 1988, pp. 76-80.

[218] R. Shillman, C. Cox, T. Kuklinski, J. Ventura, M. Eden, and B. Blesser, "A biography in character recognition: techniques for describing characters," *Visible Language*, vol. 8, pp. 151-166, 1974.

[219] R. J. Shillman, T. T. Kuklinski, and B. A. Blesser, "Experimental methodologies for character recognition based on phenomenological attributes," in *Proc. 2nd Int. Joint Conf. Pattern Recognition*, Aug. 1974, pp. 195-201.

[220] —, "Psychophysical techniques for investigating the distinctive features of letters," *Int. J. Man-Machine Studies*, vol. 8, pp. 195-205, 1976.

[221] —, "Empirical tests for feature selection based on a psychological theory of character recognition," *Pattern Recognition*, vol. 3, pp. 77-85, 1976.

[222] R. Shinghal and C. Y. Suen, "A method for selecting constrained hand-printed character shapes for machine recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 74-78, Jan. 1982.

[223] B. Shneiderman, "The future of interactive systems and the emergence of direct manipulation," *Behavior and Inform. Technol.*, vol. 1, pp. 237-256, 1982.

[224] M. Shridhar and A. Badreldin, "Recognition of isolated and simply connected handwritten numerals," *Pattern Recognition*, vol. 19, pp. 1-12, 1986.

[225] J. G. Simek and C. J. Tunis, "Handprinting input device for computer systems," *IEEE Spectrum*, vol. 4, pp. 72-81, 1967.

[226] R. Sniderman, W. Reeves, S. Patel, and R. Baecker, "The evolution of the SSP score editing tools," *Comput. Music J.*, vol. 3, pp. 14-25, 1979.

[227] Special Issue, "Papers from the Int. Workshop on the Motor Aspects of Handwriting," *Acta Psychol.*, vol. 54, 1983.

[228] C. Y. Suen, "Handwriting education—A bibliography of contemporary publications," *Visible Language*, vol. 9, pp. 145-158, 1975.

[229] —, "A study of man-machine interaction problems in character recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, pp. 732-737, Nov. 1979.

[230] —, "Handwriting generation, perception and recognition," *Acta Psychol.*, vol. 54, pp. 295-312, 1983.

[231] —, "Character recognition by computer and applications," in T. Y. Young and K. S. Fu, Eds., *Handbook of Pattern Recognition and Image Processing.* Orlando, FL: Academic, 1986, pp. 569-586.

[232] —, *Computational Studies of the Most Frequent Chinese Words and Sounds.* Singapore: World Scientific, 1986.

[233] C. Y. Suen, M. Berthod, and S. Mori, "Automatic recognition of handprinted characters—The state of the art," *Proc. IEEE*, vol. 68, pp. 469-487, Apr. 1980.

[234] C. Y. Suen and R. J. Shillman, "Low error rate optical character recognition of unconstrained handprinted letters based on a model of human perception," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 491-495, June 1977.

[235] C. Y. Suen, R. Shinghal, and C. C. Kwan, "Dispersion factor, A quantitative measurement of the quality of hand-printed characters," in *Proc. SMC Conf.*, 1977, pp. 681-685.

[236] Y. Suenaga and M. Nagura, "A facsimile based manuscript layout and editing system by auxiliary mark recognition," in *Proc. 5th Int. Conf. Pattern Recognition*, 1980, pp. 856-858.

[237] Y. Takeda, N. Tanabe, Y. Aono, and Y. Tokumitsu, "A new data tablet superimposed by a plasma display panel," in *Proc. IEEE Nat. Telecommun. Conf.*, vol. 4, 1981, pp. G5.4.1-G5.4.5.

[238] T. Tanaka and S. Kobayashi, "Entry of data and command for an LCD by direct touch: an integrated LCD panel," in *1986 SID Int. Symp. Dig. Tech. Papers*, 1986, pp. 318-320.

[239] C. C. Tappert, "Cursive script recognition by elastic matching," *IBM J. Res. Develop.*, vol. 26, pp. 765-771, Nov. 1982.

[240] —, "Adaptive on-line handwriting recognition," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1004-1007.

[241] —, "An adaptive system for handwriting recognition," in H. S. R. Kao, G. P. Van Galen, and R. Hoosain, Eds., *Graphonomics: Contemporary Research in Handwriting.* Amsterdam: North-Holland, 1986, pp. 185-198.

[242] —, "Speed, accuracy, flexibility trade-offs in on-line character recognition," IBM Res. Rep. RC13228, Oct. 1987.

[243] —, "A divide-and-conquer cursive script recognizer," IBM Res. Rep. RC14070, Oct. 1988.

[244] —, "Recognition system for run-on handwritten characters," U.S. Patent 4 731 857, Mar. 1988.

[245] C. C. Tappert, A. S. Fox, J. Kim, S. E. Levy, and L. L. Zimmerman, "Handwriting recognition on transparent tablet over flat display," in *1986 SID Int. Symp. Dig. Tech. Papers*, May 1986, pp. 308-312.

[246] C. C. Tappert and J. M. Kurtzberg, "Elastic matching for handwritten symbol recognition," in *Proc. IBM Int. Conf. Image Processing Pattern Recognition*, 1978. Also, IBM Res. Rep. RC9988, May 1983.

[247] C. C. Tappert, C. Y. Suen, and T. Wakahara, "On-line handwriting recognition—A survey," in *Proc. 9th Int. Conf. Pattern Recognition*, Nov. 1988, pp. 1123-1132.

[248] W. Teitelman, "Real-time recognition of hand-drawn characters," in *Proc. FJCC*, vol. 26, Oct. 1964, pp. 559-575.

[249] J. F. Teixeira and R. P. Sallen, "The Sylvania data tablet: A new approach to graphic data input," in *Proc. FJCC*, 1968, pp. 315-321.

[250] H. Terai and K. Nakata, "On-line real-time recognition of handwriting Chinese characters and Japanese katakana syllabary" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. 56-D, pp. 312-319, May 1973.

[251] H. H. M. Teulings and A. J. W. M. Thomassen, "Computer-aided analysis of handwriting movements," *Visible Language*, vol. 13, pp. 218-231, 1979.

[252] A. J. W. M. Thomassen, P. J. G. Keuss, and G. P. van Galen, *Motor Aspects of Handwriting: Approaches to Movement in Graphic Behavior.* Amsterdam: North-Holland, 1984.

[253] J. T. Tou, J. C. Tsay, and J. K. Yoo, "Interactive processing of Chinese characters and texts," in *Proc. 1st Int. Symp. Comput. and Chinese Input-Output Syst.*, 1973, pp. 1-28.

[254] J. A. Turner and G. J. Ritchie, "Linear current division in resistive areas, Its application to computer graphics," in *Proc. SJCC*, vol. 36, 1970, pp. 613-620.

[255] J. Vredenbregt and W. G. Koster, "Analysis and synthesis of handwriting," *Philips Tech. Rev.*, vol. 32, pp. 73-78, 1971.

[256] T. Wakahara, "On-line cursive script recognition using local affine transformation," in *Proc. 9th Int. Conf. Pattern Recognition*, Nov. 1988, pp. 1133-1137.

[257] T. Wakahara and K. Odaka, "Character deformation prediction using deformation vector field in online handwritten character recognition" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J69-D, pp. 1913-1922, Dec. 1986.

[258] T. Wakahara, K. Odaka, and M. Umeda, "Electro-luminescent display superimposed by transparent electromagnetic coupling tablet and its application to script input Japanese word processor" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J67-D, pp. 981-988, Sept. 1984.

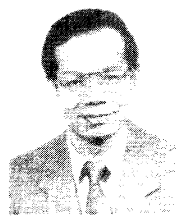
[259] T. Wakahara and M. Umeda, "Stroke-number and stroke-order free on-line character recognition by selective stroke linkage method," in *Proc. 4th ICTP*, 1983, pp. 157-162, 1983.

[260] —, "On-line cursive script recognition using stroke linkage

rules," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1065-1068.
[261] J. R. Ward, "Method and apparatus for removing noise at the ends of a stroke," U.S. Patent 4 534 060, Aug. 1985.
[262] J. R. Ward and B. Blesser, "Interactive recognition of handprinted characters for computer input," *IEEE Comput. Graphics Appl.*, vol. 5, pp. 24-37, Sept. 1985.
[263] —, "Implications of using interactive hand print character recognition for computer input," in *Proc. 1985 Trends and Appl. in Comput. Graphics Conf.*, May 1985.
[264] J. R. Ward and T. Kuklinski, "A model for variability effects in handprinting with implications for the design of handwriting character recognition systems," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, pp. 438-451, May/June 1988.
[265] J. R. Ward and M. J. Phillips, "Digitizer technology: Performance characteristics and the effects on the user interface," *IEEE Comput. Graphics Appl.*, pp. 31-44, Apr. 1987.
[266] Y. Watanabe, J. Gyoba, T. Hirata, and K. Maruyama, "A psychological approach to the human recognition of ambiguous characters," *J. Inst. TV Eng. Japan*, vol. 39, pp. 509-515, 1985.
[267] Y. Watanabe, J. Gyoba, and K. Maruyama, "Reaction time and eye movements in the recognition task of hand-written Katakana-letters: An experimental verification of the discriminant analysis of letter recognition by Hayashi's quantification," *Japan. J. Psychol.*, vol. 54, pp. 58-61, 1983.
[268] R. Whitrow and C. Higgins, "The application of *n*-grams for script recognition," in *Proc. 3rd Int. Symp. Handwriting Comput. Appl.*, July 1987, pp. 92-94.
[269] A. M. Wing, "Variability in handwritten characters," *Visible Language*, vol. 13, pp. 283-298, 1979.
[270] C. G. Wolf, "Can people use gesture commands," IBM Rep. RC 11867, Apr. 1986.
[271] —, "A comparative evaluation of gestural and conventional interfaces," IBM Res. RC 13187, Oct. 1987.
[272] C. G. Wolf and P. Morrel-Samuels, "The use of hand-drawn gestures for text-editing," *Int. J. Man-Machine Studies*, 1987.
[273] K. H. Wong and F. Fallside, "Dynamic programming in the recognition of connected handwritten script," in *Proc. 2nd Conf. Artificial Intell. Appl.*, IEEE Comput. Soc., Dec. 1985, pp. 666-670.
[274] T. K. Worthington, T. J. Chainer, J. D. Williford, and S. C. Gundersen, "IBM dynamic signature verification," in J. B. Grimson and H. J. Kuglern, Ed., *Computer Security: The Practical Issues in a Troubled World.* Amsterdam: North Holland, 1985.
[275] G. G. N. Wright, *The Writing of Arabic Numerals.* London: Univ. London Press, 1952.
[276] J. Yacyk, "Alphabetic handprint reading," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 279-282, Apr. 1978.
[277] K. Yamamoto and S. Mori, "Recognition of handprinted characters by an outermost point method," *Pattern Recognition*, vol. 12, pp. 229-236, Mar. 1980.
[278] T. Yamasaki, S. Inokuchi, and S. Yoshifumi, "Training system for well-writing based on on-line character recognition," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1039-1041.
[279] M. Yasuhara, "Experimental studies of handwriting process," *Rep. Lab. Commun. Sci.*, Univ. Electro-Commun., Japan, vol. 25-2 (Sci. and Tech. Sect.), pp. 233-254, Mar. 1975.
[280] M. Yasuhara and M. Oka, "Signature verification experiment based on nonlinear time alignment: A feasibility study," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 212-216, Mar. 1977.
[281] P. J. Ye, H. Hugli, and F. Pellandini, "Techniques for on-line Chinese character recognition with reduced writing constraints," in *Proc. 7th Int. Conf. Pattern Recognition*, 1984, pp. 1043-1045.
[282] E. F. Yhap and E. C. Greanias, "An on-line Chinese character recognition system," *IBM J. Res. Develop.*, vol. 25, pp. 187-195, May 1981.
[283] K. Yoshida and H. Sakoe, "Online handwritten character recognition for a personal computer system," *IEEE Trans. Consumer Electron.*, vol. CE-28, pp. 202-209, Aug. 1982.
[284] M. Yoshida and M. Eden, "Handwritten Chinese character recognition by analysis-by-synthesis method," in *Proc. 1st Int. Conf. Pattern Recognition*, 1973, pp. 197-204.
[285] M. Yurugi, S. Nagata, K. Onuma, and K. Kubota, "Online character recognition by hierarchical analysis method" (in Japanese), *Trans. Inst. Electron. Commun. Eng. Japan*, vol. J68-D, pp. 1320-1327, June 1985.
[286] M. J. Zobrak and T. W. Sze, "A method of recognition of hand drawn line patterns," in *Proc. 1st Princeton Conf. Inform. Sci. Syst.*, 1967, pp. 240-244.

Charles C. Tappert (M'90) received the B.S. degree from Swarthmore College, Swarthmore, PA, and the M.S. and Ph.D. degrees from Cornell University, Ithaca, NY.

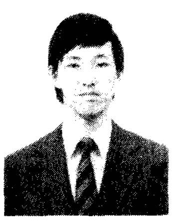
He joined IBM in 1967 and has been a Research Staff Member of the IBM T. J. Watson Research Center, Yorktown Heights, NY, since 1972. His research has primarily been in the area of pattern recognition. He worked on automatic speech recognition and speech coding for ten years. Since 1978 he has been working on tablet terminals, where his main interest has been computer recognition of handprinting and cursive writing.



Ching Y. Suen (M'66-SM'78-F'86) received the M.Sc.(Eng.) degree from the University of Hong Kong, Hong Kong, and the Ph.D. degree from the University of British Columbia, Vancouver, B.C., Canada.

In 1972 he joined the Department of Computer Science, Concordia University, Montreal, P.Q., Canada, where he became a Professor in 1979 and served as Chairman from 1980 to 1984. Presently he is the Director of the new Concordia Center for Pattern Recognition and Machine Intelligence. During the past 12 years, he was also appointed to visiting positions in several institutions in different countries. He is the author or editor of several books on pattern recognition, image processing, and computational studies of the Chinese language, and he has published many papers on pattern recognition and machine intelligence, character recognition and expert systems, text processing, and computational linguistics.

An active member of several professional societies, Dr. Suen is an Associate Editor of several journals. During the past ten years, he has been serving as Chairman of the Character and Mark Recognition Committee of the Canadian Standard Association, which developed several Canadian standards on optical character recognition. Presently, he is the President of the Canadian Image Processing and Pattern Recognition Society, Governor of the International Association for Pattern Recognition, and President of the Chinese Language Computer Society. He was also the Chairman of the International Workshop on Frontiers in Character Recognition, held in Montreal in April 1990.



Toru Wakahara (M'88) was born in Gifu, Japan, on January 30, 1952. He received the B.E. and M.E. degrees in applied physics and the Ph.D. degree in electrical engineering from Tokyo University, Tokyo, Japan, in 1975, 1977, and 1986, respectively.

Since 1977 he has been with Nippon Telegraph and Telephone Corporation where he is engaged in research on character recognition and text processing. He is presently a Senior Research Engineer, Supervisor in the NTT Human Interface Laboratories.

Dr. Wakahara is a member of the Institute of Electronics, Information, and Communication Engineers of Japan and the Audio-Visual Information Research Group of Japan.