MVA'98 IAPR Workshop on Machine Vision Applications, Nov. 17-19, 1998, Makuhari, Chiba, Japan

8—15

Extraction of Character Strings from House Maps

Takamasa Simasaki and Toyohide Watanabe*

Department of Information Engineering,
Graduate School of Engineering, Nagoya University

Abstract

In this paper, we propose an experimental extraction method of character strings from house map images, using the block information. Our method consists of two steps: the first is to recognize the block information, and the second is to extract character strings with respect to the recognized block information. In comparison with urban maps, which have often been investigated for extraction subject of character strings, house maps are characterized as (1)utilization of many different kinds of character sets; and (2)illustration with different directions of character strings or characters. Our method is applicable to these features. This paper does not only describe our method, but also evaluates the effectiveness through experiments.

1 Introduction

To extract the meaningful information automatically from map images is one of interesting and necessary subjects with a view to composing the resource data in GIS(Geographic Information System). Until today, many extraction subjects for roads, character strings, etc. have been continuously investigated on various kinds of maps. However, it is not easy to extract the information successfully because map elements are mutually intersected or overlayed.

In this paper, we address an experimental method to extract character strings such as names of house-holders, buildings, etc. from house map images. The extraction issues of character strings, as well as those of roads, have often been attacked[1] [2] [3], but in many cases these approaches were applied to symbol-oriented maps, in which character strings are not main composite elements, such as urban maps, city maps, area maps and so on: these maps represent the geographic constructions more widely than house maps. While, in house maps the character strings are important map elements to indicate individual objects in themselves. However, in comparison with urban maps the character strings in house maps are not straightly rowed, also the sizes and directions are varied individually though the character strings are not almost overlayed. Thus, it is not successful to apply directly the methods, which have been developed to extract character strings from conventionally attacked maps such as urban maps, area maps, road maps and so on, to house maps, which we address here.

With a view to extracting character strings from house map images, our idea is to make use of block information effectively: the block is defined as a region, which denotes house area. Additionally, in our house maps names of house-holders are almost assigned so as to be included in the blocks. Thus, our extraction procedure for names of house-holders first identifies individual blocks, and then distinguishes character strings on the basis of the identified blocks.

2 Framework

Our extraction procedure for names of house-holders, buildings and so on is effective and successful with regard to the blocks, which include the character strings completely: one block attends with one character string. Figure 1 shows such a house map image. In Figure 1, we can observe the following features of character strings: (1)the sizes of characters in different blocks are not always the same; (2)the directions of character strings are individually different; (3)the directions of character strings are almost located along the boundary line segments of rectangular blocks; (4)the sizes of characters are always modified so as to be included adjustably in the corresponding blocks; (5)in many cases the character strings are not touched to or overlayed over the boundary line segments.

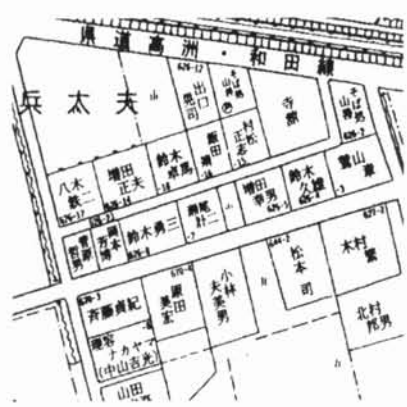


Figure 1: Example of house map image

*Address: Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan.
E-mail: {simasaki,watanabe}@watanabe.nuie.nagoya-u.ac.jp

Thus, it is not difficult to distinguish individual characters if the corresponding blocks should be identified explicitly. Figure 2 illustrates our processing flow in the extraction of character strings: the first is to recognize blocks and the second is to recognize character strings.

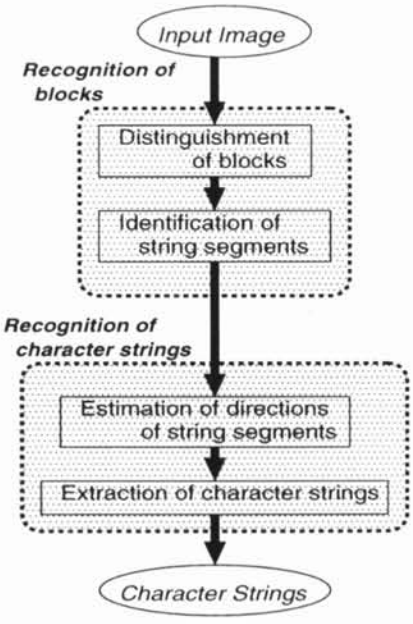


Figure 2: Processing flow

3 Recognition of Blocks

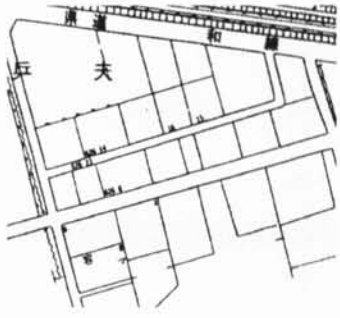
The key idea to extract the character strings is to identify them individually with respect to blocks, estimate the directions of them interpretatively as for the longest boundary line segments of each block and distinguish them correctly as words. Thus, the recognition phase of blocks takes an important role for extraction process of character strings, and is composed of two steps: identification of boundary line segments and identification of blocks. The boundary line segments define continuous foreground pixels which spread widely. This boundary line compose blocks. Thus, we use this boundary line segments to identify blocks.

[Procedure-1: Identification of boundary line pixels]

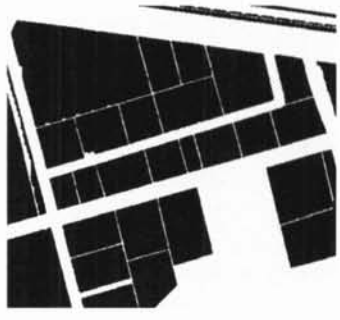
1. Label continuous foreground pixels individually.
2. Draw the minimum rectangular boxes for the labeled pixels.
3. Compute the sizes of drawn rectangular boxes.
4. Erase the continuous pixels, surrounded with the rectangular boxes whose sizes are less than 'a_0'('a_0' is a threshold value), and get the image of boundary line pixels like Figure 3(a).

[Procedure-2: Identification of blocks]

1. Label continuous background pixels of the image of boundary line pixels individually, computed in Procedure-1.
2. Compute the minimum rectangular box, using each continuous background pixel.
3. Select as blocks the continuous background pixels whose boxes have size of more than 'b_0' and less than 'b_1'('b_0, b_1' are threshold values). Figure 3(b) is a result in Procedure-2 from the original house map image in Figure 1.



(a) boundary line pixels



(b) blocks (black area)

Figure 3: Example of result in recognized block

Next, the string segments, included in blocks which were selected in Procedure-2, are identified as areas. This processing is performed according to the identification of character regions, as a preprocessing, in the ordinary character recognition.

4 Recognition of Character Strings

In this phase, individual character strings are extracted correctly according to the directions of strings as we can observe that the directions of character strings are not same in Figure 1. This phase is composed of two steps: the first estimates the directions of individual string segments according to the longest boundary line segments of blocks fundamentally; and the second rotates characters in correct directions, and cuts off character strings from the corresponding blocks, one by one. With respect to this recognition of character strings two following features are assumed.

- a character string is allocated so as not to be turned as little as possible.
- the directions of characters in a character string are consistent to that of character string.

[Procedure-3: Estimation of directions of string segments]

1. Select the longest boundary line segment from the corresponding block for focused character string.
2. Compute the angle 'θ' between the longest boundary line segment and the horizontal line of map image.
3. Estimate the direction 'α' of character (as shown in Figure 4).
4. Rotate individual characters by 'α'.
5. Rotate the character string by 'α'. In this case, the direction of character string is vertical when the longest boundary line segment is located as shown in Figures 4(b) and (c), while that is horizontal when the longest boundary line segment is done as shown in Figures 4(a) and (d).

Next, character strings are extracted after individual characters have been rotated correctly and the character strings have been rowed horizontally or vertically. The procedure is as follows:

[Procedure-4: Extraction of character strings]

1. Compute the minimum rectangular boxes which surround individual characters in string segment.
2. Call the rectangular boxes by $r_i (i = 1...n)$: n is the number of characters in the string segment, included in the corresponding block.

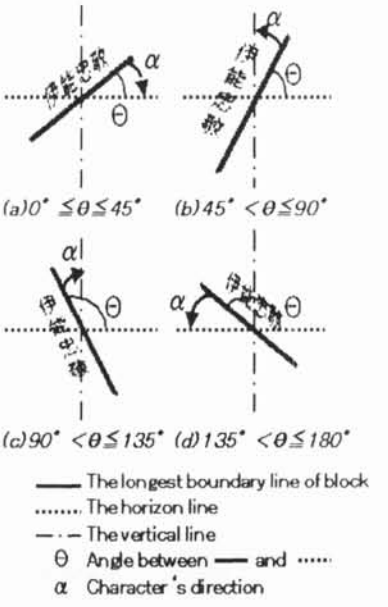


Figure 4: Direction of character

3. Define m as r_i whose ratio for horizontal length to vertical length is $0.9 \sim 1.1$. If there are several r_i's which satisfy such definition, define the average value of such r_i as m.
4. Compute l_h and l_v in the following equations:

$$l_h = \frac{r_{ih}}{m_h}$$

$$l_v = \frac{r_{iv}}{m_v}$$

   Here, r_{ih} and r_{iv} are horizontal length and vertical length for r_i, respectively. Similarly, m_h and m_v are horizontal length and vertical length for m.
5. Classify r_i into three categories in accordance with l_h and l_v.
   (a) Accept: if $0.7 \leq l_h \leq 1.3$ and $0.7 \leq l_v \leq 1.3$
   (b) Reject: if $1.3 < l_h$ or $1.3 < l_v$
   (c) Suspend: otherwise.
6. Extract character where r_i was Accept.
7. Push character whose r_i was Suspend into the integration list L.
8. Pop two different $r_j, r_k (j \neq k)$ from the integration list L, and compose another rectangular box r_l from $r_j, r_k (l \neq j, l \neq k)$. Goto 4.

After then, it is necessary to organize individually distinguished characters as a character string, in accordance with the direction. Figure 5 illustrates the computation of character string. This link procedure is well with respect to the simple rule.

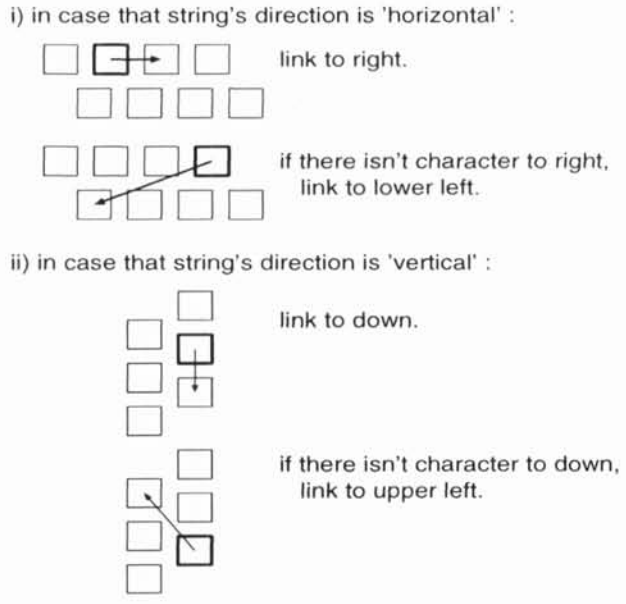


Figure 5: Link for character string

5 Experiment

We show a result of our experiment in Figure 6, which includes several failures*, for original house map image in Figure 1. Also, Table 1 shows the result for five house map images, with respect to the character strings.

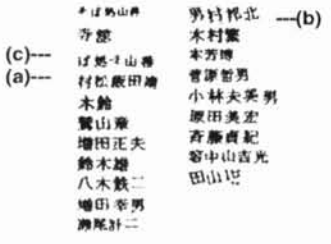


Figure 6: Experimental result from figure 1

*Failure examples – Figure6(a): incorrect recognition of two blocks as one(in Procedure-2). Figure6(b): mistake of the direction vertical for horizon(in Procedure-3). Figure6(c): failure in classification(in Procedure-4)-「そ」is Suspend(can't integrate),「-2」is Accept.

Table 1: Experimental result

	total	success ratio(%)
A	175	---
B	169	(B/A=) 96.6
C	158	(C/B=) 93.5
D	140	(D/B=) 82.8
E	91	(E/A=) 52.0

A: number of character strings in input image.
B: number of string segments.
C: number of string segments in which directions of characters are correctly identified.
D: number of string segments in which directions of character strings are correctly identified.
E: number of properly extracted character strings

6 Conclusion

In this paper, we addressed an experimental method to extract character strings from house map images, using block information. It is clear through some experiments that block information plays an important role to extract character strings from house map images, in which character strings have different directions and sizes.

Of course, our experimental result is not yet sufficient. As our future work, we must improve the extraction of character strings on our method: the longest boundary line segments of blocks are not always deterministic factor for direction of character string. We should grasps totally the direction of character string in block.

Acknowledgments

The authors are very grateful to thank Prof. T.Fukumura of Chukyo University, and Prof. Y.Inagaki and Prof. J.Toriwaki of Nagoya University, and also wish to thank our research members for their cooperations and discussions.

References

[1] O.Shiku et al.: "Character Extraction from Map Image", *Transactions of Information Processing Society of Japan*, Vol. 34, No. 2, pp. 273–280 (1993) [in Japanese].

[2] T.Watanabe and R.Zhang: "Recognition of Character Strings from Color Urban Map Images on the Basis of Validation Mechanism", *Proc. of ICDAR '97*, Vol. 2, pp. 805–808 (1997).

[3] H.Hontani and S.Shimotsuji: "Character String Extraction Based on Multi-Scale Measurement", *Proc. of ICDAR '95*, Vol. 2, pp. 644–647 (1995).