
Multiple Hypotheses Document Analysis

Tatsuhiko Kagehiro and Hiromichi Fujisawa

Central Research Laboratory, Hitachi, Ltd., Tokyo 185-8601, JAPAN
{tatsuhiko.kagehiro.tx,hiromochi.fujisawa.sb}@hitachi.com

Summary. Document layout analysis is a tough task for a document analysis and recognition system, especially when there are many variations in the layouts. Often, layout analysis requires character recognition, while character recognition requires layout analysis beforehand. It's a Catch-22, or a "chicken-and-egg" problem. This chapter discusses this kind of dilemma and presents a two-part solution that first analyzes the layout and then, using a hypothesis-driven approach, segments the numerical character line. Basically, the approach is to first generate multiple hypotheses based on low-level image processing, then to conduct layout analysis and create many candidates based on each of the hypotheses. Finally, the correct candidate is selected by the results of content recognition. Probabilistic verification is used to select the candidates that are input to the recognition module, with parameters that are learned from samples in advance. The second part of the solution, which relies on a hypothesis-driven approach for the segmentation of the numerical character line will also be presented. As a test case, these solutions were applied to the Japanese postal address recognition system. They show how tough problems involved in analyzing the surface images of mail pieces can be solved. The hypotheses-driven approach manages every possible variation, including writing orientation, printed or hand-written, address-block location, character size, and so on.

1 Introduction

OCR (Optical Character Recognition) has many uses in the automation of work in offices and factories. About 30 years ago, OCR was developed to read numeral characters written in fixed areas, and these recognition results are used to generate helpful information to automate office and factory work. As time went on, the readable character position became more variable, and various character sets (Alphabets, Kanji, etc.) could also be read by OCR. OCR continues to expand and become more adaptable, and it is hoped that in future OCR will be able to recognize documents correctly and read more types of documents.

When the design of the document is known in advance, OCR can get the positions of the characters in the input image. An example of these documents

are fixed forms. In this case, it is not necessary to analyze the layout of the document; OCR can segment the character patterns by using a priori knowledge, and recognize each character. But when OCR is used to read more varied types of documents, the positions and layout of characters can change with each type. The document types shown below are itemized by their degree of format complexity, from type 1 to type 4.

Type 1: Fixed form

Type 2: Business or name card, Cover page of paper, Contents page of book

Type 3: Address surface of mail, Cover page of magazine, Newspaper

Type 4: Handwritten memorandum, Signed board

For type 1, examples of fixed forms include those used by bank tellers and various office forms. OCR machines can read the contents of fixed forms correctly.

In type 2, the character position is strongly restricted, but not completely fixed. In this case, an OCR system can analyze the layout correctly. A little recognition error can occur and in a real workflow humans must correct these errors.

In type 3, there is a common rule in the document layout, but this rule has some variations. Type 3 documents make a good target for document analysis research, so many papers have been published [18][19][21][22][23][12]. The destination address on mail is an especially good candidate for automatic recognition, because current mail sorting machines read the destination address less than perfectly. Rejected mail requires humans to attach the correct address. The process described here represents a real solution for the use of OCR in the real world.

As for type 4, it is difficult for OCR to recognize handwritten memos and signed boards. Because humans write memos according to their own inspiration or memory, there is no format for the character allocation and sometimes no structure for the layout. There is a big demand for automatic recognition of signed boards in the real world, because they often contain very important information. But in order for OCR to read signed boards two different modules in the layout analysis must be developed completely, one to detect the position of the signed board from the scene image, the other to analyze the layout on the board. In order to recognize the kinds of documents in type 4, it will be necessary to implement a human vision approach to the computer [25].

2 Approaches to Layout Analysis

Layout analysis has a very important role in document recognition. If the layout cannot be analyzed correctly, it is impossible to understand the contents of the document. But on the other hand, the contents of the document provide useful information for analyzing the layout. This is a “Catch-22” or the

so-called “chicken-and-egg” problem. Humans can analyze the layout and understand the content in parallel, but it is difficult for the computer to imitate the human process, because computer procedures are generally sequential.

There are two approaches to document analysis, top-down and bottom-up. The top-down approach tries to estimate the layout based on a model, and extract the features which are needed to validate the model [5][16]. The bottom-up approach first extracts various features from the input image, and then tries to match these features to the model. Accordingly, if it is possible to estimate the layout in advance, the top-down approach is more effective and reliable. The bottom-up approach is generally expensive, but this approach has advantage to not output the big error, if the model of the layout is not known in advance [10].

There is an optimal approach for each type of document. The top-down approach is most effective for documents with a predictable layout structure (for example, business cards, cover pages of papers and content pages of books). Common-sense knowledge of the layout is given to the OCR system in advance, and the positions of characters are matched to this layout knowledge. After that, the contents of the document are read using the information about the layout. In this case, it is necessary to dynamically match the character positions to the layout information, because the character positions can be somewhat ambiguous [14][30].

If the supposition of the layout is difficult, or if variations of the common-sense approach can be expected, then the bottom-up approach is more effective. But a complete bottom-up approach requires large calculation costs, because all characters in the document must be recognized before the layout analysis. So the real OCR is designed to use the positions and sizes of each character pattern to analyze the layout. Usually, a rectangle circumscribed to each pattern is used to express position and size. In this method, the rectangles of the character patterns are extracted first, and then close rectangles are merged to create the character lines or blocks. This method does not require huge calculation costs and can use the layout knowledge while merging the pattern rectangles.

3 ROI in Document

There are two types of office documents. For one type, it is necessary to understand the entire contents of the document; the other requires understanding of only the important part of the document. This important part is called ROI (Region Of Interest). In fact, OCR systems are often designed to read only the ROI in documents, because the content of the ROI has sufficient information to automate the office task. For example, the ROI of an application form might be the name and address of the applicant, and the ROI of a piece of mail is the destination address and name. In the case of a scientific paper, the title, authors, and affiliations constitute a ROI. In these cases, it is not

necessary to understand the entire contents of the document. But in order to select the ROI, the whole structure of the layout must be analyzed.

The multiple hypotheses approach is an effective way to extract the ROI from a document. This approach creates a number of candidates by making some hypotheses in cases in which the layout analysis process can not decide on only one answer. If the post-process can get other useful information, the number of candidates is reduced. Finally, the most valid candidate is selected as the correct answer. The candidates are evaluated under each hypothesis which was applied during the candidate creation. For example, if the size of characters is unknown, a number of ROI candidates are created by some hypotheses regarding character size. After that, each candidate is evaluated on the condition of each character size which was used by the hypotheses. This approach can control the balance between calculation cost and recognition accuracy. If it is necessary to improve the recognition accuracy without increasing the calculation cost, unusable candidates must not be created and more effective information must be used to evaluate the candidates in each step. An OCR which adopts multiple hypotheses can be implemented quickly, because no one process in the system is responsible for outputting a completely correct answer, and recognition accuracy is increased while each process is revised step by step.

4 Target Documents for Discussion

In this chapter, the layout analysis of mail addresses is used as an example in order to discuss layout analysis and ROI extraction strategies. The layout of mail addresses is loosely restricted by common-sense knowledge. The ROI of a piece of mail is the destination address, which is needed to automate the mail-sorting task. But it is not easy to determine the ROI on a given piece of mail. It can be in a variety of positions, it can have one of two character types (printed and handwritten), there can be variations of the mail direction, and in the countries of East Asia, the character lines can be either horizontal or vertical. In addition there is the stamp area, the return address area, and sometimes advertisements on the mail, all of which create heavy obstacles to ROI extraction.

The multiple hypotheses approach is used to analyze the layout and read the destination information. After binarization of the input image, each process creates some candidates; post-processing then evaluates the candidates. In the final step, one candidate is selected as the correct answer by checking the destination point in a real address data set. This approach can control the balance between calculation cost and recognition accuracy, because the number of the candidates in each process can be controlled by setting various parameters.

The character segmentation method which uses the multiple hypotheses approach will be introduced later. Address characters on mail can be written

by machine or hand, and in either horizontal or vertical lines in the Asian countries. There can be various interpretations of the character segmentation results, because Kanji characters consist of some simple patterns, many of which can be read as other characters. In order to solve this problem, the multiple hypotheses approach creates candidates of the segmented patterns, and selects the best candidate as a correct answer.

5 Ambiguity of Mail Address Recognition

The technique of mail destination address recognition is important for automation of the mail-sorting task. But it is difficult for a machine to understand the destination address if it does not know the layout of the address in advance [20][11]. The layout of the address has several variations and much ambiguity, so the design of the address recognition system is much more difficult and complex.

In order to recognize the destination address, it is necessary to extract the area of the destination address characters. This area is called the address-block. If many mail samples are observed, it is possible to observe that there are variations in the address-block position, direction, and printed or handwritten characters, along with the return address area and possible advertisements. Some parameters for extracting the address-block can not be fixed uniformly, because of the variations and ambiguities. Many papers have been published on the topic of analyzing the layout of mail pieces [2][19][4][5][6].

After getting the address-block, it is necessary to read the character lines in the address-block. Current character recognition and segmentation modules can hardly output the correct answer. Character recognition has been studied for several years, however many researchers are continuing to improve the technique [24][29][26][27], and various approaches to character segmentation have been reported [17][1][15][13]. For example, the first of these approaches applies the similarity of the characters for recognition, the second uses the peripheral information of the character pattern, and the last employs the linguistic information of the character strings [8][9].

The character segmentation of Japanese street numbers is an especially difficult problem. In Japan, the street numbers can be written in either horizontal or vertical lines. Furthermore, the Kanji numerals “One”, “Two” and “Three” are expressed as simple horizontal lines, with the number of horizontal lines corresponding to the Kanji numerals. Thus, “One” is one horizontal line, “Two” is two horizontal lines, and “Three” is three horizontal lines. Kanji numerals written in a vertical line introduce much ambiguity into the character segmentation. In this case, it is not effective to apply linguistic information, because there are various street numbers in real addresses. So the Kanji numerals must be segmented using information about the figures and their positions.

In this chapter, two methods which can solve the above mentioned problems are described. One is the address-block extraction method, the other is the character segmentation method for Kanji numerals in a vertical line. These methods adopt the multiple hypotheses approach and use a Bayesian rule for the validation of each candidate. The Bayesian rule can combine several different confidence values, and calculate the total confidence value. If the likelihood ratio can be observed from many learning samples, the Bayesian rule can be implemented to solve real problems. In the following sections we describe a useful method which applies the Bayesian rule to real problems [31][28].

6 Address-Block Extraction from Mail Pieces

6.1 Background of Address-Block Extraction

In order to read a mail destination address correctly, it is first necessary to accurately extract the destination address lines. Because the lines of a destination address on Japanese mail can be written in various ways, extracting them is very difficult.

Moreover, the existence of the return address lines and of advertisements on pre-printed mail sometimes makes the extraction much more difficult. In this chapter, we call the area occupied by the destination address lines the “address-block”.

There are several conventional methods for extracting the address-block [4][5]. For example, in the case of handwritten mail, there is a technique for estimating the address-block by analyzing the positional relationship between character lines and ruled lines. When the address-block has ruled lines, this method is very useful, but it essentially has no power to extract an address-block without ruled lines. In the case of large-sized mail, another technique extracts an area having a uniform grey value as an address-block [6]. This technique is effective for extracting an address-block on an attached address label. However, it is not enough to use techniques that can only be applied to specific types of address-blocks.

The proposed method, which will be described later, can be adapted to various address-block types. This method extracts several candidates for the address-block by using multiple hypotheses on types and sorts them according to a confidence value. To calculate the confidence value of a candidate, an evaluation based on a Bayesian rule is executed by using a type-hypothesis. Address-block candidates are read by an address character recognition module one by one until one candidate is accepted as an address; the result is then regarded as the destination address of the mail. This recognition procedure can effectively read the destination address of mail [7].

6.2 Variations of Address-block Types in Japanese Mail

There are various types of address-block in Japanese mail. We have classified address-blocks into several types by human observation. That is, the type of mail envelope is either portrait or landscape, the direction of the written address is either vertical or horizontal, and the address is either printed or handwritten. Note that a landscape mail envelope with vertical writing is very uncommon. The total number of types is therefore six, abbreviated as follows:

P-PV: (Printed, Portrait, Vertical writing)
 P-PH: (Printed, Portrait, Horizontal writing)
 P-LH: (Printed, Landscape, Horizontal writing)
 H-PV: (Handwritten, Portrait, Vertical writing)
 H-PH: (Handwritten, Portrait, Horizontal writing)
 H-LH: (Handwritten, Landscape, Horizontal writing)

The six types of address-blocks are shown in Figure 1. We assume that the mail image is scanned in portrait orientation so that the stamp is always located in the upper left. The resolution of the image is 200 dpi.

The rectangles in Figure 1 show the address-blocks. It is difficult for a simple method to extract the address-block, because its position and area are different in each type (as shown in Figure 1). Since the type of address-block cannot be known in advance, it is not possible to adjust the extraction process for each type. Moreover, return addresses and pre-printed advertisements on the mail sometimes make address extraction even more difficult.

6.3 Mail Destination Address Reading System

To read a mail destination address, two operations must be performed: extraction of the address-block and reading of the address lines in the address-block. The address-block extraction should be a bottom-up approach, because it is performed without the knowledge of the address-block type. The address-block extraction consists of the following four steps:

1. Removal of noise by pre-processing
2. Extraction of objects
3. Creation of address-block candidates
4. Evaluation and sorting of the address-block candidates

Figure 2 shows the flow of these steps. In the pre-processing step, noise and underlines are removed from the input binary image by simple image processing. Next, connected components are derived from the image. All processing after this step is performed using connected components.

In the object extraction step, the objects (a stamp, a postal code, character lines, etc.) that may commonly exist in the mail image are extracted. The extraction uses common knowledge of all address-block types such as knowledge

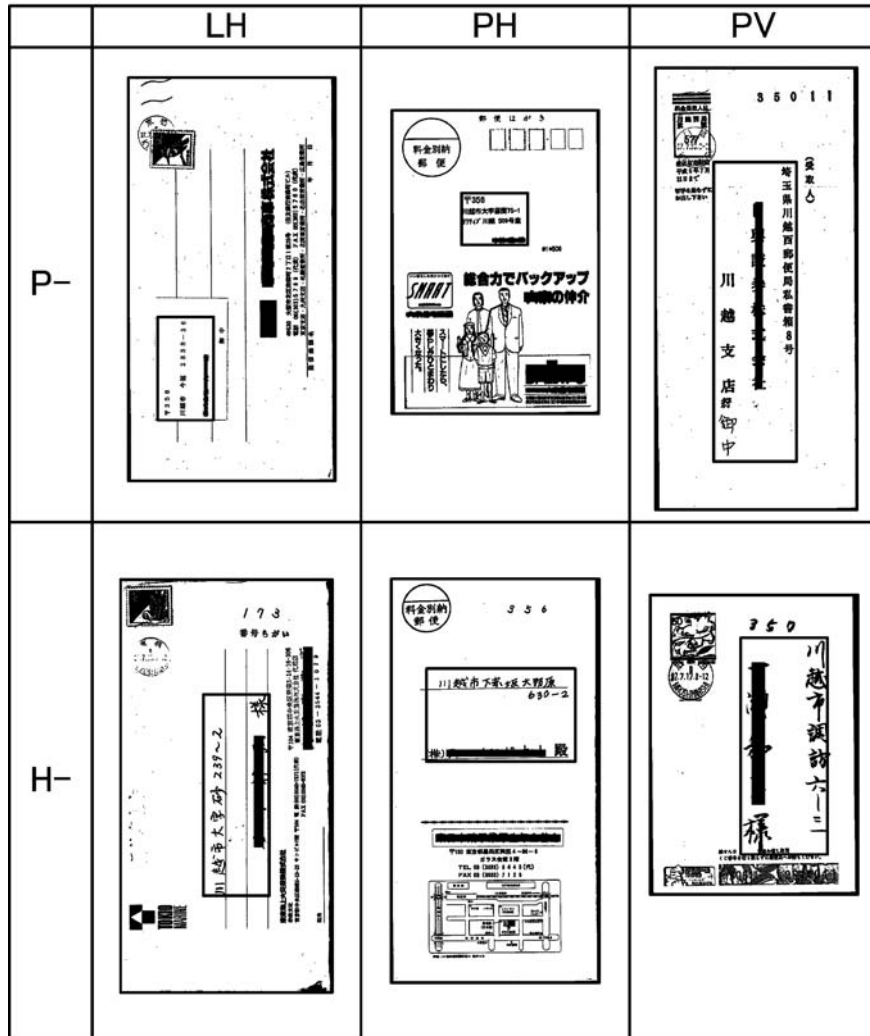


Fig. 1. Address-block Type Variations

of the position and the size of each object. In the address-block candidate creation step, several character lines extracted in the object extraction step are combined, and several candidates for the address-block are generated. This step creates several proper candidates for each mail type, because the position and the area of the correct address-block are different for each type. In the next step, one correct answer will be selected from the candidates even if there is no knowledge of the address-block type.

In the address-block candidate evaluation step, a confidence value expressing the likelihood that a candidate is the correct address-block is calculated

according to a Bayesian rule. The confidence value is calculated by using features detected in the candidate. One candidate will have several confidence values corresponding to all address-block types. The number of confidence values for one candidate is therefore the same as the number of types. Next, the candidates are sorted in order of their maximum confidence values. In the address-lines reading step, the sorted candidates are analyzed one by one, using character recognition. When the recognition result is an address, it can be regarded as the destination address of the mail. Candidate evaluation using a Bayesian rule is described in detail in the next section.

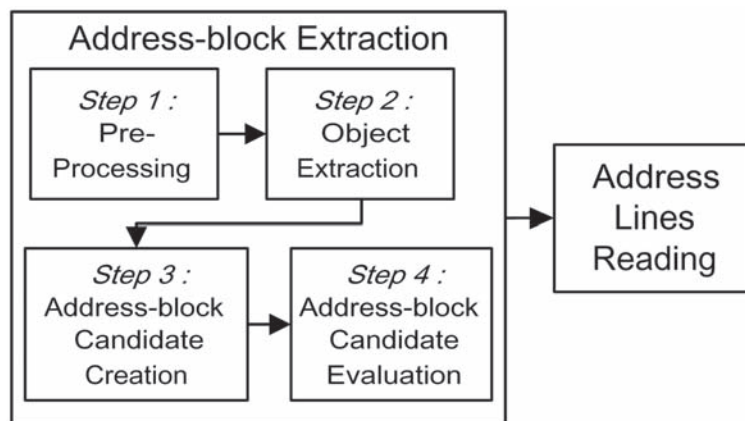


Fig. 2. Mail Destination Address Reading System

6.4 Segmentation of Character Lines by Multiple Hypotheses

Some character lines are extracted by merging some closely connected components of the patterns. These merge rules have different threshold values in response to the layout of the destination address. We call the segmentation setting for printed character the “small” character setting, and the segmentation setting for handwritten characters the “large” character setting. Figure 3 shows the character-line extraction results for each setting. In Figure 3, the printed character lines (Input Image A) are extracted correctly by the small character setting, but incorrectly by the large character setting. The handwritten character lines (Input Image B) are extracted correctly by the large character setting, but incorrectly by the small character setting. The character lines segmentation for all mail pieces can not be processed correctly by one-half of these settings.

The layout of the mail address lines is not known in advance. It is difficult to judge the layout of the destination address, because there also are advertisement and return address areas. Since the layout of the destination

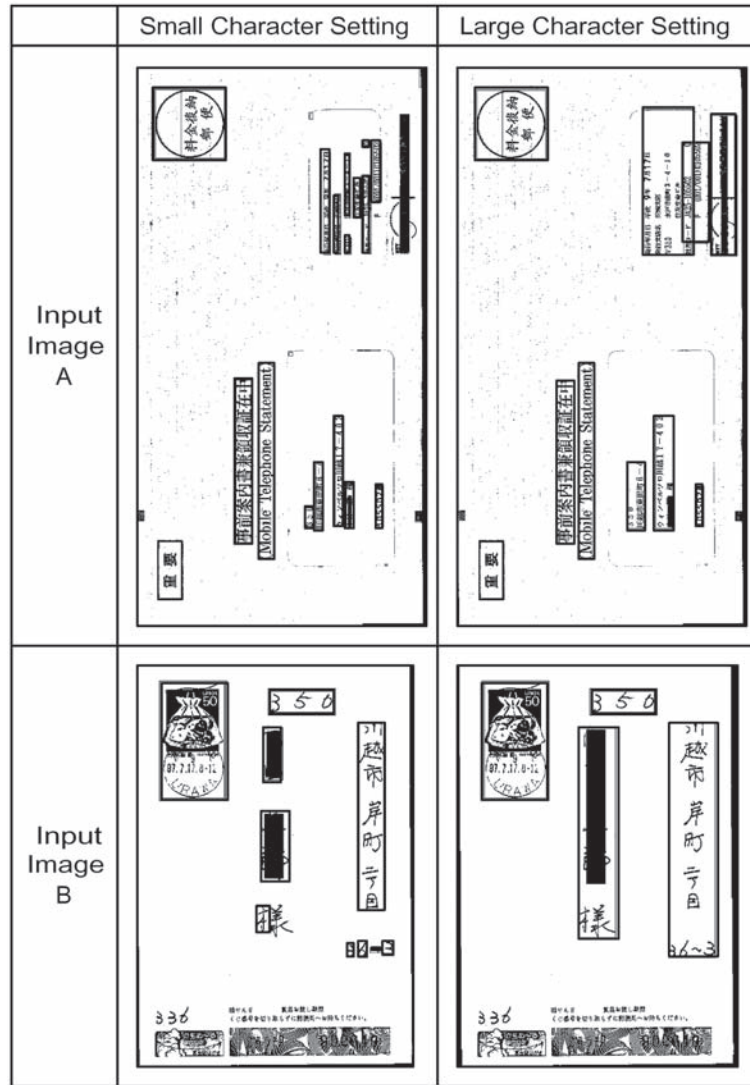


Fig. 3. Character Line Segmentation Results

address can not be decided, the segmentation settings can not be changed in response to the layout. Character lines segmentation is therefore executed twice by both settings, as shown in Figure 4. After the character lines are segmented by the multiple hypotheses, some candidates are outputted to the post-process. The correct character lines segmentation is found among these candidates wherever possible.

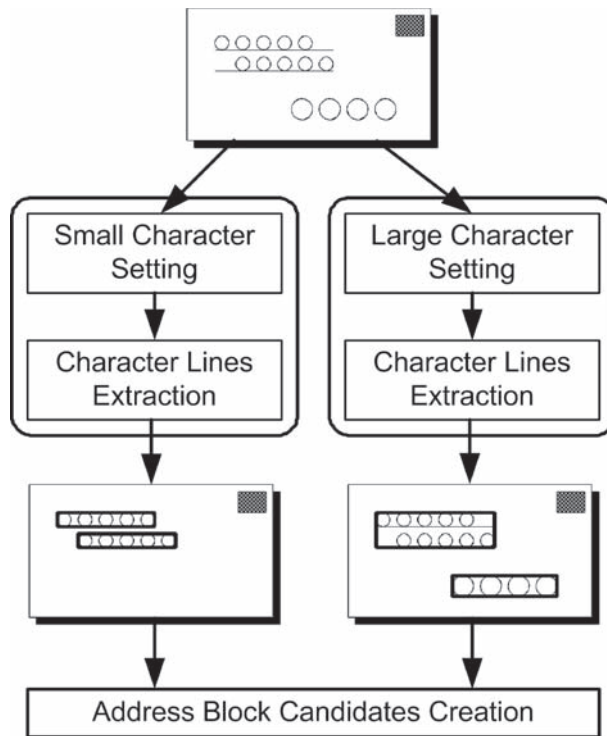


Fig. 4. Character Line Segmentation by Multiple Hypotheses

6.5 Address-Block Evaluation by Bayesian Rule

6.5.1 Calculation of Confidence Value

In the address-block candidate evaluation step, the confidence value is calculated from the following features that can be observed from the candidate.

1. Averages of height and width of character lines
2. Variances of height and width of character lines
3. Area of address-block candidate
4. Position of the candidate

Since the standard values of the features change according to the address-block type, a single set of parameters for the candidate cannot be determined. To cope with this problem, several address-block dictionaries are prepared according to each type, as shown in Figure 5. The optimal confidence values of all candidates in each type can thus be calculated. However, because the address-block type is unknown in advance, a dictionary corresponding to a certain type cannot be selected. Accordingly, an address-block candidate for

each type is assumed, and a confidence value for each type is calculated independently as each dictionary is selected and referenced. Several confidence values per candidate are thus computed and are compared with each other to select the maximum value among them. This maximum value is regarded as a representative confidence value for the candidate, because the confidence value computed from the dictionary of the corresponding type should be the highest and all confidence values for an incorrect candidate should be low.

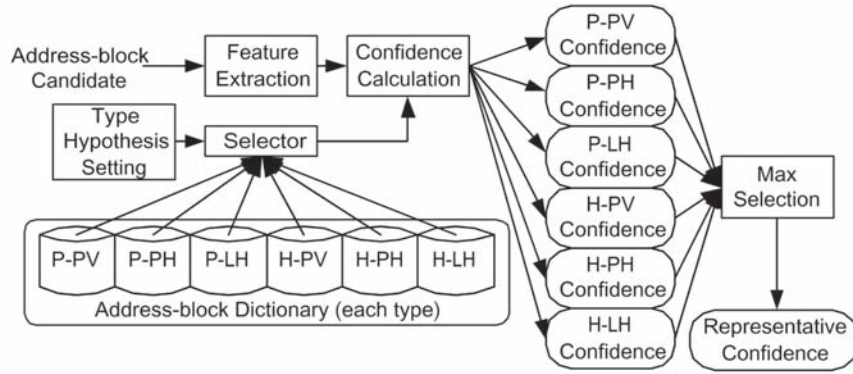


Fig. 5. Address-Block Evaluation

6.5.2 Address-Block Dictionary and Bayesian Rule

A Bayesian rule is applied to calculate the confidence value for the address-block candidate. The confidence value is calculated using the candidate features according to the Bayesian rule given in Eq. (1):

$$P(H_c|e_1, e_2, e_3, \dots, e_n) = \frac{\frac{P(H_c)}{P(\bar{H}_c)} \prod_{k=1}^n L(e_k|H_c)}{1 + \frac{P(H_c)}{P(\bar{H}_c)} \prod_{k=1}^n L(e_k|H_c)} \quad (1)$$

$$\text{Likelihood: } L(e_k|H_c) = \frac{p(e_k|H_c)}{p(e_k|\bar{H}_c)} \quad (2)$$

where c is the address-block type, H_c is the hypothesis of the correct candidate for type c , $e_1, e_2, e_3, \dots, e_n$ are the observed features of a candidate, $P(e_k|H_c)$ is a posterior probability of e_k supposing hypothesis H_c , and $L(e_k|H_c)$ is the likelihood ratio for e_k supposing hypothesis H_c as shown in equation 2.

We define $P(H_c|e_1, e_2, e_3, \dots, e_n)$ as a confidence for a candidate, where $e_1, e_2, e_3, \dots, e_n$ are observed features.

To obtain $P(H_c|e_1, e_2, e_3, \dots, e_n)$, the likelihood ratio $L(e_k|H_c)$ is required in advance. The likelihood ratio is estimated from learning data, and is stored as shown in Figure 5. It is referenced while the confidence values are calculated. The likelihood ratio should also be calculated independently according to each type. The learning data for each type of address-blocks is therefore collected from real mail images in advance.

To estimate the likelihood ratio $L(e_k|H_c)$, $P(e_k|H_c)$ and $P(e_k|\bar{H}_c)$ must be known. $P(e_k|H_c)$ is a posterior probability of e_k supposing hypothesis H_c . We consider that a posterior probability can generally be estimated from the occurrence frequency of e_k in the case of H_c . Thus, to estimate $P(e_k|H_c)$ and $P(e_k|\bar{H}_c)$, in the address-block candidate creation step, many address-block candidates are created from a learning set of mail images. Next, all candidates are judged to be correct or incorrect by human observation. Features of candidates are extracted from the address-block candidates, and the frequencies of features in correct and incorrect candidates are calculated. These frequencies are used to calculate probabilities, $P(e_k|H_c)$ and $P(e_k|\bar{H}_c)$.

Note that $P(H_c)$ and $P(\bar{H}_c)$ are prior probabilities of H_c and \bar{H}_c , and they can be obtained by calculating the ratios of the number of correct and incorrect candidates to the total number of candidates.

6.6 Experimental Results

6.6.1 Evaluation of Address-block Extraction

To evaluate the proposed address-block extraction method, we measured its accuracy rate. Two datasets were prepared: a set of 500 pieces of printed mail and a set of 500 pieces of handwritten mail. We considered 1341 learning samples that included both types of mail. This dataset is different from the evaluation dataset mentioned above. Two methods were compared in terms of their accuracy in the extraction of the address-block:

- (A) Conventional heuristic method
- (B) Proposed Bayesian rule method

In method (A), the parameter set was adjusted heuristically based on the learning data. In the evaluation of both methods, the type of test mail was not known in advance. We compared the cumulative extraction rate for the address-block in mails of the P-LH and H-PV types.

Figure 8 shows the cumulative extraction rates for the two methods. Cumulative extraction rate is calculated from the frequency of candidate rankings, including correct ones. Hereinafter, the cumulative extraction rate from the top to the 5th candidate is called the “top-five correct rate”.

In the case of the P-LH type, the top-one correct rate for method (B) was improved by 16 points up to 79% in comparison with method (A). The top-three, top-four, and top-five correct rates for methods (A) and (B) became equivalent, that is, 95%.

In the case of type H-PV, the top-one correct rate for method (B) was improved 11 points up to 90% in comparison with method (A). The top-five correct rate for (B) is still larger, that is, 91%, than for method (A).

A general experiment with all types of mail confirms that the proposed method can extract the top-five candidates, including the correct address-block, in 94% of printed mail cases and in 89% of handwritten mail cases.

Figure 6 and 7 show the input images of the P-LH and H-PV formats, and the address-block extraction results obtained by this proposed method. The rectangles in these images are the extraction results of the character lines, postal code, and stamp. In Figure 6, the first candidate is the correct answer. The others consist of some small character lines, not the destination address. These incorrect candidates are located at a valid position in the H-PV format, but the confidence value is not large because none of the other information, apart from position, is valid for H-PV format. In Figure 7, the first candidate is the correct answer, and the second candidate is the part of the return address area. It is possible to put the correct candidate in a superior position to all other candidates, if the confidence value is calculated from features of each candidate.

6.7 Evaluation of Address Reading

We measured the read rate of a destination address using the proposed method. The number of test mail samples was 13778 for printed mail and 7265 for handwritten mail. In the destination address reading system mentioned in Section 6.3, the read rates of two systems using methods (A) and (B) respectively were compared.

In the case of the printed mail, the read rates for methods (A) and (B) were equivalent, 84.5%. In the case of the handwritten mail, the read rate for method (B) was improved by 0.61pt to 67.5% in comparison with method (A).

As shown by this evaluation, the read rate improvement using method (B) for handwritten mail is larger than that for printed mail. This is because the parameter set for method (A) was heuristically adjusted for printed mail but not for the handwritten mail. If we had used a parameter set adjusted for handwritten mail, the improvement for handwritten mail would have been larger than that for printed mail. Therefore if it were possible to judge in advance whether the mail is printed or handwritten, an optimal parameter set for method (A) could be selected according to the judgment. However, it is difficult to predict the mail type in advance. Method (B), on the other hand, can cope with both printed and handwritten cases without prior knowledge of types.

6.8 Conclusion of Address-Block Extraction

A method for extracting the address-block from a mail image in various formats was developed and tested. In terms of the address-block extraction

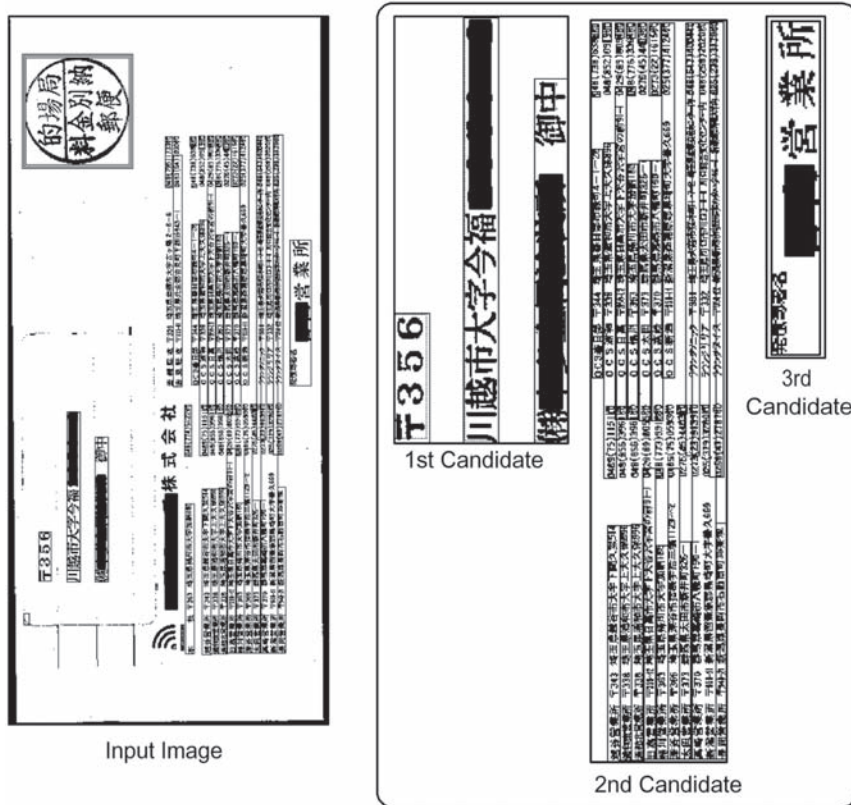


Fig. 6. Address-block Extraction Results (Printed)

accuracy of this method, the top-five correct rate was 94% for printed mail and 89% for handwritten mail. In the case of an address reading system using this method, the read rate for printed mail was 84.5% and that for handwritten mail was 67.5%. It is thus concluded the developed method can be applied to future areas such as signboard extraction and document analysis, if sufficient learning data is collected in advance.

7 Segmentation of Handwritten Kanji Numerals Integrating Peripheral Information

7.1 Background of Character Segmentation

We have developed a new method for segmenting handwritten Kanji numerals written vertically. The segmentation of Kanji numerals written vertically is difficult, because part of one Kanji numeral pattern can be read as another

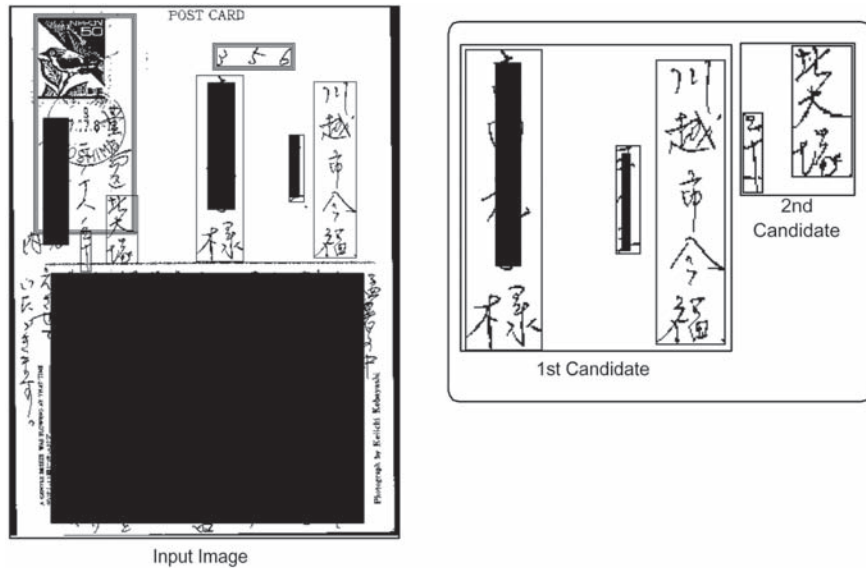


Fig. 7. Address-block Extraction Results (Handwritten)

Kanji numeral character. Figures 9 (a)-(d) show Kanji numerals and hyphens written vertically. In Figure 9 (a), there are three Kanji numerals and two hyphens, showing the Kanji numeral “three”, hyphen, “two”, hyphen, and “one” from top to bottom. The Kanji numeral “one” is expressed by one horizontal stroke, “two” is expressed by two horizontal strokes; and “three” is expressed by three horizontal strokes. Thus, if two horizontal strokes are written in a vertical direction, the pattern is ambiguous; they can be read as a pair of one’s or as a single two.

Three kinds of information can generally be used to segment characters [8].

- (1) peripheral information like size, shape, and relative positions of patterns
- (2) similarity of a candidate character pattern(s) to a valid character
- (3) word matching score

In the case of Kanji numeral segmentation, information types (1) and (2) can be used effectively, but type (3) information can not be used because unlike a string of characters, the numeral sequence has no contextual information. Type (1) information is effective for character segmentation but is not sufficient by itself. Type (2) information is particularly useful for valid segmentation, but it is risky to use it alone because segmentation in some cases is highly ambiguous, as described above.

Our method uses both information types (1) and (2) to resolve the ambiguities. The first problem we faced was the fact that the peripheral features

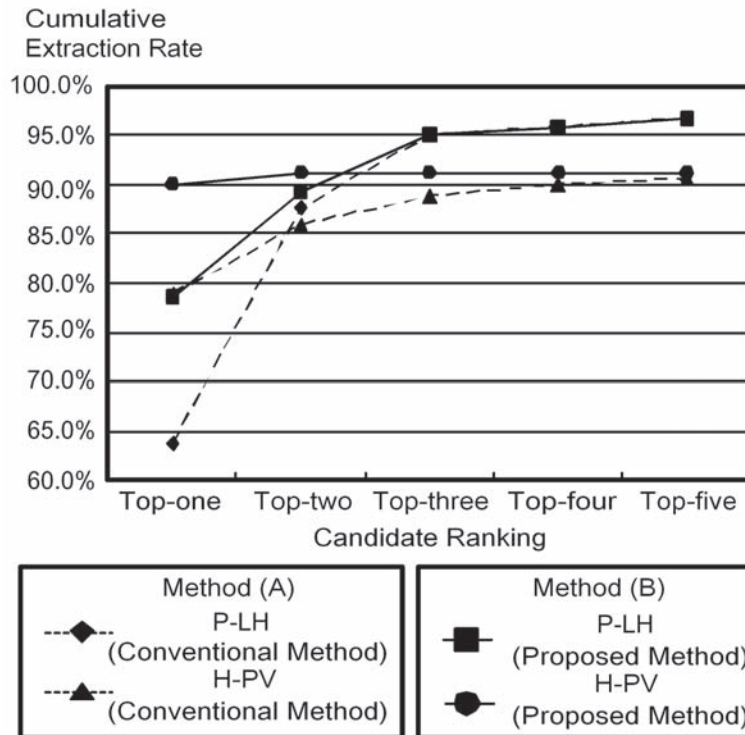
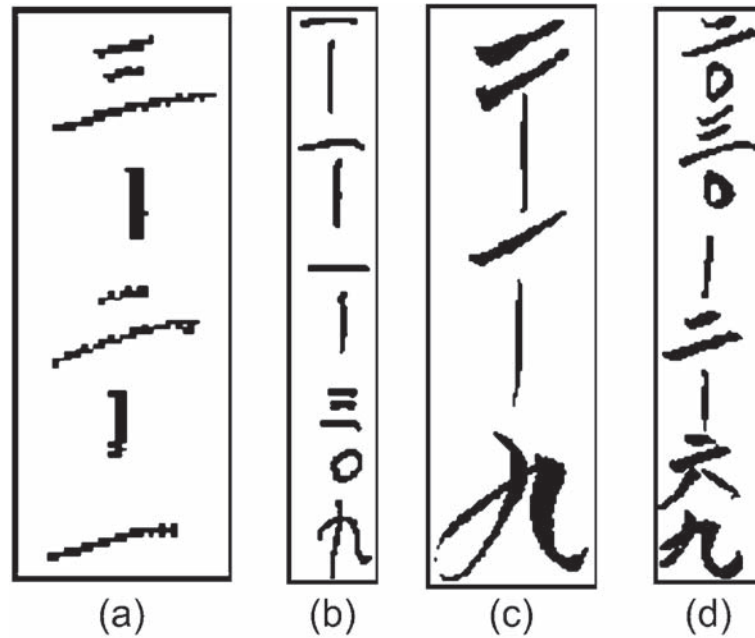


Fig. 8. Address-Block Extraction Accuracy

included in type (1) information, for example width, height, aspect ratio, and gap, do not have the same degree of usefulness for ambiguity resolution. Further, the degree of usefulness depends on the character category. To overcome these problems, we applied the Bayesian rule to obtain confidence values for each potential reading. These confidence values enable us to increase the segmentation accuracy.

7.2 Conventional Character Segmentation Method

An input image is analyzed by a pre-segmentation module that creates multiple candidates of segmented character patterns. In Figure 10, there are pre-segmented patterns of Kanji numerals. The hypotheses of the pre-segmentation are represented in terms of a graph (or a network), and one of the paths from the initial node to the terminal node is the result of segmentation. To select the optimum path among all possible paths, the links between nodes, which represent pre-segmentation candidates, are evaluated. The similarity of each pre-segmented pattern is used in the conventional segmentation method. The valid patterns (links) have a high similarity value, and the invalid



(a) 3-2-1, (b) 1-1-1-309, (c) 2-1-9, (d) 2030-2-69

Fig. 9. Kanji Numerals Written Vertically

patterns (links) have a low value. In Figure 10, one of the pre-segmented patterns is inputted into the character classification module, then the character category and similarity value of 0.9 are obtained. In this case, the similarity is high because the inputted pattern has a valid shape. The evaluation of the links between nodes eventually identifies the optimum path. This search for the optimum path can be done through dynamic programming.

However, many pre-segmented patterns of Kanji numerals could be a part of several Kanji numeral patterns. As shown in Figure 10, the pre-segmented patterns consist of one horizontal line, two horizontal lines, and three horizontal lines. All patterns can be classified as the Kanji numerals, or with a high degree of similarity. Therefore, all evaluation values are very high, so it is difficult to determine the correct path from these pre-segmentation hypotheses in the case of Kanji numerals.

7.3 Information Concerning Character Category and Peripheral Features

7.3.1 Character Category Information

When the peripheral features of Kanji numerals “three”, hyphen, “two”, hyphen, and “one” (Figure 9(a)) are determined, the aspect ratio of the Kanji

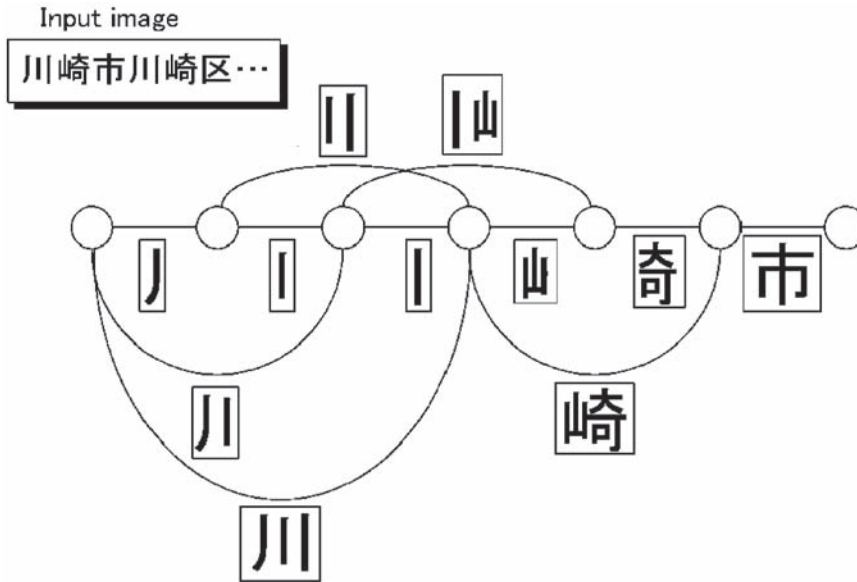


Fig. 10. Character Segmentation by Multiple Hypotheses Method

numeral “three” is about 1.0. In contrast the width of Kanji numeral “one” is much greater than the height. Also, the Kanji numeral “three” consists of three connected components, while the Kanji numeral “one” consists of one. In this way, the valid values of the peripheral features differ depending on the character categories.

The usefulness of a feature also differs depending on the character categories. For example, the usefulness of the spaces in front of and behind the characters is different for Kanji numerals “three” and “one” because the Kanji numeral “one” has the same shape as one part of the Kanji numeral “three”, but a single horizontal line can be distinguished as either “one” or as one part of Kanji numeral “three” by using information concerning the spaces. Therefore, for the Kanji numeral “one”, the information concerning the spaces in front of and behind characters is very important, and the usefulness of this feature is high.

On the other hand, the similarity given by the character classifier is not important and is not useful for Kanji numeral “one” because the similarity values for “one” and the one part of “three” is large in both cases. However, this information is very useful for other character categories.

Thus, we have to determine the usefulness of specific features for each character category, and the degrees of usefulness can then be used to calculate the link evaluation values.

7.3.2 Usefulness of Features

The degree of feature usefulness can be presented as a likelihood ratio that is given by equation 3,

$$L(e_k|H_c) = \frac{p(e_k|H_c)}{p(e_k|\bar{H}_c)} \quad (3)$$

where

c : character category assumed for a candidate

H_c : the positive hypothesis of the link being a pattern of character category c assumed for a candidate

e_k : one of the measured features, or the evidence of the hypothesis

$P(e_k|H_c)$: the probability of e_k supposing event H_c

The likelihood ratio is then evaluated for the assumed category. We computed the conditional probability $P(e_k|H_c)$ in terms of the histogram of e_k when event H_c occurred. To make this histogram, features e_k are extracted from all pre-segmented patterns, then classified into the corresponding character categories, and further divided into two classes, namely the correct and incorrect segmentation classes. So $P(e_k|H_c)$ can then be obtained from the histogram. By using the above formula, likelihood ratio $L(e_k|H_c)$ can be obtained from this $P(e_k|H_c)$.

Figure 11 shows the likelihood ratio graph of a feature extracted from the pre-segmented character patterns. This graph shows the likelihood distribution in the case of the character category of Kanji numeral “one”; the horizontal axis is the feature value of space in front of and behind the character, and the vertical axis is the likelihood ratio. Increasing the space value increases the likelihood ratio, demonstrating that the feature of the space in front of and behind characters is useful.

Thus, after extracting many features from pre-segmented patterns, we selected the most useful features for character segmentation. These features are shown below.

- (A) Pattern height (normalized by line width)
- (B) Pattern width (normalized by line width)
- (C) Pattern aspect ratio
- (D) Spaces in front of and behind characters
- (E) Number of connected components in pattern
- (F) Similarity of character classifier

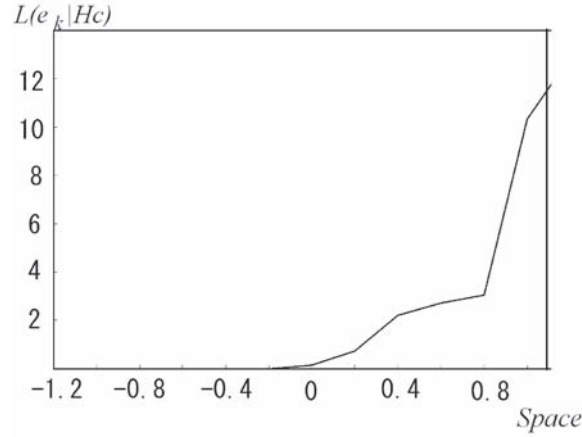


Fig. 11. Likelihood Ratio

7.4 Integrate Likelihood Ratios

The method we used to calculate the confidence value of a link by using the likelihood ratios of features is described below. We define this confidence value as the a posteriori probability $P(H_c|e_1, e_2, e_3, \dots, e_n)$, where $e_1, e_2, e_3, \dots, e_n$ is a list of measured features, or the evidence of the hypothesis. In this chapter, $e_1, e_2, e_3, \dots, e_n$ are the (A)-(F) features listed above. The confidence value can be transformed into a computable formula by applying the Bayesian rule (Bayes theorem), as shown in equation 4.

$$P(H_c|e_1, e_2, e_3, \dots, e_n) = \frac{\frac{P(H_c)}{P(\bar{H}_c)} \prod_{k=1}^n L(e_k|H_c)}{1 + \frac{P(H_c)}{P(\bar{H}_c)} \prod_{k=1}^n L(e_k|H_c)} \quad (4)$$

where

- c : character category assumed for a candidate
- H_c : positive hypothesis of category c given the candidate
- $e_1, e_2, e_3, \dots, e_n$: measured features of the candidate
- $L(e_k|H_c)$: likelihood ratio for e_k for correct category c

The usefulness of each feature is represented by the likelihood ratio. Thus, the calculated confidence value can be used for the link evaluation.

7.5 Implementation

7.5.1 Learning Stage

In the learning stage, many sample images are pre-segmented where candidate character patterns, including over-segmented partial patterns, are extracted. Peripheral features and similarity values are extracted from all of the pre-segmented patterns, and the histogram of these features is computed as described in 7.3.2 The sampled versions of the conditional density functions are then calculated, and the likelihood ratios $L(e_k|H_c)$ are pre-computed in a non-parametric way. The calculated $L(e_k|H_c)$ of each feature is stored and used in the recognition stage.

7.5.2 Recognition Stage

When an image is to be recognized, the pre-segmentation module performs segmentation as described above, and the features for each candidate are gathered. Among the features, there is a category index c given by a character classifier, and the confidence is computed for the assumed category. The computation process is depicted in Figure 12. Selection of the best path is then made using these confidence values.

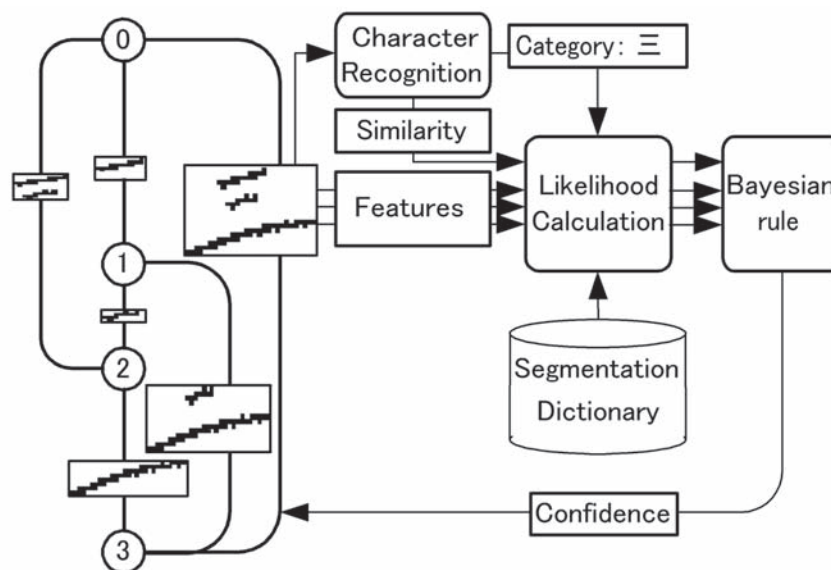


Fig. 12. New Character Segmentation Method for Street Numbers

7.6 Experimental Results

In order to measure the character segmentation accuracy, 208 sample images of character strings which were written by the general public were used for the experiment. There was no noise pattern in this input data. In the learning step 1288 samples of correct character patterns and 5082 samples of incorrect character patterns were used. This learning data set is different from the above 208 sample images. The likelihood ratio was created from the learning data set. In these samples, characters were written vertically with a ball-point pen, a writing brush, or a fiber-tipped pen.

In this experiment, we compared three methods as follows.

M1: Use similarity for confidence

M2: Use similarity and peripheral information uniformly for confidence

M3: Use similarity and peripheral information in response to character category for confidence

Table 1 shows the character segmentation and the recognition accuracy. In this experiment, a correct result for character segmentation means that all characters in the character line were segmented correctly. A correct recognition result means that all characters in the character line were recognized correctly.

Table 1. Accuracy of character segmentation and recognition

	Character segmentation	Recognition
M1	33%	30%
M2	73%	62%
M3	87%	71%

Table 1 shows that the character segmentation accuracy of M3 increased by 54 points compared to M1, and 14 points compared to M2. This means that peripheral information is effective for the character segmentation of Kanji numerals. Changing the importance level of the peripheral information in response to the character category is also effective.

With M3, character recognition accuracy increased by 41 points compared to M1, and 9 points compared to M2. The increment value of character recognition is less than character segmentation, because it is difficult to use linguistic information with the street numbers, despite the correct character segmentation results.

Figure 13 shows one example of segmentation results. In the Figure, M1 segmented each simple pattern as the characters in ImageA, ImageB, and ImageC. This means that each simple pattern has a high similarity value, because each can be read as Kanji numerals. M2 segmented those patterns which are separated by wide gaps from other patterns as shown in Figure 13 ImageB. M3 was able to segment the character patterns correctly, because

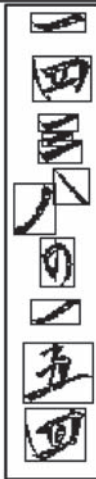


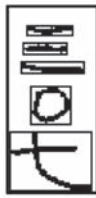
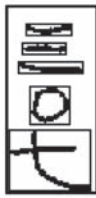
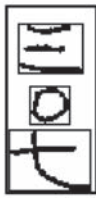

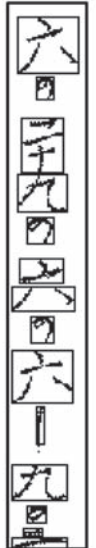
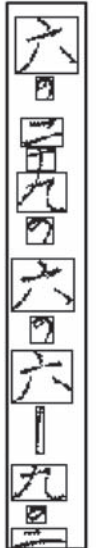
	M1	M2	M3
ImageA			
ImageB			
ImageC			

Fig. 13. Character Segmentation Results

it used information about the size, aspect, and gaps for the confidence value in response to the character category. Especially, ImageC in Figure 13 shows that M3 can solve the difficult character segmentation problem when there are many simple patterns and various sizes.

8 Conclusion

There are many approaches to document analysis, and these approaches can be divided in two types, top-down and bottom-up. Each of these two types has its good and bad points, and the most effective approach is different according to the kind of document in question. The top-down approach is best suited to well-known layouts, and the bottom-up approach is better when the layout is unknown.

OCR systems which read only the ROI (Region Of Interest) in a document can be effectively used for office automation. In this case, there are many methods which adopt the multiple hypotheses approach. If the multiple hypotheses approach is used for an OCR system, it is easy to design the whole system and adjust the balance between recognition accuracy and calculation costs.

In this discussion, a recognition system for mail destination addresses was introduced as a case in which the multiple hypotheses approach was used successfully. We described an effective method which can solve the ambiguity and variation in destination address recognition. Specifically, an address-block extraction method and character segmentation of the street numbers were introduced. These methods create a number of candidates using multiple hypotheses, and evaluate each candidate by a Bayesian rule. These evaluated candidates are then sorted by confidence value, and one correct answer is finally selected.

Future OCR systems will probably include expanded target fields and increased recognition accuracy. This will enable many more tasks in the office to be automated, freeing humans from simple, repetitive labor. Many researchers are currently working on character recognition in real scenes. For example, an embedded OCR was developed for mobile phones. We believe that the most difficult problems facing OCR will gradually be solved, and that eventually OCR systems will be able to recognize whole characters which are written anywhere in the world.

References

1. Kobayashi, Y., Yamada, K., Tsukumo, J.: A Segmentation Method for Hand-Written Japanese Character Lines Based on Transitional Information. Proc. ICPR (1992) 487–491
2. Kamimura, T., et al.: Determining Address Format Using Layout Rules for Reading Japanese Hand-written Mail. (in Japanese) IEICE PRU95-107 (1995) 25–30

3. Yeh, P.S., et al.: Address Location on Envelopes. *Pattern Recognition* (1987) Vol. 20, No. 2, 213–227
4. Wolf, M., et al.: Form-Based Localization of the Destination Address Block on Complex Envelopes. *Proc. ICDAR* (1997) 908–913
5. Wolf, M., et al.: Fast Address Block Location on Handwritten and Machine Printed Mail-piece Images. *Proc. ICDAR* (1997) 753–757
6. Yu, B., et al.: Address Block Location on Complex Mail Pieces. *Proc. ICDAR* (1997) 897–901
7. Sako, H., et al.: On The Rejection Ability Required in Multiple Hypothesis Techniques. *Proc. IWFHR* (2000) 123–132
8. Casey, R.G., et al.: A Survey of Methods and Strategies in Character Segmentation. *IEEE Trans. on PAMI* (1996) Vol. 18, No. 7, 690–706
9. Fujisawa, H., et al.: Segmentation Method for Character Recognition. *Proc. IEEE* (1992) Vol. 80 No. 7, 1079–1092
10. Gyohten, K., Sumiya, T., Babaguchi, N., Kakusho, K., Kitahashi, T.: A Multi-Agent Based Method for Extracting Characters and Character Strings. *IEICE TRNS. INF. & SYST.* (1996) VOL. E79-D, NO. 5
11. Bruckner, T., Suda, P., Block, H.U., Maderlechner, G.: In-house Mail Distribution by Automatic Address and Content Interpretation. *SDAIR96* 67–75
12. Srihari, S.N., Srikantan, G., Hong, T., Grom, B.: A General-Prupose Japanese Optical Character Recognition System. *Proc. Conference on Document Recognition* (1996) SPIE Symposium
13. Cohen, E., Hull, J.J., Srihari, S.: Control Structure for Interpreting Handwritten Addresses. *IEEE TRANS. PAMI.* (1994) VOL. 16, NO. 10, 1049–1055
14. Shinjo, H., Nakashima, K., Koga, M., Marukawa, K., Shima, Y., Hadano, E.: A Method for Connecting Disappeared Junction Patterns on Frame Lines in Form Documents. *Proc. ICDAR* (1997) 667–670
15. Dunn, C.E., Wang, P.S.P.: Character Segmentation Techniques for Handwritten Text - A Survey. *Pattern Recognition Methodology and Systems, Proceedings., 11th IAPR International Conference* (1992) 577–580
16. Tnag, Y.Y., Ma, H., Liu, J., Li, B.F., Xi, D.: Multiresolution Analysis in Extraction of Reference Lines from Documents with Gray Level Background. *IEEE TRANS. PAMI.* (1997) VOL. 19, NO. 8
17. Koga, M., Kagehiro, T., Sako, H., Fujisawa, H.: Segmentation of Japanese Handwritten Characters Using Perioheral Feature Analysis. *Proc. ICPR* (1998) 1137–1141
18. Sawaki, M.: Text-Line Extraction and Character Recognition of Document Headlines With Graphical Designs Using Complementary Similarity Measure. *IEEE TRANS. PAMI.* (1998) VOL. 20, NO. 10
19. Yen, P., Antoy, S., Litcher, A., Rosenfeld, A.: Address Location on Envelopes. *Pattern Recognitoin* (1987) Vol. 20, No. 2, 213–227
20. Srihari, S.N.: Integration of Hand-Written Address Interpretation Technology into the United States Postal Service Remote Computer Reader System. *Proc. ICDAR* (1997) 892–896
21. Palumbo, P.W., Srihari, S.N., Soh, J., Sridhar, R., Demjanenko, V.: Postal Address Block Location in Real Time. *Computer* (1992) 34–42
22. Whichello, A.P., Yan, H.: Locating Address Blocks and Postcodes in Mail-Piece Images. *Proc. ICPR* (1996) 716–720
23. Nakajima, N., Tsuchiya, T., Kamimura, T., Yamada, K.: Analysis of Address Layout on Japanese Handwritten Mail. *Proc. ICPR* (1996) 726–731

24. Liu, C., Koga, M., Fujisawa, H.: Gabor Feature Extraction for Character Recognition: Comparison with Gradient Feature. Proc. ICDAR (2005) Vol. 1, 121–125
25. Koga, M., Mine, R., Kameyama, T., Takahashi, T., Yamazaki, M., Yamaguchi, T.: Camera-based Kanji OCR for Mobile-phones: Practical Issues. Proc. ICDAR (2005) Vol. 2 635–639
26. Liu, C., Mine, R., Koga, M.: Building Compact Classifier for Large Character Set Recognition Using Discriminative Feature Extraction. Proc. ICDAR (2005) Vol. 2 846–850
27. Liu, C., Sako, H.: Class-specific Feature Polynomial Classifier for Pattern Classification and its Application to Handwritten Numeral Recognition. Pattern Recognition (2006) Vol. 39 669–681
28. Kagehiro, T., Koga, M., Sako, H., Fujisawa, H.: Address-Block Extraction by Bayesian Rule. Proc. ICPR (2004) Vol. 2 582–585
29. Ikeda, H., Furukawa, N., Koga, M., Sako, H., Fujisawa, H.: A Context-free Grammar-based Language Model for String Recognition. Int. J. Comput. Proc. Oriental Lang. (2002) Vol. 15, No. 2 149–163
30. Shinjo, H., Hadano, E., Marukawa, K., Shima, Y., Sako, H.: A Recursive Analysis for Form Cell Recognition. Proc. ICDAR (2001) 694–698
31. Kagehiro, T., Koga, M., Sako, H., Fujisawa, H.: Segmentation of Handwritten Kanji Numerals Integrating Peripheral Information by Bayesian Rule. Proc. MVA (1998) 439–442