Invited Paper

Non-keyboard Input of Japanese Text
On-line Recognition of Handwritten Characters as the Most Hopeful Approach

Masaki Nakagawa*

Among non-keyboard input methods for Japanese text, on-line recognition of handwritten characters is surveyed as the most hopeful approach. Its technical advantages are discussed in comparison with speech recognition and optical character recognition. Research activities of on-line recognition are divided into four generations including two times of hot periods. The first generation was expected to establish the technology soon, but realized how noisy and distorted human writing patterns were. The second was devoted to steady researches for recognition of such human patterns under certain constraints. The third is characterized by hot activities to make products of script recognition as well as researches to liberate writing constraints based on remarkable progresses in hardware. The latest generation studies intrinsic problems of pattern recognition in the domain of on-line recognition. The survey is aimed exhaustive. Throughout the generations, considerable efforts have been made to relax constraints on character patterns to be recognized. Stroke order free and number free recognition has been sought. At present, however, a method free from the one constraint is not completely free from the other. Other issues also remain for on-line recognition to be employed for general text input. Nevertheless, handwriting input has potential merits unmeasurable by performance or cost-effectiveness.

1. Introduction

Japanese is a language whose character set includes thousands of ideographic characters of Chinese origin called kanji and two kinds of 46 phonetic characters (among which 9 letters have a small case as well) with a voiced sound mark and a P-sound mark. One kind is called hira-gana, the other is kata-kana and they are called kana collectively. Moreover, English alphabets, numerals, Greek letters, symbols and so on can be incorporated almost freely. The history of Japanizing kanji has removed the four tones is Chinese and produced many homonyms.

Although Japanese text processing by computers has got behind, its necessity in professional business such as newspaperdom, triggered off the research and development of the Japanese text processing technology in 1950's [129, 52]. Then, this trend gradually spread into commercial business where Japanese names and addresses must be processed [107]. In 1980's, this is being strongly accelerated by the movement toward office automation (OA) and home automation [62].

The technological background for office automation and Japanese text processing had been matured by the end of 1970's. Microprocessors, large-scale IC memories, floppy disk drives, Winchester disk drives, wire-dot printers and laser beam printers were all available to produce less expensive products. As a matter of fact, the first Japanese word processor was produced in 1978 [40, 41]. Since then, the development race of low-price Japanese word processors and personal computers with word processing utilities have been driving the research and development of the Japanese text processing technology significantly. Ever continuing progress in microelectronics and input/output devices is promising that computer processing of Japanese documents can be made still more inexpensive. Consequently, the price becomes cheap, the market of Japanese text processing is expanded and then the merit of mass production can be enjoyed, which forms a positive feed-back for the spread of the Japanese text processing evolution into the entire Japanese society.

Compared with the technical developments as mentioned above, however, there still remains an intrinsic problem in Japanese input methods, i.e., a problem to input text of the large and mixed set of characters. We do not yet have any definitive input method which is easy to learn, easy to use, and efficient as well.

*Department of Computer Science, Division of Electronic and Information Science, Faculty of Technology, Tokyo University of Agriculture and Technology, 2-24-16 Naka-machi, Koganei, Tokyo 184, Japan

2. Japanese Input Methods

2.1 Overview

Input methods currently available or being studied could be divided into four groups: sight-typing, touch-typing, kana-kanji automatic translation, and applications of pattern recognition. Sight-typing is a method that uses a full keyboard of kanji and kana. The point of sight-typing is that anybody can use it without training. But, it is not efficient to search for a character among thousands of keys even if they are arranged in some order. Touch-typing [43, 122, 22, 63] is performed on a kana or English keyboard where two or more keys correspond to a single kanji. Touch-typing would be the most efficient and inexpensive method among others. But the problem is that one must get trained. Kana-kanji automatic translation [55] is the mainstream at the present time, since anybody can use it with a keyboard after a little training and high performance of more than 100 character input/minute can be expected for well-trained typists with their private translation dictionaries [109]. But, a sentence must be chopped into clauses for better translation and the translation is often obstructed by homonyms. Takahashi shows that non-professional users have a wall of 30~35 character input per minute [109]. Moreover, human stress of this method is being socially recognized and scientifically studied by H. Yamada et al. [123, 87]. In any case, smooth and perfect translation from the phonetic characters to the ideograms can not be expected until both syntactic and semantic researches on Japanese text processing bear fruit. As applications of pattern recognition, we can enumerate speech recognition, Optical Character Reader/Recognition (OCR) for handwritten characters and their on-line recognition.

2.2 Non-keyboard Methods

Speech recognition [11, 75, 76, 23, 77] is one of the easiest methods for users. But, both the problem of voice to phonogram transformation and that of phonogram to ideogram transformation must be solved. The current level of speech recognition is restricted to a very limited number of words by a specific speaker with word by word input under noiseless circumstance. This by no means provides a natural method for users to input text. Syllable by syllable recognition theoretically enables a user to input any sequence of phonetic characters but it is technologically even harder and less natural for him. Speech recognition is not just confined to the problem of speech signal processing but it requires fundamental studies in relation with linguistics, cognitive science, artificial intelligence and so on. Although almost all problems in pattern recognition have such nature, it is most eminent in speech recognition. Such an interdisciplinary research project has just been begun as one of Japanese national projects at ATR Interpreting Telephony Research Laboratories [76, 101]. For restricted applications, on the other hand, developments of LSI and DSP (digital signal processor) is favorable to make speech recognition cost-effective.

OCR [128, 104] is an off-line means to input a large amount of Japanese text at a time. One can write text anywhere and later he only has to input that text to OCR. As for the OCR technology, extensive researches were made from 1971 to 1980 under the PIPS (Pattern Information Processing System) project of MITI [2, 80] and a high-speed OCR for multi-font Japanese printed characters was developed as one of its fruits [95]. But OCR for handwritten kanji patterns was left for later researches. Intensive studies are being continued [44, 110, 124, 125]. Moreover, the recent development of low-price image scanners is favorable for the OCR method. But, intrinsically, statistical methods pose limitations to topological variations of handwriting and syntactic methods are troubled with segmentation.

On-line recognition of handwritten characters is more feasible than OCR and still provides the most natural way to input Japanese text. But the speed to input text is limited by that of writing (about 30~40 characters/minute) [105, 134]. Each has its merits and demerits, thus has its raisons d'être. It is probable that future systems need to make several input methods available each of which is suitable for particular usages and presents its superiority for various kinds of workers to prepare, edit, modify and refer shared documents. Extensive variety of input/output methods and their systematization would be essential for better man-machine communication.

In summary, on-line recognition of handwritten characters is the most feasible method that we can expect. As a matter of fact, extensive researches have been made and some have been turned into products. As non-keyboard input of Japanese text, therefore, this survey focuses on the history of on-line recognition.

3. Advantages of On-line Recognition

The advantages of on-line recognition are as follows: (1) Non-professionals can use it without training in much the same way as they write a manuscript with a pen. (2) On-line handwriting is suitable to input various line drawings which often appear in Japanese documents as well as text. (3) On-line, real-time and interactive communication between a user and a system is suitable to input, edit and format documents. (4) Rejection could be notified and fixed by virtue of interactive processing, although misidentification may be slipped in. (5) Technological progress and price-down of tablet digitizers are remarkable [27] so that they can be expected as one of common and standard input devices to input script and drawings. Conventional tablets provide a trace of pen motion at more than 100 points/sec. with 0.1 mm resolution. Furthermore, there would be little technical difficulties to produce portable and handy tablets. (6) Input speed of on-line handwriting is surely slower than those of well-trained kana-kanji translation, touch-typing and OCR. But, on-line handwriting dose not disturb user's thinking so that input can be simultaneously made with it. Therefore, overall time required to produce documents would not be much greater than others for end-users who must make and input text for themselves.

Moreover, it has the following technical advantages against OCR.

(1) Pen's down/lift information is available to separate strokes easily form one another, so that the difficult problem of segmentation can be avoided. (2) The information about pen motion and order of strokes to write a character can be utilized.

The number of characters and the variety of character patterns, however, are yet serious obstacles to realize on-line recognition of Japanese characters. Moreover, rough, distorted and simplified character patterns or old style of patterns are often written in one's own stroke order. Because of unexpected pattern variations, pattern recognition of even a much smaller set of pattern categories remains to be a non-trivial problem.

4. The State of the Art

4.1 A Bird's-eye View

Orthodox classification of pattern recognition methods is either statistical or syntactic, although it could be argued that the statistical and syntactic approaches are not completely contradictious concepts.

Statistical (decision-theoretic or discriminant) approaches extract statistical quantities of features from a sampled pattern and consider partitioning of the feature space to best discriminate each pattern category from one another. Due to its nature, better robustness for random noises on patterns can be expected. Well-established mathematics of multivariate analysis is available.

In syntactic (structural) approaches, on the other hand, each pattern is structurally described by discrete nature of syntactic features (pattern primitives) with their composition rules. Recognition of each pattern is usually realized by parsing a pattern (sentence of the pattern primitives) according to a set of syntax rules (grammar). These approaches are advocated for recognition of structures of patterns or structural patterns.

In pattern recognition, ad hoc or heuristical methods are also sometimes useful and yield good results for some particular applications in spite of their computational economy. In general, they pick up most distinctive features of patterns to be recognized and decide a category of an input pattern after a sequence of deterministic tests. But, they are not generally applicable for other domains. Moreover, serious problems are often exposed in learning, increase of input pattern categories to be recognized and so on.

In the field of on-line recognition of handwritten Japanese characters, some of the early researches adopted often rather heuristic methods which examined structural features. This was partly because hardware constraint was too severe to apply the theory of syntactic recognition straightly and partly because pure syntactic recognition was premature for practical applications. At present, "structural" and "syntactic" are almost interchangeably used, since hierarchical structures are best described by syntax. But, this was not true for the researches having been made in our field.

Later in the paper, various distortions and variations of character patterns are treated, so terminology for referring them is defined as follows: A character pattern is termed as a "simplified pattern" when some strokes or radicals are simplified or omitted according to common rules or customs. We use a "running stroke pattern" or "cursive pattern" to designate a character pattern with consecutive strokes unstably connected. When, it is deformed with stroke connected and distorted, we call it a "deformed pattern". A stroke may be "chopped up" or "broken" because of fast writing. Such a stroke is also termed a "skipping" stroke. Character pattern distribution is formed by random noises, geometrical distortions, stroke connections, omissions, breaks, pattern simplifications, deformations, wrong stroke order and whatever else that is applied to a character pattern.

4.2 The First Generation

It seems that there have been two hot periods in the researches on on-line recognition of handwritten Japanese characters. The first period was formed around 1970, preceded by the opening research by Groner [16] which attempted on-line recognition of numerals on the RAND tablet. Most people expected that technology would soon be matured to a practical level [99], but the researches were not turned into products. We review them from the beginning.

Hanaki et al. at NEC corp. attempted on-line recognition of 86 categories of kata-kana and alphanumerals by a heuristic method [17]. They approximated an input pattern by 8-directional straight-line segments as shown in Fig. 1 and examined it according to a decision tree where tested were (1) the number of strokes (a stroke is a pen movement from pen-down to pen-lift), (2) the first, last or largest directional change between two consecutive segments in a specified stroke, (3) directional gap between the first and last segments in a stroke, (4) position of a crossing point of specified segments (5) direction of a principal segment in a stroke, and (6) positional relation between a specified pair of a point and a segment.

Furukawa and Mita also at NEC studied a structural and syntactic method for the educational set of kanji characters [15]. An input character pattern was segmented into 8-directional vector sequence and consti-

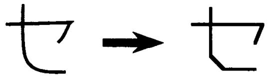


Fig. 1 8-directional segmentation of an on-line pattern.

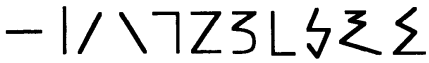


Fig. 2 Primitive stroke set.

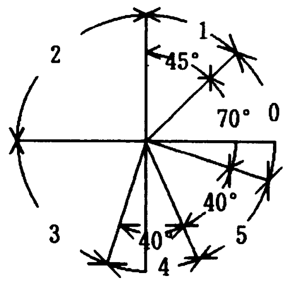


Fig. 3 Non-uniform quantization boundaries for segmentation.

tuent strokes were identified into one of 13 primitive strokes by segment accepting state transition (basically automaton). Then, it was examined with respect to a feature description table for kanji patterns of the same stroke number. In the table, a very neat description about each character pattern was provided in terms of units (subpatterns), order free strokes in a unit, 3 ways of stroke connection and 8 ways of stroke positional relation within a unit, as well as 4 ways of unit relation. An input pattern must satisfy all of the features specified in the description table, although very restricted distortions and noises were allowed in the identification of each feature.

Kashioka and Iwaoka at Kyoto Univ. [38] represented an input character pattern by a sequence of 8-directional segments for each stroke and 8-directional untouched pen movements between strokes. A segment and an untouched pen movement both had one-bit length information, i.e., long or short. The grammar to describe standard patterns for the educational set (961 char.) of kanji was provided with 50 kinds of strokes and 220 subpatterns (nonterminals). A subpattern was a sequence of smaller subpatterns or strokes as well as in-between pen movements. A subpattern could be composed in plural ways. A stroke was a class of segment sequences and an automaton to accept it was implemented. An input sequence of segments and untouched pen movements was identified by the top-down analysis using this grammar. Other features such as relations between strokes were used when necessary. This was a pure syntactic approach. Therefore, the problem of this approach was that of the syntactic methods, i.e., lack of robustness against variations of segments to form a stroke and other random noises.

Terai and Nakata at Hitachi Ltd. attempted a structural method for kata-kana recognition with 7 primitive strokes [111]. Then, they extended this method to recognize the educational set of kanji and kata-kana with 11 primitive strokes as shown in Fig. 2 [112, 113]. It was claimed that this was the first attempt to recognize a mixed set of kanji and kana. Each input stroke was described by a sequence of 8-directional straight-line segments and identified rather heuristically by means of the following 5 kinds of information: (1) stroke length, (2) turn features with respect to the first segment, (3) a sequence of maximum and minimum X-coordinates, (4) a sequence of segments, and (5) direction of a straight stroke. An input character pattern was roughly classified by the number of strokes to form it and distinctive curvilinear strokes included. Then, it was tested according to a decision tree which determined a character uniquely at each terminal leaf. Items tested at the tree nodes were: (1) stroke order of a specified curvilinear stroke, (2) stroke sequence between specified stroke orders (positions in a stroke sequence), (3) stroke set (order free) between those, (4) existence of a specified subpattern, (5) positional relation of specified two strokes, and (6) length relation of those.

They also implemented a similar but stroke-order free recognition procedure for upper case alphabets and numerals with 24 primitive strokes, and a statistical method for hira-gana where X-, Y-waveforms of an input hira-gana character pattern were matched with standard patterns. As to the matching, an elastic matching by Dynamic Programming and a linear matching with Euclidean distance were compared [113].

Another syntactic method is reported by Inoue et al. [28] at NTT which employs the same primitive strokes as Terai and Nakata's method for the recognition of 996 kanji characters. Its identity lies in standard pattern representation in terms of a sequence of subpatterns, registration of stroke order variations for a subpattern and dissimilarity matching rather than perfect match. The dissimilarity is simply the count of stroke dismatch. Input data points are segmented into 6 directional straight-line segments by non-uniform boundaries as shown in Fig. 3. Non-uniform directional quantization is also reported in on-line recognition of handwritten alphabets by Ito and Chui [36]. Stroke identification is made by automata accepting allowable sequences of segments. 90% recognition is reported to be realized after a little training for handprinted character patterns.

Statistical approaches to on-line recognition were opened by Fujisaki et al. [12] at the Univ. of Tokyo, although their method was not really statistical. For handwritten numerals, pen motion was decomposed into two waves with respect to a pair of orthogonal axes and the first several Fourier coefficients were computed for each wave. Then, the order of the largest coefficient and that of the second largest coefficient were adopted as features of the wave. Two pairs of these features for the two coordinates were examined in reference to registered features for ten numerals.

Extensive enhancement was made by Arakawa and Masuda at NTT [6, 7, 8, 9]. For alphanumerals, the first p Fourier coefficients were computed for each of $x(t)$ and $y(t)$ of a stroke, and a stroke was represented by a vector of $2p$ elements. They divided alphanumerals into one-stroke and two-stroke patterns and prepared standard patterns of $2p$ and $2\times 2p$ elements, respectively. Recognition of an input pattern was realized by the Bayes classifier and the effect of a full covariance matrix between elements (the Maharanobis distance) was shown against its simplifications such as the diagonal matrix of variances, although the time complexity of $O(p^2)$ was required for the full covariance matrix. Moreover, improvement of the recognition rate was shown as the number of the coefficients was increased up to 10 [6]. In [8], they applied this method for hira-gana and kata-kana separately, and settled the covariance matrix to that with the Fourier coefficients assumed dependent within a stroke but independent over a stroke. The method was expected to be also applicable for kanji patterns, but eventually superseded by a different scheme based on the research by Odaka et al. [81], which will be described soon.

Aoki, Inokuchi and Sakurai at Osaka Univ. presented a statistical method for handwritten and possibly cursive hira-gana patterns [5]. Given an input pattern, its stroke sequence as well as untouched pen movement vectors between them are approximated by a sequence of 20 8-directional segments of almost equal length and statistically matched with standard templates. Due to the statistical nature of the employed features, learning of weight vectors are performed. Auxiliary features of the stroke number, the number of crossing points, the number of α-type loops with their positions, and the number of φ-type loops, are utilized for fine classification by a decision tree.

A characteristic of the first generation is the dominance of structural and syntactic approaches. This is technologically because the hardware limitation was very serious. Syntactic description of standard patterns takes less space than statistical matching templates. And, deterministic classification does not spend as much CPU time as statistical classification.

In addition to the above reasons, however, there are the following intrinsic reasons: (1) Extraction of pattern primitives is comparatively easier so that the difficult problem of segmentation can be avoided; (2) Distinctive features can be explicitly expressed; (3) Hierarchical structures of kanji patterns could be reflected in their recognition.

But, the structural and syntactic approaches as well as the statistical approaches did not satisfy the level that we hoped to achieve. Structural and syntactic approaches are often too sensitive or unstable to erroneous features in input patterns. Patterns like kanji include much redundant information, so that some features may be defective in identifing them. Those approaches, especially when a decision tree or an automaton is applied, however, are likely to reject input patterns due to a single defective feature. We also experienced this problem in our early researches with Pre-JOLIS (Japanese On-Line Input System) and JOLIS-0 [67, 68, 69], so that similarity was introduced.

The problem of pattern primitive errors reflected in the case of handwritten Japanese character recognition is serious, since serifs originated from traditional brush writing often appear at the beginning and end of a stroke, with the result that strokes are frequently misidentified. This entails that beautiful patterns with serifs are difficult patterns for machines to read.

Until recently, computers were far more expensive than human labor cost, so that users had to adapt themselves to communicate with computers. Design of applications as well as systems software were based on this balance of values. In our application as well, it was supposed that users should write "correct" patterns of right stroke order and correct stroke number so that machines could recognize them. But, users could not bear and continue to write machine readable characters.

Human communication is carried out by means of very ambiguous, distorted, noisy and erroneous patterns, so that it is extremely difficult for us to narrow down this wide range of tolerance to a hairbreadth and very rigid band of expressions for computers.

The researches of the first generation were too optimistic for pattern distortions and variations. Therefore, it might be more correct to conclude that input patterns turned out not to satisfy the quality expected by the machine recognition of those days rather than the machine recognition did not meet the requirement.

4.3 The Second Generation

Between the two hot periods, steady researches were continued by the several groups.

Ikeda et al. at Kyoto Univ. attempted a structural method for about 2000 mixed characters of kanji, kata-kana, hira-gana, alphabets and numerals [13, 25, 26]. They divided this set into 5 classes, i.e., one-stroke, two-stroke, three- to six-stroke, seven- to ten-stroke and more than ten stroke characters, and examined basically the following two types of features in a different way and different priority for each class:

(1) a sequence of strokes each of which has been determined by a restricted DP-matching with 177 registered distinct shapes of strokes. (2) a sequence of 24-directional vectors each of which links the middle point of a stroke to that of the next stroke as shown in Fig. 4. In a fewer stroke patterns, the stroke shape possesses more importance, so that the former type of features are mainly employed. In many stroke patterns, on the other hand, the principal features lie in the relation of strokes so that the latter description is primarily considered.

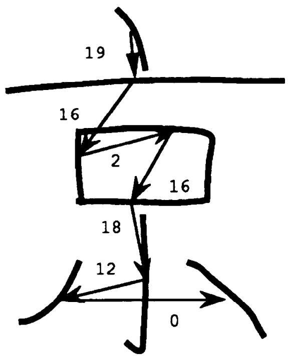


Fig. 4 Vector sequence to link a sequence of middle points of strokes.

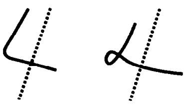


Fig. 5 Similar strokes which produce distant vector sequence (large geometrical distance).

For the identification of stroke shape, an input stroke is approximated by a chain of 24-directional vectors, normalized into 11 pieces of 24-quantized directions in a chain, and then matched with registered sequences by a restricted DP-matching.

To design an unified approach to a wide range of patterns is clearly very difficult, but the switch of recognition methods poses a serious problem for distorted patterns especially when stroke connection occurs due to cursive writing.

Nevertheless, their work is worth noting in structural approaches. A serious problem of structural and syntactic approaches is the lack of robustness to pattern distribution. Misidentification of structural elements due to irregular hand motion, serifs, sampling errors of input data points, preprocessing errors and so on must be coped with. In the structural and syntactic approaches to on-line recognition of handwritten characters, they first introduced the concept of similarity among primitive strokes rather than just geometrical distance, although they defined it heuristically. If two different shape of strokes often appear at the same place of the same character pattern, a small similarity value is empirically assigned between them even though they may have a geometrically large distance as such a pair shown in Fig. 5 does. Consequently, they could reduce the size of the character pattern dictionary to 1/2~3/4.

Hanaki's group also published a structural and heuristic method for kanji recognition [19]. Uniqueness of their method is a representation scheme of a character pattern by a string of symbol pairs each of which denotes positional relation of a stroke with

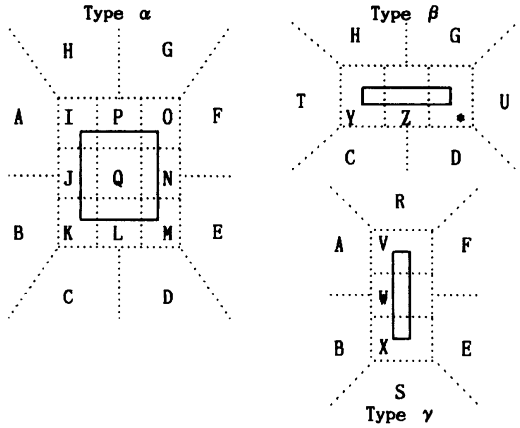
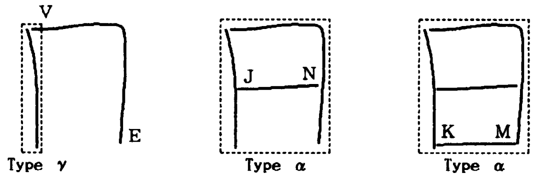


Fig. 6 Three types of reference zones (solid lines) and sub-zones (dotted lines) with their sub-zone symbols.



The string of symbol pairs, VE JN KM, is generated.

Fig. 7 An example of symbol string generation steps.

respect to a reference zone. Suppose a stroke subsequence $s_1 s_2 \ldots s_i$ has been written to form a full sequence $s_1 s_2 \ldots s_i \ldots s_n$ of a character pattern. For a stroke s_i, the reference zone is decided as a surrounding rectangle for the strokes up to the previous stroke i.e. $s_1 s_2 \ldots s_{i-1}$. There are three types of reference zones each of which has a different partitioning into sub-zones as shown in Fig. 6. Then, the stroke s_i is represented by a sub-zone name of a selected zone type where s_i is started and another where s_i is finished. This process goes to the last stroke until a string of sub-zone pairs is generated. The reference zone type for the first stroke is defined exceptionally by the first stroke itself. But the sub-zone pair for the first stroke with respect to the defined zone type can be omitted. An example of this representation scheme is shown in Fig. 7.

The recognition is made by a decision tree on the basis of complete match. A decision tree is prepared for each character set made of the equal number of strokes. For up to 5 stroke patterns, an arc of the decision tree correspond to a stroke sub-zone pair, while for more stroke patterns, an arc corresponds to a unit and the unit examines constituent strokes. The employment of units is to confine stroke variations in a unit and represent kanji patterns economically.

Clearly this scheme poses very strict constraints on pattern distortions and variations.

Kitahara and Isomichi [50] at Hiroshima Univ. set up 10 primitive strokes for the common set of kanji (1926 char.) and represented standard patterns of primitive stroke sequences by a directed graph with an arc corresponding to a stroke. A directed graph is a tree where arcs can be joined. Each stroke in an input pattern is approximated by $4 \sim 9$ coordinate points. Then, it is compared with arcs going out from the node until which the graph has been traversed, and the most similar arc is chosen. This graph traversing goes until the last input stroke. Comparison of the point representation of an input stroke with that of a standard stroke specified by an arc is made by DP-matching of a directional gap between a vector which links points (with some points possibly skipped) in the input stroke and another vector in the standard stroke. They have automated the directed graph from sample patterns. For 80 groups of 176 character patterns, directions from the start or the end of a stroke to that of another are examined as distinctive features.

Unlike a decision tree or automaton in the earlier researches, erroneous strokes can be accepted as far as the overall distance (sum of the distances of DP-matching) is within the threshold. But, the classification by a tree can never recover the misselection of an arc made at a certain node. Even if backtracking could be made, a misselection made near the root of the tree would be extremely fatal.

The research by Saeki and Matsushita is also syntactic [90]. The main purpose of this research is to establish a method to input 2-D representations such as line drawings and mathematical formulae. Associated with this problem is the recognition of attributed characters and symbols. Consequently, a writing box is not employed in which a character is written. For 62 alphanumeric characters (including both upper and lower cases) and 10 symbols, an input stroke is split into segments at sharp corners or cusps and each segment is recognized by a decision tree according to the turn direction, positional relation among its beginning point, end point and local X-, Y-maximum points, and so on. A stroke is recognized by another decision tree according to its constituent segment sequence. Cutting out of a character pattern (character segmentation) and its recognition is made again by a decision tree method which returns a character code when its constituent stroke sequence is traversed. When an input stroke does not match prescribed strokes or stroke relations associated with arcs are not satisfied, that input stroke is taken as the first stroke of the next character pattern and the preceding strokes are supposed to form a character pattern. These strokes, however, may not constitute a character pattern, then a backtrack is made until a character pattern is formed. Time lag of more than 0.8 sec. between strokes is also employed for character segmentation. They report human adaptation can realize from 97 to 100% recognition after 3 sessions, without the system learning.

Kato et al. at Fujitsu published a brief paper [39]. They represented a kanji pattern by a matrix M whose (i, j) element m_{ij} specified the positional relation ($x_i \gtreqless x_j$, $y_i \gtreqless y_j$) between the middle (x_i, y_i) of the i-th stroke and that (x_j, y_j) of the j-th stroke. A matrix representation of an input pattern is matched with templates and the most matched template is chosen. Progress after the paper, however, was not reported by them. Recently, Ishii has elaborated the method, which is described later.

Morishita et al. at Fujitsu investigated the nature of pattern variations under the direction of usual but neat handwriting [60]. Sample patterns thus collected were subjected to the recognition method by Kato et al. where only such standard patterns were prepared to detect pattern variations that represent right stroke number and order patterns. Stroke number variations from the standard are nearly 26% on average of 5 individual subjects, where the stroke number decrease is 24.7% while the increase is only 1.2%. Stroke number deviations within each person is still significant, i.e., 18% on average. On the other hand, stroke order variations and whatever else account for 4% within an individual and 99.6% of the total patterns remain within the 5th candidate. In summary, stroke number varies even within each individual, but stroke order is rather stable as far as the writer is specified.

As an interactive means to input Japanese text, it is true that on-line recognition may expect human adaptation to write characters to be recognized. For this purpose, algorithm transparency about how characters must be written to be recognized or why not recognized, is claimed to be important. Morishita et al. have shown that the recognition of Kato's method is improved by the registration of unrecognized patterns by more than 10% after the first session, but not thereafter [61]. They consider that this limitation has been caused by the lack of transparency as to misrecognition or rejection so that users could not adapt themselves to the system.

In this sort of researches, however, clear distinction must be made between the effect of machine "learning" and human adaptation.

Odaka et al. at NTT showed the superiority of the feature point representation of strokes against the Fourier coefficient representation described before for kanji patterns [81]. They first attempted an extension of the Fourier coefficient method for kanji patterns. To represent rather straight strokes of kanji patterns by the Fourier coefficients, however, higher order of coefficients are necessary. Moreover, many strokes of a kanji pattern multiply a large sequence of the coefficients by the stroke number with the result that time and space complexities for matching become too large. Therefore, they normalized each stroke length, catenated strokes and in-between pen movements, decomposed an entire character pattern into X-, Y-waveforms, and represented the character pattern by the catenation of the first $20 \sim 30$ Fourier coefficients for

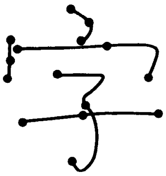


Fig. 8 Feature point representation of an on-line pattern.

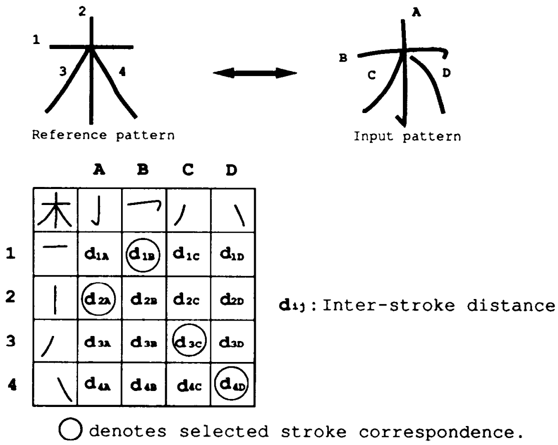


Fig. 9 Inter-stroke distance matrix.

X-wave form and those for Y-waveform. Then, they applied the Bayes classifier of a simplified variance matrix. Recognition experiments, however, revealed the following problems of this method:

(1) Local and distinctive features of a kanji pattern are not well expressed in the Fourier coefficients. (2) The method is fatal for stroke order variations since stroke reordering produces a completely different sequence of coefficients.

A better performance was realized by the following method, which was basically statistical but somehow structural. A stroke is represented by three points (two end points and a middle point) as shown in Fig. 8, and they are grouped into a sequence by stroke order to represent an input character pattern. This sequence is then statistically matched with standard patterns, where the covariance matrix for the classification is again simplified by only diagonal elements of variances.

They showed the relation between the former representation by the Fourier coefficients and the latter by feature points. The sum of the squared difference between the corresponding Fourier coefficients of two patterns is equivalent to the mean square distance between corresponding feature points of the two patterns. This implies that the two representations are equivalent at the extreme simplification with uniform variances and zero covariances among each set of features. However, the existence of non-uniform variances and non-zero covariances in the covariance matrix of the Fourier coefficients and that of feature points, the finite number of the Fourier coefficients in the former representation and the variable number of feature points in proportion to the number of strokes in the latter make them differ from each other both theoretically and practically.

They reported that the latter scheme realized a higher recognition rate by 1/10 recognition time with 55~70% memory space in comparison with the former. Their revised paper [83] reported that the method worked for kata-kana and even for hira-gana when 6 points were sampled to represent a stroke in hira-gana patterns. More than 99% recognition was reported for each set of character categories. For a mixed set of kanji, kata-kana and hira-gana characters, the scheme that a single stroke character pattern is approximated by 6 points, a stroke of 2 or 3 stroke patterns by 5 points and that of more than 3 stroke patterns by 3 points, presents again more than 99% recognition. For a single stroke character pattern, a full covariance matrix is employed (i.e., Maharanobis distance is measured).

This method would be robust against geometrical distortions of strokes but might be unstable against topological distortions or style variations. Especially, a hard limitation is expected for cursive and connected strokes, since their topologies can not be represented by only 3 or even 6 points. In addition, stroke order variations are fatal.

A research to remove the stroke order constraint is succeeded by Odaka et al. [82, 84]. When a system is demonstrated for many people, the constraint on stroke order is very serious. A stroke matching along with its sequence to form a character pattern inherently constrains the stroke order of input patterns. In order to remove the constraint, therefore, either all possible stroke sequences must be registered in the dictionary or stroke order information is abandoned. They have chosen the latter. This implies that all the combinations of stroke matching have to be considered. With respect to every stroke in a standard character pattern, the distance to every stroke in an input pattern, collectively inter-stroke distance matrix (ISDM), is calculated as shown in Fig. 9. Then, the standard stroke to input stroke correspondence that produces the least inter-character distance (ICD) is chosen, where the ICD is the sum of inter-stroke distances (ISD's) of a one-to-one stroke correspondence. Because of n^2 stroke matching where n is the number of strokes, however, the recognition time gets 20 times as much as the preceding method. Moreover, the increase of recognition rate from the above is not clear probably because sample patterns are too good so that the recognition rate has been already saturated even by the preceding method.

The research associated with JOLIS-1 is also placed around this period just before the third generation [68, 69, 71, 72]. Its major result is the introduction of stochastic dissimilarity among primitive strokes and its effect shown in relation to the dictionary expansion of standard character patterns.

4.4 The Third Generation

The next hot period was formed around 1983 ~ 1985 after the advent of Japanese wordprocessors and word-processing software for personal computers. An announcement of a wordprocessor with on-line handwriting input by Matsushita Electric Industrial Co. [79, 57, 116] stimulated other computer manufacturers to make products in the OA movement, such as SHARP, EPSON, SANYO, Hitachi, Toshiba and so on. The announcement was also enthusiastically welcomed. In fact, it was selected as one of the 10 major products in the year 1985 by the NIKKEI paper. It was expected to provide an ordinary Japanese who would not like to use a keyboard with a very natural way to input Japanese test. But, the expectation could not be satisfied. It could not realize what is most important, i.e., a satisfactory level of recognition performance. In a year or so, all the manufacturers substantially retreated.

The recognition scheme employed by Tomimoto et al. [115] is basically similar to that by Furukawa and Mita [15] and the scheme by Inoue et al. [28] in the first generation, although ambiguous strokes are represented by or-strokes. The method is classified in syntactic approaches and it defines 28 primitive strokes. Data points of an input stroke are segmented into a sequence of 8-directional straight-line segments. This segment sequence is matched with a standard segment sequence of each primitive stroke. The primitive stroke is chosen to which the cumulative length of mismatched input segments is shortest. Between successive strokes, 4 kinds of off-strokes of 8-directional are derived which interlink from the start or the end of a preceding stroke to that of a succeeding stroke in 4 combinations.

Kanji patterns are basically represented by a sequence of strokes. More than 300 kinds of subpatterns may replace stroke subsequences, thus reduce the size of pattern representations by 40%. The notable features of the representation is the specification of stroke variations by or-strokes, liberated stroke order in order-free stroke subsequences and the designation of inter-stroke relations by the off-strokes. Distinctive features are employed when necessary.

Kanji patterns are recognized by matching of codes for strokes and whatever else and the standard character pattern of the least number of mismatched codes is chosen. For hira-gana character patterns, on the other hand, the stroke identification is skipped and the segment matching is made over a character pattern. The standard character pattern is chosen to which the total length of the mismatched input segments is minimal. For a mixed set of character patterns, if an input character pattern is made of less than 7 strokes, it is first subjected to the hira-gana recognition then to the kanji recognition. Otherwise, the kanji recognition is directly applied.

For handprinted character patterns that do not include running or skipping strokes, 96.8% recognition is reported for kanji and 95.4% for hira-gana. In this method, the similarity concept was incorporated but it was oversimplified. Pattern variations could be prescribed but those unexpected were hard to recognize. The method is characterized to be an effort to make a product rather than an attempt to attack various pattern variations.

The problem of recognition time in the stroke order free algorithm by Odaka et al. [82, 84] was settled by the employment of hardware logic [85]. The recognizer realized 6 ~ 500 msec. recognition time per character. It was combined with a full-keyboard wordprocessor and named AESOP. The report on AESOP by the NIKKEI paper actually preceded that of Matsushita's wordprocessor. But, it was not until the latter that people could try the taste of on-line handwriting wordprocessing.

A brief report on a handwriting input terminal was also made from NEC by Ito et al. [37]. It also employs hardware logic to implement a DP-matching method by Yoshida and Sakoe [132]. The method is elaborated as presented below.

Ooi et al. at Toshiba corp. present another stroke reordering [88, 96, 21]. By comparing direction from the start to the end of a stroke and coordinate values of the feature points (starting point, middle point and end point) of a stroke to dictionary representations, disordered strokes may be ordered into the correct sequence. 99.5% recognition is reported but the method would not be able to cope with running or skipping strokes. Unfortunately, the detail of the method is not presented.

Some fundamental researches to relax constraints on character patterns to be recognized have overlapped with the rush of products or attempts to produce them. Advance of microelectronics has greatly improved hardware environments to make software attempts which require heavy and large complexities of time and space. Development of low-cost yet better tablets has also lightened the problem of human interface in hardware.

At NTT, their recognition method was further elaborated to accept connected and broken strokes as well as wrong stroke order patterns by Wakahara et al. [117]. First, a stroke is approximated by feature points at regular intervals rather than by a fixed number of points. A stroke in a standard pattern has been also approximated by feature points at the same intervals. Second, an ISDM (inter-stroke distance matrix) is calculated with regard to a standard character pattern assumed, where an input stroke and a standard stroke is matched from their starting points to the end of the shorter stroke with the city-block distance between each corresponding pair of feature points accumulated. The remaining feature points of the longer stroke are left at this stage. Third, whether it is an input pattern or a standard pattern, every stroke in the fewer stroke pattern is coupled with one of the strokes in the more stroke pattern without many to one correspondence so as to yield the least ICD (inter-character distance). Fourth, an uncoupled stroke in the more stroke pattern is catenated to a matched stroke either preceding or succeeding which yields better inter-stroke matching. Then, the number of strokes becomes equal. This process is termed as selective stroke linkage. Fifth, an ISD (inter-stroke distance) is recalculated between a pair of corresponding strokes by DP-matching with the city-block distance of matched feature points. The total sum is taken as the final ICD.

A standard pattern of right stroke number and order is prepared per character category for the total of 1851 kanji character categories. Input patterns of 126 categories which contain stroke connection variations of 5 common types and arbitrary stroke order variations are subjected to recognition experiments with the result that 98.9% correct recognition is reported. Input patterns of correct stroke number and order are recognized 99.8% correctly.

In order to recognize deformed character patterns, Wakahara and Umeda introduce knowledge about how strokes are likely connected as stroke linkage rules [118, 119] in addition to the feature point representation of standard patterns employed in the previous system. Actually, strokes in standard character patterns are classified into three according to where and how they appear in deformed patterns.

The previous method is slightly simplified and used for the preselection of candidates. The processes up to the selective stroke linkage are applied with only the both ends of a stroke taken as its feature points. Then, an ISD is recalculated with the end points of a linked stroke. The total sum of the ISD's between corresponding strokes is a preliminary ICD between an input character pattern and a standard character pattern. According to this distance, from the top to the 50th least distant standard patterns are selected as candidates.

Finer selection starts from the generation of stroke connected patterns from each standard pattern. Strokes are connected so as to make the same number of strokes as the input pattern and conform to stroke linkage rules. Then, an ISDM (inter-stroke distance matrix) is calculated to accept stroke order variations between the input pattern and a generated pattern, where the ISD is measured by DP-matching with the city-block distance of matched feature points. The ICD is obtained in the same way as the previous method. ICD's are calculated between the input pattern and all the generated patterns from a standard pattern and the least value is taken as the definitive ICD between the input pattern and the standard pattern.

This method is applied to the same sample patterns as the previous method and the recognition rate has been improved from 98.88% to 99.42%. Deformed patterns written without any instruction or direction are reported to be recognized 95% correctly. It is apprehended, however, that running stroke patterns generated from wrong stroke order or unexpected stroke connection would be hard to recognize.

Kimura and Odaka at NTT extract the hidden side of pen movement as character pattern features [45, 46]. Among dark strokes which link between real strokes, some are turned into running strokes, but others stay and present important information on character pattern structure. Each dark stroke in an input pattern is matched with those in a standard pattern and the best correspondence from input dark strokes to standard dark strokes is determined. Some standard dark strokes may remain unmatched. Here, the distance between dark strokes is defined as the sum of the city-block distance between the beginning points and that between the end points. An ICD is the sum of the distances of the best dark stroke correspondence. A sample pattern per category and per person is collected from three people for 214 character categories normally made of 10 strokes. Two subjects present rough patterns and more than 80% of them include stroke connections. By this rather simple method, a correct answer is nominated within the 50th candidate for more than 95% input patterns. The method may be useful for preselection of candidates.

As for a standard pattern, real strokes between matched dark strokes must have been turned into a running stroke. Then, the matching between an input stroke and corresponding standard strokes may contribute to express the ICD more correctly. But, it is shown that deformation of standard strokes due to running hand works contrary to the above expectation.

Stack DP-matching has been proposed by Yoshida and Sakoe at NEC corp. for the recognition of running hand patterns with limited stroke order variations [133]. It first approximates an input pattern by a sequence of 16-directional straight-line vectors. Pen movement while it is off, is also represented by a vector and interleaved with pen-on vectors. Given an input sequence of vectors $a_1a_2...a_n$, it is matched with a standard pattern $b_1b_2...b_m$. Standard patterns may share common substructures.

Matching is basically DP-matching where the distance $d(i, j)$ between the vectors a_i and b_j is their directional difference with the penalty regarding pen's on/off condition. "Stack" is prefixed to the method since it employs a stack to handle multi-way alternatives in a vector subsequence.

9270 sample patterns of 881 educational kanji character and 46 hira-gana character categories were collected from 10 people for experiments. Provided that wrong stroke order patterns, that is 7.4% of the total, are registered in the standard patterns, they report 98.3% correct recognition for the above character set except 27 characters of 12 groups, each of which has an identical standard representation. Finer classification by the Yes/No decision is added that examines stroke length balance, stroke crossing and so on, with the result that 99.8% is reported for the total character set.

For running and skipping (being chopped) stroke patterns, pen pressure as well as coordinate value is employed in DP-matching by Sato et al. currently at Nagoya Insti. of Tech. [97, 98]. When a pen is running or skipping, pen pressure ranges continuously within a small value or 0 (during pen off), while it is digitized on a paper as a pen touch/untouch sequence. A stroke sequence of an input pattern is linked by straight-line pen movements with 0 pen pressure, the size is normalized around the center of gravity and then coordinate value and pen pressure is sampled at regular intervals.

Pattern matching is made by DP-matching where distance between sampled points of an input pattern and those of a standard pattern is defined by:

$$(1-\mathrm{Wp})|\text{coordinate distance}|^2 + \mathrm{Wp}|\text{difference of pen pressure}|^2$$

where Wp: weight.

For 416 kanji characters made of a small number of strokes and 143 characters of hira-gana and kata-kana, the highest of 97.1% correct recognition is reported with the weight Wp for the pen pressure adjusted, while 96% without the pen pressure.

Stroke number varies in 25% of kanji and 10% of kana patterns even within patterns presented by a single person. Those stroke number variations were correctly recognized, but there were 2% style or stroke order variations that the method could not cope with.

Pen pressure information is effective if it is employed for each person, but its variations are large among people with the result that standard patterns must be prepared for each individual.

The recognition time of an input character pattern is not presented but it is suspected that a practical use of the method for text input would be difficult without a dedicated hardware such as a DP-matcher.

In relation to the above researches with DP-matching, the work by Ishigaki and Morishita [29, 30, 31, 32] at Fujitsu is worth mentioning here, although their work extends over the fourth generation. They propose an elastic matching of feature points with backtracking rather than DP-matching. Feature points are edge points and corner points. Suppose that a sequence of feature points $a_1a_2...a_n$ is matched with another sequence $b_1b_2...b_m$. First, a_1 is paired with b_1. Then, assume that the matching proceeds and a_i is successfully paired with b_j. The next matching pair is chosen from either a_{i+1} and b_l $(j<l\leqq m)$ or a_k $(i<k\leqq n)$ and b_{j+1}. A feature point which satisfies certain conditions is treated as an indispensable feature point and must be paired with some point. Suppose a_k and b_l are chosen for pairing. The distance of this feature point matching is defined by the weighted sum of (1) directional gap between $\overrightarrow{a_ia_k}$ and $\overrightarrow{b_jb_l}$, (2) the length difference between them, and (3) the directional gap between $\overrightarrow{a_ka_{k+1}}$ and $\overrightarrow{b_lb_{l+1}}$. When a plural set of pairs produce the distance less than the threshold, the pair of the smallest distance is selected as the next correspondence and the others are saved on a stack. When no pair produces the distance less than the threshold, backtracking is made. If no possible alternative is left on the stack, the matching is quitted.

They report 94.2% correct recognition for stroke connected patterns. Backtracking occurs only less than 0.5 times for a character matching. Matching with a wrong standard character pattern is quitted after 4 or 5 times of feature point matching. These facts enable the system to recognize an input pattern in about 1 sec. They claim that the indispensable feature points bound the matching so that superfluous matching cases are pruned.

An orthodox syntactic approach is taken by Yurugi et al. at Oki Electronic ltd. [135]. They prepare 76 primitive strokes and 671 fractional subpatterns, among which 206 are character patterns by themselves, for 3280 character categories including 3059 kanji, hira-gana, alphabets and symbols.

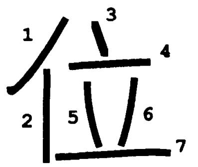
A fractional subpattern is recognized by the unordered set of constituent strokes and a character pattern by that of subpatterns respectively, so that the stroke order in a subpattern and subpattern order to form a character pattern are free. Stroke number variations are, on the other hand, provided in the subpattern definitions.

They report 95% recognition rate for 543 character categories which covers all the subpatterns. This method could be regarded as a typical example of syntactic approaches. The method should work well for ideal patterns but expose its defects against distortions and variations which have not been expected in the dictionary. Especially stroke transformations due to serif, running hand, and so on seem to be fatal.

4.5 The Fourth Generation

Recently, Ishii at Fujitsu extended the approach by Kato et al. [39] and constructed a stroke order free recognition scheme for character patterns not including stroke connection [33, 34, 35].

Ishii first studied inter-stroke relation of a kanji pattern by means of the up/down or right/left relation between any combination of strokes. A stroke is actually represented by a feature point which may be the start, middle or end point of the stroke. Sample patterns of 200 kanji categories were collected from 40 people under the condition of correct stroke number and order. He has shown that at least either the up/down or the right/left relation is very stable between any pair of feature points and if whichever stabler is chosen as the relation between feature points, the stability of the relation is more than 99% regardless of the kind of the feature points. Moreover, it has been shown that the stability is raised to 99.99% provided that the kind of feature points can be selected from the three, i.e., the start, middle or end point, for each pair of strokes. This



$$\begin{pmatrix} ** & RU & LU & LU & LU & LU & LU \\ LD & ** & LD & LD & LU & LU & LU \\ RD & RU & ** & RU & RU & LU & RU \\ RD & RU & LD & ** & LU & LU & LU \\ RD & RD & LD & RD & ** & LD & RU \\ RD & RD & RD & RD & RU & ** & RU \\ RD & RD & LD & RD & LD & LD & ** \end{pmatrix}$$

R(right), L(left), U(up), D(down)

Fig. 10 A kanji pattern and its structure matrix.

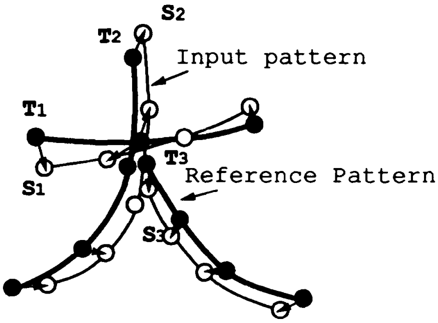


Fig. 11 Deformation vector field.

implies that the three kinds of feature points complement one another to represent inter-stroke relation of a kanji pattern.

Then, he has defined the structure matrix whose (i, j) element represents the up/down and right/left relations between the feature points of the i-th stroke and the j-th stroke in a character pattern as shown in Fig. 10. It is proved that a structure matrix can be converted to a tree description if either the up/down or the right/left relation is stable. This tree description denotes horizontal or vertical decompositions of a kanji pattern hierarchically into a single stroke, so that it does not reflect stroke order but purely represents inter-subpattern relation and inter-stroke relation. The conversion is shown to be 99.5% correct for 40 people's patterns of 198 categories, if a structure matrix with respect to a kind of feature points can be switched to another with respect to a different kind when the decomposition of a kanji pattern by the former gets blocked. According to the tree description of a correct character category, an input character pattern of the right stroke number but wrong stroke order can be reordered into the right sequence.

Under the condition of the correct stroke number but stroke order free, sample patterns of 203 categories made of 10 strokes were collected and subjected to the stroke reordering method and recognized 99.5% correctly by a simple stroke to stroke matching with the Euclid distance. This recognition was realized by twice the memory size and 1.5 times the processing time of the recognition without stroke reordering.

The worrying problem is the robustness against pattern variations. Extension of the method for running stroke patterns is anticipated.

A seek for the ultimate recognition of deformed patterns are continued at NTT by Wakahara et al. [120]. As the result of DP-matching in the previous method, a set of vectors is derived each of which links from a feature point in a stroke connected pattern generated from a standard pattern to a matched feature point in an input pattern. This set of vectors is termed as a deformation vector field (DVF). An example is shown in Fig. 11. In the previous method, an ICD (inter-character distance) was the total sum of the city-block length of every vector. This time, the stroke connected pattern selected as the reference pattern is further deformed with deformation prediction superimposed. Deformation is predicted by considering continuity, consistency, or temporal causality of the time sequence vectors in a DVF. Then, the ICD is calculated between this and the input pattern. The recognition rate of deformed patterns has been improved from 95.2% to more than 97% due to any of the three kinds of deformation prediction.

Yet another method with DVF is proposed by Wakahara [121]. Around each feature point in a reference pattern, a local Affine transformation (LAT) is defined which approximates neighboring deformation vectors by the least square approximation. The remaining vectors are further approximated by an even localized Affine transformation. That is, each deformation vector in a DVF can be expanded into a series of LAT's. An ICD is calculated between an input pattern and a reference pattern with low-order series of LAT's superimposed. More then 95% recognition is reported for the sample patterns to which the method without DVF [118, 119] achieved 91.9% recognition.

The latest two methods are based on DVF as the results of DP-matching. They seem to be very powerful but it takes a considerable amount of processing time. It is reported to take 190 sec. on a 1 MIPS computer to select 10 candidates before the stage with DVF starts. Further processing time is needed for the deformation of standard patterns. To realize the methods as an online input medium of Japanese text, it is essential to wire down the algorithm. Another problem associated with them is that they can not recover wrong correspondence by the DP-matching which determines DVF.

Kimura and Miyahara at NTT relax the constraints on stroke number in Ishii's method by the combination with NTT's stroke order and number free algorithm [49]. Given an input pattern, stroke connected patterns are generated from a standard pattern so as to produce the same number of strokes as the input pattern. Then, the stroke order free matching is carried out by means of ISDM (inter-stroke distance matrix) so that the correspondence between input strokes and standard strokes is made. Once the correspondence is obtained, the input strokes can be ordered according to the standard stroke sequence. For this ordered input strokes, Ishii's structure matrix is generated.

Ishii employed the structure matrix to order input strokes of a character pattern, while they define the distance between the structure matrix of an input pattern and that of a reference pattern such that it reflects the proportion of mismatched elements between the two matrices.

The ISDM of NTT expresses the dissimilarity of stroke shape and position while the structure matrix distance represents the difference in positional *relations* of strokes. Their correlation is shown relatively small, so that they may complement each other. In their experiments, the former produces better recognition than the latter by itself, but the latter can recognize 68% of the sample patterns which cannot be recognized by the former. The proper combination of them may produce better performance. As a matter of fact, the recognition of running stroke patterns has been improved from 91% correct recognition and 4.9% misrecognition with ISDM alone to 96% recognition and 2.3% misrecognition by the weighted sum of the both distances.

The last but hopefully not the least is a syntactic approach under the JOLIS (Japanese On-Line Input System) project by our group at Tokyo Univ. of Agri. and Tech.

Presentation of JOLIS begins with preprocessing of sampled data points. But a general description must precede a particular method. Therefore, a brief survey on preprocessing is inserted here. Then, the story of JOLIS follows.

The aim of preprocessing is to reduce fluctuation of user's pen motion and quantization noises of a tablet digitizer. Even unsatisfactory preprocessing adopted in JOLIS-0 and succeeded by JOLIS-1 provided them with increased performance in comparison with Pre-JOLIS. But, deformation of stroke shape by preprocessing was sometimes serious.

Smoothing and thinning have been used to reduce the above noises [16, 17, 111, 7, 9, 13, 90].

Smoothing is a process to average irregular noises. There are two types of smoothing, auto-regressive type and moving-average type. The former extracts topological features more robustly than the latter. Moreover, it tends to remove small spurious loops which often appear in place of sharp cusps. On the other hand, however, the effect of smoothing accumulates to the end of a stroke and smoothed data points get gradually behind from their original data points because of the auto-regressiveness. This is fatal in such a case of discriminating patterns shown in Fig. 12.

The other process common in preprocessing is thinning. Thinning thins out superfluous data points sampled from slowed down pen movement.

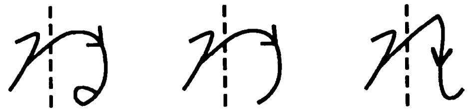


Fig. 12 Strokes likely damaged by auto-regressive smoothing.

The order of smoothing and thinning affects the result. Groner [16] applied smoothing and thinning in this order. The problem with this order is that the smoothing effect is affected by pen speed. Namely, fast pen movement with sparse data points is strongly smoothed but slow movement with dense data points is not much smoothed. This implies, conversely, that slower and sluggish part in a stroke tends to have higher fidelity to an original pen track while faster and running part may have less fidelity.

On the other hand, the opposite order employed by Terai and Nakata [111], S. Fujiwara et al. [13] and Hanaki et al. [17] has the problem that sharp cusps are dulled. At a cusp or sharp corner, pen movement normally slows down and many data points are sampled. If smoothing is applied first, it preserves the sharp shape owing to a sufficient number of data points around there as mentioned above. But if thinning is applied first, those data points are thinned out and just a few points represent the sharpness. The result is that the weight of raw data points forming a cusp is greatly decreased, and the succeeding smoothing eventually dulls it.

In either case, deformation is observed in distinctive features. The problem may be summarized as how to transform time variant data stably to space variant data. Preprocessing is a front end process and its performance determines the recognition accuracy of a whole pattern recognition system to a great extent.

Aizawa et al. of the JOLIS group present a signal processing approach toward optimum preprocessing in the sense of directional stability of preprocessed data points and their fidelity to original data points [4]. Conventional preprocessing methods have been evaluated with the two measures, and the incorporation of adaptive sampling which thins out sampled data points while stressing slowed down pen movement, as well as interpolation has been proved to improve preprocessing in the both measures. Consequently, their preprocessing is composed of four stages: (1) adaptive sampling, (2) interpolation, (3) smoothing (moving average type) and (4) thinning. By this preprocessing, original time variant data are transformed into almost time invariant and space variant data with feature points stressed. An example is shown in Fig. 13.

Nakagawa et al. show the quantitative evaluation of stochastic dissimilarity [72, 73]. JOLIS-1 incorporates stochastic dissimilarity in its syntactic recognition scheme to recognize a mixed set of kanji, hira-gana, kata-kana and several symbols. JOLIS-1 classifies each

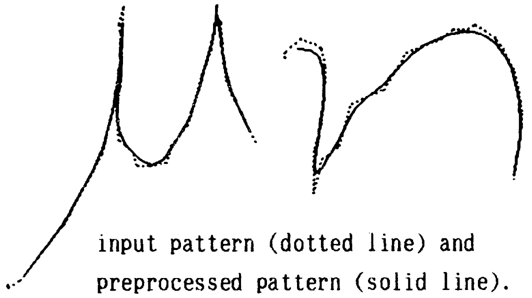


Fig. 13 Result of preprocessing.

input stroke into one of primitive strokes and then, recognizes an input character pattern by dissimilarity matching of its stroke sequence with those of registered patterns. Character dissimilarity is obtained from stroke dissimilarity which is stochastically defined by the probability of stroke (mis)identification with respect to a certain character pattern dictionary. When the character pattern dictionary is small, the dissimilarity plays the principal role to cover distortions or variations. As it becomes large, on the other hand, the role of the dissimilarity decreases and multiple standard representations take over the role. Consequently, the distribution of each category is better approximated.

As a matter of fact, the stochastic dissimilarity has a notable effect for a small dictionary of character patterns. The recognition rate has been improved by 1.6~2.0% from about 90% in comparison with the heuristic dissimilarity. As standard patterns are registered, on the other hand, the effect of the stochastic dissimilarity becomes less notable.

Another important fact revealed in evaluation experiments is the deficiency of a set of 29 primitive strokes to represent patterns composed of a few strokes. The information derived from a character pattern is roughly proportional to the number of strokes so that character patterns composed of a few strokes only convey a small amount of information by means of their stroke sequences. Dissimilarity is effective only when significant information over noise is sufficiently high.

They also show the effect of private dissimilarity and dictionary. Customization seems to be essential for pattern recognition systems to work for a general application of text input.

Aizawa et al. consider the intrinsic problem of pattern primitive selection [3]. The problem of how to define primitive strokes is a particular case of the general problem of pattern primitive selection in syntactic pattern recognition. So far, almost all attempts in syntactic approaches have been keeping away from this problem and defining a set of primitives empirically or intuitively as we did in JOLIS-1. Consequently, whether feature extraction is adequate and stable depends on designers' art. We may expect such art for classification of a small number of pattern categories. But it is quite difficult for the Japanese character set because of its number and variety of patterns. For a certain class of character patterns, extracted features get likely inadequate or noisy, so that the robustness by means of stochastics, (dis)similarity, or whatever else, is not realized.

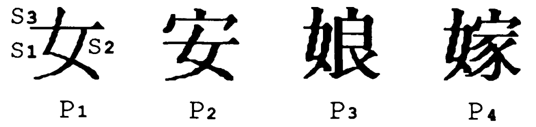


Fig. 14 Some Patterns including P_1.

Considering pattern occurrence as an information source, pattern primitive identification is viewed as an information channel. Then, a pattern primitive set should be defined to maximize the information transmitted while keeping its set size computationally small. 134 pattern primitives have been thus selected and their identification procedure has been realized for 2265 categories of alphanumerals, Greek letters, kana and kanji patterns.

The identification procedure conveys 6.7 bit per pattern primitive while that of JOLIS-1 only transmitted 3.8 bit. They also present semi-automatic generation of the pattern primitive dictionary and the dictionary of character patterns composed of those pattern primitives.

In the researches with JOLIS, we have gained the following knowledge on style variations and distortions:

(1) There are wide variations among people in the stroke order to form a character pattern, but it is considerably stable for each person. (2) Wrong stroke order patterns and simplified/deformed patterns appear in only ten or some more fractional subpatterns. Moreover, the way that a subpattern is distorted is consistent among all the character patterns which contain it. Fig. 14 depicts an example. People who write P_1 in the order of s_3, s_1 and s_2, consistently use this order in writing of P_2, P_3, P_4 and so on. (3) It is hard to expect where strokes are likely to be connected. Stroke connection is unstable even for one person.

The above three guide us to utilize stroke order information in application of an elastic matching such as DP-matching for running stroke patterns and to pursuit the advantage of syntactic approaches in the hierarchical representation of character patterns with subpatterns shared for wrong stroke order patterns and simplified/deformed patterns.

Komoda et al. present the advantages of hierarchical representation of kanji patterns quantitatively [51]. Standard patterns are organized into grammatical structures with 472 fractional subpatterns. Consequently, the number of stroke and subpattern symbols in the character pattern dictionary is reduced to 1/2 and the work of character pattern dictionary construction and maintenance is lightened in proportion to the above.

Grammatical representation and matching with stochastic dissimilarity realize a combination of syntactic and statistical approaches. Their approach is claimed to be syntactic even if parsing is not performed, since representation is more fundamental than implementation.

Sawai et al. present an attempt to recognize running stroke patterns in the syntactic approach [100]. It follows basically JOLIS-1's syntactic recognition scheme with stochastic dissimilarity. It first identifies each stroke in an input character pattern to one of the primitive strokes of JOLIS-1 with running strokes possibly unidentified or misidentified. Then, it applies the dissimilarity matching with standard pattern representations in the character pattern dictionary.

The key difference from JOLIS-1 is the introduction of 8-quantized direction of pen movement (dark stroke) between strokes in the character pattern representation and a kind of elastic matching to treat stroke connection.

In the course of matching an input pattern with a standard pattern representation, if an input stroke satisfies certain conditions, it is assumed to be a running stroke and catenation of standard strokes and dark strokes are made until the same number and sequence of turns are formed. A dark stroke is generally visualized as a straight segment when two strokes around it are catenated. But, if its visualization forms an opposite turn to its corresponding turn in the input stroke, the dark stroke is extinguished. If a proper stroke forms an opposite turn by its catenation, on the other hand, all the catenations for the input stroke is cancelled and the input stroke is taken as a badly distorted stroke from the corresponding standard stroke rather than a running stroke. The heuristic assumption is to drastically cut down matching cases and dispense with DP-matching or exhaustive search with backtracking.

The heuristic matching and that with the unit of strokes rather than a smaller unit such as segments have realized the recognition time of about 2.0 seconds per input character on a 16-bit microprocessor. For 323 kanji and hira-gana character patterns which contain running strokes, 55% of the testing patterns have been correctly recognized.

5. Looking Back the Recent Researches

Some approaches ignore the order of strokes and treat them as a set rather than an ordered sequence in order to accept wrong stroke order patterns [82, 84, 117, 118, 119]. Matching of the two sets of strokes for input and standard patterns is made by arbitrary correspondence of strokes.

Whatever features may be extracted from a stroke. As far as a character pattern is represented by an ordered sequence of strokes, and fractional subpatterns are not shared among character patterns, to accept wrong stroke order of a subpattern implies registrations of wrong stroke order patterns to all the character patterns which include the subpattern. Then, they preferred to abandon stroke order information. Clearly, this is not annoyed by wrong stroke order, but would not reject complete disorder of strokes. More serious is that these approaches must consider the combinatorial order of stroke connections to recognize running stroke patterns. Consequently, they need to mark stroke pairs which are likely to be connected. But, it is clear that unexpected stroke connections cannot be dealt with.

Invisible features such as stroke order, pen speed and so on are not explicitly employed for human recognition with the result that distortions from the standard are unlikely corrected. But, this does not mean that the stroke order is completely random. There are wide variations from person to person, but the stroke order is pretty stable for each person. In pattern recognition, almost all features are individually unstable, but they provide enough and even redundant information to classify patterns as a whole. Unless a feature is completely random, it could be utilized.

Other approaches apply DP-matching between an input pattern and a standard representation of data points [97, 98], or of straight-line segments [133], to recognize running stroke patterns. A possibility is considered in DP-matching that a pen movement between strokes is turned visible. Until their researches, dark strokes between real strokes had been utilized just to code the positional relations of real strokes.

Recent researches with DP-matching apply it to an entire pattern to recognize cursive patterns, while Kitahara and Isomichi [50] as well as Ikeda et al. [13, 25, 26] adopted DP-matching for just stroke identification. This fact reflects the innovation of microelectronics and a very wide availability of high performance CPU's. But, even with the current hardware level, DP-matching over an entire character pattern is too heavy for real-time recognition without dedicated hardware.

When a system is demonstrated for many people, the constraint on stroke order is very serious. But, when a system is personalized, the constraint on stroke separation gets fatal. Considerable efforts have been made to relax the constraints on stroke order and number. At present, however, a method free from the one is not completely liberated from the other.

In recent studies, syntactic approaches are not just syntactic and statistical ones are not only statistical. On one hand, syntactic approaches attempt to incorporate statistical nature of pattern distribution. On the other hand, statistical approaches try to reflect the structure of patterns in the representation of standard patterns so that they can expect pattern variations. Classification into either syntactic or statistical might be misleading.

The JOLIS approach introduces (dis)similarity between pattern primitives based on their confusion matrix. It provides the syntactic approach with better robustness. But, the assumption of either independent or Markov occurrence of pattern primitives contradict with syntactic rules which prescribe occurrence of pattern primitives in a certain context.

A series of researches at NTT is remarkable extension of statistical approaches, and it is beyond the scope of them. But, it is worth noting that the extensions have not been made for nothing. In orthodox statistical approaches, a wrong stroke order pattern is considered to have a different covariance matrix of statistical features from that of a right stroke order pattern. The former can not be obtained from the latter just by the replacement of the matrix elements. In the NTT approach, the covariance matrix of features is simplified to only diagonal elements of variances and the simplified matrix (not a matrix but a vector) is used for any stroke order pattern with the variances interchanged according to stroke reordering. The serious simplification has been made.

For stroke order free and number free recognition, the inter pattern distance has been further simplified. The city-block distance or the distance by DP-matching do not reflect covariances or variances of features. Therefore, the methods are no more statistical in a strict sense.

The distinction between "statistical" or "syntactic" has become vague and might be meaningless. It is important that the simplification is proved safe for target patterns as well as it introduces benefits of the other approaches.

To attack noisy and distorted patterns usually entails the increase of standard patterns for them. A registration of a pattern variation for a subpattern implies registrations of all the character pattern variations including it. Combination of subpattern variations produces a combinatorial number of character patterns. Stroke order free matching methods work equivalent to standard patterns of all the permutations in stroke order. Too many standard patterns per category thus generated eventually may hinder recognition of one another, if adequate information is not extracted. To elaborate a recognition method for bad patterns might damage recognition of good patterns.

A pity in the research history of this field is that almost all the researches have been destined to report more than 99% correct recognition rate since the early beginning. But, clearly it depends on the quality of patterns. In fact, Kimura and Miyahara shows how the recognition is degraded according to the liberation of handwriting condition [48]. The problem is that we still don't share benchmark data of sample patterns in order to evaluate various approaches on the common basis. Unlike our field, benchmark data is provided for the OCR researches by Electro Technical Laboratory of Japan by S. Mori et al. [59, 93].

6. Miscellany

In the OCR researches, the use of linguistic information has been studied by Osada et al. at Kyushu Univ. [89] and Shinya et al. at NTT [102]. The method is clearly applicable for on-line recognition if it is effective. On a wordprocessor or personal computer, a Japanese language dictionary is normally implemented for kana-kanji translation, so that it would be available for such purposes. But, to realize really helpful assistance of recognition by linguistic information, a very deep and tough problem of natural language understanding must be studied. The current level of linguistic information processing is a practical compromise for the realization of Japanese wordprocessors, although it truly made a significant impact. Moreover, a kind of error correction may turn the result of recognition even more unreliable, unless a sufficiently high level of recognition has been achieved.

Writing box free recognition has been attempted by Murase et al. at NTT [64]. Judging from the result that the recognition rate can be damaged to nearly 50% by the relaxation of writing constraints reported by Kimura and Miyahara [48], however, a relaxation of this constraint seems to be too early for a practical sense. It would also hold for the above linguistic approaches as well as this approach that the level of a single character recognition must be improved much more for distorted and noisy patterns.

As an application of on-line recognition techniques, computer aided instruction (CAI) systems to teach correct handwriting of Japanese characters have been suggested by Yamasaki et al. [126, 127] and by Baskerville and Shirai [10].

The system by the former group is designed for Japanese children to learn to write well-balanced characters through personal lessons. For this purpose, even local and topological fine features of character patterns are stored in the character pattern database so that input patterns can be tested for those fine features. This can be done, since a character category provided by a learner is the one designated by the system.

Baskerville and Shirai have proposed a structural recognition method based on primitive strokes and their relations for foreign people to learn correct writing of Japanese characters [10]. They claim that structural features which are meaningful for people must be extracted so that distortions in those features are to be explained to and understood by learners. In other words, statistical quantities may be useful for pattern recognition but their deviations from the standard may not be directly fed back to them for correct writing.

Owing to the very wide spread of personal computers and price-down of small tablet digitizers, the environment is getting more and more favorable for the research and development of this sort of systems.

7. Conclusion

As the most hopeful non-keyboard input method, on-line recognition of handwritten Japanese characters has been surveyed. The survey was aimed exhaustive. Research activities have been divided into four generations including 2 times of hot periods. The first generation was to know how noisy, distorted and erroneous human writing patterns were. The second generation between the hot periods started researches to recognize such human patterns under certain constraints. The third generation of the second hot period is characterized by the fuss about products of script recognition in the OA movement as well as researches to liberate writing constraints based on the remarkable progresses in hardware. The latest generation studies intrinsic problems of pattern recognition in the domain of on-line recognition of handwritten characters. The number and variety of Japanese character patterns are ever challenging.

Throughout the four generations, considerable efforts have been made to relax constraints on character patterns to be recognized. Namely, stroke order free and number free recognition methods have been sought. At present, however, a method free from the one constraint is not completely free from the other.

In recent studies, syntactic approaches are not just syntactic and statistical ones are not only statistical. On one hand, syntactic approaches attempt to incorporate statistical nature of pattern distribution. On the other hand, statistical approaches try to reflect the structure of patterns in the representation of standard patterns so that they can expect pattern variations. Classification into either syntactic or statistical might be misleading.

Fundamental studies are being continued, but there still remains a considerable amount of work to be done for on-line recognition to be employed for general text input. Benchmark data of sample patterns are essential to make steady steps based on reliable evaluations.

The related issues of text editing, formatting and the more global issue of human factors are left untreated. Once the key technology of recognition has been established, numerous ideas can be implemented. As a matter of fact, such an attempt is found even in the first generation. Text editing on a tablet with conventional correction marks has been a target of researchers since Terai and Nakata [114].

Both syntactic and semantic processing of Japanese text would be helpful for general pattern recognition methods to work as text input methods. To study a structure which accommodates cognition of character patterns, their recognition, understanding of the language and cooperative interactions among them, is intrinsically important and ultimately contributes to construct a highly personalized or customized user interface in the future.

Handwriting input as well as voice input has potential merits which can not be measured by performance or cost-effectiveness of text input. These merits would be considered even more important as the technology progresses.

References

1. Adachi, H., Hirano, K., Kaneko, M. and Sato, Y. Improvement of Online Recognition of Cursive Writings (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1522.
2. Agency of Industrial Science and Technology ed. 大型プロジェクトパターン情報処理システム研究開発成果発表会論文集 (in Japanese), (財)日本産業技術振興協会 (1980).
3. Aizawa, T., Nakagawa, M. and Takahashi, N. Information Theoretic Pattern Primitive Selection for On-line Recognition of Japanese Characters and Semi-automatic Generation of the Pattern Primitive Dictionary (in Japanese), *Trans. IECEJ*, **J69-D**, 10 (Oct. 1986), 1459–1470.
4. Aizawa, T., Nakagawa, M. and Takahashi, N. Signal Processing Approach toward Optimum Preprocessing for On-line Recognition of Handwritten Japanese Characters (in Japanese), *Trans. IEICE*, **J70-D**, 3 (Mar. 1987), 581–591.
5. Aoki, N., Inokuchi, S. and Sakurai, Y. On-Line Recognition System for Handwritten Hirakana (in Japanese), PRL75-28, *IECEJ* (July 1975), 58–68.
6. Arakawa, H. and Masuda, I. On-line Recognition for Handwritten Alphanumeric Characters (in Japanese), AL72-68, PRL72-68, *IECEJ* (Oct. 1972), 1–8.
7. Arakawa, H. and Masuda, I. Preprocessing for On-line Character Recognition (in Japanese), PRL73-84, *IECEJ* (Jan. 1974), 79–86.
8. Arakawa, H. and Masuda, I. On-line Recognition for Handwritten Characters (Alphanumerics, Hiragana, and Katakana) (in Japanese), PRL74-66, *IECEJ* (Feb. 1975), 117–126.
9. Arakawa, H. and Masuda, I. On-line Recognition of Hand-printed Characters (in Japanese), *Trans. IECEJ*, **J59-D**, 11 (Nov. 1976), 809–816.
10. Baskerville, D. and Shirai, H. A System to Recognize and Critique Hand-Written Japanese Characters (in Japanese), PRL83-11, *IECEJ* (June 1983), 1–8.
11. Chiba, S. Speech Input and Output for Japanese Language (in Japanese), *J. IECEJ*, **63**, 7 (July 1980), 723–728.
12. Fujisaki, H. and Nagai, S. 手書き数字のオンライン認識の一方法 (in Japanese), *National Convention Record 1971, IECEJ*, (Mar. 1971), 124.
13. Fujiwara, S., Ikeda, K., Tominaga, Y. and Kiyono, T. On-line Recognition of Handwritten Character by General Counter Vector Sequence Method (in Japanese), *Trans. IPS Japan*, **17**, 3 (1976), 191–199.
14. Fukamachi, Y., Agui, T. and Nakajima, M. Recognition of on-line KATAKANA characters using attributed grammars, (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1523.
15. Furukawa, S. and Mita, Y. On-line Recognition of Handwriting Chinese Characters (in Japanese), PRL73-22, *IECEJ* (May 1973), 1–9.
16. Groner, G. F. Real-Time Recognition of Handprinted Text, *Proc. FJCC* (1966), 591–601.
17. Hanaki, S., Temma, T. and Kiji, K. Realtime Character Recognition from a Remote Terminal (in Japanese), J. *IPS Japan*, **11**, 10 (Oct. 1970), 577–583.
18. Hanaki, S., Temma, T. and Yoshida, H. An On-line Character Recognition Aimed at a Substitution for a Billing Machine Keyboard, *Pattern Recognition*, **8** (1976), 63–71.
19. Hanaki, S. and Yamazaki, T. On-line Recognition of Handprinted Kanji Characters, *Pattern Recognition*, **12** (1980), 421–429.
20. Hanaki, S., Temma, T., Yoshida, H., Arakawa, T., Suzuki, M., Seki, T. and Kikuchi, Y. Online Realtime Character Recognition System (in Japanese), *Trans. IECEJ*, **J64-D**, 5 (1981), 411–418.
21. Hidai, Y., Ooi, K., Nakamura, Y. and Kurosawa, Y. On-line Handwritten Character Recognition System on a Personal Computer (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1528.
22. Hiraga, Y., Ono, Y. and Yamada, H. An Assignment of Keycodes for a Japanese Character Keyboard, *Proc. 8th Intl. Conf. on Computational Linguistics*, Tokyo (Sep. 1980), 249–256.

23. ICHIKAWA, A. Application of Speech Equipment (in Japanese), *J. IPS Japan*, **24**, 8 (Aug. 1983), 1011–1019.

24. ICHIKAWA, T. and YOSHIDA, J. Recognition of Handprinted Characters with Association Activated (in Japanese), *Trans. IECEJ*, **J59-D**, 1 (1976), 17–24.

25. IKEDA, K., YAMAMURA, T., MITAMURA, Y., FUJIWARA, S., TOMINAGA, Y. and KIYONO, T. On-line Recognition of Handwritten Characters Utilizing Positional and Stroke Vector Sequences, *Proc. 4th IJCPR*, Kyoto (Nov. 1978), 813–815.

26. IKEDA, K., YAMAMURA, T., MITAMURA, Y., FUJIWARA, S., TOMINAGA, Y. and KIYONO, T. On-line Recognition of Handwritten Characters Utilizing Positional and Stroke Vector Sequences, *Pattern Recognition*, **13**, 3 (1981), 191–206.

27. IKEDO, T., OHZONE, K. and SAITO, Y. Large-Sized Digitizer for High Resolved Position Determination (in Japanese), *Trans. IECEJ*, **J61-D**, 5 (1978), 346–352.

28. INOUE, K., KOSUGI, M. and ITO, H. On-line Real-time Recognition of Handwriting Chinese Characters (in Japanese), PRL73-20, *IECEJ* (May 1973), 1–9.

29. ISHIGAKI, K. and MORISHITA, T. オンライン手書き文字認識における画数に依存しない特徴点の対応づけ方式 (in Japanese), *The 30th Annual Convention IPS Japan* (Mar. 1985), 1257–1258.

30. ISHIGAKI, K. and MORISHITA, T. A Stroke-Number-Free Character Recognition using Indispensable-Feature-Point (in Japanese), *The 33rd Annual Convention IPS Japan* (Oct. 1986), 1633–1634.

31. ISHIGAKI, K. and MORISHITA, T. A High-Speed Successive-Feature-Point Matching using Directed-Acyclic-Graph (in Japanese), *The 34th Annual Convention IPS Japan* (Mar. 1987), 1843–1844.

32. ISHIGAKI, K. and MORISHITA, T. An Evaluation of an Online Hand Written Character Recognition Method with Classified Test Patterns (in Japanese), *The 34th Annual Convention IPS Japan* (Sep. 1987), 1953–1954.

33. ISHII, Y. Positional Stability of Stroke Representative Points of Online Handwritten Kanji Characters (in Japanese), *Trans. IECEJ*, **J68-D**, 12 (Dec. 1985), 2107–2115.

34. ISHII, Y. Structure Analysis of Kanji Characters by means of Stroke Representative Points (in Japanese), *Trans. IECEJ*, **J69-D**, 4 (Apr. 1986), 558–568.

35. ISHII, Y. Stroke Order Free Online Handwritten Kanji Character Recognition Method by Means of Stroke Representative Points (in Japanese), *Trans. IECEJ*, **J69-D**, 6 (1986), 940–948.

36. ITO, M. R. and CHUI, T. L. On-line Computer Recognition of Proposed Standard ANSI (USASI) Handprinted Characters, *Pattern Recognition*, **10** (1978), 341–349.

37. ITO, T., UEHARA, K., KIHARA, K., KOBAYASHI, S., YOSHIDA, K. and SAKOE, H. CR-100 オンライン手書漢字入力ターミナル (in Japanese), *National Convention Record 1983*, *IECEJ*, (Mar. 1983), 2290.

38. KASHIOKA, S. and IWAOKA, H. On-line Classification of Handprinted Chinese Characters (in Japanese), A71-119, IT71-104, *IECEJ* (Mar. 1972), 1–6.

39. KATO, O. et al. A Handwriting Input System For Japanese, *Proc. IFIP 80*, Tokyo (1980), 689–694.

40. KAWADA, T. et al. Japanese Word Processor JW-10, *Proc. of COMPCON'79* (1979).

41. KAWADA, T. and AMANO, S. Japanese Word Processing (in Japanese), J. *IPS Japan*, **21** (1980), 894–901.

42. KAWAKAMI, A. et al. Human Factors in Rainputto Keyboard for Kanji Input, *Proc. 1st UJCC* (1972), 292–295.

43. KAWAKAMI, A. and KAWAKAMI, T. Touch Typewriting for Kanzi-kana Input (in Japanese), *J. IPS Japan*, **15**, 11 (Nov. 1974), 863–867.

44. KIMURA, F., HAI, T., YASHIMURA, M. and MIYAKE, Y. On Extracting Subpattern and Classification of Handprinted Chinese Characters (in Japanese) *Trans. IECEJ*, **J64-D**, 11 (Nov. 1981), 1037–1044.

45. KIMURA, Y. and ODAKA, K. Corresponding Points Extraction for On-line Cursive Character Classification (in Japanese), PRL84-68, *IECEJ* (Jan. 1985), 19–26.

46. KIMURA, Y. and ODAKA, K. Feature Point Extraction for Online Cursive Character Recognition (in Japanese), *National Convention Record 1985*, *IECEJ* (Mar. 1985), 1531.

47. KIMURA, Y. and TOIDA, T. Online Similar Character Recognition for Large Character Set (in Japanese), *National Convention Record 1987*, *IECEJ* (Mar. 1987), 1481.

48. KIMURA, Y. and MIYAHARA, S. The Properties of Characters on Different Writing Restrictions (in Japanese), PRU87-30, *IECEJ* (July 1987), 19–26.

49. KIMURA, Y. and MIYAHARA, S. On-line Handwritten Character Recognition Using Inter-stroke Distance Matrix and Structure Matrix (in Japanese), PRU87-72, *IECEJ* (Dec. 1987), 45–54.

50. KITAHARA, Y. and ISOMICHI, Y. Online Recognition System of Handwritten "Joyo Kanji" by Method of DP Matching Based on Directional Cosine (in Japanese), *Trans. IECEJ*, **J64-D**, 11 (1981), 1013–1020.

51. KOMODA, C., NAKAGAWA, M. and TAKAHASHI, N. Grammatical Representation of Hierarchical Structure of Kanji Patterns and Its Advantage for On-line Recognition of Simplified, Distorted and Wrong Stroke Order Patterns (in Japanese), *Trans. IEICE*, **J70-D**, 4 (1987) 777–784.

52. KUBOTA, S. 新聞通信報道界におけるニュース情報処理動向について (in Japanese), 日本語情報処理シンポジウム報告集, (1978), 113–122.

53. KUWAHARA, K. 大分類の段階がほぼ完成した手書き漢字認識の研究, NIKKEI ELECTRONICS (Dec. 7, 1981), 148–167.

54. LEA, W. A. Ed. Trends in Speech Recognition, Prentice-Hall (1980).

55. MAKINO, H. Japanese Input Method by Kana-Kanji Translation (in Japanese), *J. IPS Japan*, **23**, 6 (June 1982), 529–535.

56. MASUDA, I. On-line Recognition of Hand-printed Characters (in Japanese), *J. IECEJ*, **61**, 2 (1978), 125–129.

57. MIKI, S., FUJITA, T., TOMIMOTO, T., KANNO, M. and HATTORI, M. Word Processor with Online Handwritten Character Recognition (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1525.

58. MORI, K. Machine Printed Character Recognition (in Japanese), *J. IECEJ*, **61**, 2 (1978), 107–114.

59. MORI, S., YAMAMOTO, K., YAMADA, H. and SAITO, T. 手書教育漢字のデータベースについて (in Japanese), 電子技術総合研究所彙報, **43**, 11 (1979), 752–773.

60. MORISHITA, T., KATO, O., FUJITA, T. and TANAHASHI, J. Some Variation Analyses of On-line Hand-written Characters (in Japanese), PRL80-90, *IECEJ* (Feb. 1981), 57–62.

61. MORISHITA, T. et al. An Experiment for Japanese Handwriting Input System Using Script (in Japanese), WGJI12-4, *IPS Japan* (Nov. 1983).

62. MOTO-OKA, T. Trends of Office Automation Technology (in Japanese), J. *IPS Japan*, **22**, 10 (Oct. 1981), 914–921.

63. MURAYAMA, N. 2-Stroke Method for Japanese Text Input (in Japanese), *J. IPS Japan*, **23**, 6 (June 1982), 552–558.

64. MURASE, H., WAKAHARA, T. and UMEDA, M. Online Writing-Box Free Character String Recognition by Candidate Character Lattice Method (in Japanese), *Trans. IECEJ*, **J68-D**, 4 (1985), 765–772.

65. NAGAI, H. Their Tendency and Technical Problems in the Input-Output Equipments for Japanese Language (in Japanese), *J. IPS Japan*, **10**, 5 (1969), 320–332.

66. NAGAO, M. An Overview on Japanese Language Information Processing, *J. IECEJ*, **63**, 7 (1980), 702–706.

67. NAKAGAWA, M., KIMURA, S. and TAKAHASHI, N. A Japanese On-Line Input System (JOLIS-0) and Its Successors, *Proc. of ICC*, Hong-Kong (1980), 8-3.1 ~ 16.

68. NAKAGAWA, M., KIMURA, S., MANABE, T., AOKI, K. and TAKAHASHI, N. Design and Implementation of Japanese On-Line Input System (JOLIS)-1 (in Japanese), WGJI3-1, *IPS Japan* (1982).

69. NAKAGAWA, M., AOKI, K., MANABE, T., KIMURA, S. and TAKAHASHI, N. On-Line Recognition of Handwritten Japanese Characters in JOLIS-1, *Proc. 6th ICPR*, Munich (1982), 776–779.

70. NAKAGAWA, M., MANABE, T., AOKI, K., IKEDA, Y. and TAKAHASHI, N. On-Line Handwritten Character Recognition As a Japanese Input Method, *Proc. Intl. Conf. on Text Processing with a Large Character Set*, Tokyo (1983), 191–196.

71. NAKAGAWA, M., IKEDA, Y., AIZAWA, T., KOMODA, C. and TAKAHASHI, N. An Evaluation of Japanese On-Line Input System (JOLIS)-1 (in Japanese), WGJI16-1, *IPS Japan* (July 1984).

72. NAKAGAWA, M., AIZAWA, T., KOMODA, C., IKEDA, Y. and TAKAHASHI, N. Syntactic Pattern Recognition with Stochastic Dissimilarity in Japanese On-Line Input System (JOLIS)-1/1.5, *Proc. 8th ICPR*, Paris (Oct. 1986), 1059–1062.

73. NAKAGAWA, M., IKEDA, Y., AIZAWA, T., KOMODA, C. and TAKAHASHI, N. Stochastic Dissimilarity in Syntactic Approach to On-Line Recognition of Handwritten Japanese Characters (in Japanese), *Trans. IEICE*, **J71-D**, 5 (May 1988), 800–811.

74. NAKAMURA, H. Health, Labor and Social Problems around In-

put Methods (in Japanese), J. *IPS Japan*, **23**, 6 (June 1982), 565-568.

75. NAKATA, K. Technical Trends of Speech Information Processing (in Japanese), *J. IPS Japan*, **24**, 8 (Aug. 1983), 968-975.

76. NAKATA, K. The Past, the Present and the Future of Speech Processing (in Japanese), *J. IEICE*, **70**, 4 (April 1987), 332-338.

77. NAKATSU, R. Speech Recognition Technology (in Japanese), *J. IPS Japan*, **24**, 8 (Aug. 1983), 984-992.

78. NAKAYAMA, T. and KUROSU, M. Research on Evaluation Method of Japanese Text Entry Systems (in Japanese), J. *IPS Japan*, **26**, 11 (1985).

79. NIKKEI ELECTRONICS, 透明タブレットを用い手書き2294字種を99.5%オンライン認識 (in Japanese) (Aug. 29, 1983), 98-102.

80. NISHINO, H. PIPS (Pattern Information Processing System) Project —Background and Outline—, *Proc. 4th IJCPR*, Kyoto (Nov. 1978), 1152-1161.

81. ODAKA, K., ARAKAWA, H. and ISHII, A. Feature Descriptions of Handwriting Chinese Characters for On-line Recognition System (in Japanese), PRL77-58, *IECEJ* (Jan. 1978), 49-58.

82. ODAKA, K. and MASUDA, I. Some Results on Stroke Order Free On-line Character Recognition (in Japanese), PRL79-28 (July 1979), 77-84.

83. ODAKA, K., ARAKAWA, H. and MASUDA, I. On-line Recognition of Handwritten Characters by Approximating Each Stroke with Several Points (in Japanese), *Trans. IECEJ*, **J63-D**, 2 (Feb. 1980), 153-160.

84. ODAKA, K., WAKAHARA, T. and MASUDA, I. Stroke Order Free On-line Character Recognition (in Japanese), *Trans. IECEJ*, **J65-D**, 6 (June 1982), 679-686.

85. ODAKA, K., WAKAHARA, T. and HASHIMOTO, S. On-Line Handwritten Character Recognizer (in Japanese), *Trans. IECEJ*, **J65-D**, 8 (Aug. 1982), 951-958.

86. ODAKA, K. A Basic Stroke Construction Method Using Approximated Patterns (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1539.

87. OKADOME, T., ONO, Y. and YAMADA, H. Interference between Structural Elements of Typing Task (in Japanese), *Trans. IPS Japan*, **27**, 3 (Mar. 1986), 304-312.

88. OOI, K., HIDAI, Y., NAKAMURA, Y. and KUROSAWA, Y. A Stroke Order Free On-Line Character Recognition (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1527.

89. OSADA, K., MAKINO, Y. and HITAKA, T. Character Recognition by Using Japanese Context (in Japanese), *Trans. IECEJ*, **J67-D**, 4 (1984), 521-527.

90. SAEKI, M. and MATSUSHITA, T. 2次元データ入力のためのオンライン手書き文字認識 (in Japanese), AI23-3, *IPS Japan* (1981).

91. SAITO, T. A Study of Skill Acquisition Process on Kanji Keyboards (in Japanese), *Trans. IECEJ*, **J65-D**, 8 (1982), 975-980.

92. SAITO, S. and NAKATA, K. 音声処理の基礎 (in Japanese), オーム社 (1981).

93. SAITO, T., YAMADA, H. and MORI, S. 手書き文字データベースの解析(III) (in Japanese), 電子技術総合研究所彙報 **42**, 5 (1980), 385-434.

94. SAKAI, T. and NAKAGAWA, S. A Speech Understanding System for Simple Japanese Sentences in a Task Domain. *Trans. IECEJ*, **E60**, 1 (Jan. 1977), 13-20.

95. SAKAI, K. et al. An Optical Chinese Character Reader, *Proc. 3rd IJCPR* (1976).

96. SAKATA, T., HIDAI, Y. and OOI, K. Refinement of Dictionary of On-line Character Recognition System, *National Convention Record 1985, IECEJ* (1985), 1526.

97. SATO, Y. and ICHIHARA, T. Feasibility Study on Online Character Recognition of Cursive Writings Using Feature of Writing Pressure (in Japanese), PRL83-23 (July 1983), 1-8.

98. SATO, Y. and ADACHI, H. Online Recognition of Cursive Writings (in Japanese), *Trans. IECEJ*, **J68-D**, 12 (1985), 2116-2122.

99. SAWAI, H. 手書き文字オンライン認識活発に研究進む (in Japanese), NIKKEI ELECTRONICS (May 7, 1973), 46-59.

100. SAWAI, R., NAKAGAWA, M. and TAKAHASHI, N. On-line Recognition of Handwritten Japanese Characters with Running Strokes (in Japanese), *Trans. IEICE*, **J70-D**, 5 (May 1987) 946-957.

101. SHIKANO, K. Speech Recognition and Language Knowledges (in Japanese), 自然言語処理シンポジウム, *IPS Japan* (Jan. 1988), 99-113.

102. SHINYA, M., MEGURO, S. and UMEDA, M. Post Processing in Character Recognition Using Pattern Features as well as Lexical and Morphological Information (in Japanese), *Trans. IECEJ*, **J67-D**, 11 (Nov. 1984), 1349-1355.

103. SHUTOH, M. and ITOH, H. Pen-touch Input Method (in Japanese), *J. IPS Japan*, **23**, 6 (1982), 536-542.

104. SUEN, C. Y., BERTHOD, M. and MORI, S. Automatic Recognition of Handprinted Characters—The State of the Art. *Proc. IEEE*, **68**, 4 (April 1980), 469-487.

105. SUMINAGA, Y. くずし字など筆記制限を緩和する方向に進むオンライン手書き漢字認識 (in Japanese), NIKKEI ELECTRONICS (Dec. 5, 1983), 115-133.

106. TAKAHASHI, N. The Expectation of Information Processing of Japanese Language, *J. IPS Japan*, **20**, 1 (1979),44-49.

107. TAKAHASHI, N. Recent Trends on Input/Output Technology for Japanese Text Processing (in Japanese), *J. IPS Japan*, **20**, 10 (1979), 933-940.

108. TAKAHASHI, N. The Current State and Future Trend of Japanese Text Input Systems (in Japanese), *J. IPS Japan*, **23**, 6 (1982), 518-528.

109. TAKAHASHI, N. A Feasibility Test Made for the Preparation of the Official Examination System of Professional Word-processor Operators by The Japanese Chamber of Commerce and Industry (in Japanese), WGJI20-4, *IPS Japan* (Feb. 1985).

110. TANAKA, N., SHIONO, M., SANADA, H. and TEZUKA, Y. Recognition of Handprinted KANJI Characters by Dynamic Directional Matching Method (in Japanese) *Trans. IECEJ*, **J68-D**, 1 (Jan. 1985), 56-63.

111. TERAI, H. and NAKATA, K. An On-line Recognition of Japanese KATAKANA Characters (in Japanese), *J. IPS Japan*, **13**, 3 (Mar. 1972), 145-153.

112. TERAI, H. and NAKATA, K. On-line, Real-time Recognition of Handwritten Chinese Characters and Japanese Katakana Syllabary (in Japanese), *Trans. IECEJ*, **56-D**, 5 (1973), 312-319.

113. TERAI, H. 日本語文字パターン情報のオンライン認識に関する研究 (in Japanese), *Ph.D Thesis*, the Univ. of Kyoto (1974).

114. TERAI, H. and NAKATA, K. Text Editing System Using On-line Real-time Character Recognition (in Japanese), *J. IPS Japan*, **15**, 6 (June 1974), 419-427.

115. TOMIMOTO, T., SHIMA, K. and OHTA, H. On-line Recognition of Handprinted Japanese Characters (in Japanese), PRL82-37, IECEJ (Oct. 1982), 49-56.

116. TOMIMOTO, T., MIKI, S. and OHTA, H. Analysis of Reference Data for On-line Character Recognition (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1524.

117. WAKAHARA, T., ODAKA, K. and UMEDA, M. Stroke Number and Order Free On-Line Character Recognition by Selective Stroke Linkage Method (in Japanese), *Trans. IECEJ*, **J66-D**, 5 (May 1983), 593-600.

118. WAKAHARA, T. and UMEDA, M. On-Line Cursive Script Classification by Stroke Linkage Rules (in Japanese), PRL83-30, *IECEJ* (Sep. 1983), 9-16.

119. WAKAHARA, T. and UMEDA, M. On-Line Cursive Script Classification Using Stroke Linkage Rules (in Japanese), *Trans. IECEJ*, **J67-D**, 11 (1984), 1285-1292.

120. WAKAHARA, T. and ODAKA, K. Character Deformation Prediction Using Deformation Vector Field in Online Handwritten Character Recognition (in Japanese), *Trans. IECEJ*, **J69-D**, 12 (Dec. 1986), 1913-1922.

121. WAKAHARA, T. Online Handwritten Character Recognition Using Local Affine Transformation (in Japanese), *Trans. IEICE*, **J71-D**, 2 (Feb. 1988), 379-386.

122. YAMADA, H. 日本語テキスト入力法の人間工学的比較 (in Japanese) 日本語情報処理シンポジウム報告集 (1978), 1-33.

123. YAMADA, H. The History of Typewriter and the Input of Japanese Text (in Japanese), *J. IPS Japan*, **23**, 6 (June 1982), 559-564.

124. YAMAMOTO, K., YAMADA, H., SAITO, T. and OKA, R. Recognition of Handprinted Chinese Characters and Japanese Cursive Syllabary, *Proc. 7th ICPR*, Montreal (July 1984), 385-388.

125. YAMAMOTO, K., YAMADA, H., SAITO, T. and SAKAGA, I. Recognition of Handprinted Characters in the First Level of JIS Chinese Characters, *Proc. 8th ICPR*, Paris (Oct. 1986), 570-572.

126. YAMASAKI, T., INOKUCHI, S. and SAKURAI, Y. Training System for Handwriting Chinese Characters Using On-Line Character Recognition Techniques (in Japanese), *Trans. IECEJ*, **J65-D**, 10 (Oct. 1982), 1211-1218.

127. YAMASAKI, T., INOKUCHI, S. and SAKURAI, Y. Training System for Well-Writing of Chinese Characters Based on Their Local Structure (in Japanese), *Trans. IECEJ*, **J67-D**, 4 (1984), 442-449.

128. YASUDA, M., NAKANO, Y. and FUJIMOTO, Y. Recognition

Techniques of Handwritten Characters, *J. IECEJ*, **61**, 2 (1978), 115-124.
129. YASUDA, T. Computer and Typesetting in Japanese Newspaper, *J. IPS Japan*, **10**, 5 (Sep. 1969), 340-347.
130. YASUHARA, M. A Synthetic Model of Handwriting Letters (in Japanese), *J. IPS Japan*, **11**, 10 (Oct. 1970), 584-591.
131. YOSHIDA, K. Stroke Order Free Online Character Recognition by Optimum Stroke Correspondence Matching (in Japanese), *National Convention Record 1985, IECEJ* (Mar. 1985), 1529.
132. YOSHIDA, K. and SAKOE, H. ストローク数に存在しないオンライン手書き文字認識の一手法 (in Japanese), *National Convention Record 1982, IECEJ* (Mar. 1982), 1357.
133. YOSHIDA, K. and SAKOE, H. Online Character Recognition by Stack DP Matching Method (in Japanese), PRL83-29, *IECEJ* (Sep. 1983), 1-8.
134. YOSHIDA, K. et al. Japanese Text Input by Online Handwritten Character Recognition (in Japanese), WGJI18-3, *IPS Japan* (Nov. 1984), 1-9.
135. YURUGI, M., NAGATA, S., ONUMA, K. and KUBOTA, K. Online Character Recognition by Hierarchical Analysis Method (in Japanese), *Trans. IECEJ*, **J68-D**, 6 (1985), 1320-1327.

(Received September 13, 1989)