Historical Review of OCR Research and Development

SHUNJI MORI, MEMBER, IEEE, CHING Y. SUEN, FELLOW, IEEE, AND KAZUHIKO YAMAMOTO, MEMBER, IEEE

Invited Paper

In this paper research and development of OCR systems are considered from a historical point of view. The paper is mainly divided into two parts: the research and development of OCR systems, and the historical development of commercial OCR's. The R&D part is further divided into two approaches: template matching and structure analysis. It has been shown that both approaches are coming closer and closer to each other and it seems they tend to merge into one big stream. On the other hand, commercial products can be classified into three generations, for each of which some representative OCR systems are chosen and described in some detail. Some comments on recent techniques applied to OCR such as expert systems, neural networks, and some open problems are also raised in this paper. Finally we present our views and hopes on future trends in this fascinating area.

Keywords*— *Optical character recognition; feature extraction; template matching; structural analysis; learning; Chinese character recognition; practical OCR systems.

I. Introduction

The history of science and technology does not flow like a straight canal, but is usually tangled like a meander. We will describe not only the main stream, but also the resulting impact, as when oxbow lakes are generated after a meander changes its direction. The history of OCR research, like that of speech recognition, is comparatively old in the field of pattern recognition. In the early days of pattern recognition research, almost everyone took the subject of OCR. One reason was that characters were very handy to deal with and were regarded as a problem which could be solved easily. However, against what was the expectation of many people, after some initial easy progress, great difficulty in solving this problem surfaced. Hence, people diversified their interests over a wide range of topics in the pattern recognition field, for example, image understanding and 3-D object recognition. Of course there were practical demands for such research. A new field always gives benefit to its pioneers, but research on these topics of pattern recognition seems to be confronted with a strong barrier. In this sense, the topic of OCR is not so exceptional, but rather is universal in that it includes essential problems of pattern recognition which are common to all other topics. In this sense, we have written its history from as general a point of view as we could. Actually the problem is most profound and indeed we realized this when writing this monograph.

On the other hand, research cannot exist without its applications in engineering. Fortunately, market demand for OCR is very strong even though word processors are prevalent. For example, a dozen leading companies in Japan sell or are preparing to sell hand-printed Kanji character readers. So far these sophisticated machines are not prevalent yet, but it is certain that if the price and performance meet the requests of the users, these machines will be widely used in offices as a very natural man–machine interface. The accumulation of OCR knowledge is reducing the gap between the users and makers, which is also helped by the rapid development of computer technology.

This historical review is roughly divided into three parts. The first part is a prelude. The second and third parts constitute the main body of the paper, i.e., research and products respectively. The research part not only has its own right to exist; it also provides a preparation for the products part, so that the reader can understand the products more easily and deeply in terms of technological development. The research part is further divided into two approaches: template matching and structure analysis. This paper shows that the two approaches are converging. That is, the template matching approach has been absorbing structure analysis techniques and now the two approaches seem to be on the verge of fusion. On the other hand, we classify commercial products into three generations, for

Manuscript received March 18, 1991. C. Y. Suen's research was supported in part by the National Networks of Centers of Excellence program of Canada and by the Natural Sciences and Engineering Research Council of Canada.

S. Mori is with the Research and Development Center, Ricoh Company, Ltd., 16-1, Shinei-cho, Kohoku-ku, Yokohama, 233 Japan.

C. Y. Suen is with the Department of Computer Science, Concordia University, 1455 de Maisonneuve Bld. W., Montreal, Quebec, H3G 1M8, Canada.

K. Yamamoto is with the Electrotechnical Laboratory, 1-1-4, Umezono, Tsukuba Science City, Ibaraki, 305 Japan.

IEEE Log Number 9203434.

0018–9219/92$03.00 © 1992 IEEE

each of which we choose representative OCR's and describe them in some detail. Finally we comment on expert system and neural network applications to OCR.

The description might be biased toward research and development in Japan, but it reflects the fact that research and development in OCR has been particularly active and prosperous in Japan. For many years the Electrotechnical Laboratory (ETL), in Ibaraki, has played a key role in developing OCR technology in Japan. Another reason is that many important papers have not been translated into English and so we thought that this would be a good opportunity to introduce some of them to the international community. Nevertheless, in writing this paper we have found several reference books and review papers which have been very useful, among them the books by Ullman [1], Sakai and Nagao [2], Pavlidis [3], and Mori and Sakakura book [4] and papers by Suen *et al.* [5], Schurman [6], and Couindan and Shivaprasad [7]. Other books and review papers are referred to at the appropriate places, but it is difficult to read all the numerous papers related to the topic. We thus avoided biographical description, and pursued the streams of research. Here, we do not mention the very important research fields of document analysis and cursive script recognition because other papers in this issue make reference to them. However, we might have missed some important papers or patents. Actually we found a few new papers which were very useful in preparing this paper. We would appreciate criticism on this paper from readers and we hope that it will be useful to researchers in advancing the technology of OCR.

II. Dawn of OCR

In 1929 Tausheck [8] obtained a patent on OCR in Germany and in 1933 Handel [9] did the same in the U.S. These are the first concepts of the idea of OCR as far as we know. At that time certain people dreamed of a machine which could read characters and numerals. This remained a dream until the age of computers arrived, in the 1950's. However, we think their basic idea is worth mentioning, because it is still alive. In this sense we introduce Tausheck's patent. The principle is template/mask matching. This reflects the technology at that time, which used optical and mechanical template matching. Light passed through mechanical masks is captured by a photodetector and is scanned mechanically. When an exact match occurs, light fails to reach the detector and so the machine recognizes the characters printed on paper.

Mathematically speaking, the principle is the axiom of superposition, which was first described by Euclid as the seventh axiom in the first volume of *Elements*. For humans, however, E has the same meaning as an $\mathcal{E}$ in the sense of pattern. Therefore, what is the principle of their equivalence? So far no general solution has given and yet it is the principal and central problem in pattern recognition. The seventh axiom is the first principle proposed for equivalence of shape. We will return to this problem later. We will see that the principle of superposition has been realized by employing more advanced technologies of hardware such as cathode ray tubes and analog electric circuits. Actually, this original work is the origin of the main stream of OCR technology. The "template matching method" is in a broad sense the principle of superposition.

III. Age of Cut and Try

The first commercial computer, UNIVAC I, was installed and began to work at the Bureau of Statistics in the U.S. in 1951. In terms of hardware, electronics was the basis of the age of computers. First of all, electronics made engineers regard OCR as a possible reality. However, there were strong limitations in terms of the quantity and complexity of the hardware.

A. Template-Matching Methods

The basic reduction in complexity was archieved by projecting from two-dimensional information onto one. This approach was taken by Kelner and Glauberman [10] using a magnetic shift register in 1956. An appropriately placed input character is scanned vertically from top to bottom by a slit through which the reflected light on the printed input paper is transmitted to a photodetector. It is a simple calculation using only algebraic addition to obtain a value which is proportional to the area of the black portion within the slit which segments the input character. Then the sampled values are sent to the register to convert the analog values to digital ones. Template matching is done by taking the total sum of the differences between each sampled value and the corresponding template value, each of which is normalized. The machine was not commercialized.

We note here a very important point in the matching process, concerning the general problem of registration. The template matching process can be roughly divided into two processes, i.e., superimposing an input shape on a template and measuring the degree of coincidence between the input shape and the template in the two cases mentioned above. Projection is taken horizontally or vertically, which makes the superimposing processes position invariant in one direction. This is clearly illustrated in Fig.1. When the slit is long enough to cover the input numerals, there is no change in the value of black area projected on the x axis, even if the numerals are moved vertically. However, we need to detect the starting and ending points of an input numeral to register it against the corresponding template. This is done very easily because numerals are all simply connected and there is enough space in each interval between neighboring numerals. Actually, the projection technique has been broadly used to segment the input character string and picture region of the documents, for example, in current OCR's. Such processing is called preprocessing in the terminology of OCR.

The two topics mentioned above tell us that in essence characters contain two-dimensional information. If we want to reduce the dimension to one, then we have to distort the shapes of the characters so that the machine can recognize it. Such distortion may be allowed for numerals which have a small number of characters. In this sense, MICR has a very limited application, although it is widely used by

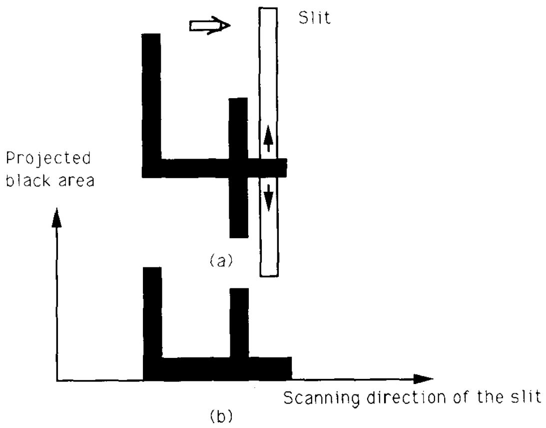


Fig. 1. Illustration of 2-D reduction to 1-D by a slit. (a) An input numeral "4" and a slit scanned from left to right. (b) Black area projected onto x axis, the scanning direction of the slit.

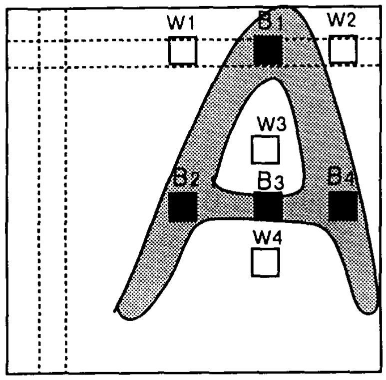


Fig. 2. Illustration of the peephole method.

banks. However, it has naturally a major problem from the man–machine interface point of view. So people proceeded to deal with two-dimensional information.

When looking at two-dimensional information, it seems natural to use optical techniques to perform template matching. Actually a very sophisticated OCR was made combining electronics and optical techniques by Hannan [11] (RCA group) in 1962. At that time RCA had the most advanced electron tube technology in the world, which was fully employed in the OCR research work. Hannan concluded his paper as follows: "In summary, the test results of this program proved that the RCA optical mask-matching technique can be used to reliably recognize all the characters of complete English and Russian fonts (91 channels are necessary)." However, no announcement was made for a commercial RCA OCR based on the techniques. The great experiment ended without a successor.

It is very natural that the advent of computers influenced the design of OCR with respect to hardware and algorithms. We introduce a logical template matching method. The simplest one is called the peephole method. First of all we assume that an input character is binarized.*Binarization* is an important preprocess in OCR technology. Ideally an input character has two levels of density, i.e., black and white, commonly represented by 1 and 0 respectively . However, real data are not always so. We will discuss this problem later. Here we note that binarization is not an easy matter to deal with.

Imagine a two-dimensional memory plane on which a binarized input character is stored and registered in accordance with some rule, with the character positioned at the top right corner, for example, as shown in Fig. 2. Then obviously for an ideal character, which has a stroke of constant size and width, black portions are always black and the same is true for the white background. Then appropriate pixels are chosen for both black and white regions so that the selected pixels can distinguish the input character from characters belonging to other classes. Looking at Fig. 2, a so-called logical matching scheme is easily constructed, which is called the peephole method.

The first OCR based on the peephole method was announced by Solatron Electronics Group Ltd. [12] and was called ERA (Electric Reading Automation) in 1957. The characters read were numerals printed by a cash register. The reading speed was 120 characters/second (chs), which was very high. This was due to the simple logic operations used. The total number of peepholes was 100, which is considerably greater than the ideal number of $\lceil log_2 10 \rceil = 4$ which would be needed to obtain stable recognition for real data.

At ETL an OCR was designed based on the same scheme by Iijima *et al.* [13] in 1958. However, the design was more systematic than ERA using three-level logic and so it was more efficient. The characters that could be recognized were 72 alphanumerals; 10×12 meshes were used. The total number of peepholes used was 44 at 10 pixels/character. Logic circuits for the ETL Mark IV computer were used. Actually the OCR was one component of a larger system which was planned at ETL at that time. It was to be an input device for a machine translator.

Autocorrelation: As we mentioned above, two-dimensional template matching has a weakness in registration. Researchers became aware of that and began to devise new methods which are shift invariant. Two methods attracted attention in particular. One is based on an autocorrelation method and the other is based on a moment method. The latter is very ambitious, aiming at both shift and rotation invariance and will be discussed later together with the Fourier series, a method which is also shift invariant.

An exact formulation of the autocorrelation method is easily derived; therefore simulation can be done before the hardware is made. This method is commonplace now. This was the first time that this approach could be taken. For example, in 1958, the IBM 7090 was announced. Rapidly this powerful computer became available at certain research centers. Horowitz and Shelton [14] of IBM did very extensive research on the autocorrelation method in 1961. They proposed a very large special-purpose shift register machine for calculating autocorrelation exactly and with high speed. In the same year, Sato *et al.* [15] of the

Japanese Government's Radio Wave Research Laboratory carried out a very systematic simulation of the autocorrelation method. The results were unfortunately disappointing. The difference between "R" and "B" is only 0.4% when normalized with a maximum output value. In addition, the differences between character pairs "K" and "R," "A" and "V," and "U" and "D" are less than 1%.

B. Structure Analysis Method

The principle underlying template matching is really only appropriate for the recognition of printed characters. However, we have another set of hand-printed/handwritten characters that need consideration. The variation of shape of handwritten characters is so large that it is difficult to create templates for them. A so-called structure analysis method has been applied to handwritten character recognition. However, in the early stages of OCR development, we note that some very primitive methods were considered in addition to template matching. The weakness of these methods was due to the constrained hardware resources of that time. We include those methods in the structural analysis method, as described in the following subsection. However, it will be shown that these methods mark a continuation of logical template matching in terms of concept. Actually, contrary to the above description, these simple methods were applied to stylized fonts, while some were applied to the recognition of constrained hand-printed characters.

In the case of the structural analysis method, there is no mathematical principle. Rather it is still an open problem and there is no sign that it will be solved in the near future. Hence, our intuition has been the most reliable weapon in attacking this problem. However, there appear to be certain informal strategies that can be used in structure analysis. First of all, we give a very general and basic idea of the conceivable strategies. Since a structure can be broken into parts, it can be described by the features of these parts and by the relationships between these parts. Then the problems are how to choose features and relationships between them so that the description gives each character clear identification. Feature extraction, therefore, has become the key in pattern recognition research .

1) Slit/Stroke Analysis: A specific description will be given along this line. It has already been mentioned that the peephole method is considered a kind of template matching method. Now we try to extend it to the structure analysis method. The peephole is not always limited to a single pixel. Instead, it can be expanded into a slit or window whereby it is not necessary to fix at a specific position on the two-dimensional plane. The logical relationship between two pixels can be extended to a general relationship between them. The above description is illustrated in Fig. 3.

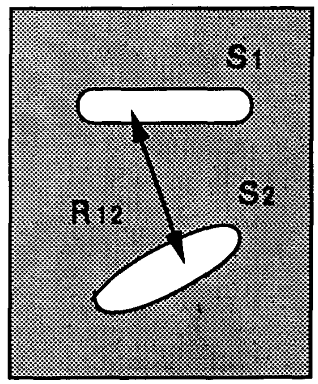


Fig. 3. Extension of the peephole method to structure analysis.

Perhaps the simplest example is the so-called cross counting technique, in which scanned lines are regarded as slits. The feature of the slit is the number of black regions in it. Rohland [16] proposed this technique in 1954, in which a vertical scan was primarily used. In 1961, Weeks [17] used this approach in a simpler fashion. In this method scanning is taken in four directions, i.e., vertical, horizontal, and two orthogonal diagonals, for each of which six equally spaced and parallel lines are used to cover a character. A crossing count, defined as the number of times each of the six rasters crossed black, is made. However, three or more crossing counts are regarded as three and so counts of 0, 1, 2, and 3 are possible. Thus, an input character is coded into a $6 \times 6 \times 4 = 144$ bit binary pattern, which is used for recognition based on a statistical decision method. The statistical decision method is a very important theoretical approach to pattern recognition in general. However, this has already been reasonablly well established and so we recommend that readers refer to the excellent book by Duda and Hart [18], to Highleyman's paper [19], and to Nagy's paper [20] concerning OCR in particular.

Here we give a rigorous description of the above scheme in a mathematical sense. The slits are usually convex regions, within each of which we can detect both topological and geometrical features. Connected components and black areas are typical features. The former is just the cross count and the latter is the length or width of the black portion. These features are given as a function of slit shape and size, and its position. In the above cases, the slit is simply given by a straight line due to a simple scanning mechanism. However, we need not be restricted to this method of scanning.

Actually Johnson [21] in 1956 and Dimond [22] in 1957 used a sonde as a slit, which is shown in Fig. 4. For numerals there seems to be two basic points around which they are written. Therefore, it is very effective to span some sondes/slits centering the two points. Thus we can count crossing times in each slit and, on the basis of this, easily distinguish the numerals. The scheme can become tougher by assigning weight to each sonde, which was done by Kamensky [23]. However, the two-point scheme is not generally applicable to many characters and so a more general scheme, free of specific points, was considered by Glucksman [24], which will be mentioned later.

So far only topological features within the slit have been described, but geometrical features have been extensively used. In particular, we note that what is referred to as run-length coding is very typical and is a special case of slit

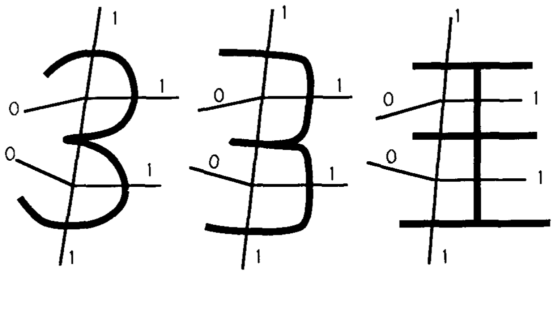


Fig. 4. Illustration of the **sonde** method; part (c) shows its weak points.

analysis. Actually run-length coding is regarded as a first step for more advanced structure analysis, which will be discussed in detail. Here, a very simple version of the geometric feature of a slit is introduced. Suppose there exists a base line such as a left/right vertical line. Then the geometrical feature is the distance from the base line to the first black pixel. If the slit is scanned from top to bottom of the character, for example, then we have a single value function, which is easy to analyze. In this case it is a convex function. Such approaches were taken by Doyle [25] in 1959. This approach can be easily expanded and will be discussed later.

Finally, as an example of the generalization of the slit/stroke analysis method, we can introduce a prototype based on an asynchronous reading method which is independent of scanning speed. This work was started by Sakai *et al.* [26] in 1963. On the other hand, the slit structure analysis approach is very effective so long as the number of categories of characters to be read is limited and the character shapes are mechanically designed as E-13B fonts in the MICR (magnetic ink character reader) used in processing checks. Some companies designed special fonts for OCR so that they could make use of the slit structure analysis, which will be described in the next section.

2) Hybrid of Template Matching and Structural Analysis: So far we have roughly classified the recognition methods into two classes, namely, template matching and structure analysis . However, there is an intermediate class between the two. In general, it is difficult to draw a clear boundary to separate the classes in the space of these recognition methods. At any rate, this trend is natural, because both methods have their own advantages. For example, template matching is very sensitive to positional change, but it can be very strong in the sense of global matching. On the other hand, structural analysis has the advantage of detecting the local stroke features of characters. From this the so-called zoned feature method was born. The general idea is that pixelwise matching is made loose, replaced by subregionwise matching, and the matching objects are local features within the subregion. Specifically, as the first step, an input character is set into a frame which is roughly partitioned into subregions, in each of which some local features are detected. Therefore the sensitivity to the position of the template matching becomes loose, because what is of concern is whether or not particular features lie within certain subregions.

For example, suppose that an input character is squeezed into a 15×15 array in which 5×5 pixels are taken as a subregion. Therefore, the array is partitioned into roughly a 3×3 array, in each of which four kinds of direction features, i.e., vertical, horizontal, and two orthogonal diagonal directions, are detected. Thus, an input character is coded into $3\times3\times4=36$ bit patterns. This is a tremendous reduction of $2^{15\times15}$ binary patterns. This example is a real one, with U.K. patent being granted to researchers at NEC [27]. A similar idea was proposed by Munson [28]. We can also find a similar idea in the study by Greanias *et al.* [29], based on which a famous OCR, the IBM 1287, was built. This machine also has the typical characteristics of the structure analysis method, i.e., a contour following method, and so it will be described in detail later.

IV. Commencement of Research

At the beginning stage it was thought that it would be easy to develop an OCR, and the introduction of a very rigorous reading machine was expected in the 1950's. Roughly speaking, the 1950's and the early half of the 1960's, the age of cut and try, were periods when researchers imagined an ideal OCR, even though they were aware of the great difficulty of the problem. Actually this is an instance of a common phenomenon which occurred in the research field of artificial intelligence in general. So some resorted to basic research, and engineers took a more systematic approach than ever. Such a trend was seen in both the template matching and the structure analysis method. Naturally it is difficult to draw a line to separate these research efforts, both chronologically and semantically. For example, basic research was done toward the end of the 1950's, as will be seen later, and research on the autocorrelation method could be regarded as basic research.

There are varieties of data for pattern recognition in general. Character data are the only specific ones in this sense. A digital picture, for example a bubble chamber photograph, represents another set of data. In such data, picture description is very important and takes precedence over identification. It was thought by some that character recognition was not very interesting, because its main purpose was class identification and so the approach was too engineering-oriented and not scientific. At that time it might have been true, but this is a misunderstanding. A description of character shape is very important when we want to proceed further. So basic research was needed and was done.

A. Template Matching Approach

1) Application of Information Theory: In logical template matching, an approach which IBM had taken for their OCR systems, the construction of the logic functions is quite tedious. Naturally a research effort on automatic generation of discriminate logic functions began at the IBM Watson Laboratory. Kamentsky and Liu [30], [31] gave such a scheme based on information theory. They gave a measure of discriminating power of a logic function, f, which was chosen randomly. Its configuration is $f(x_i, x_j, \cdots x_n)$, for which $x_i, x_j, \cdots x_n$ are randomly chosen pixels whose values are either 1 or 0. The function f has canonical form. Extensive experiments were done and this scheme seemed to be promising. Their scheme was elaborated by introducing the concept of distance. So an automatic design system for the discriminating functions was implemented on the powerful IBM computer. The IBM engineers anticipated the power of the program and applied it to their OCR design. We will continue this story later, which will provide a lesson.

2) Normalization: Iijima realized the importance of normalization in the template matching approach through his experiments and those of his group. Therefore, he thought that they should do basic research on this problem. He wrote a series of papers relating to basic research on pattern recognition, which was divided roughly into two parts. One is the theory on normalization of patterns [32] in 1962; the other is theory on feature extraction [33] in 1963. Both are very important and have exerted great influence on pattern recognition research and OCR development, particularly in Japan.

Here we introduce the theory of normalization and the related work which was done by Iijima's group. Actually the concept of blurring was first introduced into pattern recognition research by his work, whereas it was widely attributed to Marr [34] in the West. Iizima's idea was derived from his study on modeling the vision observation system. According to his model of the vision system, blurring was necessarily introduced and so normalizations of displacement and blurring were studied, in particular.

Setting reasonable conditions for the observation system, he proved that the mathematical form of the transformation must be a convolution of a signal $f(x')$ with a Gaussian kernel. That is a blurring transformation. At the first stage of their research, they tried to recover the blurring effect of an observation system, i.e., normalization of blurring [35]. However, through the experimental study of OCR, Iijima got the idea of blurring an input character rather than recovering the blurring effect. The research along this line has been carried out by Iijima and Yamazakii [36]. Since then, blurring has become a well-known preprocessing technique in Japan.

Based on this theory, Iijima advocated the analog approach to pattern recognition. Actually he derived a partial differential and integral equation for recovering displacement by introducing time [37]. This is a dynamic equation and has a form such that its behavior is controlled by the average of $f(x, t)$. The equation was actually implemented to a hardware system with only five input terminals and it was confirmed that the equation certainly works well [38]. They referred to this class of equations as a globally associated and dependent network. However, the implementation was too expensive and another approach was taken.

Mathematically speaking, the normalization can be considered a linear operator. On the other hand, data are given in discrete form. Therefore, the operator is represented necessarily by a matrix, and the desired property of the operator is given by the eigenvalue distribution on an appropriate set of eigenfunctions, which construct a subspace in a Hilbert space. Based on this idea, Mori gave a displacement matrix by d, where d takes on any real value, say 0.5 [39]. At that time, such a shifting filter had meaning, because it was very expensive to have a retina (2-D sensor) with reasonable resolution. Actually they designed a 9×9 retina. However, there was still the problem of implementing the above filter using an analog circuit because the fan-in number to the operational amplifier was too large. He therefore proposed using a much simpler approximating system as follows [40]:

$$Af(x) = f(x+d), \quad A \approx (1-d)E + dP^{-1}, \quad 0 \le d \le 1.$$

where A is the shift filter matrix, and E and P are the identity matrix and the circulant matrix respectively. A circulant matrix can be regarded as a shift operator for a vecor under the circular boundary condition (see [41]). However, in order to make the filter we need a variable resistance element and so he suggested using an FET. The special-purpose FET was made by Mitsubishi, and Funakubo implemented the system, which was quite effective [42].

Behind these formulations, serious consideration was given to the fact that we should treat continuous functions even if they appeared in discrete form. In this sense, Mori called such functions background functions. Therefore, when we displace an input pattern by, say, 0.3 units, the background function appears real. In this sense, it is not merely an interpolation technique. Spatial network was the name given to this approach, which had been developed theoretically by Ogawa and Igarashi [43]. Now, this approach exists in practice owing to highly developed computer technology. For example, a character is represented by a continuous function and the exact formula of curvature can be easily calculated from the continuous function, which is usually a spline [44].

On the other hand, Amari introduced another view of normalization, i.e., as a process that can be done after feature extraction [45]. He investigated the condition that such a scheme holds, assuming linear feature extraction, and proved that for an affine transformation, moment features have so-called admissible properties. This meant that normalization could be conducted on the feature space. From the practical point of view, this idea provides a very important insight into normalization, in which we need not insist on using the moment feature.

3) Karhunen-Loeve Expansion: Iijima's second paper was devoted to the theory of feature extraction based on the consideration that for a given normalized pattern set denoted by $D = \{h(x, \alpha)\}$, where α is an assigned/indexing number of the individual pattern, we should choose basic functions for a coordinate system of the pattern representation space. More specifically, he constructed the following functional (mapping functions to real values):

$$J[\varphi(x)] = \frac{\int_D \omega(\alpha)(h(x, \alpha), \varphi(x))d\alpha}{\| \varphi(x) \|}$$

where $\omega(\alpha)$ is an appearance probability of α. He derived the integral equation in which the function maximizes the above functional using the variation method. However, the above formulation is the same as the Karhunen-Loeve expansion [46]. He is very good at solving integral equations and did it independently of Karhunen's [47] and Loeve's [48] work, which was done in the 1940's. K–L expansion is very attractive not only for pattern recognition but also for data compression. Therefore, many researchers used it. Among them, Watanabe was a pioneer and Fukunaga [49] was very active with him in this field. Watanabe's work was done independently of Iijima, although their work was done in almost the same period. We will show later the more advanced development of K-L expansion, which was done by both researchers from different angles. At any rate, the application of K-L expansion to OCR was explored extensively at ETL by Iijima and his group and later at the Toshiba Central Research Center. First we will introduce the work done at ETL.

Before that, the reader might note that K-L expansion is identical to principal component analysis, which was originated by Hotelling [50] in 1933. Since then it has been widely used in the field of statistical data analysis. Therefore, when a data set of $[h(x, \alpha)]$ is given by vectors, we can construct the covariance matrix and solve its eigenvectors, which form the coordinates of the given pattern space. However, we notice that the integral representation is very convenient for the theoretical work and so in this sense they are not exactly the same thing.

Now, returning to our real problem, suppose that an $N \times N$–dimensional data set is given. Say N is equal to 50; then $N \times N$ becomes the large number 2500. We must handle a 2500-dimensional vector and solve the 2500×2500 covariance matrix. This is a huge problem even if the available computer power is great. Some way must be considered to reduce its huge dimension. In this respect, we recommend Gonzalez and Wintz's book [51], which deals with data compression by K-L expansion. Here we show another approach, where a class-based correlation can be used instead of taking a pixel-based one. Note that we change the notation slightly for our convenience and that $h^r(x)$ is canonical, i.e., constructed such that the average value of $h^r(x)$, $r \in L$ in the set of L is zero. Notice that L denotes also the number of classes.

Therefore we introduce the following covariance matrix:

$$C_{nr} = \omega_n \omega_r \int_S h^n(x) h^r(x) dx.$$

Thus solving this eigenvalue problem, orthogonal eigenvectors φ_m, $m = 1, 2, \cdots L-1$, were obtained. L is usually a small number, say 36 for alphanumerals. So it is very easy to calculate eigenvectors and eigenvalues. Thus, say a 50×50–dimensional pattern space is reduced to 36, which is a very compact space. However, we need to test this theoretical result in the real field. Noguchi [52] conducted the experiment on printed Katakana data. Katakana is one of the subsets of Japanese letters and consists of 46 characters. The feature extraction rate is defined in general in terms of eigenvalues as follows:

$$T_N = \frac{\sum_{n=1}^{N} \lambda_n}{\sum_{n=1}^{L-1} \lambda_n},$$

where λ_n is the nth eigenvalue and the eigenvalues are arranged in order of magnitude. Note here that the sum up to $L-1$ is taken because one freedom is used by taking the canonical form. According to the experiment, $T_N = 0.8$ when $N = 10$ and 0.9 when $N = 20$. Therefore this scheme seemed very promising and it was expected that 31×43 dimensions of the image plane could be drastically reduced, to around 30. However, they had to realize how big a problem "noise" can create. In particular, displacement noise was the most serious, followed by the width of a stroke. On the other hand, as mentioned above, Iijima's group was constructing an OCR called the ETL pilot model [53]. They intended to make an all-analog and parallel OCR and so the analog shift filter was implemented. They conducted a systematic simulation experiment of feature extraction based on K-L expansion. The retina was a 9×9 array and 0.25 shifts in eight directions were allowed to be an acceptable displacement noise. Eventually, after several trials, they adopted only the first eigenvector and the sign of the projection value to the axis. Therefore, their discrimination scheme was the tree structure. As a result, the discrimination tree needed a total of 43 nodes. The performance of the OCR was tested using sets of 150 printed numerals and C, S, T, X, Y, and / (one font) printed on various printing conditions for each class. The recognition rate was 96%. Naturally the question arose, "Why not use similarity itself?" In that case we can use templates of more than two for each class, say three; then the necessary templates would amount to 48 (16×3). The number 48 is almost the same as 43; i.e., the complexity of the tree structure is almost the same as the similarity system in which three templates are used for every class.

Actually Iijima seriously investigated the structure of similarity in Hilbert space. First of all, he found a very important property of similarity related to blurring. Intuitively $S(f_0, f) \rightarrow 1$ seems to be reasonable when the blurring increases, but he proved that when canonical transformation is done $S(h_0, h) \rightarrow \delta < 1$. That is, even if heavy blurring is applied, essential differences between different patterns remain. Blurring is equivalent to low-pass filtering and suppressing the high-frequency part. Therefore, we can reduce sampling points by taking advantage of blurring. Furthermore, Iijima introduced the so-called multiple similarity method [54]. The method was considered to overcome the weakness of similarity against displacement noise. He investigated the trace of patterns in Hilbert space, when displacement noise is expressed by the first and second terms of its Taylor expansion expressed by $h_0(\boldsymbol{r} + \boldsymbol{d}), \boldsymbol{d} = \boldsymbol{i}d_1 + \boldsymbol{j}d_2$. The pattern $h_0(\boldsymbol{r} + \boldsymbol{d})$ moves on the same definite locus as $\boldsymbol{d}$ moves from zero. Therefore,

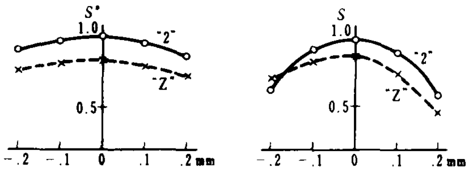


Fig. 5. Comparison of multiple similarity with ordinary similarity against displacement noise. The x axis is scaled with the displacement in millimeters. Note that in the case of ordinary similarity a reversal occurs at the displacement of 0.2 mm, and the standard width of a stroke of printed Roman is 0.35 mm.

it is definitely impossible to compensate the displacement noise by only one template. However, since the locus is known, we can avoid the loss incurred by the displacement noise by introducing more than one template and defining a new similarity in terms of the new templates. He defined the new similarity $S^*(h_0, h)$ as follows:

$$S^{*2}(h_0, h) = S^2(\varphi_0, h) + S^2(\varphi_1, h) + S^2(\varphi_2, h).$$

, where φ_0 is normalized h_0 and both φ_1 and φ_2 are constructed by the partial derivative h_0 with respect to x and y respectively. Then the following equations hold:

$$(\varphi_i, \varphi_j) = \delta_{i,j}. \qquad (i, j = 0, 1, 2).$$

The effectiveness of the multiple similarity is shown in Fig. 5. This method has become the main algorithm of Toshiba's OCR systems. Iijima's theory is not so easy to understand, but his recent book [55] is readable, although it is written in Japanese. For those who are good at linear algebra, it is a good way to illustrate the method using the concept of vector space. That is, a template vector is replaced by a subspace which is spanned by more than one orthogonal base vector. The matching is just a projection to this subspace. Actually this is almost the same as the multiple similarity method and is called the subspace method. Watanabe [56] proposed the method almost at the same time as Iijima. The subspace method has been developed by many researchers and is still an active field. We recommend Oja's fine book [57] for reference.

4) Series Expansion: In general, the K-L expansion is a special case of a series expansion. In the case of the K-L expansion, orthogonal vectors are generated from a data set. Here, we describe some series expansions obtained using built-in functions. The most typical ones are moment and Fourier expansions, while Walsh and Hadamard expansions have also been studied (see Pratt's book [58]).

a) Moment: Humans can recognize characters when viewing from any direction, with the exception of a few characters, such as "6" and "9." Recognition of shapes, independent of position, size, and orientation in the visual field, has been a goal of research. From a practical point of view in OCR in particular, orientation invariance is not as important as position and size. However, it is a very interesting research goal and work on it has been continuing since Hu's theoretical work [59]. However, we are obliged to omit the related research work, because of the limitations of space and the pessimistic prospects of using many higher order moments in practice.

When we limit the invariance to only position and size, and/or slant, the moment method is not too bad. In 1962, a first systematic experiment was conducted by Alt [60] of the National Bureau of Standards (NBS), in which one font of alphabet was used which was normalized in position, size, and slant. Therefore, moments up to the second order were used as the normalization factors, and the third to sixth moments (22 kinds) were used for classification. This was a pilot program and no recognition rate was given for the basic experiment; however he was optimistic.

In spite of the positive perspective of Alt, very litle work followed [61]. In 1987, two and a half decades after Alt's work, a very systematic and reliable experiment on the moment method was reported by Cash and Hatamian [62] at Bell Laboratories. They used central moments only up to the third order and so ten moments (ten-dimensional vector) were used for classification. The data set tested consisted of six different machine-printed fonts (62-class alphanumerals). They compared the performances of three typical similarity measures: Euclidian distance, cross correlation, and Mahalanobis distance. The results were as follows: All three similarities achieved a recognition rate over 95% for all six fonts, and the weighted and normalized cross correlation measure produced the best recognition rates: 99% for four of the six fonts for the high-quality data sets. However, the print quality seems to have been good and no rejection class was taken. Therefore, an exact evaluation from a practical point of view is difficult.

b) Fourier series: Fourier series expansion is the most popular and so naturally had been applied to character recognition systems. In the early 1960's, such research had been started by Cogriff [63]. In general, concerning the studies that followed, refer to Zahn and Roskie's paper [64]. The representation of Fourier coefficients of a boundary can be divided in two ways. The first is based on the cumulative angular function which is expanded to a Fourier series. The set $\{A_k, a_k, |k = 1, 2, \cdots\}$ is called the Fourier descriptor (FD) for a curve, where A_k and a_k are the amplitude and phase of the kth-harmonic term. The other representation was proposed by Granlund [65] and developed by Persoon and Fu [66]. A point moving along the boundary generates the complex function $u(s) = x(s)+jy(s)$, which is periodic with period S. The FD(CF) now becomes

$$a_n = S^{-1} \int_S u(t) exp(-j2\pi S^{-1} xnt) dt.$$

A fully theoretical study was conducted in particular by Zahn and Roskies. In particular, FD's have invariant properties with respect to position and scale, but we have to note that they depend on the starting point of a boundary tracing.

Now let us examine FD's properties for real characters. In the case of the cumulative angular function (CAF) expansion, a reconstruction experiment was done using Munson's data set, consisting of a 24×24 binary matrix.

It was shown that the reconstruction of a hand-printed numeral "4" is almost sufficient using up to ten harmonics (ten amplitudes and ten phases). However, the reconstructed image is not closed, which is a general feature of CAF. It was demonstrated that CF is considerably better than CAF [4]. That is, in the case of CF, reconstructed figures are closed and five harmonics are almost sufficient to differentiate between "2" and "Z." On the othr hand, at least ten harmonics are necessary in the case of CAF. The experiment used a 64×64 binary matrix.

Persoon and Fu did a recognition experiment using Munson's numeric data set. The error rate was 10.6% with no rejection class. Conspicuous misclassification occurred between "8" and "1," which was due to the fact that only the outer boundary was traced. Taking this fact into account, we may say that it was a good result, but the size of the data was so small that we could not evaluate it definitely. However, it took 14 seconds per character using CDC 6000, the most powerful computer in terms of scientific calculation at that time.

In spite of this demonstrated potential, it was more than a decade before the first systematic and reliable experiments were conducted, by Lai and Suen [67]. In their experiment, FD's were used as global features and local boundary features, such as concave and convex types, were also used as local features. The classification scheme used was a binary decision tree. The data used were Suen's data set, consisting of 100 000 hand-printed alphanumeric characters in a 64×32 binary matrix. A separate experiment was also performed in which only FD's were used, up to five harmonics, i.e., ten features. The results were recognition rates of 81.74%. However, when the combined features were used, a 98.05% recognition rate and an error rate of 1.95% were achieved. From this experiment we can conclude that FD's are very useful for global feature extraction and powerful for distortion. However, they are not so good for local feature extraction, which coincides with the reconstruction experiments shown before. Their experiment was done with careful preprocessing, starting point normalization, and extraction of inner structure (holes).

On the other hand, Fourier expansion can be applied to the 2-D plane directly. Hsu *et al.*[68] did it using a circular harmonic expansion and formalized a rotation-invariant amplitude, which was proved by a pilot experiment. It is interesting to compare this method with moment invariant method, because the circular harmonic expansion is also a powerful tool for theoretical work.

5) Feature Matching: Mathematically speaking, matching in linear space is equivalent to a correlation/inner product between two vectors. So far we have taken the implicit meaning of template matching. Now we try to expand the scheme a little. That is, each element of a vector cannot be constrained to be a scalar value, but it can be a vector. In other words, each element can have a structure, i.e., a local feature such as direction or gradient. However, we note that the structure element can be represented by a vector. Therefore, the expansion means that a higher-dimensional vector is constructed and a usual inner product is done with the high-dimensional vectors. In this sense, there is no change in mathematical form, but semantics of the element change.

Actually such expansion was done very clearly at an early stage back in 1965 by Spinrad [69]. As described later, he extracted primitive stroke features, and then tried to match simultaneously the arrangement of the strokes and their attributes also. The direction of a stroke was quantized into eight and the position of the center of a stroke was also quantized directionwise into 16 viewing the center of gravity of the whole set of strokes, such as N (north) and NE (north-east). Thus, an 8×16 matrix was constructed and at each entry of the matrix a 3×3 submatrix was set, in which the row is the quantized distance from the center of gravity and the column is the quantized stroke length. Thus the total number of the elements was $8 \times 16 \times 3 \times 3 = 1152$. Therefore, 1152-dimensional vectors were constructed, and the correlation between the unknown input vectors and each template vector, corresponding to each class, was measured. The above operation is a somewhat mechanical expansion of the correlation, but we consider "correlation" deeply apart from its mathematical form.

Correlation on a 2-D plane might convey the idea that face to face matching takes place. However, this is an illusion. This is nothing but point to point matching and a summing up of these matchings. On the other hand, the 2-D plane has its natural topology; i.e., each point on it has its neighborhood. A blurring preprocess can be interpreted as a technique to incorporate the neighboring system with correlation. The reduction of a 2-D grid plane is also one such technique. The preprocessings mentioned above are of the lowest level and are also basic. A higher-level one is a gradient in which, for example, a 3×3 neighbor system is included into one element of a vector [70]. Now we introduce work based on this principle which was done by Yasuda [71]. He attempted a one-mask recognition system for hand-printed characters where each mask for a class was constructed as follows:

1) Direction at each pixel of each sample image is extracted and quantized into four directions.
2) Size normalization is done.
3) Blurring is done.
4) An average is taken for all the samples processed.

Figure 6 gives an example of the mask for "2," in which the top row shows four mask components corresponding to four quantized directions. From the second to the bottom the blurred and size-normalized masks are shown. The blurring factor increases from the second to the bottom masks. The data set consists of unconstrained hand-printed characters written by 25 subjects. The number of classes is 41, i.e., numerals, alphabet, and some symbols. The sample size is 8200 (41×200). In the first experiment, only the numeral data set was used. No rejection class was allowed. As a result 97.1% correct recognition rate was achieved using only a single mask for each class. The secondary experiment carried out on the total data set showed a drop in the error rate of 1.4% and variations in classes are

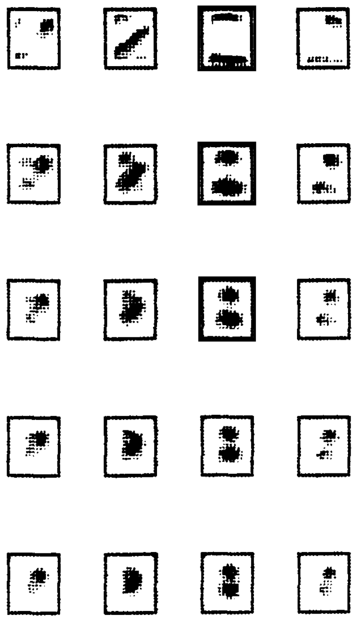


Fig. 6. Four directional feature masks of "2" with their blurred ones being normalized [71].

conspicuous. "N," in particular, gave a 22% error rate. The experiment demonstrated the power of directional matching with blurring.

Feature matching is still sensitive to stroke positions. In this sense, considerable efforts have been made for normalization. However, so-called linear normalization is not enough and so-called nonlinear normalization has been used [72]–[74]. The basic idea is to measure busyness of lines/strokes and relocate the strokes so that the busyness become uniform based on the measurement. Tskumo *et al.* gave an interesting comparison study of normalization methods.

6) Nonlinear Template Matching: In K-L expansion, we constructed a subspace as a template instead of a one vector template. In feature matching, we constructed a very blurred template on a feature plane, which means that it creates many templates belonging to one class implicitly. Now we introduce another method to generate a set of templates explicitly and to match against the set, which was done by Sakoe [75] in 1974. First of all, he assumed that a character consists of a sum/concatenation of vectors. We can easily formalize such a shape as

$$C = \{(c_1, c_2, ..c_k, ..c_K).(i_K, j_K)\}.$$

where c_k denotes the kth line segment/vector and is further expressed as $c_k = (d_k, l_k)$. Here d_k and l_k are the direction and length of the vector respectively and (i_K, j_K) denote the coordinates of the terminal point. The above expansion is very flexible when each c_k and (i_K, j_K) are changed. Therefore, we need to impose some constraints on it to represent a set of templates. The constraints are described as ranges of direction, length, and terminal points of a vector. Besides the above constraints, a weight, w_k to each kth vector is set, which is used to connect separated line segments by setting it to zero. Here, a set of the templates is denoted as B.

Now we need to consider how to define "matching" against the template set. First an input image on the $I \times J$ matrix is defined simply as follows:

$$A = \{a(i, j)\}, \qquad i = 1, 2, \cdots, I \quad j = 1, 2, \cdots, J,$$

where $a(i, j)$is multilevel in general. Next, such practical representation has also to be made for the conceptual set of templates, too. Thus, the similarity between A and B (set), is defined as follows: $S(A, B) = max\{(A, B') \mid B' \in B\}$, where $(A, B\prime)$ denotes inner product of vectors A and $B\prime$. Here we note that a maximum template is chosen from the template set B scanning B in the range specified. By searching for the best match, we can get the best approximate pattern $B\prime$ from the set B.

By the way, Sakoe also estimated the size of the set of templates of "0," for example, as $2^{4\times5\times5} = 1640000$ where K was set to 8, the variance of length is four pixels, and the variance of position of the terminal point is 5×5. This number is enormous, which shows its degree of flexibility in some sense, but we face too big a problem on how to search the set to find the best match. Fortunately, there is an efficient technique for solving the problem, which is called dynamic programming (DP) and was invented by Bellman [76], [77]. Thanks to an additiveness of the matching estimate as shown, we can apply DP to the problem. Sakoe named the method rubber string matching and solved the problem using DP and conducted an experiment on hand-printed numerals. We will show only the results. He made standard masks represented by B, manually, with a small pilot experiment to accommodate B's. Thus 12 masks were made. On 2000 samples, the correct recognition rate was 99.5% when no rejection was allowed.

The so-called DP matching has been broadly used in the pattern recognition field. The first application to image analysis was made by Kovalevsky [78] in 1967, in which segmentation of a character string on noisy background was tried. DP matching was successfully applied to speech recognition. Sakoe was an expert in this field and built a speech recognition system at NEC based on his DP matching method [79]. With regard to the application to image analysis, we recommend Kovalevsky's book. Recently, Yamada [80] has been very active in DP matching applications to both character recognition and map recognition.

In DP matching, order is also very important because of the nature of its multistage processing. In his case, the contour of an image was approximated by a polygon, in which we can establish a very natural order of line segments along the contour. The details and results will be presented later. At any rate, an interesting aspect of DP matching lies in the fact that both template and input images are described structurally, and matching is done componentwise numerically. In this sense, the DP matching implies the two natures of template matching and structural analysis.

7) Graphical Matching: Graphical matching is a very useful technique which can be applied to all areas in the field of pattern recognition and even other fields. Stroke segments and their relationships are represented by a graph in strict mathematical sense. Therefore, graph isomorphism and subgraph isomorphism provide a basic matching theory [81]; actually they are used when the number of nodes is small. Otherwise it still has the problem of complexity. Strict and general formalization from a more practical point of view was given by Ambler *et al.* [82]. There are some algorithms which can find cliques mechanically, but this is NP-complete in general. Therefore, a practical method was considered, namely the relaxation method invented by Rosenfeld *et al.* [83]. This was first applied to shape matching by Davis [84]. The relaxation method originates from Waltz's work [85], i.e., a filtering algorithm to prune incompatible candidate labeling in labeling problems in so-called block-world understanding. Here we gave only a historical introduction, due to limited space and because graph matching is not specific to character recognition. We recommend Ballard and Brown's fine book [86], along with [3]. We note that graphical matching is an independent matching algorithm as well as DP matching. However, for our convenience, the two matching techniques were included in the template matching approach.

B. Structure Analysis Approach

We have quickly reviewed the slit/stroke analysis methods, which constitute a set of the simplest methods of structure analysis. A common feature of these methods is that they look at a character only partially. Because of the limited use of hardware at that time, researchers tried to construct their algorithms for reading machines as simply as possible. Therefore, only partial features of a characters were detected and simple relationships were used. This trend was exactly the same as that of template matching. However, as said before, character information is two-dimensional in essence. Therefore, we need to look at the full structure of a character. Actually researchers became aware of that fact from bitter experience.

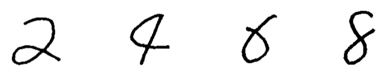
There are several viewpoints to systematically see the complete structure of a character. These are classified as follows:

- thinning line analysis,
- bulk decomposition,
- consecutive slits analysis/stream following,
- contour following analysis,
- background analysis.

Among these views, thinning line analysis has been most intensively investigated. Certainly this is a very important approach and based on it many OCR systems have been made. The analysis is higher than the other analyses in terms of abstraction, except for bulk decomposition. Bulk decomposition can be regarded as being at the same level as thinning line analysis, except bulk decomposition is applicable to the general shape. In the context of OCR, bulk decomposition was considered to have overcome some of the defects of thinning line techniques. We will describe these topics when we discuss the year 1980.

1) Thinning Line Analysis: Humans look at a character as a linelike object. This view is most intuitive, at least for adults. Here, "linelike" needs to be defined strictly. A well-known definition of a line is Euclid's, which has no width. Naturally we cannot accept such a definition in digital picture representation. So a one bit width is necessary and the connectivity of a line needs to be defined. A regular grid plane is not homogeneous. Such basic definitions of the geometric concept of a digital plane were first given by Rosenfeld and Pfaltz [87] in 1966 in their epoch-making paper.

We will not go into this problem, although this is very important and gives a base for the thinning process. Instead, we will show that when looking at a character with abstracting eyes, it is not too hard to analyze the structure of a character. In 1960 Sherman [88] regarded characters as consisting of abstracted lines and constructed a graph. In his graph he ignored a node which has two outgoing lines (degree of node is 2). Therefore, feature nodes are endpoint, branching point, crossing point, and so on. The adjacency matrix represents these relationships, but so loose that many different character classes belong to one class. For example, the numerals



are all placed in the same class. However, this topological view is very important, because it can absorb terrible variation of character shape.

Along this line, Beun [89] of Philips did an experiment on unconstrained numerals. He used other features, such as node position relationship, to resolve the degenerated class. The experimental results were a 91.7% correct recognition rate and a 2.6% substitution rate. He stated in his paper "According to personal experience, we knew that how easy to deceive ourself and others in such experiment." This statement means two things. First, the difficulty of doing unconstrained character recognition was shown. At that time, researchers were not aware of this fact, but a few engineers were. The very ambitious of Sherman indicates that. People gradually become conservative as they gained experience. Now we have roughly three classes of hand-printed qualities, i.e., unconstrained, loosely constrained, and constrained classes. For the constrained numerals, several kinds of rules were considered so that they could be easily recognized by a machine, as mentioned in connection with second-generation OCR's. The most reasonable target is the loosely constrained class. This is because people dislike being constrained and prefer having their freedom. Too much freedom, however, causes confusion among people, and Suen [90] revealed this point by conducting psychological experiments. The other thing is that we need some common data in order to compare researchers' experimental results. Therefore several data sets have been made so far. See the paper by Suen *et al.* [5].

Now we shall describe a very important preprocessing, called thinning. While it may be regarded as preprocessing, it should be treated as more than preprocessing. At any rate, it is needed to obtain abstracted lines. Actually Beun did it [89]. An observed line usually has a width greater than that of a pixel. So the line is eroded from both sides keeping some constraints so that the line is not broken and shortened.

The idea of thinning was suggested by Kirsch [91] in 1957 and programs to describe bubble chamber pictures were written by McCormick [92] and Narashimhan [93]. For medical image applications, Rutovitz [94] tried it, as did Deutch [95] for character recognition. However, the first systematic and rigorous algorithm was given by Hilditch [96] in 1969. Since then, more than 30 variations of thinning algorithms have been proposed, some of which were compared by Tamura [97]. However, it is well known that we cannot obtain perfectly abstract lines for real lines, which include acute corners and/or intersections. This is a basic problem which cannot be avoided because of the local operations used. Therefore, for its actual application some postprocessing is done. Since it is an iterative process, it is time consuming. However, thinning is a basic preprocessing in OCR technology as well as in recognizing drawings and this is widely used in many OCR systems, as we will see. Research on this continues.

Now we move to the analysis of a set of lines after the thinning process. As mentioned above, the analysis is divided into two parts. Suppose the line set is a graph; one method is to detect nodes/singular points, and the other is to describe edges/arcs. For both purposes, a 3×3 window/mask has been extensively used. Although it is comparatively easy to detect simple and local segments, it is not so easy to obtain more global segmentation of lines. We will discuss this point further below.

Concerning the description of arcs, there is a simple technique, called chain encoding or Freeman code, developed by Freeman [98]. It is also based on a 3×3 window. We can set a 3×3 window at any point on a line, as shown in Fig. 7(a). We assume that the binary image is scanned from the left and the bottom. The scan meets a black pixel eventually. Then the line coding starts, following the line, and looking at the neighbor from the center, we can decide the next moving pixel uniquely as shown. Eight such local directions are encoded, as shown in Fig. 7(b). For example, the line is coded as 221100. This encoding scheme is naturally applicable to contour following and is extensively used. In some sense, this is the counterpart of run-length coding, which is used to code a total plane image. Both are the first stage in describing the shape in general, from which we need to describe the shape more globally. For the above example, the reader can guess easily that there exists a corner looking at the chain code.

2) Bulk Decomposition: The work of Grimsdale *et al.* [99] is surprisingly basic and valuable considering the time when the work was done. This is the first paper which dealt

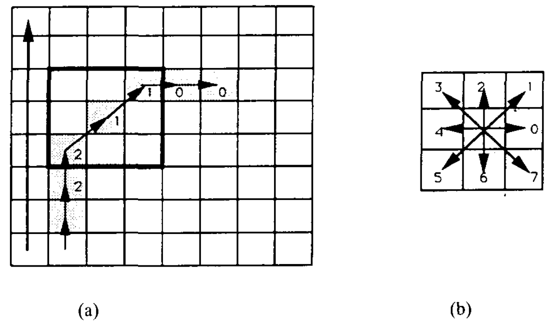


Fig. 7. Illustration of chain encoding: (a) a simple example; (b) the coding rule.

with bulk decomposition systematically. The concept of bulk decomposition is intuitively clear; i.e., a letter "L" can be regarded as consisting of vertical and horizontal lines, for example. The decomposition can be regarded as a counterpart of principal component analysis in template matching. However, it is very difficult to deal with the problem from the theoretical point of view. So they virtually began the decomposition from run-length coding. Consecutive black runs are analyzed and connected if certain conditions are satisfied. Otherwise, a new part/segment of a character is generated.

Their method is considerably practical in terms of algorithm. The first is that they took noises, breaks/gaps, and isolated small blobs into account. The second is that some proof for the rotation problem is provided. After finding each segment in a character, the final description of the character is done. This consists of two parts: first attribute description of each segment, i.e., length, slope, and curvature; second, describe the relationship between individual segments, table of joints. In this stage, merging is done so that overlapped segments are combined. Thus, it can be seen that the basic processes of structural analysis such as feature growing, called macroprocess and merging, are clearly mentioned. However, naturally it took 60 seconds for 4K instructions in their computer at that time. So their work ended up as basic research.

The work of Grimsdale *et al.* was too sophisticated at that time. So a more engineering oriented method was considered by Spinrad [69]. His method is very simple and can be regarded as an application of the slit method. It consisted of eight directed slits on a frame within which a character image was set. For example, a vertical slit is moved from the left to the right on the frame, then for a letter "D," at some movement, the slit intersects the vertical stroke of the image of "D." Thus, we can detect its vertical stroke. On the other hand, concerning the classification, his method was based on feature matching in general, which has already been described.

After Spinrad, Pavlidis [100] extended his approach from the theoretical point of view. He proposed the kth integral projection. All the black runs for any scan can be labeled by number, for example, from the left: 1, 2, 3, and so on. The kth integral projection is just the sum of the kth labeled black runs. The projection method is very resistant against noise, i.e., ruggedness of boundaries of character images. In the case of Roman alphanumerals, it is not so effective, but for Kanji the expansion of projection/profile is essentially important and broadly used in Kanji OCR. Pavlidis continued his research on bulk decomposition rigorously. As mentioned before, the abstraction level of bulk decomposition is higher than boundary/contour decomposition. Therefore, if we once succeeded in that, we could obtain a very simple and powerful description of shape in general. Pavlidis has stated the problem in his representative survey paper [101]: Thus regardless of formalism one is faced with the task of segmenting S ("irregularly" shaped subset of the Euclidian plane) into some kind of "regularly" shaped subsets.

This is a fundamental problem in pattern recognition. Regularity is very hard to prove, even if some theory were possible. However, we need some guidance which is got from a general strategy/criterion. We pick up some parts from the paper: They must conform with our intuitive notions of "simpler" components of a "complex" picture and have a well-defined mathematical characterization. This is a very important remark which should be applied to any type of shape decomposition. Along this line, he proposed convex decomposition, but it was somewhat rigid. Several researchers challenged this decomposition problem. The work continues; however, it is still in the realm of basic research. Concerning "regularity," see also the paper by Simon in this issue.

3) Stream Following Analysis: The idea of stream following seems to have been first described by Rabinow [102] in 1962. He used the term "watch bird" to describe a curve tracing circuit. However, in the same year exactly the same idea was reported in the *RCA Review* by Sublette and Tults [103], where a more concrete form and labeling technique was fully used to follow streams. The strict description of stream following was given by Perotto [104] in 1963, who was not aware of the forerunners. In any event, Rabinow and Perotto intended to make a reading system for handwritten characters, and Sublette *et al.* constructed a software system for reading multifont characters. A common point is that their target character set is rich in variation, which reflects the nature of stream following. Another interesting aspect is that RCA took a two-way strategy to OCR systems: One is an analog/hardware oriented approach, mentioned before; the other is a digital/software approach. At that time, the latter was inferior to the former, due to the slow speed of 1 chs using the large computers of that time. It was hard to test their system.

Before illustrating stream following, it may be helpful to the readers to note the order of description on a two-dimensional plane. Although we can take any coordinates on the plane, humans seem to take few coordinates unconsciously. A coordinate system is important for character recognition, in particular. In other words, humans assume natural orders on the plane, such as from top to bottom and right to left. Such an order is very important for describing shape.

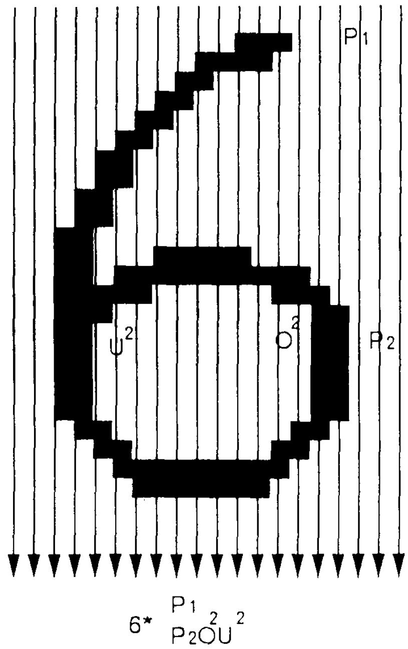
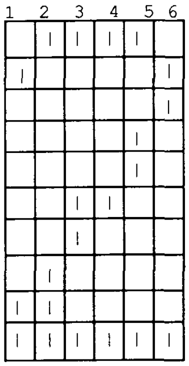


Fig. 8. Illustration of the stream following method. The image of "6" is scannned from right to left vertically.

Fig. 8 shows one example given by Perotto, which uses the order mentioned above. The order of raster scanning is from the top to bottom and from the right to left. Suppose that a slit moves from the right to left. First, there is no black component (b component) appearing in the slit, which is labeled P, as an abbreviation for principle. The coordinates of the position are stored (center point). The b component is followed. Again at once, two b components appear in the slit, both connecting to the b component detected previously. Then the state is labeled D, denoting branching of order 2. The two b components are followed. Immediately a new black component appears in the slit, which is labeled P, and its coordinates are stored. Thus, the three b conponents are followed. At the position close to the left, the state D merges to one b component, whose state is labeled U, denoting union of order 2. Finally, we find no more b components in the slit, which brings to an end the following process of the b component. According to the coordinates of the two P's, the first and the second P are labeled P_2 and P_1 respectively. Thus, "6" is labeled as shown at the bottom of Fig. 8. The order of the labels is important and reflects the order specified at the beginning.

As can be seen, from the description this method is very simple and is very strong against variations of shape. Perotto called the description a morphotopological description. However, we need to note because of the simplicity, considerably different shapes are identified as the same. Therefore, we need another raster scanning horizontally, in order to distinguish among "U," " $\cap$ ", and " $-$." For "+" and "T," diagonal scanning is necessary. Nadler [105] proposed such scanning, but it was found to be unnecessary.



LL	LR	RL	RR	C2	C1	Label
*	->	<-	*	1	2	
<-	<-	*	*	2	1	
*	*	*	*	1	1	
*	*	*	*	1	1	
*	*	*	*	1	1	
*	*	*	*	1	1	
*	*	*	*	1	1	
*	*	*	*	1	1	
*	<-	*	<-	1	2	

Fig. 9. Schematic illustration of the principle of feature extraction used in the NEC system.

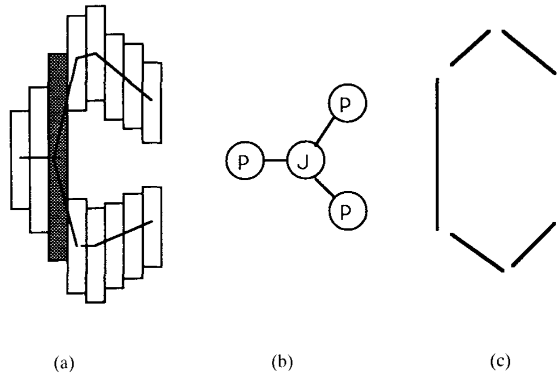


Fig. 10. Illustration of vectorization method based on LAG.

The description of stream following is based only on the topology of each slit. However, scanning gives run length, and so we can use profiles to extract the geometric attribute of b components. In fact, Uyehara [106] proposed a stream following method in the same year as Perotto with one modification. The term *stream following* itself was coined by him. However, there is no noise removing process and it only works well for ideal patterns. Subletts *et al.* worked on the stream following method.

Nadler [107], in 1974, gave a simple algorithm of stream following analysis, in which he used a small window of 2×1. The final description is graphlike and intuitive, being slightly redundant. The algorithm is given by the transition table/diagram of an automaton whose number of state is eight, while the original paper described a seven-state automaton. On the other hand, in 1969 Hoshino and Miyamoto [108] presented a paper on feature extraction, which was based on stream following analysis with a systematic geometric feature extraction. A schematic illustrative figure is shown in Fig. 9. The decision logic is a simple sequential symbol matching keeping the order mentioned above [109]. This is the main algorithm of the first NEC postal code number reader, which will be described in detail later. From the general point of view in image processing some algorithms had been developed based on the same idea and use run length encoding [110]–[112]. Among them the paper by Agrawala *et al.* is pedagogical and fit for computer processing, i.e., a sequential approach.

4) Vectorization: A new method of thinning using vectorization should be introduced [113] . However, we can regard it as a kind of bulk segmentation and it includes stream following. Therefore, we place its description here. First of all we note that the abstraction level of vectorization is higher than thinning and bulk segmentation . Connected lines are separated, a segmented bulk is represented by a line, and the relationships between the vectorized lines are described as a graph.

The vectorization is based on what is referred to as a LAG (line adjacency graph) [114], which is easily obtained from run length encoding, as shown in Fig. 10(a). It can be seen that the LAG exists implicitly in stream following and also in the work of Grimsdale *et al.* The LAG is transformed to a compressed LAG, called c-LAG (shown in Fig.10(b)), in which nodes of degree (1,1), where one segment is connected above and below, are mapped into single nodes. Junction nodes are connected by more than one path. Each path is examined by each segment width and the collinearity of the center of each segment. The path is then divided further into groups of segments, each of which has the same width and collinearity approximately. Merging is done for two collinear groups even if they are separated by another group between them. For example, imagine a vertical line of "E." The last example gives a feature of intersection implicitly. This method is based on run length encoding and so is very fast compared with the usual pixel-based thinning method. Based on the feature extraction and representation, Kahan *et al.* [115] had developed a recognition system for printed characters of any font and size.

5) Contour Following Analysis: We mentioned before that raster scan sets a natural order in a 2-D plane. Contour following also gives a very suitable natural order invariant to rotation. In other words, feature events along a contour of a closed shape is a circulant description. However, in a real application, for example, a starting point is fixed from the leftmost point. The contour following direction is also fixed clockwise looking at the black side. We note here that the term of contour following is used in a broad sense, i.e., including both boundary tracing and line (thin) following.

Historically speaking, the first contour following mechanism was given by using an analog device, called a flying spot scanner. This was developed by Greanias *et al.* [29] and was fully used in the IBM 1287 OCR system. A beam motion is attenuated within a black region so that the beam draws a cycloid traversing the boundary. At that time, it was very fast and flexible. Furthermore, it could absorb the coarseness of the boundary. On the other hand, its computer counterpart was very slow and suffered from boundary noise. We have already mentioned the computer-based mechanism of contour following using chain encoding. Other methods had been considered, but it became a standard method, although readers should be careful of some pathological cases. Readers can refer to Pavlidis's book [114].

The IBM 1287 was a very fine OCR system, but it was eventually replaced by an all-computer system except for input and output systems and optomechanical systems, along with electronics-oriented general trend. Coarseness on the grid plane is the first problem of contour following. Some smoothing techniques have been developed. A 3×3 window averaging process is one of them. A direct averaging technique on a contouring curve was done by Freeman [116] and Gallus *et al.* [117]. The latter is an extension of the former, both of which used chain encoding and defined local curvature based on chain encoding. They are theoretically very interesting, but are omitted here for reasons of space.

The second problem is feature selection along a contour. The third problem is how to segment a contour. The third is the most difficult and fundamental problem, as mentioned before. However, we can avoid the segmentation problem by introducing normalization of character image size and zoning the frame within which the image is squeezed or expanded. This is an ad hoc method. We introduced the zoned feature method before. That is based on a contour being segmented by forcing it into subregions of, say, 3×4 pixcels. The direction of contour following is taken as a feature. Thus, all the features describe direction and position, i.e., what direction is the contour and which partition of the 3×4 subregions is the contour in. Such features are sequentially checked from the starting position, along the contour. Concrete description is given later. However, we note that such normalization does not always work well when considering large shape variation/distortion of characters.

Therefore, the segmentation problem has been seriously considered. The simple idea is to use curvature maxima points, and this was accelerated by the famous psychological evidence by Attneave [118]. Although much work has been done, the first reliable technique was given by Rosenfeld and Johnston [119]. Analytically a curvature is given by a function of local derivatives, but actually it has the global nature suggested by Freeman. This was extensively examined by Freeman and Davis [120]. The corner feature plays a crucial role. For example, sometimes differentiating between "O" and "D" and between "5" and "S." So considerable effort has been spent in this area.

However, the methods developed do not always give stable and consistent results. Therefore an alternative approach was taken, one based on a polygonal approximation [121], which is global and very strong against noise. Theoretically speaking, the two approaches mentioned above are closely related [122]. So considerable work has also been done on polygonal approximation methods. Here we give a few representative works (Ramer [123], Pavlidis and Horowitz [124]). In particular, the work of Pavlidis *et al.* is very general. It is not constrained to make a connection. In this sense, it gives a global approximation which meets human intuition. Other polygonal approximation algorithms can be found in [3] and [4].

In fact, Pavlidis and Ali experimented with unconstrained handwritten numerals of the first file of IEEE data base

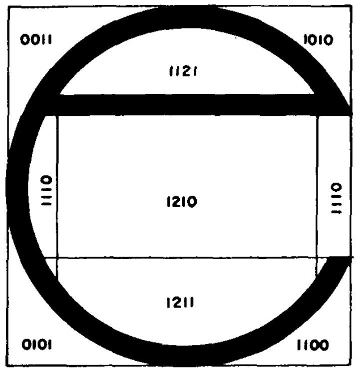


Fig. 11. Ternary coded regions. The four corners, however, are merged by taking binary code [24].

1.2.2, prepared by Munson. Input images were reasonably approximated by the polygon, i.e., not too fine and not too coarse. Therefore it is easy to detect global convexes and concavities, which are very important features. Naturally, holes are also so and are easily obtained in the course of contour following. They achieved reasonable recognition results. However, it is somewhat hard to estimate the result because Munson's data are so sloppy and small.

6) Background Analysis: A basic idea of background analysis was given by Steinbuch [125] in 1957. He assumed a potential field which is caused by electric charge whose distribution is just the same as the density distribution of character image. A simplified scheme was proposed by Kazmierczak [126] in which the potential gradient was coded.

As mentioned before, independent of them and without any analogy to physics, Glucksman took an approach of background analysis. His method is considered an extension of the sonding one, where each pixel of the background, four directed sondes, are spanned along the x and y axes. More specifically, each background pixel takes a four-digit code. This may be a binary, ternary, or higher-order code. An example is shown in Fig. 11, in which the background region is segmented into mainly seven subregions. The four corner regions have a more complex configuration, but this is meaningless and so binary code is used. As a result these subregions are effectively merged. However, no explicit region description was attempted. Classification was done using the feature vector, each element of which is a number of the ternary coded pixels. The histogram of the background features, ternary codes, was used. Theoretically, the feature vector has 81 dimensions, but 30 were sufficient in practice. An experiment on alphabetic characters of nine fonts and 52 classes of uppercase and lowercase letters was done. The total number of samples was 26 643, a large number at that time. The data sets were provided by Casey of the IBM Watson Research Laboratories. The correct recognition, rejection, and misclassified rates were 96.8%, 0.3%, and 2.9% respectively.

We continue the history, turning to algorithms. Munson of SRI [127], [28] proposed an interesting method, also

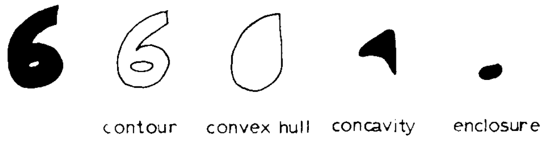


Fig. 12. Schematic illustration of Munson's idea [128].

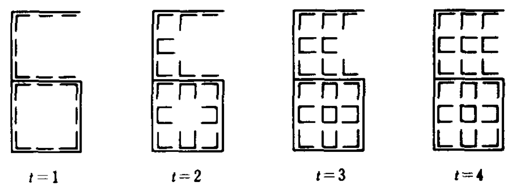


Fig. 13. Schematic issustration of the macroprocessing in the field effect method.

based on background analysis using contour following. A character image's boundary is traced and some points are marked as extreme points. Using these extreme points, a convex hull is constructed and connected regions adjacent to both the boundary and convex hull are detected as concavity regions, as shown in Fig. 12 [128]. Enclosures are detected when tracing the boundary. Concerning the detection of the extreme points, however, only a simple procedure was given. That procedure involves tracing a boundary, clockwise with a right hand system. Points turned to the right were searched and consecutive extreme points were connected with a straight line so that they lie inside the image.

Such a procedure is local and generates many extreme points; therefore some filtering technique must be provided to find global extreme points. No further structural analysis was done and both concavity and enclosure were used as a part of the components of the feature vector. This was based on which classification was done, as in Glucksman's case. The other components were profile and "spur," a kind of stroke. An experiment on a hand-printed 46-character FORTRAN alphabet was conducted, which gave a 97% correct reading rate for test data and a 3% of error rate. The data size was not specified. It is interesting to compare Glucksman's approach with Munson's. The former used only the background feature, but the latter as a part of the method. Both experiments obtained high recognition rates. We note that Glucksman's ternary coding reflects the relationship among concavities and holes implicitly, but in Munson's case, background features were used in isolation.

In the U.S. further developments were not made. On the other hand, a series of R&D efforts for OCR had been done nationwide in Japan as part of a national project, called PIPS (Pattern Information Processing System). The center of the research was ETL and it was conducted by Mori's group. The first task performed was to create a hand-printed character data base. Mori felt strongly about the need to have such a data set. A committee for it was chaired by Prof. Hiramatsu of Denki University and the principal related companies participated in the committee. Up to ten sets of data sets from alphanumerals to Kanji were assembled with the cooperation of the companies in which Yamada's and Saito's technical contributions were great. The management work was done by Mori and Yamamoto. The data sets are rich in their varieties and their volumes, and were made public internationally, which contributed to the basic R&D of OCR in Japan.

At the beginning of the OCR project at ETL, Mori established three guidelines to perform the research. These guidelines were to take structural , two-dimensional, and analog approaches. As a structural approach, they took background analysis. Three versions were developed. The first one, called system I, was considered by Mori and Oka [129], [130], in which a potential field was constructed merely to reduce the dimension low from a 60×60 to a 10×10 binary image. On the reduced plane each pixel has analog value virtually and at each 3×3 window local field was determined and labeled. However, system I was too primitive. The next version, called system II, was proposed by Mori [131], in which all 3×3 local potential fields were labeled using three digits: 0, 1, and 2. These digits just correspond to potentially different viewing from the center value. These three-level labels are transformed to symbols when their interactions take place in order to use rules which are represented by symbols in the two-dimensional configuration. Thus the iteration proceeds on the symbol field until no further change is possible. A schematic diagram of the macroprocessing is shown in Fig.13 as a typical example. Its merits are a very simple structure of masks and astonishingly strength against background and boundary noise.

However, it was not still analog in its complete sense and the research was further pushed, mainly by T. Mori, S. Mori, and Yamamoto [132], [133]. The key lies in the three-level labels, which are intermediate between analog and symbolic representations. Therefore, we can pass the symbols and use the three-level labels directly by means of transferring the digits themselves. Now the movement to system III was easy. Only the construction of the exact formalization remained, which was done by T. Mori. We skip the formulation, which is described in the cited references. A readable one appears in the book [4], which is written in Japanese.

The important point is that many rules are replaced by only two dynamic equations of density propagation. This is the strong point of the analog representation, which will be discussed later. However, the representation allowed 60 s/ch to be processed using a large scale computer of that time, for example, the TOSBAC 5600/GE 6000. Therefore, two kinds of small-scale experiments were conducted. One was for the ISO-B font data, which were very thin or of very dark print quality and consisted of 46 classes. The amount of data tested was 1300 characters and masks were made for 1, 2, 6, 8, D, Q, V, and W classes considering typical pairs which were hard to differentiate, such as (1, I), (2, Z), and (6, G). The correct recognition rate was 100%. The other was for constrained hand-printed numerals, where

2100 samples were written by 21 subjects. The correct recognition, rejection, and error rates were 98%, 0.2%, and 1.8% respectively.

On the other hand, as they proceeded to the more distorted data sets of unconstrained hand-printed alphanumeric characters, they encountered the same problem as Glucksman's case. Some subregions occurred within the same concavity region. The next step was to merge apparently separated field subregions which occur in a somewhat complex situation. The merging was easily done because the apparently separated field subregions are connected through a white line. The final step was to describe the relationship between field regions which are represented by representative fields. For example,"A" has two field regions, a hole and a concavity opened to the bottom. The relationship between the hole and the concavity regions was obtained by expanding propagation of the field regions and so if two field regions are adjacent, then the two collide with each other. The length of the collision line is an analog representation of the relationship between the field regions. An example can be seen in the book [4].

On the other hand, they found that the propagation equation includes two modes: One gives nonuniform fields as shown above; the other uniform fields, each of which is represented by any point in the field. This is similar to the special case of Glucksman's method, which is a binary coding system. At that time, it was very hard to pursue the analog system even on a computer, because it consumed too much time in its computer simulation. Therefore, they tried to make a simple version of the field effect method. One way is to use Glucksman's binary coding system. However, it generates counterintuitive regions, as mentioned before. Therefore, they needed to be merged and altered in order to meet with our intuition. Such processing was done by Zhan and Mori [134] based on a linguistic method proposed by Narashimhan [135], [136]. In 1977, Komori *et al.* [137] of NTT gave a very simple equation for these production rules. Furthermore they constructed global fields, called concentrated codes, including black strokes. Some examples are shown in Fig.14, where PS and PH stand for slant and horizontal strokes respectively. A recognition experiment was done in which a histogram of the global fields/codes was used for the classification as well as Glucksman's case. The data set was 10 000 hand-printed numerals written by 350 untrained writers. The correct recognition, rejection, and error rates were 99%, 0.6%, and 0.4% respectively.

Returning to the OCR project at ETL, the development of practical algorithms was needed, which was conducted in two ways. One was one microcomputer/programmable OCR; the other was to develop an algorithm assuming some hardware implementation to obtain a reasonable speed. Yamamoto and Mori [138] took this approach using Munson's idea. As mentioned before, in Munson's case, some difficulty was anticipated in finding global extreme points. The problem in practice was how to construct a convex hull . The definition of convex is well known: Let U be a subset of a linear space L. We say that U is convex if $\{\boldsymbol{x}, \boldsymbol{y}\} \in \boldsymbol{U}$ implies that $\boldsymbol{z} = [\lambda \boldsymbol{x} + (\lambda - l)\boldsymbol{y}] \in \boldsymbol{U}$ for all $\lambda \in [0, 1]$.

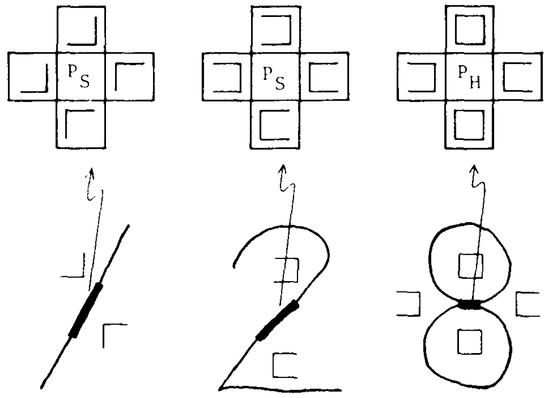


Fig. 14. Examples of the concentrated codes [137].

Table 1 Results of the Experiments on the Outermost Point Method

Set	Test Set			
Data	K	N	A	K+N+A
Mask	K	N	A	K+N+A
Correct	99.7	99.6	99.9	99.0
Reject	0.26	0.4	0.1	0.9
Error	0.04	0.0	0	0.1

The letters K, N, and A stand for katakana, numeral, and Roman alphabet, respectively [138].

This is not a constructive definition and it is of no use to construct a convex hull. Fortunately there is a constructive definition. Imagine a boundary point of the subset U of 2-D Euclidean space and a tangential line passing through it. Then further suppose a half plane whose edge is just the tangential line and includes larger U than that of the other opposite half plane. We can imagine such a half plane at every boundary point of U and select only such half planes that include only U. Then we can construct a convex hull of U by finding an envelope of the edges of the half planes. Their idea of the outermost point method is based on that constructive definition of a convex hull.

Once a convex hull is constructed, it is easy to detect concavities. However, the attributes of a concavity are important and they take three kinds of attributes, i.e., closeness, open direction, and area. The closeness is a measure of concavity, such as shallow or deep concavity. Very systematic experiments had been done using ETL data sets, ETL-3 and ETL-5. Four sets were arranged, i.e., numerals (ND), alphabet (AD), Katakana (KD), and mixed data sets of ND, AD, and KD. First of all semiautomatic masks were generated, which we will discuss later. ND, AD, and KD consist of 2000, 5200, and 9200 respectively, among which 1000, 3900, and 6900 were used as learning data respectively. The rest were used as test data sets. The results are shown in Table 1, which shows the high performance of the method. The reading speed was 0.7 s/ch using the TOSBAC 5600.

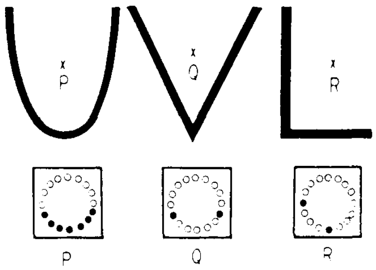


Fig. 15. Schematic illustration of the cell features.

In another direction, Oka continued basic research based purely on the field effect method. One drawback of the method was the difficulty in detecting geometric features. Actually the analog type of FEM had been combined with the Hough transformation to construct a recognition system [139]. The usual Hough transformation on the (ρ, θ) plane was not used; however, they applied edge to detect a line direction. The work [140] was done independently of O'Gorman and Clowes's [141]. At any rate, Oka [142] solved this problem using edge propagation. The general framework is based on a cellular automaton and each cell has a certain number of intracells according to the resolution of edge direction. Some illustrative and schematic examples are shown in Fig.15. The cells P, Q, and R are located at the points P, Q, and R. The system was suitable for LSI and implemented, as will be described later.

7) Syntactic/Linguistic Approach The syntactic or linguistic approach to pattern recognition has attracted many researchers in this field. In particular, Fu advocated the application of this approach to pattern recognition and developed it very actively with his colleagues [143], [144]. The approach has been included partially and/or implicitly as we have already described and will be mentioned further in real OCR systems. The basic idea has been explained before. Here we introduce it from a practical point of view. By scanning the labeled plane in a certain order, as already explained, we can construct a string of labels. The problem is how to construct an automaton which accepts the string as belonging to a class. A language whose grammar accepts the string must be constructed. A syntactic approach makes use of formal language theory, but a real image is not so simply recognized, as might be expected. Some typical problems were described by Ali and Pavlidis [145] as follows:

1) The direct syntactic analysis is faced with the need to handle the effects of noise, which causes rather complicated strings.
2) The parsing of the whole boundary requires the use of context-sensitive grammars for a description of the complete object.

In order to overcome these problems they used polygonal approximation and syntactic technique in a hierarchical manner. The former has already been mentioned. Later we will introduce another approach to problem 1. For the latter, they assumed that any closed boundary can be expressed by concatenations of semiglobal features/labels: Arc, Trus, Line, and Break. Here "Trus" means a sharp protrusion or intrusion and "Break" indicates the rest of regular shape; a Break is like a short segment having no regular shape. Thus, circular representation of a boundary is definitely segmented in this frame so that the context-sensitive problem is avoided. The production rule is very simple and intuitive, and the number of rules is 9. For example,

$$\text{TRUS} \rightarrow \text{STROKE} \mid \text{CORNER}, \text{CORNER} \rightarrow \text{LINE1+LINE2}$$

$$[\text{the-angle-of(LINE1, LINE2)} > \theta].$$

The experiment was conducted using Munson's data set, which was also used in their first version, mentioned before. The performance was reported as follows: error rate and rejection rate were 1.46% and 1.04% on the design set of 480 characters; error rate and rejection rate were 3.57% and 3.45% respectively on a testing set of 840 characters. Comparing the first version, they achieved considerably better results; the error rate of the first version was 14.7% on a testing set of almost the same size. However, their success lies mainly in the appropriately rough representation of the polygonal approximation and so it seems to be not easy to extend the method beyond numerals to a larger character set such as alphanumerals and Katakana. Concerning problem 1 above, Abe formalized a so-called penalty automaton [146]. To avoid the noise problem, counting and thresholding mechanisms were introduced into the automaton formulation.

Besides the above-mentioned problems, further difficulties exist. Given that an image is two-dimensional, a simple concatenation is not enough to represent a 2-D image; therefore, various natural high-dimensional generalizations of strings have been considered, such as Webs, plex structure, and trees. As far as we know, no concrete application of these has been reported in any OCR system. For more information we recommend the very readable book written by Gonzalez and Thomason [147].

8) Algebraic Approach: Operators for 2-D concatenation have been used by Shaw [148] in picture description language (PDL). In his scheme, a primitive is abstracted very generally by a line segment which has two points: head and tail. Four kinds of operators were defined on the primitives. We can construct letters of any shape using these operators and vectorlike primitives. In other words, we first need to determine the primitives to be used so that they are appropriate for the actual problem of representing characters. Mori [149], [150] investigated the problem from an algebraic point of view. His basic consideration was that a character set gives humans some feeling of roundness or linelike feature. This roundness or linelike feature must be reflected by the primitives that are used to generate the character set. Another guideline was that the primitives must be simple, intuitive, and strict in the mathematical sense, as described by Pavlidis (previously mentioned). He therefore introduced four kinds of monotone functions, so-called L-type primitives assuming contour following. A set of the primitives was treated as a set of integers (0,1,2,3) and a concatenation of the primitives can be represented by a "+" operator as mod 4. He derived mathematical shape descriptions based on the primitives. The point is that topological/geometrical properties are represented very simply and exactly by an algebraic form.

On the other hand, Nishida and Mori [151] further developed this line, in which the scheme is applicable to both boundary and thinned line. The primitives are four monotone functions with head and tail, including both horizontal and vertical line segments. The remaining two are "/" and "\" if schematically represented.

On these primitives, four kinds of binary operations are denoted as

$$a \xrightarrow{j} b, \qquad j = 0, 1, 2, 3,$$

where j is a characteristic number of the operator, and a and b are primitives. The operators have very intuitive meaning as well as mathematical rigor. In choosing appropriate monotone functions a and b, we can construct concavities (convexes), for example, which are open to the right, bottom, left, and top, corresponding to 0, 1, 2, 3, respectively, with a very flexible manner due to only the restriction of the monotone functions. A line segment is represented by such links of primitives. If the links are smoothly connected by a right-hand system, then they are represented by one sequence of the primitives, called a primitive sequence. That is, a line segment is compactly and hierarchically represented. In Mori's work, only quasi-topological features were considered, but both geometrical and quasi-topological features were systematically integrated, introducing singular point decomposition. Many mathematical formulas were derived, some of which are extensions of classical geometry, for example, the polygon.

Based on the theory and a new thinning algorithm [152], which is stable for simple figures of structure, a recognition experiment was conducted on a data set of 13 400 characters written by 220 subjects. The data set can be considered loosely constrained data with a variety of writing styles. Training and test sets were disjoint, selected in the ratio 1:1 in a randomized manner from the data set. Recognition, rejection, and error rates were 98.7%, 1.0%, and 0.3% respectively for the test data set. The total number of masks used was 46, in which that of "8" was 10, which was conspicuously large. This reflects the complexity in the shape of the numeral "8." They did another experiment removing poor-quality samples from the data set. This resulted in a data set containing 11 500 samples. The two methods were compared, the new method, the algebraic and quasi-topological method (AQT), based on the thinning method, and the contour following analysis. For the former, the recognition and error rates were 99.1% and 0.1% respectively using 37 masks. For the latter, the recognition and error rates were 97.6% and 0.4% respectively using 58 masks. Based on these experiments, they claimed superiority of the AQT method over the contour analysis method. We note that its flexibility is high, comparing the number of masks of the former with that of the latter.

V. Commercial OCR Products

A. Current Status

During the past ten years, a number of optical character readers have disappeared from the market place. They have been replaced by new and more powerful ones which appeared in many different forms, among them small hand-held wands with the shape of a handgun, hand-held scanners like a sweeper, page readers, flat-bed scanners, and document readers integrated with mechanical transport devices [153]–[157]. The prices of these machines range from $500 to over a million dollars. The majority of them are PC-based and cost between $2000 and $10 000 (software packages excluded). Because of space limitation, this section mainly deals with the machine capabilities and gives details of the specifications of different OCR products and lists of manufacturers, especially those in America and Europe that can be found in recent surveys compiled by the authors mentioned above.

Hand-held scanners and wands are often used to read UPC codes, magnetic data, and materials (such as tickets, price tags, checks) where severe constraints have been imposed on the printing of the codes and characters, e.g. bar codes, magnetic inked characters, and OCRA characters. They are mostly limited to the reading of numbers and just a few additional letters or symbols. Typical OCR machines can read a few 10 and 12 pitch typewritten fonts, including OCRA, OCRB, and others in single and double spacing. Depending on the complexity of the layout of the text and number of fonts treated, their reading speeds vary from five to ten pages per minute. With a scanning resolution varying from 150 to 400 DPI (dots per inch), normally these optical readers can tolerate a skew of $2 \sim 3$ degrees, or a quarter of an inch per line. A number of them can output files in the formats of common word processors in addition to the ASCII format. Most are able to reject unrecognizable characters or documents and allow manual key reentry of such characters or documents afterwards. Since the recognition rate depends heavily on the quality of the print and the preparation of the document, the resolution of the scanner, the fonts used, the complexity of the layout and format, and factors other than the recognition algorithm, many products do not state their recognition rate. Those which state such a rate, generally say that it is over 99.9% when operated under optimal conditions. Sometimes such a specification can be very misleading.

More sophisticated optical readers can function both as image scanners and as OCR machines. Such combined functions allow them to capture both pictorial and text information for storage, transport, and retrieval purposes. Software management systems also allow the user to rapidly build a full text and graphics data base from the scanned input. These readers can process documents in multifonts or omnifonts which can be typewritten, typeset, and printed by dot-matrix, line, and laser printers. They can recognize characters with font sizes from eight to 24 points at least, spaced between four and 12 lines per inch, in different formats including intermixed text and graphics. Some of them can read a large variety of fonts in regular pitch and proportional spacing, including those used in most common office correspondence and popular publisher fonts, for example OCRA, OCRB, Gothic, Courier, Elite, Pica, Orator, Narrator, Avant Garde, Helvetica, Times, Times Roman, Palatino, Bookman, Monaco, Geneva, and New Century Schoolbook. With the introduction of narrow range scanners, measuring 3 to 6 inches wide, columnar scanning is now available. With this, the optical reader can recognize multiple columns or sections of a page or mailing lists. Some are equipped with software for spell checking, and for flagging suspicious characters or words. Some can also read special symbols plus a variety of languages, including many European languages which contain vowels with different types of accents, and Oriental languages which contain thousands of characters written in different fonts.

It is interesting to note that several companies have produced low-cost software which enables users with scanners to apply it to read their documents. Common packages cost in the neighborhood of only $500. This has proved to be very convenient for a number of owners of optical scanners. Some of the optical readers are equipped with an automatic document feeder so that they can read many documents automatically. Apart from working with Personal or Apple/Macintosh computers, some optical readers can also be integrated to more advanced computers, such as PS/2, Sun workstations, and a variety of other host systems. Some have resolutions up to 600 DPI, and can read documents in different gray levels and colors. A number of them support SCSI, RS-232, and Ethernet for communication with other pieces of equipment.

One of the challenging areas of optical character recognition is the recognition of handwritten materials. However, in spite of many years of intensive research in this area [158], optical readers with such capability are rare. Even if they exist, they typically can read only hand-printed digits which must be legible, well-written, and consistently placed. Very few products can read alphanumeric handprints though some Japanese machines can read standardized alphanumerics and Katakanas hand-printed in preprinted boxes. In this respect, standardization of hand-printed characters is very important. On this topic see the paper by Suen *et al.* [159]. Recognition of characters written without any constraint is still quite remote. For the time being, recognition of handwritten characters, signatures, and cursive script seems to belong only to on-line products where typically writing tablets are used to extract real-time information and features to aid recognition. See, for example, the paper by Wakahara *et al.* in this issue. Although no complete OCR product exists, people are using OCR machines knowing that the OCR industry has become profitable recently. Now let us trace the history of OCR systems in terms of commercial products.

B. The Generations of OCR

Several commercial OCR's appeared in the beginning of the 1960's; these were outputs of the age of cut and try. The first generation can be characterized by the constrained letter shapes which the OCR's read. These simple methods were effective. The most typical one was the NCR 420 [160], for which a special font for numerals and five symbols, called NOF or bicodes, had been designed. The next typical one is the Farrington 3010 [161] of Farrington Electronics Inc. The story was the same as the NCR's OCR; i.e., they also used a special font, called Selfchek 12F,7B. IBM was also very active in developing OCR, recognizing that OCR was a very important input device for the computer. The first commercialized OCR was the IBM 1418 [162], which was designed to read a special IBM font, 407. An input character was fed to a 17 column by 10 row shift register, which made the machine resistant to vertical position variation. The letter shape looked more natural than the two fonts mentioned above. The recognition method was logical template matching; however, positional relationship was fully employed. The logic was quite complex in order to cope with document variations. The design stance of using computer technology, logic in particular, was inherited by subsequent IBM OCR systems, i.e., IBM 1428, IBM 1285, IBM 1287, and even the special-purpose OCR, the IBM 1975. In Japan, by the end of the 1960's, some mainframe companies announced their OCR's of the first generation. They included the Facom 6300A from Fujitsu and the H-852 from Hitachi, both of which used the stroke analysis method. On the other hand, the N240D-1 of NEC was somewhat similar to the IBM 1417 and also read the 407 font.

The second generation of OCR systems can be characterized by hand-printed character recognition capabilities. At the early stage they were restricted to numerals only. Such machines appeared in the middle of the 1960's and early 1970's. We introduce a few typical systems here. The first and a famous OCR system was the IBM 1287, which was exhibited at the World Fair in New York in 1965. The system was a hybrid one in terms of hardware configuration. It combined analog and digital technology. We have already mentioned it as the analog device of contour following.

The first automatic letter-sorting machine for postal code numbers was developed by Toshiba [163]. The three big companies in Japan participated in the development of the machine and Toshiba was the first winner, with the work done by Genchi, K. Mori, Watanabe, and Katsuragi. Their method is a typical structual analysis approach and is very famous. We recomend their paper, which is readily available. Two years later, NEC developed an automatic OCR sorter for postal code systems and participated in this market as well as Toshiba. The algorithm developed by NEC was based on stream following analysis, which was described above. Here we add some explanation in relation to Toshiba's system. In contrast to Toshiba's approach, an image was considerably reduced to a 6×10 matrix and size normalization was performed. Using consecutive run length encoding information, as the first step, each pixel was labeled row by row from bottom to top. A semilocal feature plane was constructed, with a crossing count pair of consecutive rows. These features were macroprocessed along each row and the resultant semifeatures were represented by 16 kinds of labels. In this sense, the scheme of the macroprocessing is similar to that of Toshiba's, but simpler. They continued the development toward better performance and the series of the machines reached NAS50 in 1975, which read numerals and alphabetical characters printed in type (multifont) as well as unconstrained numerals. For the printed characters they basically used template matching. For the numerals, they improved the algorithm taking a higher resolution of a 16×20 matrix.

The performance was reported as follows: At the early stage, they could not obtain satisfactory results. The correct sorting rate and the error sorting rate were about 60% and $5 \sim 6\%$ in 1968. However, they had improved machines to the level of $92 \sim 93\%$ correct sorting rate and $1 \sim 2\%$ error sorting rate by 1974. Here we would like to point out that the above reveals a very important element in developing OCR systems. Both a sympathetic manager and the user are key factors in developing this kind of technology. Both Toshiba and NEC people have been fortunate in this respect, aside from their intelligence and great efforts.

The second generation of OCR can also be characterized by the recognition capability of a set of regular machine-printed characters as well as hand-printed characters. In this respect, a very powerful OCR system, called RETINA [164], was developed by Recognition Equipment Inc. This system read both hand-printed and machine-printed characters. When hand-printed characters were considered, the character set was constrained to numerals, and the following letters and symbols: C, S, T, X, Z, +, and -. On the other hand, the system that could read printed characters could recognize 40 letters: a complete uppercase alphabet, numerals, and four special symbols. The user specified the font to be read by the machine. A key feature of RETINA, in terms of OCR systems, lies in its parallelism, which is associated with the name RETINA. The processing speed was very high at 2400 chs. Concerning the recognition of hand-printed characters, strokes are matched, and line crossing and corner types are also taken into account. Positional relations of detected features are examined to identify an input character. RETINA was used in practical applications in spite of its great expence.

Early in the 1970's, one big barrier to the use of OCR began to break. The first OCR which aimed at high performance and low cost was made by Hitachi [165], [166], called H 8959. As stated before, a flying spot scanner was very expensive at that time. The flying spot scanner was replaced by a laser scanner, and a special-purpose OCR processor was used instead of wired logic circuits, resulting in considerable cost reduction. This OCR was primarily for hand-printed numerals and some symbols, which was the same set for RETINA. The algorithm used belonged to thinning line analysis and actually the machine at first employed a regular thinning algorithm. The thinned line was represented by chain encoding using 3×3 masks. Singular points, such as end points and branching points, were also identified by counting matched mask(s). If the number of matched masks was one, two, three, or four, then the end point, normal line point, branching point (T type), and intersection point (X type) were identified respectively. Thus the thinned character line is decomposed into branching line segments whose terminal points are singular except for a simple loop in which the starting point was used. The coordinates of those singular points of the branching line segments were registered. Then each branch segment was further divided into four kinds of monotone line segments, called four-quadrant coding: upward right and vertically up; upward left and horizontally left; down right and vertically down; and down left and horizontally right [167]. This was a partial application of four kinds of monotone functions (not strict) as primitives, mentioned before (strict).

For a loosely constrained data set written with pencils by 60 Hitachi employees, with a sample size of 1200, the correct recognition rate and the error rate were 95.7% and 1.3% respectively. For a data set written by the same persons who had received training, a 100% recognition rate was obtained. These experiments revealed how much the recognition rates depend on the quality of the writer's group and that, in the case of a cooperative group, the recognition performance can be high without imposing severe constraints.

Hitachi improved the H-8959 by applying DP matching to branch matching. The result was the H-8957 [168], in which the branch was represented by chain encoding. The matching is nonlinear (DP matching) and so many-to-one and one-to-many are allowed. The algorithm used for DP matching is a typical and well-known method in speech recognition. The machine was primarily for FORTRAN reading, and so 47 characters of hand-printed alphanumerals and symbols were read, and context checking was taken. The recognition performance was reported: 0.06% error rate and 0.20% rejection rate were obtained with a sampling of 26 400 characters produced by six trained writers. This was so good that we can suppose that the quality of the data was exceptionally good, being close to constrained hand-printed characters.

Third-Generation OCR Systems

By the end of the 1960's, the next targets for document readers were for poor-print-quality characters, and hand-printed characters for a large category character set, such as Chinese characters. By document, we mean that it includes words such as names, addresses, and commands. These targets have been achieved partially and such commercial OCR systems appeared roughly during the decade from 1975 to 1985. Low cost and high performance are always common objectives for these systems. The dramatic advances of LSI technology were of great help to the engineers who were engaged in the development of OCR systems. Although this was common to all electric systems, in general LSI was especially important for pattern recognition systems. In the early days, the barrier had to do with the large number of components. This is being solved in general by high-speed CPU's, large available memories, and ROM's in particular, rather than by making highly parallel machines.

1) Print-Quality Proofreader To overcome the print quality problem, much work was done. We will pick two cases as representative examples. One is the ASPET/71 (TOSHIBA OCR-V100) and the other is the IBM 1975 [169]–[171] . They are in strong contrast to each other. First of all, the former is an analog type and the latter is a digital type. The name ASPET/71, derived from the analog spatial parallel system made by ETL and Toshiba, was given by Katsuragi, an excellent researcher at Toshiba and contributed significantly to the construction of the system, but died very soon after the project ended at very young age. The 71 denotes 1971, the year when the system was built completely. The algorithm is based on correlation, but a special one, called the multiple similarity method, which was mentioned in detail before. The ASPET/71 was built from analog circuits, but the actual commercial product, the OCR-V100, used digital technology fully. On the other hand, the algorithm of the IBM 1975 is based on logical matching and also was mentioned in detail before.

A problem faced by both parties of engineers was noise arising from the poor print quality of the characters. However, methods for handling the problem were very different. It was solved by theory in the former, but by human intuition in the latter. In correlation matching it is easy to make masks/dictionary by taking the average of the data belonging to the same class, which is the strong point of template matching in general. But in the case of logical matching, the average has no meaning, because it is done in linear space. So considerable effort was devoted to automatic mask making at IBM, as mentioned before. However, there was a big difference between laboratory data and field data. In fact the IBM 1975 was developed specifically to meet the needs of the U.S. Social Security Administration, which involved reports printed by more than 256 typewriter fonts with an uncontrolled and wide range of print quality. As a result, the engineers who developed the IBM 1975 gave up using the automatic mask making theory after many unsuccessful trials and relied on intuition. They wrote: "the designer imagined as generalizations of the few specific examples which he had on hand." In some sense, we may say that their approach was structural analyzed logical matching. They actually listed the features which they noticed as line segments (short, long, horizontal, vertical, slanted, etc.), line endings, various corner curvatures, gap in lines, relative positions of line segments, and so on.

ASPET/71 was planned as one product of the Ultrahigh Speed Computer System Development Project, promoted by the Japanese Government, and was achieved by cooperation between ETL and Toshiba. The great success of the project led Toshiba to develop a commercial OCR system based on the multiple similarity method, which has already been described.

Concerning the IBM 1975, besides the recognition system designed by human intuition, considerable effort had been paid to preprocessing. Binarization was crucial, particularly in logical matching, so an adaptive video thresholding system was used as a linear function of V. V was defined as the average of all video samples greater than some low threshold value within a predetermined area. Width normalization was also done, where the stroke width measurement was fed back for threshold adjustment. Postprocessing was also implemented to create a file name. (The data consisted of names and associated digits: social security number, FICA-taxable wages.)

In the recognition system, features were first measured, each of which was detected by a single-output Boolean function of about 30 inputs. A total of 96 feature detection circuits were used and so a 96-position binary feature vector was constructed for each input character. The decision process was done by comparing the feature vector of the input character with a set of reference vectors/masks and determining which reference most closely matches the input feature vector. The reference structure was ternary, including the case "don't care." Each reference was stored in ROM, where 192 (2×96) bits were used to store the main portion of the reference and 48 additional bits were used to store the reference control information. Approximately 2000 references were stored and 150 to 200 of them were processed for each input character. Remember that 256 fonts were used! The control information used a very typical technique of computer applications. A two-level decision process was specified in which a few character classes are first chosen by means of a selected group of reference vectors, and then a large number of reference vectors are used to decide among these classes.

2) hand-printed Document Reader: In September 1975, the Labor Market Center of the Labor Ministry of Japan formally decided to use Katakana OCR in its total employee insurance system in 1980. This contract attracted the competition of almost all the big electronics and electrical engineering companies. The R&D for Katakana OCR became very active and a related story can be found in the paper by Suen *et al.* [5]. That paper was written in 1979, and so a final contractor had not yet been chosen. The story will be continued here. The winner was not just any company, it was NTT. At that time, NTT was a public corporation and a telephone and telegram monopoly. The system planned by the Labor Market Center was very much involved with communication systems. NTT therefore developed OCR systems very rapidly and succeeded in developing a high-performance Katakana OCR system in time. As a result, the NTT OCR was adopted and its manufacturing was committed to NEC and OKI.

The recognition system of the NTT Katakana OCR [172] was based on background analysis, as described before. The basic idea was that relations between semiglobal features should be represented in the same form as the semiglobal features. In other words, the global features should be the natural results of macroprocessing of features. Returning to Glucksman's method, he used the ternary coding, but the system mixed semiglobal features and global features. NTT therefore first took the binary coding. According to the value of the center pixel, i.e., white or black, the codes were classified into 16 code sets or four code sets, respectively, called white point first-order features or black point first-order features, respectively. The former can be inferred from the Fig. 14, where concavities and loops are schematically represented. The latter codes were symbolized as P_V, P_H, P_S, and P_I, where V, H, S, and I denote vertical, horizontal, slant, and internal. Some representations of the global features, called secondary features, are shown also in Fig. 14 . We note here that the total coding is reduced from 16+4 to 16 in order to facilitate and economize the hardware implementation. The final stage is to construct masks using the secondary features. They made them manually, but the work was easy because the features were globally macroprocessed. However, their scheme was not enough to separate some closed categories, where the difference lies in only the geometrical feature. Contour following was used to detect an exact geometrical feature using black first-order features, which are used as auxiliary features. Concerning hardware implementation, see [173].

A large-scale simulation experiment was conducted in which a data set was written by 2000 students and clerks who were instructed to write characters within boxes keeping specified writing rules of the Japanese handprint standard. The total number of characters collected was 188 000 ((4000/category) and the number of categories was 47 (Katakana set except one letter). Half of the data set was used as learning data and the rest were used in the test. The recognition results were reported with a correct recognition rate and an error rate of 97.1% and 1.0% respectively. The number of masks used was 234 (5/category). Currently the machine, called the DT-OCR100C, reads 65 kinds of letters, numerals, Katakana, and symbols. The total number of document types handled is greater than 100.

3) Software Package as an OCR: Thanks to highly developed computer technology, very recently the recognition part of OCR has become fully implemented in software packages, which work on personal computers. Present PC's based on the 386, for example, are comparable to the large-scale computers of the early days. Of course, there are some limitations in such OCR software, in particular in its reading speed and the kinds of character sets that can be read. However, regarding printed character sets used in the West, some commercial products have appeared which have impressive performance. Here, we introduce some of them, but unfortunately no technical information is available and so the following description is based on catalog data.

The Caere Corporation sells OmniPage and Typist. Typist includes a hand-held scanner which is used with typical PC's. The PC computers use the 286 or 386 microprocessors with AT or PS/2 bus, with 640K base, and have 2 MB of expanded memory and hard disk. OmniPage works in almost the same system environment. Both OmniPage and Typist can read 11 European character sets, all nonstylized fonts including italics and boldface from 6 to 72 point sizes, and dot-matrix alphanumerics. OmniPage can read characters with an average speed of 40 cps and also underlined characters. It provides a proofing tool which can display a small bit map image of the word or character in question, adjacent to the questionable occurrence when a user jumps to the uncertain character. Some page segmentation capability is provided such as automatically differentiating columns, graphics, and text. The price of Typist is only $595 for a PC with AT bus and PS/2 bus as of 1991. This is certainly a surprising price considering its high performance.

One competitor to Omnipage is the Discover series of Kurzweil (Xerox Image System, Inc.). It reads printed characters in almost the same system environment, with $10 \sim 40$ cps. They claim almost the same performance/specifications as Omnipage. However, character size is limited to $8 \sim 24$ points and it imposes a specification on character pitch as a minimum of one dot intercharacter spacing after scanning. Regarding page segmentation, its performance is not clear. Another competiter is WordScan, produced by the Calera Corporation. Some technical infomation relating to this OCR system is contained elsewhere in this issue (refer to Bokser's paper). There are other OCR software products for reading printed characters, but for European languages, the above three seem to be major players in this market so far. Regarding other products and detailed information for OCR users we recommend [174].

4) Kanji Recognition Machine: The dominant characteristic of Kanji is its large number of categories ($2000 \sim 4000$). Kanji was transferred from China to Japan as a subset of Chinese characters ($6000 \sim 50000$)). Because of space limitations, two kinds of specific products will be mentioned. For a general and technical description, the reader can refer to [176].

Commercial products: Toshiba is a pioneer in Kanji OCR, which announced a Kanji OCR, the OCR-V595, in 1983, based on multiple similarity method, which was described in detail before from a historical point of view. Now Toshiba's ExpressReader 70J is popular as a reader for Japanese printed characters. It has omnifont reading and page segmentation capability. For the latter, see the paper by Tsujimoto *et al.* in this issue. Concerning Kanji it reads 4000 kinds of characters, which is virtually enough to deal with almost all kinds of characters. The character size is $6 \sim 40$ point and reading speed is $70 \sim 100$ chs, which is the highest among Kanji OCR commercial products. Its correct recognition rate is 99.5% according to the catalog.

Next we introduce two commercial products of Kanji OCR whose origin is based on the feature matching and field effect method, which was explained before. However, they are very different in terms of implementation and way of utilization. The first, the XP-70S, made by the Fuji Electric Company, is based on Yasuda's algorithm. The algorithm is based on feature matching of two kinds of feature fields: stroke directional field and edge propagation field [175]. For generation of these feature fields, Fuji developed special LSI [177], which consists of CMOS at a clock speed of 5 MHz. This gives a speed of about 115 chs. This was developed for document processing which has been printed or typed. The kinds of character sets read are rich: Roman alphabet (52 letters), Hirakana (82), Katakana (83), numerals (10), Greek letters (19), Kanji (2965), and special symbols (152). The machine is a multifont reader. Reading speed is 30 chs. The user can have his own special letters by invoking its learning mode. The machine can display a maximum of ten candidate characters, and these can be corrected manually. The price is 4.5 million yen at the time of writing. Postprocessing is also available.

The second product is the CLL-2000, made by Sanyo Electric Co. Ltd. [178]. This is a desktop hand-printed character reader which is used together with a word processor. It is small in size, and low in price (2 million yen at the time of writing). The architecture of the machine is very interesting. It consists of 21 16-bit microprocessors (8086) and so it is a multi-CPU system, which made the low cost possible. The algorithm implemented is Oka's and was explained before. It has a cellular architecture; i.e., each pixel has a cell which has eight intracells. A character is written inside a box 1 cm $\times$ 1 cm, which is sampled to a 96×96 matrix. After size normalization, it is reduced to a 60×60 matrix, on which local edge detection is done and the matrix is further reduced to a 30×30 matrix. Therefore, the feature field needs $(30\times30\times8) = 7200$ bytes. The problem is the large number of categories, about 2400, and so the dictionary needs $7\text{K}\times2400 = 16.8$ Mbytes, which is too big in aiming at low cost. So drastic dimension reduction was done by reducing the 30×30 matrix to 7×7. Thus the size of the dictionary was reduced to about 1 Mbytes. We note that in spite of such drastic reduction of the dimension, the recognition performance does not drop too much. The CLL-2000 reads hand-printed Kanji, Roman alphabet, Hiragana, Katakana, numerals, and symbols: a total of 2377 letters. The reading speed is 2 chs. The correct recognition rate is 93%, but the machine can list ten candidates with a correct recognition rate of 99.6%. However, the data set was collected by the company. When a manuscript is read, the processing speed is twice as fast as humans, including the time to make corrections.

VI. Learning

Learning is a basic function of pattern recognition in animals in general and in human beings in particular. Therefore, considerable research effort has been spent and neural networks have recently attracted a lot of attention. Everyone is aware of the situation, and is interested in the near-term contributions of neural networks. However, such predictions need serious consideration and lengthy study, which is beyond the scope of this paper. In this respect we recommend the tutorial paper by Amari [179], who is known as the inventor of back-propagation and a distinguished theoretician. We can in a restricted way learn from the perspective of automatic design. In this respect, we discuss an expert system application used to design OCR. Concerning neural network applications to OCR, we will make some comments at the end of this section from the engineering point of view.

A. Automatic Design

The review paper on character recognition written by Govindan *et al.* in 1990 says that "No attempts are known to the authors in the topic of automated designs dealing with the design of recognizers suitable for structurally different character sets." Certainly research on automatic design based on a structural analysis approach is rare, in spite of its crucial importance. Naturally, however, total automatic design, which includes preprocessing, feature extraction, structure description, and mask making, is not known to the authors, and we think such total automatic design is impossible. This would insinuate an automatic creator. We are not God! The so-called connectionist seems to aim at such a learning machine. Anyway we treat this topic in a restricted way and so we can introduce very successful research which was done mainly by Yamamoto [180], [181]. We restrict automatic design to automatic mask making, which is a very tedious and laborious process, especially when dealing with unconstrained hand-printed characters. In this sense, we also need to have an expert system.

At ETL Yamamoto *et al.* became aware that topological features such as continuity, hole, and quasi-topological features are very effective in classifying characters. According to these features, a character set can be divided into a set of subclasses, each of which is denoted as ω_{jk}, where j and k mean the jth kind of letter and kth subclass in the j class respectively. Automatic mask making is divided into the following two steps:

1) taking correspondence between the input line segments and the mask's line segments;
2) taking statistics on each feature axis in the correspondence.

Step 1 is crucial. Here we assume contour following and so as an example we start it from the top left. Then because of shape variations, we cannot always obtain a stable order of line segments . However, fortunately, after the classification mentioned above is done, within each subclass the order of the line segments is very stable. Therefore, we can do step 1 easily. Since step 2 is well known, we do not mention it here. Experiments on Katakana character sets have revealed its usefulness; the average number of learned masks is 3.7 per class and naturally a 100% correct recognition rate was obtained for a 10 000 training data set. A key to automatic mask making is the good qualitative description of the shape being used. In this respect, singular points and a quasi-topological description [151] are good candidates for this purpose and in fact it is shown that automatic mask making is done effectively and strictly in both practical and mathematical senses [182]. On the other hand, another approach was taken by Baird [183], who used a Bayesian classifier by mapping structural features to numerical vectors. Usual statistical learning is done in the feature space.

B. Expert System

For sloppy data, we have to consider the construction of some kind of expert system. It is said that one human expert is assigned per famous novelist, whose writing material cannot be understood by ordinary people, without elaboration by the expert. Thus, for totally unconstrained characters some expert systems have been developed [184], [185]. Among them, Suen *et al.* recently introduced a multiple-expert system [186]. Very encouraging results have been reported. The key lies in an intelligent way of combining the expertise of these experts to reinforce their strengths and to suppress their weaknesses. More details can be found in the paper by Suen *et al.* in this issue. See also Srihari's paper in this issue.

C. Neural Network

Many different methods have been explored during the past four decades by a large number of scientists to recognize characters. A variety of approaches have been proposed and tested by many researchers in the field, as stated. Recently, the use of neural networks to recognize characters and different types of patterns has resurfaced. In Krzyzak *et al.* [187] and Le Cun *et al.* [188], back-propagation networks composed of several layers of interconnected elements are used. Each element resembles a local linear classifier which computes a weighted sum of its input and transforms it into an output by a nonlinear function. The weights at each connection are modified until a desired output is obtained. In a sense, it acts like a black box which makes use of the statistical properties of the input patterns and modifies its decision functions to produce a desired output. Owing to the lack of shape features, its output is sometimes unpredictable; hence apart from modifying the back-propagation model, some geometrical features were added to the network to enhance the recognition rate. Thus, from our experiences and informal information, when we compare a neural network OCR with a conventional OCR at the top level in this field, we cannot find any evidence that a neural network is superior to a conventional one. Some, however, advocate its superiority. One point dampening high expectations for the neural net is its poor capability for generality.

However, neural networks have been studied by researchers for several decades and some new features have been found and applied [189]. This is because each method has its own advantages and disadvantages, and it is better for us to have more information at our disposal than miss some new advancement. Recent research results call for the use of multiple features and intelligent ways of combining these various methods [190], [191]. The use of complementary algorithms to reinforce each other has also been suggested by Nadal *et al.* [192].

Here we quote Amari: "It is desirable to apply a neural network to broad area where it is effective..... However, backpropagation is so simple in terms of theory that it has its own limitation which can not be deep. It is important that we should not impose too much expectation on backpropagation. I am sure a higher level theory will appear sooner or later, because of the research being done by many researchers.... I can not now see any perspective on the new neural net theory" [179].

How does the human brain recognize different characters? Which parts of the characters is it looking for? What kinds of features do we get when we see a character? Is there some way we can extract character recognition knowledge from humans, who have become such superb character recognizers? These are some of the questions which remain unanswered. Obviously a lot more research should be done in this field before we can make computers read documents reliably and intelligently.

VII. Conclusions

We have described the stream of research and development of OCR systems, which has two large tributaries, template matching and structure analysis. We showed that both tributaries have grown to a concrete OCR technology and are merging to become a wide stream which constitutes an essential part of pattern recognition technology. On the other hand, we pursued the development of commercial OCR's and classified them into three generations. We also described some representative OCR's in some detail to show the historical development of OCR from both academic and industrial points of view.

We described many methods, some of which are somewhat related to each other and some of which are more or less independent. The important point is that we should make these methods more precise in the sense of an exact science, not a mere accumulation of empirical knowledge. The above statement is very much related to a shape model in which we need to establish an exact and flexible mathematical model of shape including a noise model which can be independent of the shape, but is intrinsically related to a given shape. The above two statements might have the relation of "chicken and egg," but it is necessary to attack both of these problems at the same time in order to establish a shape recognition technology on scientific ground. Based on these efforts, we will be able to make a machine which will approach human performance on shape recognition.

It is clear and real that, in order to make such a machine, some combination of the methods mentioned so far should be brought together. Multiple approaches and complementary algorithms will be integrated. Performance of each method and models of characters will be known by the reading machine. On the other hand, a new method is still naturally expected and because of the rapid development of modern computer technology, we can expand our creative space to a more sophisticated method, such as a morphological approach, which incorporates human expertise and the direct use of multilevel images.

In practice, it is very important that a machine gain the confidence of its users. If a serious user writes a character well, then a machine has to read it with 100% accuracy with low-cost machine/software. This means that we should specify the performance of the OCR. So far, the specification has been very loose. It sometimes does not give any confidence to a user. However, this is not an easy matter and in fact, it is very much related to the problems mentioned above. In practice, it is also related to standardization of character shapes in hand-printing. People are not so sensitive to certain confusing shapes such as "O" and "D," and so engineers should point out such issues so that users can understand the real situation. Naturally, such a standard will be less restricted, but more precise in the sense that it points out "key features" to be appreciated naturally by humans. In this connection it is very useful to have the cooperation of elementary school teachers to help children to form good habits in writing characters properly and legibly, and pay special attention to their distinctive features.

The R&D of OCR is also moving toward "word recognition," using contextual knowledge such as addresses and names. In fact such development, of postal address and name reading machines, is already such a trend. This necessarily leads the R&D of OCR to document analysis, in which characters constitute one component. Thus the R&D of OCR will expand its applications to a total document reader, posing the greatest challenge to researchers in this field.

The history tells us that OCR technology has been built by many researchers over a long period of time, consisting implicitly of something like a worldwide human research network. In such an invisible forum, people have made efforts, with "competition and cooperation," to advance the research effort. In this sense, international conferences and workshops are being organized to stimulate the growth in the area. For example the International Workshop on Frontiers in Handwriting Recognition and the International Conference on Document Analysis and Recognition will play a key role in the scholarly and practical arena.

ACKNOWLEDGMENT

The authors would like to express their thank to Prof. T. Pavlidis for encouraging them to write this extensive survey paper, to the members of the Japanese Document Standardization Committee for their important contributions, and to Dr. P. Hart and J. Cullen of the Ricoh Research Center in the U.S. for their valuable comments. They would also like to thank their colleagues, staff, and students who have contributed significantly to these research efforts during the past two decades.

REFERENCES

[1] J. R. Ullman, *Pattern Recognition Techniques.* London: Butterworths, 1973.
[2] T. Sakai and M. Nagao, *Characters and Figures Recognition Machine.* Tokyo: Kyoritsu, 1967.
[3] T. Pavlidis, *Structural Pattern Recognition.* New York: Springer, 1977.
[4] S. Mori and T. Sakakura, *Fundamentals of Image Recognition vol. I, vol. II*, vols. I and II. Tokyo: Ohm, 1986 and 1990.
[5] C. Y. Suen, M. Berthod, and S. Mori, "Automatic recognition of hand-printed characters—The state of the art," *Proc. IEEE*, vol. 68, pp.469–487, Apr. 1980.
[6] J. Schurmann, " Reading machines," in *Proc.6th IJCPR*, 1982, pp.1031–1044.
[7] V. K. Govindan and A. P. Shivaprasad, "Character recognition - A review," *Pattern Recognition*, vol. 23, no.7, pp. 671–683, 1990.
[8] G. Tauschek, "Reading machine," U.S. Patent 2 026 329, Dec. 1935.
[9] P. W. Handel, "Statistical machine," U.S. Patent 1 915 993, June 1933.
[10] M. H. Glauberman, "Character Recognition for business machines," *Electronics*, pp. 132–136, Feb. 1956.
[11] W. J. Hannan, "R. C. A. multifont reading machine," in *Optical Character Recognition.* G. L. Ficher *et al.*, Eds. McGregor & Wemer, 1962, pp.3–14.
[12] ERA, "An electronic reading automaton," *Electronic Eng.*, pp. 189–190, Apr. 1957.
[13] T. Iijima, Y. Okumura, and K. Kuwabara, "New process of character recognition using sieving method," *Information and Control Research*, vol. 1, no. 1, pp. 30–35, 1963.
[14] L. P. Horwitz and G. L. Shelton, "Pattern recognition using autocorrelation," *Proc. IRE*, vol. 49, no. 1, pp. 175–185, 1961.
[15] M. Sato, K. Yoneyama, and Y. Ogata, "Recognition of printed characters," *T. R. Radio Wave Research Laboratory*, vol. 7, no. 33, pp. 489–492, Nov. 1961.
[16] W. S. Rohland, "Character sensing system," U.S. Patent 2 877 951, Mar. 1959.
[17] R. W. Weeks, "Rotating raster character recognition system," *AIEE Trans.*, vol. 80, pt. I, *Communications and Electronics*, pp. 353–359, Sept. 1961.
[18] R. O. Duda and P. E. Hart, *Pattern Recognition and Scene Analysis.* New York: Wiley, 1973.
[19] W. H. Highleyman, "Linear decision functions with applications to pattern recognition," *Proc. IRE*, vol. 50, pp. 1501–1514, June 1962.
[20] G. Nagy, "Optical character recognition-Theory and practice," in *Handbook of Statistics*, P. R. Krishnaiah and L. N. Kanal, Eds., vol.2. Amsterdam: North-Holland, 1982, pp. 621–649.
[21] R. B. Johnson, "Indicia controlled record perforating machine," U.S. Patent 2 741 312, Apr. 1956.
[22] T. L. Dimond, "Devices for reading handwritten characters," in *Proc. Eastern Joint Computer Conf.*, 1968, pp.207–213.
[23] L. A. Kamentsky, "The simulation of three machines which read rows of handwritten Arabic numerals," *IRE Trans. Electron. Comput.*, vol. EC-10, pp. 489–501, Sept. 1961.
[24] B. A. Glucksman, "Classification of mixed-font alphabetics by characteristic loci," in *Dig. 1st Ann. IEEE Comput. Conf.*, Sept. 1967, pp.138–141.
[25] W. Doyle, "Recognition of sloppy hand-printed characters," in *Proc. Western Joint Comput. Conf.*, 1960, pp.133–142.
[26] T. Sakai, M. Nagao, and Y. Shinmi, "A character recognition system," in *Proc. Annual Conf. IECE Japan*, 1963, p. ,450.
[27] NEC (Nippon Electric Company), "Improvements in or relating to character recognition apparatus," U.K. Patent 1 124 130, Aug. 1968.
[28] J. H. Munson, "Experiments in the recognition of hand-printed text: Part I, Character recognition," in *Proc. 1968 Fall Joint Comput. Conf., AIIPS Conf.*, vol. 33, 1968, pp. 1125–1138.
[29] E. C. Greanias, P. E. Meagher, R. J. Norman, and P. Essinger, "The recognition of handwritten numerals by contour analysis," *IBM J. Res. Develop.*, vol. 7, pp. 2–13, 1963.
[30] L. A. Kamentsky and C. N. Liu, "Computer-automated design of multifont print recognition logic," *IBM J. Res. Develop.*, vol. 7 , no. 2, pp. 2–13, 1963.
[31] L. A. Kamentsky and C. N. Liu, "A theoretical and experimental study of a model for pattern recognition," in *Computer and Inf. Science*, J. T. Tou and R. H. Wilcox, Eds. New York: Spartan Books, 1964.
[32] T. Iijima, "Basic theory on normalization of pattern," *Bulletin of Electrotechnical Laboratory*, vol. 26, no. 5, pp. 368–388, 1962.
[33] T. Iijima, "Basic theory of feature extraction for visual pattern," *T. R. on Automata and Automatic Control, IECE Japan*, July 1963.
[34] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. Lond. B*, vol. 207, pp. 187–217, 1980.
[35] M. Awaya and N. Funakubo, "Blur recovering circuit in observed images," in *Proc. Annual Conf. IECE Japan*, Apr. 1970, p. 2816.
[36] I. Yamazaki and T. Iijima, "Sampling of a character image," *J. of Trans. of Electron. Commun. Eng. Japan*, vol. 51-C, no. 9, pp. 428–429, Sept. 1968.

[37] T. Iijima, S. Mori, I. Yamazaki, and M. Yoshimura, "Automatic normalization circuit—A proposal of globally reciprocally dependent network," *T. R. on Automata and Automatic Control, IECE Japan*, Oct. 1963.
[38] I. Yamazaki, "Research on observation and normalization in character recognition," *Researches of the Electrotechnical Laboratory*, no. 726, Apr. 1972.
[39] S. Mori, "Theory of linear spatial network," *T. R. on Automata and Automatic Control, IECE Japan*, Oct. 1963.
[40] S. Mori, Internal Report, Electrotechnical Laboratory, June 1966.
[41] H. C. Andres and B. R. Hunt, *Digital Image Restoration*. Englewood Cliffs, NJ: Prentice-Hall, 1977, appendix B.
[42] N. Funakubo, "Studies on spatial network in printed character recognition system," *Researches of Electrotechnical Laboratory*, no. 778, Dec. 1977.
[43] A. Igarashi, E. Ogawa, and T. Iijima, "Topological analytic foundation of linear spatial network," *Trans. of IECE Japan*, vol. 53-C, no. 6, pp. 393–401, June 1970.
[44] Y. Yasumoto and C. Medioni, "Corner detection and curve representation using cubic B-splines," *Trans. of IECE Japan*, vol. J70-D, no. 12, pp. 2517–2574, Dec. 1957.
[45] S. Amari, "Invariant structures of signal and feature space in pattern recognition problems," *RAAG MEMORS* vol.4, pp.19–32, 1968.
[46] S. Watanabe, "Karhunen-Loeve expansion and factor analysis-theoretical remarks and applications," in *Trans. 4th Prague Conf. Info. Theory, Statis. Decision Funct., Random Process*, 1965, p.635.
[47] H. Karhunen, "On linear methods in probability," 1947 (English translation by I. Selin), The Rand Corporation, Dec. T-131, Aug. 11, 1960.
[48] M. Loeve, *Fonction Aleatories de Seconde Ordre*. Paris: Herman, 1948.
[49] K. Fukunaga, *Introduction to Statistical Pattern Recognition* New York: Academic Press, 1972.
[50] H. Hoteling, "Analysis of a complex of statistical variables into principal components," *J. Edu. Psychol.*, vol. 24, pp. 417–441 and 498–520, 1933.
[51] R. C. Gonzalez and P. Wintz, *Digital Image Processing*, 2nd ed. Reading, MA: Addison-Wesley, 1987.
[52] Y. Noguchi and T. Iijima, "Pattern classfication system using equi-variance characteristic papameters," *Trans. Info . Pross. Soc. Japan*, vol. 11, pp. 107–116, 1971.
[53] N. Funakubo, A. Iwamatsu, T. Suzuki, S. Mori, and T. Iijima, "ETL OCR pilot model," *Bulletin of the Electrotechnical Laboratory*, vol. 34, no. 1, Jan. 1970.
[54] T. Iijima, "Character recognition theory based on multiple similarity method," in *Proc. Annual Conf. IECE Japan*, Apr. 1970, p. 2710.
[55] T. Iijima, *Theory of Pattern Recognition*. Morishita Publishing, 1989.
[56] S. Watanabe and N. Pakvasa, "Subspace method of pattern recognition," in *Proc.1st Int. J. Conf. Pattern Recognition*, Nov.1973, pp. 25–32.
[57] E. Oja, *Subspace Methods of Pattern Recognition*. Research Studies Press, 1983.
[58] W. K. Pratt, *Digital Image Processing*. New York: Wiley-Interscience, 1978.
[59] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Trans. Inform. Theory*, vol. IT-8, pp.179–187, Feb. 1962.
[60] F. L. Alt, "Digital pattern recognition by moments," in *Optical Character Recognition*, G. L. Fischer *et al.*, Eds. Washington, DC: McGreger & Werner, 1962, pp. 159–179.
[61] T. E. Southard, "Method of recognizing single strings of non-touching rotated characters," Master's thesis, Ohio State University, Report AFOSR-TR-75–0177, Aug. 1974.
[62] G. L. Cash and M. Hatamian, "Optical character recognition by the method of moments," *CVGIP*, vol. 39, pp. 291–310, 1987.
[63] R. L. Cosgriff, "Identification of shape," Rep. 820–11, ASTIA AD 254 792, Ohio State Univ. Res. Foundation, Columbus, Dec. 1960.
[64] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Trans. Comput.*, vol. C-21, pp. 269–281, Mar. 1972.
[65] G. H. Granlund, "Fourier preprocessing for hand printed character recognition," *IEEE Trans. Comput.*, vol. C-21, pp. 195–201, Feb. 1972.
[66] E. Persoon and K. S. Fu, "Shape discrimination using Fourier descriptors," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 170–179, Mar. 1977.
[67] M. T. Y. Lai and C. Y. Suen, "Automatic recognition of characters by Fourier descriptors and boundary encoding," *Pattern Recognition*, vol. 14, nos. 1–6, pp. 383–393,1981.
[68] Y. Hsu and H. H. Arsebault and G. Apr., "Rotation-invariant digital pattern recognition using circular harmonic expansion," *Appl. Opt.*, vol. 21, no. 22, pp. 4012–4015, Nov. 1982.
[69] R. J. Spinrad, "Machine recognition of hand printing," *J. Info. and Cont.*, vol. 8, pp. 124–142, 1965.
[70] S. Mori, *Fundamentals of Characters and Figures Recognition Techniques*. Tokyo: Ohm, 1983.
[71] M. Yasuda, "Research on character recognition systems based on correlation," Doctoral thesis, Tokyo University, 1981.
[72] Y. Yamashita, K. Higuchi, Y. Yamada, and Y. Haga, "Classification of hand-printed Kanji characters by structural segment matching method," *Tech. Rep. IECE Japan*, PRL82–12, no. 12, p. 25, 1982.
[73] H. Yamada, T. Saito, and K. Yamamoto, "Line density equalization—A nonlinear normalization for correlation method," *Trans. IECE Japan*, vol. J67D, no. 11, pp. 1379–1388, 1984.
[74] J. Tsukumo and H. Tanaka, "Classification of hand-printed Chinese characters using nonlinear normalization and correlation methods," in *Proc. 9th IJCPR*, Nov. 14–17, 1988, pp.168–171.
[75] H. Sakoe, "hand-printed character recognition based on rubber string matching," *T. R. IECI Japan*, PRL74–20, pp. 1–10, 1974.
[76] R. Bellman, *Dynamic Programming*. Princeton University Press, 1957.
[77] R. Bellman and S. Dreyfus, *Applied Dynamic Programming*. Princeton University Press, 1962.
[78] V. A. Kovalevsky, "An optimal algorithm for the recognition of some sequences," *Cybernetics*, vol. 3, no. 4, 1967.
[79] H. Sakoe, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, pp. 43–49, Feb. 1978.
[80] H. Yamada, "Contour DP matching method and its applications to hand-printed Chinese character recognition," in *Proc.7th IJCPR*, 1984, pp. 389–392.
[81] J. R. Ullman, "An algorithm for subgraph isomorphism," *J. Ass. Comp. Mach.*, vol. 23, no. 1, pp. 31–42, Jan. 1976.
[82] A. P. Ambler, H. G. Barrow, C. M. Brown, P. M. Burstall, and R. J. Popplestone, "A versatile computer-controlled assembly system," *Artificial Intelligence*, vol. 6, pp. 129–156, 1975.
[83] A. Rosenfeld, R. Hummel, and S. Zucker, "Scene labeling by relaxation oprations," *IEEE Trans. Syst., Man., Cybern.*, vol. SMC-6, pp. 420–433, 1976.
[84] L. Davis, "Shape matching using relaxation techniques," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, pp. 60–72, Jan. 1979.
[85] D. I. Waltz, "Generating semantic descriptions from drawings of scenes with shadows," Ph.D. dissertation, AI Lab, MIT, 1972.
[86] H. Ballard and C. M. Brown, *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
[87] A. Rosenfeld and J. L. Pfalz, "Sequential operations in digital picture processing," *J. Ass. Comp. Mach.*, vol. 13, pp. 471–494, 1966.
[88] H. Sherman, "A quasitopological method for the recognition of line patterns," in *Info. Process., Proc. UNESCO Conf.* (Paris), 1959.
[89] M. Beun, "A flexible method for automatic reading of handwritten numerals," *Philips Tech. Rev.*, vol. 33, no. 4, Part I, pp. 89–101; Part II, pp. 130–137, 1973.
[90] C. Y. Suen, R. Shinghal, and C. C. Kwan, "Dispersion factor: A quantitative measurement of the quality of hand-printed characters," in *Proc. Int. Conf. Cybern. and Society*, Sept. 1977, pp.683–685.
[91] R. A. Kirsch, L. Cahn, C. Ray, and G. L. Urban, "Experiments in processing pictorial information with a digital computer," in *Proc. Eastern Computer Conf.*, 1957, pp. 221–229.
[92] R. H. McComick, "The Illinois pattern recognition computer—Illiac III," *IRE Trans. Electron. Comput.*, vol. EC-12, no.5, 1963.
[93] R. Narasimhan, "Labeling schemata and syntactic descriptions of pictures," *Information and Control*, vol. 7, pp. 151–166, 1964.
[94] D. Rutovitz, "Data structures for operations on digital images," in *Digital Processing*, C. G. Chang *et al.*, Eds. Washington, DC: Thompson, 1968, pp. 105–133.

[95] E. S. Deutsch, "Computer simulation of a character recognition machine," *Post Office Elec. Eng. J.*, vol. 60, pp. 39–44 and pp. 104–109, 1967.
[96] C. J. Hilditch, "Linear skeleton from square cupboards," in *Machine Intelligence IV*, B. Meltzer and D. Michie, Eds. Edinburgh, University Press, 1969, pp. 403–420.
[97] H. Tamura, "A comparison of line thinning algorithms from digital computer view point," in *Proc. 4th Int. Joint Conf. Patt. Recog.*, 1978, pp. 715–719.
[98] H. Freeman, "Boundary encoding and processing," in *Picture Processing and Psycholopictorics*, B. S. Lipkin and A. Rosenfeld, Eds. New York: Academic Press, 1970, pp. 241–266.
[99] R. G. Grimsdale, F. H. Sumner, C. J. Tunis, and T. Kirburn, "A system for the automatic recognition of patterns," *Proc. Inst. Elec. Eng.*, vol. 106B, pp. 210–221, Dec. 1958.
[100] T. Pavlidis, "Computer recognition of figures through decomposition," *J. Info. Control.* vol. 12, pp. 526–537, 1968.
[101] T. Pavlidis, "Structural pattern recognition:Primitives and juxtaposition relations," in *Frontiers of Pattern Recognition*, S. Watanabe, Ed. New York: Academic Press, 1972, pp. 421–451.
[102] J. Rabinow, "Developments in character recognition machines at Rabinow Engineering Company," in*Optical Character Recognition*, Eds. G. L. Fischer *et al.*, Eds. McGreder & Werner, 1962, pp. 27–51.
[103] L. H. Sublettle and J. Tults, "Character recognition by digital feature detection," *RCA Review*, pp. 60–79, Mar. 1962.
[104] P. G. Perotto, "A new method for automatic character recognition," *IEEE Trans. Electron. Comput.*, vol. EC-12, pp. 521–526, Oct. 1963.
[105] M. Nadler, "Structual codes for omnifont and handwritten characters," in *Proc. 3rd IJCPR*, 1976, pp.135–139.
[106] Uyehara, "A sream-following technique for use in character recognition," in *1968 IEEE Int. Conv. Record*, part 4, Mar. 1963, pp. 64–74.
[107] M. Nadler, "Sequentially-local picture operaters," in *Proc. 2nd IJCPR*, 1974, pp. 131–135.
[108] S. Hoshino and M. Miyamoto, "Feature extraction on hand-printed numerals," in *Proc. Ann. Conf. IECE Japan*, 1969, p. 92.
[109] K. Kiji *et al.*, "Recognition of the postal code number," NEC Tech. Rep., pp. 30–38, 1969.
[110] C. B. Shelman, "The application of list processing techniques to picture processing," *Pattern Recognition*, vol. 4, pp. 201–210, 1972.
[111] R. D. Merril, "Representation of contours and regions for efficient computer search," *Commun. Ass. Comput. Mach.*, vol. 16, pp. 69–82, 1973.
[112] A. K. Agrawala and A. V. Kulkarni, "A sequential approach to the extraction of shape features," in *Proc. CGIP*, 1977, pp. 538–557.
[113] T. Pavlidis, "A vectorizer and feature extractor for document recognition," *CVGIP*, vol. 35, pp. 111–127, 1986.
[114] T. Pavlidis, *Algorithms for Graphics and Image Processing*. Rockville, MD: Computer Science Press, 1982.
[115] S. Kahan, T. Pavlidis, and H. S. Baird, "On the recognition of printed characters of any font and size," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9. pp. 274–288, Mar. 1987.
[116] H. Freeman, "On the digital computer classification of geometric line patterns," in*Proc. Natl. Elect. Conf.*, vol. 18, 1962, pp. 312–324.
[117] G. Gallus and P. W. Nourath, "Improved computer chromosome analysis incorporating preprocessing and boundary analysis," *Phys. Med. Biol.*, vol. 15, pp. 435–445, 1970.
[118] F. Attneave, "Some informational aspects of visual perception," *Psychol. Rev.*, vol. 61, pp. 183–193, 1954.
[119] A. Rosenfeld and E. Johnston, "Angle detection on digital curves," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-5, pp. 610–614, Nov. 1975.
[120] H. Freeman and L. S. Davis, "A corner finding algorithm for chain coded curves," *IEEE Trans. Comput.*, vol. C-26, pp. 297–303, 1977.
[121] T. Pavlidis and F. Ali, "Computer recognition of hand written numerals by polygonal approximations," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-5, pp. 610–614, Nov. 1975.
[122] C. de Boor, "Good approximation by spline with variable knots," *ISNM*, vol.1, pp. 224–256, 1972.
[123] U. E. Ramer, "An iterative procedure for the polygonal approximation of plane curve," *CGIP*, vol. 1, pp. 244–256, 1972.
[124] T. Pavlidis and S. L. Horowitz, "Segmentation of plane curves," *IEEE Trans. Comput.*, vol. C-23, pp. 860–870, Aug. 1974.
[125] K. Steinbuch, "Automatische Zeichnerkenung," *SEL Nachrichten*, Heft 3, p. 127, 1958.
[126] H. Kazmierczak, "The potential field as an aid to character recognition," in *Proc. Int. Conf. Inf. Processing*, June 1959, p. 244.
[127] J. H. Munson, "The recognition of hand-printed text," in *Proc. IEEE Pattern Recognition Workshop* (Puerto Rico), Oct. 1966, p. 115.
[128] M. D. Levine, "Feature extraction: A survey," *Proc. IEEE*, vol. 57, pp. 1391–1419, Aug. 1969.
[129] S. Mori. R. Oka, A. Iwamatsu, and T. Saito, "Reading system," Japan Patent 822 486, Dec.1970.
[130] R. Oka and S. Mori, "Character recognition system-I," *T. R. IECE Japan*, PRL-71–1, Jan.1971.
[131] T. Mori, "Fundamental research with respect to feature extraction for hand-printed character recognition," *Res. Electrotech. Lab. Tokyo*, no. 762, June 1976.
[132] T. Mori, S. Mori, and K. Yamamoto, "Feature extraction method based on field effect method," *Trans. IECE Japan*, vol. 57-D, no. 5, pp. 308–315, 1974.
[133] S. Mori *et al.*, "Recognition of hand-printed characters," in *Proc. 2nd IJCPR*, Aug. 1974, pp. 233–237.
[134] Q. Zhan, "A linguistic approach to pattern recognition," Master's thesis, Waseda University, 1972.
[135] R. Narashimhan, "Syntax-directed interpretation of classes of pictures," *Commun. Ass. Comput. Mach.*, vol. 9, pp. 166–173, 1966.
[136] R. Narashimhan, "On the description, generation, and recognition classes of pictures," in *Automatic Interpretation and Classification of Images*, A. Grasselli, Ed. New York: Academic Press, 1969.
[137] K. Komori, T. Kawatani, K. Ishii and Y. Iida, "A feature concentration method for character recognition," in *Proc. IFIP Congress 77* (Toronto), Aug. 1977, pp. 39–34.
[138] K. Yamamoto and S. Mori, "Recognition of hand-printed characters by outer most point method," in *Proc. 4th IJCPR*, 1978, pp.794–796.
[139] T. Mori *et al.*, "Hand-printed alphanumerals and special symbols recognition system," *Trans. IECE Japan*, vol. 58-D, no. 8, pp. 442–449, 1975.
[140] T. Mori, S. Mori, and Y. Monden, "Pattern recognition based on coordinats and edge direction distribution," *T. R. IECE Japan*, PRL-73–11, Nov. 1973.
[141] F. O'Gorman and M. B. Clowes, "Finding picture edge through co-linearity of feature points," in *Proc. 3rd IJC on AI* (Kyoto), Aug. 1973, p. 543.
[142] R. Oka, "Cellular feature extraction from patterns," *Trans. IECE Japan*, vol. J65-D, no. 10, pp. 1219–1226, Oct. 1982.
[143] K. S. Fu, *Syntactic Method in Pattern Recognition.* New York: Academic Press, 1974.
[144] K. S. Fu, *Syntactic Method in Pattern Recognition Applications*. New York: Springer-Verlag, 1977.
[145] F. Ali and T. Pavlidis, "Syntactic recognition of handwritten numerals," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 537–541, July 1977.
[146] R. Abe, "Synthesis of an automaton which recognizes pattern destorted as strings of symbols," *T. R. IECE Japan*, PRL74–5, pp. 43–54, 1974.
[147] R. C. Gonzalez and M. G. Thomason, *Syntactic Pattern Recognition.* Reading, MA: Addison-Wesley, 1978.
[148] A. C. Shaw, "The formal picture description scheme as a basis for picture processing system," *Infom. and Contr.*, vol. 14, pp. 9–52, 1969.
[149] S. Mori, "A nonmetric model of hand-printed characters," *Res. Electrotech.Lab. Tokyo*, vol. 798, Aug. 1979.
[150] S. Mori, "An algebraic structure representation of shape," *Trans. IECE Japan*, vol. J64-D, no. 8, pp. 705–712, Aug. 1981.
[151] H. Nishida and S. Mori, "Structural analysis and description of curves by qusi-topological features and singular points," in *Pre-Proc. IAPR Workshop on SSPR*, 1990, pp. 310–334.
[152] T. Suzuki and S. Mori, "A description method of line drawings by cross section sequence graph and its application to thinning," *T. R. IECE Japan*, PRU90–22, June 1990.
[153] RTUA, *Buyer's Guide*, Recognition Technologies Users Association, U.S.A., vol. 13, no. 1, pp. 38–48, 1990.
[154] AIM, *The Source Book*, Automatic Identification Manufactures, Inc., U.S.A., p. 64, 1990.

[155] S. Impedovo, L. Ottaviano, and S. Occhingro, "Optical character recognition—A survey," *Int. J. Pattern Recognition and Artificial Intelligence*, 1991.
[156] C. Y. Suen, "Character recognition by computer and applications," in *Handbook of Pattern Recognition and Image Processing*, T. Y. Young and K. F. Fu, Eds. Boca Raton, FL: Academic Press, 1986, pp. 569–586.
[157] C. Y. Suen, "OCR products and manufacturers," Center for Pattern Recognition and Machine Intelligence, Concordia University, Montreal, 1991, in preparation.
[158] C. Y. Suen, Ed., *Frontiers in Handwriting Recognition*. Concordia University, Montreal: CENPARRMI, Apr. 1990.
[159] C. Y. Suen and S. Mori, "Standardization and automatic recognition of hand-printed characters," in *Computer Analysis and Perception*, vol. 1, *Visual Signals*, C. Y. Suen and De Mori, Eds. Boca Raton, FL: CRC Press, 1982, pp.41–53.
[160] R. K. Gerlach, "Wide-tolerance optical character recognition for existing printing mechanisms," in *Optical Character Recognition*, G. L. Fischer *et al.*, Eds. McGregor & Wemer, 1962, pp. 93–114.
[161] C. C. Heasly, Jr, and G. L. Fischer, Jr, "Some elements of optical scanning," in *Optical Character Recognition*, G. L. Fischer *et al.*, Eds. McGregor & Wemer, 1962, pp. 15–26.
[162] E. C. Greanias, "Some important factors in the practical utilization of optical character readers," in *Optical Character Recognition*, G. L. Fischer *et al.*, Eds. McGregor & Wemer, 1962, pp. 129–146.
[163] H. Genchi, K. Mori, S. Watanabe, and S. Katsuragi, "Recognition of hand-printed numerical characters for automatic letter sorting," *Proc. IEEE*, vol. 56, pp. 1292–1301, Aug. 1968.
[164] I. Sheinberg, "The INPUT2 document reader," *J. Pattern Recog. Soc.*, vol. 2, pp. 167–173, Sept. 1970.
[165] S. Yamamoto, M. Yasuda, Y. Miyamoto, and M. Tsutsumi, "Design of handwritten numeral recognition," *Trans. IECE Japan*, vol. 53-C, no. 10, pp. 891–898, 1970.
[166] T. Sano and T. Hananoi, "Type H-8959 optical character reader," *Hitachi Review*, vol. 54, no. 12, pp. 1077–1082,1972.
[167] M. Oka, S. Yamamoto, and S. Kadota, "Pattern feature detection system," U.S. Patent 3 863 218, Jan. 28,1975.
[168] T. Fujimoto *et al.*, "Recognition of hand-printed characters by nonlinear elastic matching," in *Proc.3rd IJCPR*, Nov. 1976, pp. 113–117.
[169] R. B. Hennis, "The IBM 1925 optical page reader, PartI: System design," *IBM J. Res. Develop.*, vol. 12, no. 5, pp. 346–353, 1968.
[170] M. R. Bartz, "The IBM 1925 optical page reader, Part II: Video thresholder," *IBM J. Res. Develop.*, vol. 12, no. 5, pp. 354–363, 1968.
[171] D. R. Andrews, A. J. Atrubin, and K. C. Hu, "The IBM 1925 optical page reader, Part III: Recognition logic development," *IBM J. Res. Develop.*, vol. 12, no. 5, pp. 364–371, 1968.
[172] K. Komori, T. Kawatani, K. Ishii, and Y. Iida, "hand-printed Katakana characters recognition by feature concentration method," *Trans. IECE Japan*, vol. J63-D, no. 11, pp. 962–968, Dec. 1980.
[173] S. Mori and C. Y. Suen, "Automatic recognition of symbols and architecture of the recognition unit," in *Computer Analysis and Perception*, vol. 1, *Visual Signals*, C. Y. Suen and R. De Mori, Eds. Boca Raton, FL: CRC Press, 1982, pp.17–40.
[174] S. Diehl and H. Eglowstein, "Tame the paper tiger," *Byte*, pp. 220–241, Apr. 1991.
[175] M. Yasuda, K. Yamamoto, H. Yamada, and T. Saito, "An improved correlation method-hand-printed Chinese character recognition in a reciprocal feature field," *Trans. IECE Japan*, vol. J68-D, pp. 353–360, Mar. 1985.
[176] K. Yamamoto, H. Yamada, and T. Saito, "Current state of recognition method for Japanese characters and database for research of hand-printed character recognition," in *Proc. Int. Workshop Frontiers in Handwriting Recognition* (Chateau de Bonas, France), Sept. 1991, pp.81–99.
[177] M. Kishi, T. Kiuchi, Y. Maruko, A. Yoshida, and Y. Hongo, "A feature extraction processor," in *Proc. Anal. Conf. IECE Japan*, 1988, D-208.
[178] H. Matsumura, K. Aoki, T. Iwahara, H. Oohuna, and K. Kogure, "Desk-top optical handwritten character reader CLL-2000," *Sanyo Tech. Rev.*, vol. 18. no. 1, pp. 3–12, Feb. 1986.
[179] S. Amari, "Mathematical foundation of neurocomputing," *Proc. IEEE*, vol. 78, pp. 1443–1463, Sept. 1990.
[180] K. Yamamoto, T. Mori, S. Mori, and J. Shimizu, "Machine recognition of Katakana and numerals," *Trans. IECE Japan*, vol. 59-D, no. 6, pp. 414–421, 1976.
[181] K. Yamamoto, "Recognition of hand-printed Hiragana characters by concave and convex features and automatically merging the dictionary," *Trans. IECE Japan*, vol. J65-D, no. 6, pp. 774–781, June 1982.
[182] H. Nishida and S. Mori, "An approach to automatic construction of structural models for character recognition," in *Proc. First ICDAR* (Saint-Malo, France), 1991, pp. 231–241.
[183] H. S. Baird, "Feature extraction for hybrid structural/statistical pattern classification," *CVGIP*, vol. 42, pp. 318–333, 1988.
[184] R. M. Brown, T. M. Fay, and C. L. Walker, "Hand-printed symbol recognition system," *Pattern Recognition*, vol. 21, no. 2, pp. 91–118, 1988.
[185] J. J. Hull *et al.*, "A blackboard-based approach to handwritten Zip Code recognition," in *Proc. U.S. Postal Service Adv. Techol. Conf.*, 1988, pp. 1018–1032.
[186] C. Y. Suen, C. Nadal, T. A. Mai, R. Legault, and L. Lam, "Recognition of totally unconstrained handwritten numerals based on the concept of multiple experts," in *Proc. Frontiers in Handwritting Recognition* (CENPARMI Concordia University), 1990, pp. 131–140.
[187] A. Krzyzak, W. Dai, and C. Y. Suen, "Unconstrained handwritten classification using modified backpropagation model," in *Proc. Frontiers in Handwritting Recognition* (CENPARMI Concordia University), 1990, pp. 155–164.
[188] Y. Le Cun *et al.*, "Constrained neural network for unconstrained handwritten digit recognition," in *Proc. Frontiers in Handwritting Recognition* (CENPARMI Concordia University), 1990, pp. 145–151.
[189] K. Fukushima and N. Wake, "Handwritten alphanumeric character recognition by neocognitron," *IEEE Trans. Neural Networks*, vol. 2, pp. 355–365, May 1991.
[190] J. J. Hull, A. Commike, and T. K. Ho, "Multiple algorithms for handwritten character recognition," in *Proc. Frontiers in Handwritting Recognition* (CENPARMI Concordia University), 1990, pp. 117–124.
[191] T. K. Ho, J. J. Hull, and S. N. Srihari, "Combination of structural classifier," in *Pre-Proc. IAPR Workshop on SSPR* , June 1990, pp.123–136.
[192] C. Nadal, R. Legault, and C. Y. Suen, "Complementary algorithms for the recognition of totally unconstrained handwritten numerals," in *Proc. 10th IJCPR*, June 1990, pp. 443–449.
[193] S. Amari, *Nikkei AI Fall Special Issue on Neural Network Applications*, p. 33, 1990.

Shunji Mori (Member, IEEE) was born in Hokkaido, Japan, in 1934. He received the B.E. degree in 1956 and the Dr.Engineering degree in 1978 from Hokkaido University.

From 1957 to 1983 he was with the Electrotechnical Laboratory of the Ministry of International Trade and Industry of Japan. There he served as head of the Pattern Processing section from 1969 to 1981 and as supervisor of the Pattern Information Processing System (PIPS) from 1971 to 1973. In 1975 he was a visiting researcher at the Computer Vision Laboratory, University of Maryland, for half a year and from 1981 to 1982 he was a visiting professor in the Department of Computer Science, Concordia University, Montreal. From 1983 to 1987 he worked for Nippon Schlumberger as head of the Modeling Department of System Engineering. Currently he is head of the Artificial Intelligence Research Center of the Research and Development Central Research Center of Ricoh Company, Ltd. His research interests include pattern recognition, image processing, and knowledge engineering.

Dr. Mori is a member of AAAI, the Institute of Electronics, Information, and Communication Engineers of Japan, the Information Processing Society of Japan, and the Chinese Language Computer Society.

Ching Y. Suen (Fellow, IEEE) received the M.Sc.(Eng.) degree from the University of Hong Kong and the Ph.D. degree from the University of British Columbia, Canada.

In 1972, he joined the Department of Computer Science of Concordia University, Montreal, Canada, where he became Professor in 1979 and served as Chairman from 1980 to 1984. Presently he is the Director of CENPARMI, the new Centre for Pattern Recognition and Machine Intelligence of Concordia. During the past 15 years, he has been appointed to visiting positions at several institutions in different countries.

Dr. Suen is the author or editor of several books, with titles ranging from *Computer Vision and Shape Recognition* and *Frontiers in Handwriting Recognition* to *Computational Analysis of Mandarin and Chinese.* His latest book, *Operational Expert System Applications in Canada,* is published by Pergamon. Dr. Suen is the author of many papers and his current interests include expert systems, pattern recognition and machine intelligence, optical character recognition and document processing, and computational linguistics.

An active member of several professional societies, Dr. Suen is an Associate Editor of several journals related to his areas of interest. He is a former president of the Canadian Image Processing and Pattern Recognition Society, governor of the International Association for Pattern Recognition, and president of the Chinese Language Computer Society. Prof. Suen received the 1992 ITAC/NSERC award for his contributions to pattern recognition, expert systems, and computational linguistics.

Kazuhiko Yamamoto (Member, IEEE) received the B.E. degree in 1969, the M.E. in 1971, and the Dr.Engineering in 1983, all from Tokyo Denki University, Tokyo, Japan.

Since 1971 he has been with the Electrotechnical Laboratory of MITI studying hand-printed character recognition. He participated in the development of OCR in the Pattern Information Processing System Project form 1971 to 1979. From 1979 to 1980 he was a visiting researcher in the Computer Vision Laboratory, University of Maryland, where he worked on computer vision. He has been head of the Image Processing Section at ETL since 1986. His research interests include pattern recognition and artificial intelligence.

Dr. Yamamoto is a member of the Institute of Electronics, Information, and Communication Engineering of Japan and of the Information Processing Society of Japan.