IAPR Workshop on CV - Special Hardware and Industrial Applications OCT.12-14, 1988, Tokyo

AUTOMATIC READING SYSTEM FOR PRINTED DOCUMENTS

Teruo Akiyama
NTT Human Interface Laboratories

1-2356 Take Yokosuka-shi
Kanagawa 238-03, Japan

Norihiro Hagita
NTT Basic Research Laboratories

3-9-11 Midori-cho Musashino-shi
Tokyo 180, Japan

ABSTRACT

A system for automatically reading Japanese and English printed documents including different types of layout objects such as headlines, text lines and charts is proposed. This paper attempts to define these features, and describes a prototype system which satisfies the design objectives.

The system has two stages. In the first stage, document image segmentation is carried out using three basic features and also knowledge of document layout rules. Furthermore, projection profiles of a text line are employed to extract a wide variety of characters. In the second stage, multifont character recognition is performed using feature vectors which represent stroke complexity, direction, connectivity and relative location of individual characters. A prototype system combining these two stages is described. The system consists of three special hardware units each connected with the one host computer.

Recognition experiments with the prototype system are conducted for a range of printed documents. It is confirmed that the method proposed here is capable of reading different types of printed documents containing text lines and charts at an accuracy rate of 95 to 97 percent.

1. INTRODUCTION

Huge amounts of printed materials are published every day, such as newspapers, journals, magazines and office documents. However, the printing is styled for easy reading not for easy machine recognition. Various layout objects, such as, headlines, text lines and charts, are used to guide the reader in clearly distinguish articles and provide him with an indication of article content. To access the knowledge contained within these documents, an accurate automated document entry system is needed that can recognize the differing styles of document layouts. With the advent of such an automated system, the space required for documents in offices and the time needed for information retrieval will be dramatically reduced.

There are some data entry systems in commercial use for printed documents. However, they are difficult to use for documents whose layout structure is complex such as newspapers. An automatic reading system for Japanese printed materials has been proposed[1], but it cannot easily handle documents that include charts, pictures or tables.

In this paper, a complete prototype system which can read a variety of Japanese and English printed documents is proposed. In sections 2 and 3, we focus on document layout structure and character recognition algorithms for automatic document reading. Next, a prototype system for automatic document reading, in which these algorithms are combined, is described. Finally, we discuss the results of experiments using typical printed documents.

2. LAYOUT STRUCTURE RECOGNITION

2.1 Document Layout Structure

Documents have three different types of areas, they are, headline areas, text line areas and chart/ picture areas, named graphic areas. These areas include headlines, text lines and charts as layout objects. Fig.1 shows the hierarchical decomposition of documents. In order to assist the development of the layout recognition algorithm, the following assumptions are made,

(1) document images have already been binarized,
(2) the three types of areas mentioned above can be expressed as non-overlapped rectangular areas, and
(3) background of the document is white.

2.2 Basic Features

When humans read documents, the document image is apparently segmented into different area types, such as headlines, text lines and graphic areas. This division may use some features which reflect global properties of the document and detailed shape of document components. From this understanding, three basic document features were developed.

(1)Projection profiles

A projection profile is obtained by counting black pixels in horizontal(PPh) or vertical(PPv) raster scan lines(Fig.2(a)) over rectangular areas of the document image. Projection profiles reveal rough positions and sizes of layout objects, and also determines degree of document skew.

(2)Crossing counts

A crossing count is obtained by counting the points at which the pixel value turns form 0(white)

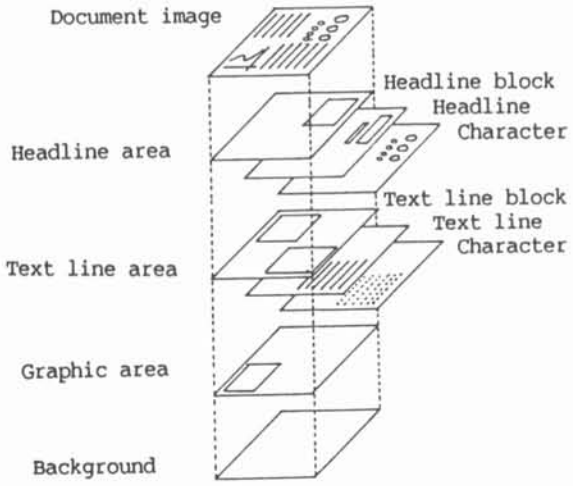


Fig.1 - Document layout structure.

to 1(black) in horizontal(CCh) or vertical(CCv) raster scan lines(Fig.2(b)) over rectangular areas of the document image. Crossing counts are used to measure the complexity of document image.

(3)Circumscribed rectangles

A circumscribed rectangle, which contains a cluster of contiguous black pixels in a document is used to exactly represent the cluster's position and size. It can be obtained by a contour line tracing process. Each circumscribed rectangle is expressed by four values, its height, width and the coordinates of its top left corner(Fig.2(c)).

2.3 Pre-processing

(1)Detection of document orientation

The values of projection profiles whose direction parallels that of text lines, fluctuate more than those that run at right angles to text lines. By comparing the squared sum of horizontal projection profiles (PPh) with that of the vertical profiles (PPv), orientation of the text lines is discovered.

(2)Skew normalization

In the skew detecting process, a document image is divided into document-wide swaths of equal height. The degree of document skew is obtained by calculating the arctangent of phase shift "α" between projection profiles from adjoining swaths (Fig.3). Skew normalization is carried out by rotating the image using an affine transform operation.

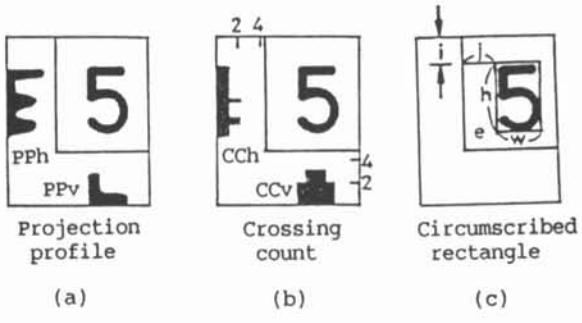


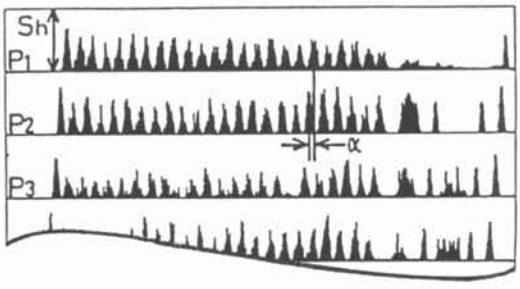
Fig.2 - Basic three features.

2.4 Area Extraction

In area extraction, the three basic features, mentioned in 2.2, are utilized. Common knowledge about layout rules, which are incorporated not as rules[2][3] but as procedures, will help area extraction. The flow of area extraction process is shown in Fig.4. This process has four steps as follows.

(1)Field separator and candidates extraction

Firstly, the thicknesses of text lines are estimated using histograms of circumscribed rectangle widths. Candidates close to the maximum value of the histogram are selected as text line characters. Rectangles whose width exceed the range, and which have low crossing counts are extracted as candidates for headline character components. Rectangles whose width exceed the range, and which have high crossing counts are candidates for elements included in graphic areas. On the other hand, circumscribed rectangles which have high ratio of height to width, and which have small crossing counts are extracted as field separators.



θ = arctan(α /Sh)

Fig.3 - Skew(θ) detection.

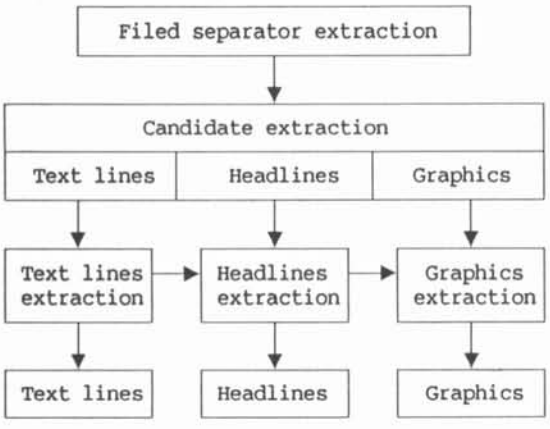


Fig.4 - Area segmentation flow.

(2)Text line area extraction

Text line area extraction is performed based on the features of projection profiles and crossing counts. Text blocks including text lines, are extracted by a segmentation process using field separators and blank areas both of which separate layout objects in a document. At beginning of the segmentation process, a rectangular area that just covers all text line candidates is chosen. Then, this area is successively divided into small sub-areas each of which includes a set of text lines, that is a text block. Segmentation process is terminated when crossing counts of all sub-areas become lower than a predefined limit. Each text line is extracted as a circumscribed rectangle which merges adjacent text line characters.

(3)Headline area extraction

Both horizontal and vertical direction headlines are possible in the vertical printing style, although only horizontal direction headlines are found in the horizontal printing style. Headlines, composed of aligned characters, are extracted using a circumscribed rectangle merging operation. If adjacent rectangles are located in the horizontal direction, the headline is horizontal. The converse is true for vertical headlines.

(4)Graphic area extraction

Candidates of graphic areas, extracted in process (1), includes charts and/or pictures. Text lines next to these areas are considered captions. Therefore, the areas and text lines are merged and extracted as graphic areas.

2.5 Character Segmentation

In order to accurately extract individual characters from a Japanese text line, we have to extract touched, irregular pitched and multi component characters. To cope with these problems, some useful properties of printed Japanese characters can be utilized as follows.

(1) The height/width ratio of full pitch characters is nearly equal to 1.
(2) Most katakana and hiragana characters do not touch each other.
(3) Touching characters have a constant pitch.

With these considerations, an individual character segmentation algorithm has been developed[4]. In this algorithm, isolated characters and single component characters are extracted first, next, remaining characters are extracted using the information of previously extracted character's pitch and position. In order to extract a pair of half pitched characters which is extracted as single full pitch character, examination by alphanumeric character recognition techniques is carried out. If each component in an extracted area is recognized as an alphanumeric character, segmentation results are modified and the area is divided into two areas each of which includes a half pitched alpha-numeric character.

3. Character Recognition

Printed documents to be read usually include different types of character sets, such as Kanji characters, Kana characters and Alpha-numeric characters. Additionally, they may be in many character fonts. Our research has confirmed that characters can be effectively recognized based on the ratio of black pixels, number, direction, connective relation and relative position of character strokes[5]. From this, several character recognition methods based upon "Direction Contributivity" have been developed. The following section outlined the method discovered in our original paper[1][5]. The direction contributivity is a quadruple component vector which is described as $d = (d_1, d_2, d_3, d_4)$. At a black pixel point P in the stroke shown in Fig.5, each of the eight directional run-lengths, L_m, are obtained. Each component of the vector, d_m $(m = 1, 2, 3, 4)$, is calculated by equation (1).

$$d_m = \frac{L_m + L_{m+4}}{\sqrt{\sum_{j=1}^{4} (L_j + l_{j+4})^2}} \qquad (1)$$

A character recognition method, which included the two stages of pre-classification and identification, was constructed in order to reduce computation time[1]. In the pre-classification stage, about four percent of 2253 categories are selected as candidates for each input character using the 64 component dimensional vector, the G-DCD (Global Direction Contributivity Density) and L-DCD (Local Direction Contributivity Density). PDC (Peripheral Direction Contributivity), which is a 1152 component vector, was utilized in order to uniquely recognize characters. The features, G-DCD, L-DCD and PDC, are based upon direction contributivity[5]. The PDC vector $P_{tmn}(k)$ can be calculated as follows (Fig.6).

(step1) Scan the unknown character from one direction, $t = 1$.
(step2) For the first contour edge, $n = 1$, determine the d_m vector for each black pixel on the characters leading edge.
(step3) Each of the four components of the $d_m (m = 1, 2, 3, 4)$ vector are mapped onto a different axis as shown in Fig.6.
(step4) The components are averaged into 12 equal areas $(k = 1, 2, ..., 12)$ to create the PDC feature vector.
(step5) Repeat steps 2,3 and 4 for successive contour edges, $n = 2, 3$.
(step6) Repeat steps 2,3,4 and 5 for successive scanning directions, $t = 2$ *to* 8.

The method is capable of reading not only printed Japanese characters but also hand-written characters. The dimensional reduction of PDC vector for printed characters will be possible by means of orthogonal transformation, because it has been confirmed that di-

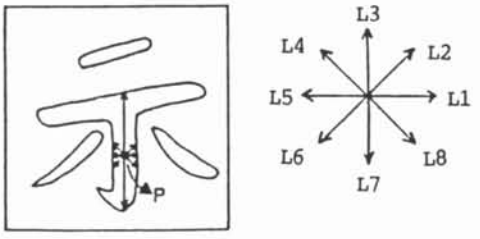
mension of PDC vector can be reduced from 1152 to about 100 while still preserving the same hand written character recognition accuracy[6].

4. PROTOTYPE SYSTEM

A prototype automatic document reading system was constructed with three special hardware units. The system contains an area extractor, feature extractor and classifier each of which is connected to the one host computer shown in Fig.7. The host computer controls each hardware unit. General output devices were used for checking processing results. The bit-map display can show original document images (Fig.8), result of area extraction (Fig.9) and result of individual character extraction (Fig.10). These results can also be confirmed through printed output. First, area extraction of layout objects and individual character segmentation was performed using the area extractor, and then recognition vectors were extracted by the feature extractor for each character. Finally, the classifier selected one character as the recognition result for each extracted character candidate.

4.1 Area Extractor

Although a stand-alone type system for document layout recognition has been proposed[7], the area extractor developed here is designed as a attached processor in order to permit algorithm modification. It performs following document image processes under the control of host processor, (1)computation of the three basic features, (2)rotation operation, (3)run-length operation, (4)contour trace operation, (5)pixel counting operation, and (6)logical operation (for image). Each character image, 64x64 pixel, extracted by the area extractor is sent to the feature extractor via the host computer.



(a) (b)Scan direction.

Fig.5 - Direction contributivity.

4.2 Feature Extractor

The feature extractor is capable of extracting L/G-DCD and PDC features that are used for the prototype system. Some operations for extracting features, such as number of black/white pixel counting and run-length computing, are accelerated by means of parallel processing using multiple ALUs. Currently, it has a $\sqrt{\ }$ look up table for high speed computing of feature vectors. There are four ALUs for PDC feature vector computation; two vector components of the eight directions are computed by each ALU. In order to choose best number of vector components, it is possible to change the number of parameters.

4.3 Classifier

The classifier assigns each feature vector extracted by the feature extractor to some character category. It is composed of two accumulating units each of which has a sorter and distance calculator, and each unit is connected to reference pattern memory. The distance calculator is capable of computing various kinds of distance between feature vectors, e.g., the city block distance or Euclidean distance, weighted Euclidean distance, and it can also compute similarity. The sorter arranges the distance values in an ascending order, or of similarity in a descending order. Next, it selects

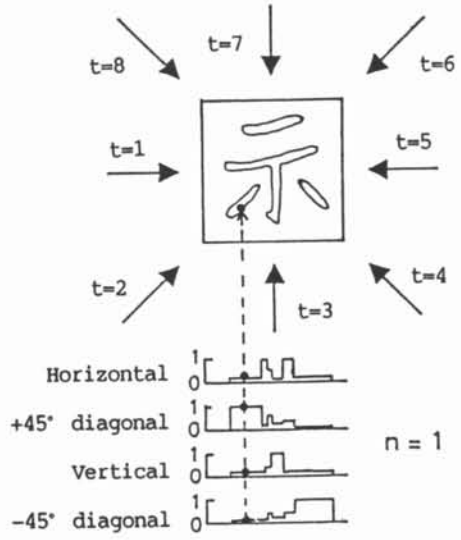


Fig.6 - PDC(Peripheral Direction Contributivity)

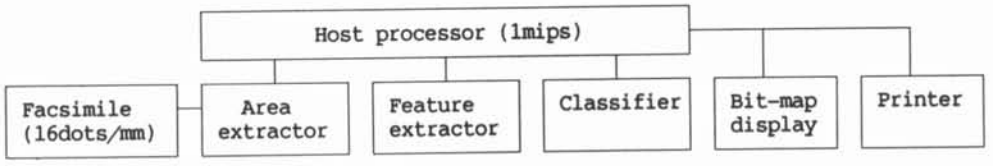


Fig.7 - System configuration.

Fig.8 - Bit-map display.

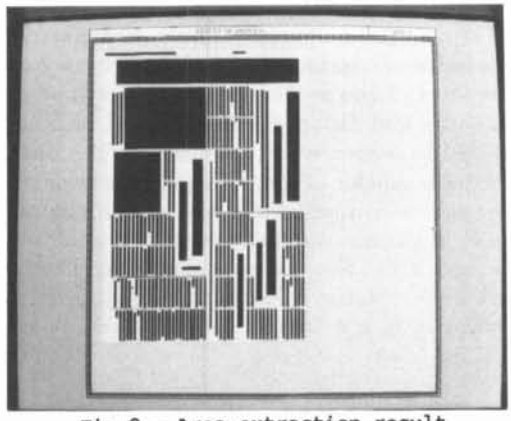


Fig.9 - Area extraction result.

Fig.10 - Character segmentation result.

some candidates for pre-classification or chooses only one candidate for identification. A pair of accumulating units makes parallel processing and pipe-line processing possible.

5. EXPERIMENTS AND DISCUSSION

In this section, the results of experiments conducted using 33 documents from four different printed materials, as shown in table-1, are described. All documents, except the paperbacks, included graphic areas.

5.1 Text Line Extraction

An example of layout structure recognition result is shown in Fig.8. The low extraction rate of text lines for paperbacks was due to the existence of "rubi" characters, i.e., attached kana characters which provide Kanji pronunciation. Such characters make thickness of text lines slightly greater. Accordingly, character sequences which contain "rubi" are divided into several parts, though it was single line. For improving these results, a top-down approach using knowledge of document models should be introduced.

5.2 Character Segmentation

Extraction rates for vertically printed documents were higher than those for horizontal ones, because of the stability of vertical character pitch. All documents with irregular text pitch suffered from segmentation errors. The segmentation algorithm for merged characters with proportional pitch in English text lines remains to be developed. Through these experiments, it was found that segmentation error came from the following causes:

(1) Half pitched character adjoins multi-component character.
(2) There are half pitched characters which cannot be recognized.
(3) There are proportional pitched characters which are merged.
(4) Character pitch cannot be estimated from projection profiles of a text line because it contained only a small number of characters.

Segmentation accuracy will be improved by re-segmentation utilizing feedback control of recognition

Table 1. Experimental results.

	Newspaper articles	Paperbacks	Journals (Japanese)	Journals (English)
Data number	2	10	19	2
Text line number	344	179	1201	201
Character number	5358	5763	23562	7228
(a)Text Line extraction rates(%)	100	95.5	99.9	100
(b)Character extraction rates(%)	99.5	99.8	97.7	99.6
(c)Character recognition rates(%)	96.7	99.5	98.8	97.6
(d)Document recognition rates(%) (d)=(a)× (b)× (c)	96.2	94.8	96.4	97.2

results. Actually, in the experiments with the Japanese journal, modification of segmentation results using half-pitched character recognition was carried out, and 341 out of 727 previously non-extracted characters were extracted correctly. Thus, effectiveness of feedback control in the character segmentation process has been confirmed.

5.3 Character Recognition

Twelve typefaces of Japanese characters including alpha numeric were used as the reference pattern for constructing a feature vector dictionary. Preclassification by G/L-DCD features, and classification by PDC features was carried out, and city block and weighed Euclidean distance were utilized in order to measure vector distance. In the multi-font character recognition experiments, 96.7 to 99.5 percent of the correctly segmented Japanese characters were recognized. Most of the recognition error in the Japanese documents occurred because of smashed and blurred character images. In the English documents, 76.6 percent of recognition error occurred because of insufficient accuracy, i.e., "5"-"s" and "h"-"b". Modification of the reference pattern of those characters by learning will help to improve recognition rates. The recognition algorithm proposed here has no capability of distinguishing the same shaped characters, such as "V" and "v", or the number "1" and small character of L, "l". In order to solve this problem, post-processing using linguistic information is needed.

5.4 Processing Time

If the processing speed without specialized hardware is taken as 1, the relative processing speeds with specialized hardware, for area extraction, character segmentation and character recognition were 19, 3 and 15, respectively. Processing without specialized hardware performed with a 1-mips host processor. Effectiveness of the specialized hardware in character segmentation was less than for other operations, because the proportion of image processing was smaller. The processing time needed for an A4 size (297x210mm) document layout structure recognition was about 80 seconds, and for character segmentation was about 0.1 seconds/char. The system can read each Japanese character about in about 1 second.

6. CONCLUSION

In this paper, a complete prototype system for automatically reading printed documents was described. The system first analyzes layout structure of input document images using the three basic features of projection profiles, crossing counts and sizes/positions of circumscribed rectangles. It then recognizes printed multi-font Japanese characters after segmenting individual character images from text lines. The character recognition process is based on G/L-DCD and PDC feature vectors which represent stroke complexity, direction, connectivity and the relative location of individual characters. The effectiveness of the system has been confirmed in experiments using several types of document. Accordingly, the prototype system proposed here, will no doubt, greatly contribute to the creation of a practical automated document entry system in the near future. For improving the capability of the prototype system, heuristic or structural analytical approaches using AI techniques are necessary.

ACKNOWLEDGEMENT

The authors wish to thank Dr. Isao Masuda and Dr. Seiichiro Naito for their kind support in this research. They would also like to thank Mr. Isao Kawashima, Executive Manager of the Human Interface Laboratories and Dr. Akihiro Hashimoto, Executive Manager of the Basic Research Laboratories for their encouragement.

REFERNCES

[1] I.Masuda, N.Hagita, T.Akiyama, T.Takahashi and S.Naito: "Approach to Smart Document Reader System", Proc.CVPR'85, pp.550-557, 1985.

[2] K.Kubota, O.Iwaki and H.Arakawa, "Document Understanding System", Proc.ICPR'84, pp.612-614, 1984.

[3] D.Niyogi and S.N.Srihari, "A Rule-based System for Document Understanding" Proc.AAAI'86, pp. 789-793, 1986.

[4] T.Akiyama, S.Naito and I.Masuda: "A Method of Character Extraction from Format-unknown Document Images", Proc.ICTP'83, pp.85-90, 1983.

[5] N.Hagita, S.Naito and I.Masuda: "Handprinted Kanji Characters Recognition based on Pattern Patching Method", Proc.ICTP'83, pp.169-174, 1983.

[6] N.Hagita and I.Masuda: "Design Principles of Feature Vectors for Recognition of Large Character Sets", Int.Conf.SMC, pp.826-830, Oct, 1987

[7] K.Inagaki, K.kato, T.Hiroshima and T.Sakai, "MACSYM: A Hierarchical Parallel Image Processing System for Event-driven Pattern Understanding of Document", Pattern Recognition, Vol.17, No.1, pp.85-108, 1984.