스캔된 지도상의 가옥 추출 방법
House Detection on the Scanned Topographic Map

장항배*, 박종암*, 권영빈**
Hang-Bae Chang, Jong-Am Park and Young-Bin Kwon

요 약 지도에서의 정보추출은 GIS의 구축을 위하여 매우 필요한 사항이다. 본 논문에서는 특수목적으로 제작된 지도내의 가옥을 인식하는 방법을 제안하고자 한다. 가옥을 추출하기 위하여 윤곽선 추출 기법을 사용하였으며 격자에 붙은 가옥을 처리하기 위하여 런 길이 코딩 방법을 사용하였다. 도로와 경계선에 접합되어 있는 가옥을 처리하기 위하여 형태학적 연산을 수행하였으며 형태학적 연산에서 발생하는 오인식을 없애기 위하여 한글 문자부를 제거하는 작업을 수행하였다.

Abstract Extracting information of maps is necessary to establish the GIS. In this paper, a house recognition method on the scanned topographic map is described. A contour detection method is used to extract houses from the scanned maps and RLE (run-length encoding) method is used for manipulating houses touching grid lines. To handle houses touched to roads and borderlines, morphological operation is used. To remove misrecognition occurred by morphological operation, the regions which contain characters on the map are also automatically eliminated.

keyword : map recognition, run-length encoding, morphological operation, 8-connected contour, house detection

1. Introduction

The application of computers tends to expand due to improvement of computers and computer-related devices. Along with the tendency, the request for related software is also being increased. Especially, the demand of automation using computers is rising. Extraction of information out of maps is highly necessary to establish the Geographic Information System(GIS). But, existing maps are often made by hand. There are lots of various symbols, lines, characters on the maps. Thus it's not easy to recognize existing maps. The necessity of extraction of map information and layered structure construction have not been emphasized until recent establishment of nation-wide GIS. Thus, in this paper, software by automatic recognition of map information is implemented.

There are some methods for inputting of geographic information. One is done manually, and the other is an automatic recognition of paper-based maps. The manual process can have fewer errors because people do it. However, an inputting process is a quite routine and difficult work. It requires a quite a long time, too. But the automatic recognition is hard to be used generally because there are maps that have too various shapes of elements[1]. Thus, the automatic recognition technique is usually used for maps with restricted forms.

Lines on maps are represented with set of lines and points, and these elements represent one information according to a target object. Unlike general documents that have distinct target objects, there are overlapping among target objects, because lines that are parts of each target object on maps overlap one another. There are lots of cut-off of lines because symbols and characters are drawn on lines. Those leave many subjects to researchers

* 중앙대학교 컴퓨터공학과 xtel@cvlab.cse.cau.ac.kr japark@cvlab.cse.cau.ac.kr
** 정회원, 중앙대학교 컴퓨터공학과 교수 ybwkon@ripe.chungang.ac.kr

who do map recognition, although there are many researches being done. They are about lines that make part of one object, gathering points and lines efficiently, and the methods for recognizing assembled target objects[2].

There are lots of researches being done in foreign countries, and it's also in Korea. Making of digital maps is on progress at the Department of National Geography, but many parts are still done manually. And the status of current research and development that is based on the result of map recognition is in its early stage. For the road detection, character based detection is proposed by Watanabe[7]. For the point detection, a morphlogical method is proposed by Shieh[8]. For the automatic interpretation of topographic maps, the IGN takes lots of research. From the results of the Deseilligny, hierarchical interpretation on the topographic map is introduced. Using the peeling onions strategy, roads and buildings are detected[9,10]. The ETH uses a template matching for the raster map, but its results were not quiet well[11]. In this paper, a method for recognizing houses on the topographic maps and its vector representation is presented.

2. Map Recognition and Processing

2.1 Characteristics of Target Maps and Objects

To design a map automation system, map organization information is necessary. The maps used in this paper are made for military purpose, and it is 20"x20" with scale ratio of 1/50,000. Figure 1 shows top-left corner of the original map. The illustrated map that is dealt in this paper has tics on its corners, and longitude lines and latitude lines are drawn as lines with grid style shape.

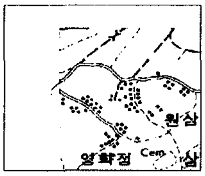


Figure 1. A part of input map

And it doesn't have symbols for crop areas, etc., which is different from general maps. There are many symbols for factories, churches, schools, buildings and houses, and English/Korean characters and numbers. The houses, which are to be extracted, are drawn as black points. What tic means is that line elements which have right angle on them, as you see in the top-left side of Figure 1. The point location where the line is curved becomes the base of coordination. There is almost nothing special with the characteristics of other objects.

2.2 Object Recognition Techniques

In many cases, the houses which are to be extracted can exist as isolated ones. But they can touch grid lines, characters, lines which represent roads, etc., or the houses can be connected one another. Before houses are extracted, removal of lines like grids, and thick lines for characters is required.

Figure 2 shows a procedure for separation of objects from houses in order to extract jointed houses by classifying the houses on maps as non-touched and

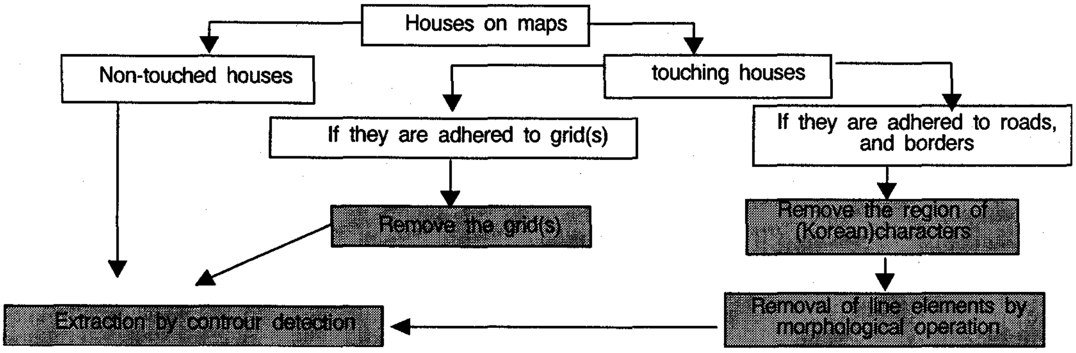

Houses on maps
Non-touched houses
touching houses
If they are adhered to grid(s)
If they are adhered to roads, and borders
Remove the grid(s)
Remove the region of (Korean)characters
Removal of line elements by morphological operation
Extraction by controur detection

Figure 2. The steps for extracting houses

touched ones. The shaded ones are implemented in this paper and will be explained in this section.

2.2.1 Extraction of Line Elements

Line elements are important because they represent critical information on maps, documents, and images. And for dividing regions, extracting main walls of houses, analyzing documents and measuring slopes, the line elements provide very important information[2][3].

Houses that touching other elements are usually connected to grid lines on the map. So, it is required to remove the grid lines. The grids consist in line elements; thus they contain of vertical and horizontal line elements. Eliminating the line elements can separate the touching houses.

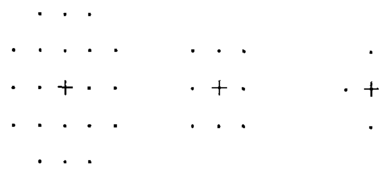
Run-Length Coding is originally used for image compression, and it's done by representing repetitive pixels as one code by counting the occurrence number of the specific pixels.

In this paper, a grid is composed of vertical and horizontal lines. In order to extract straight lines, black pixels may be transformed into white pixels depending on the threshold value. To be transformed, the gray value of a pixel should be greater than the threshold value. If the grid line is eliminated without any condition, it's possible that the houses which should be detected can't be extracted, because the information for the houses is lost when the houses touch the grid very closely. Thus, it's necessary to restore the lost information. As a solution, we use a method which determines whether any elements are contacted to grid lines or not, and if there are, the elements are not deleted while grid lines are being eliminated.

2.2.2 Morphological Operation

Mathematical morphology is to get preferred data by doing mathematical operations on image data and transforming images[4]. The morphological operation is used to remove borderlines, like roads. The two basic morphological operations used are the *erosion* and the *dilation*.

To separate and extract houses that are connected to borderlines very closely, we use a rectangular plate among plates as shown in Figure 3 and do open operation. Open operation is to minimize and eliminate roads and other borderlines, expand the image, and as a result to recover the original image. The result image may be different from the original one, so another expansion is applied to get closer image to the original.



A circle A rectangle A cross

Figure 3. Simple plates

2.2.3 Isolation of Connected Components

Before explaining the connected component, we need to explain a neighbor. A neighbor or neighbors are the pixels around a pixel. There are 8-neighbor pixels around a specific pixel. These 8 pixels are called 8-neighbors. And among these 8-neighbors, we call the up/down and left/right pixels from the center pixel 4-neighbors. Figure 4 shows the neighboring pixels. If there is a pixel Q at one of the 4 neighboring positions of the pixel P, and it has the same color as that of pixel P, then the pixel Q is called, "connected as 4-connection". And if the pixel Q is at one of the 8 neighboring positions of the pixel P, and it has the same color as that of pixel P, it's called, "connected as 8-connection"[3].

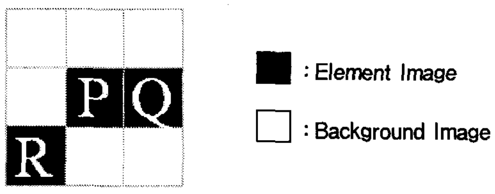
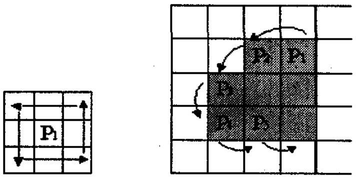


Figure 4. 4-neighbors and 8-neighbors

The contour detection is a differentiation procedure to the target pixels from the background pixels, thus it gets the closed polygons/curves which represent border points. The closed polygons and curves are pixels of 8-connection or 4-connection. The border lines which are got by outline extraction give information, for example, the size of area, the circumference, the shape, and the inclusion relation of connected components.

When contours are extracted, we should check 8-neighbors of a center pixel, and the 8 neighbors are stored in sequence as codes are called chain codes.



(a) CCW tracing (b) tracing procedure

Figure 5. Tracing the 8-connected contour

In this paper, contour detection method is used to get 8-neighbors. The image files are scanned from top to bottom, and if a black pixel is extracted, pixels are checked in 8-neighbors in counter clockwise(CCW) order. The border pixels among white pixels and black pixels are changed to have other predefined color, and as a result, the borderlines are detected[5][6]. The directions of border pixels and x, y coordinates are stored in the linked list, and in the upper linked list we store the minimum and maximum x, y coordinates in the set, and the number of pixels on the outline.

2.2.4 Elimination of Korean Characters

In this paper, the mathematical morphology is used which has erosion and dilation operation. But, before the operations are applied, the thick and large region of characters should be erased, because they can be recognized as houses when they are not removed completely. Usually the thick characters are the names of regions that represented by various sizes. The sizes of characters are similar if they have same size property of font. It's often with English characters that one character has one connected component set, but with Korean characters, it's not. Korean characters tend to have two, three or four connected component sets. Because the Korean characters on maps are usually the names of regions, it's rare that there are 4-connected neighbor set. To extract thick character strings, the contour detection method is used again. But it is necessary to recover the image data located at memory space.

As we mentioned before, it is difficult to decide that the connected component set is a part of Korean characters when we have only the sizes of connected components and the numbers of pixels which consist of contours. Thus, we choose a technique that eliminates Korean characters using the relations among the connected components. The Korean characters on maps can be classified as follows;

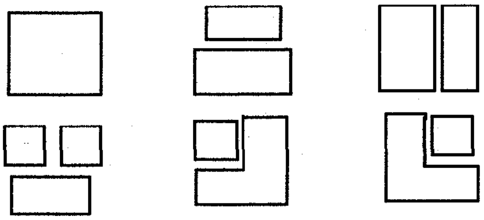


Figure 6. Various style of Korean Character Connection

Using the maps, it is possible to assume that more than one Korean characters are usually in the near place. So, we can first determine whether the elements we are trying to remove are one character or a part of character.

It can be known that the sizes, the distances, and the angles of the connected component sets are different from those of others, by studying the characters among sets that are thought as a part of a character.

So, in this paper, to eliminate the region of Korean characters, we first choose the components set which is assumed as a part of character, and search for 3 connected components which are the closest to the chosen component. The reason we search 3 components is that we should determine the fourth type of Korean character that can be shown in Figure 7. For the 4th type of Korean characters, an inverse triangle can be formed when the center point of each connected components are linked. The other types of Korean characters may found horizontal, vertical and diagonal links or point as shown in Figure 7. But we cannot decide the Korean character itself because the touching of symbols may produce the same component types. Thus if we find out first 4th type of Korean character, the size information can be fitted. Then the similar size component may become a Korean character. After doing that we eliminate the character string by examining the closest distance among element sets and the size of them.

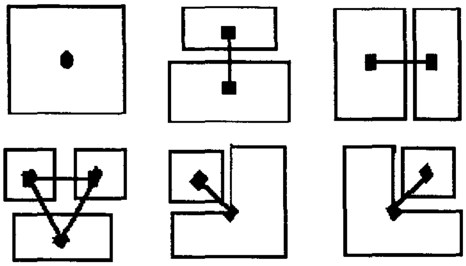


Figure 7. Sizes and distance relations between/among elements

2.2.5 Region Filling

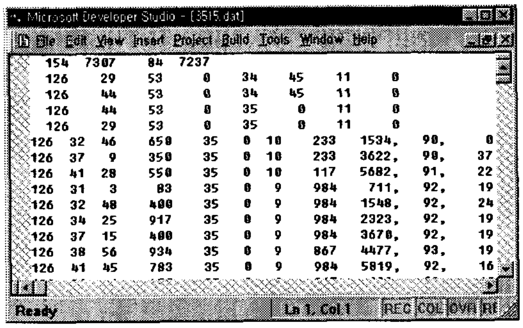
To extract houses from maps, the character strings should be changed to have other color values, which prevent them from being included in the extraction operation. Filling regions means that changing neighboring pixels to a same predefined value. Briefly speaking, the operation fills a region with a predefined value when the 4 neighbors are examined and a pixel that has the same value as that of the center pixel is found.

2.3 Coordinate Transformation

In the result data, the direction and the center latitude and longitude coordinates of extracted houses should be stored. But when jobs were done manually after automatic operation was done, absolute value of x, y were added to the data, because data are not indexed with real latitude and longitude values on the image data.

So, the result data are composed of 3 parts as shown in Figure 8. First two values are the real x, y values for the tic. Second, there are 4 sets of real latitude and longitude values that were given. From the fifth column, the center position of houses as degree, minute, 1/1000 second, the x, y coordinate in the image data, and the direction of houses per 1 degree are produced.

The calculation is done by comparing the latitude and longitude of tic with the absolute coordinate on an image, and the error was maximum 0.13 seconds after the first extraction of contour, but after second trial, it was 0.26. The reason is because the value of pixel is integer, and the value of second is real number, thus the truncation error was came from the procedure of transforming integers to real numbers and real numbers to integers. The direction of houses is got by calculating average value after getting the vector value of bottom/up sides of houses. Because the house is very small, error propagation is somewhat occurred.



Microsoft Developer Studio - [3515.dat]
File Edit View Insert Project Build Tools Window Help
Ready
Ln 1, Col 1
REC COL OVR RE

```
  154  7307    84  7237
  126    29    53     0    34    45    11     0
  126    44    53     0    34    45    11     0
  126    44    53     0    35     0    11     0
  126    29    53     0    35     0    11     0
126  32  46   650    35   0  10   233  1534,  90,    0
126  37   9   350    35   0  10   233  3622,  90,   37
126  41  28   550    35   0  10   117  5682,  91,   22
126  31   3    83    35   0   9   984   711,  92,   19
126  32  48   400    35   0   9   984  1548,  92,   24
126  34  25   917    35   0   9   984  2323,  92,   19
126  37  15   400    35   0   9   984  3670,  92,   19
126  38  56   934    35   0   9   867  4477,  93,   19
126  41  45   783    35   0   9   984  5819,  92,   16
```

Figure 8. An example of house detection result

Figure 9 shows the scanned 400-dpi image that was magnified and means a house. The direction of houses is in-between 0 ° ~ 90 ° , the base point which is required to calculate vector values are got using chain code. After getting 4 corner positions, we get two angles like Figure 9, and the average of the two angles is used as the direction of the given house.

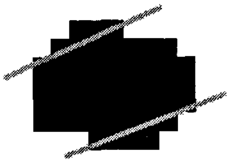


Figure 9. House zoom-in and its direction

3. Recognition Result and Discussions

The program was written on a Pentium 166Mhz PC, MS-Windows 95 and the Visual C++ was used. The data for the experiment is 1/50000 scale map for military purpose by the Department of National Geography. The data is a 400-dpi TIFF image. The amount of data for the map is about 8 Mbytes, and 64MByets of memory space is required to manipulate the raw data, because transforming bit to byte is required.

The recognition result, shown in Table 1 is about a map that has grids and moderately touched houses. It shows that the recognition ratio is flat while the operation is going on. But the purpose of this paper is not for full automation, but for reducing the manual jobs. The extraction of houses by hand needs more complicated procedure than elimination of mis-recognized result. Thus, in this paper, we have in mind how to minimize the number of non-recognized houses. So, although the result shows that the recognition ratio is not getting better after some stage, it can give better efficiency than when it is fully done by hand.

Table 1. Recognition Result

	Isolated houses only		After removing of grid		After character removal operation		After morphological operation	
	number	Recog. (%)	Number	Recog.(%)	Number	Recog.(%)	Number	Recog.(%)
Correct	1,148	74.93	1,271	81.74	1,370	84.99	1,384	85.43
Not recog.	377	24.61	254	16.33	155	9.62	141	8.70
Wrong	7	0.46	30	1.93	87	5.39	95	5.87
Sum	1,532	100	1,555	100	1,612	100	1,620	100

Because the quality of image affects the size of houses very much, with images which have enough quality, it gives good result. But with ones which have low quality, the result can get worse. So, it is necessary to add algorithm that changes the threshold value for the size of houses automatically.

4. Conclusion

This paper is dealt with the house detection on maps to automate the inputting process.

In this paper, a detection method using 8-direction chain code is used, and to extract information about components that are connected to others, the run-length encoding is implemented. In addition to them, by combining the closest neighbors, the regions of Korean characters are extracted, and the thin lines are removed by the morphological operation.

As you see in the result, the success of recognition is akin to the quality of images, thus it is required to add an algorithm that changes the threshold value automatically by calculating the sizes of houses whenever new maps are scanned. By doing so, it can be more robust.

The main interest of this paper is detection of houses, so others are not differentiated. But houses, characters, roads, borderlines of administrative district, and islands, etc. are required to be saved as layered vector data. And with that data, the map should be reconstructed. It is also necessary to build DB of characters on maps using character recognition method.

Because the content and the scale of maps are different according to the purpose of maps, it is difficult to compare this paper with other existing map recognition methods. And unlike other recognition trial, there are lots of complex elements, for example, symbols, characters, and lines. Thus, there is a limitation for providing general map recognition only using the result of this paper. But it is thought that we will be able to get better result if we are successful in getting more interests and better supports.

5. References

[1] R. Kasturi et al., "Map Data Processing in Geographic Information System", IEEE Computer, Dec. 1989, pp.13-21.

[2] G. K. Myers, et al., "Verification-Based Approach for Automated Text and Feature Extraction from Raster-Scanned Maps", 1st GREC, 1995, pp.190-203.

[3] G. Maderlechner and H. Mayer, "Conversion of High Level Information from Scanned Maps into Geographic Information Systems", 3rd ICDAR, 1995, pp.253-256.

[4] I. Pitas, *Digital Image Processing Algorithm*, Prentice Hall, 1993.

[5] H. Bassmann and P. W. Besslich, Ad Oculus: *Digital Image Processing*, International Thomson Publishing, 1995.

[6] Min-Ki Kim, et al., "Automatic Region Labeling of the Layered Map", 1st GREC, 1995, pp.179-189.

[7] M. Nishijima and T. Watanabe, "An Automatic Extraction Approach of Road Information on the Basis

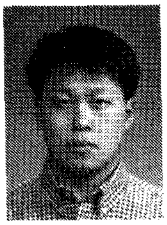
of Recognition of Character Regions", In Lecture Notes on Computer Science(Springer-Verlag), Vol. 1024, 1995, pp. 173-180.

[8] J. S. Shieh, "Recursive Morphological Sieve Method for Searching Pictorial Point Symbols on Maps", 3rd ICDAR, 1995, pp. 931-935.

[9] M. P. Deseilligny et al., "Topographic Maps Automatic Interpretation : Some Proposed Strategies", In Lecture Notes on Computer Science(Springer-Verlag), Vol. 1389, 1998, pp. 175-189.

[10] F. Dupont et al., "Automatic Interpretation of Scanned Maps : Reconstruction of Contour Lines", In Lecture Notes on Computer Science(Springer -Verlag), Vol. 1389, 1998, pp. 194-206.

[11] S. Frischknecht and E. Kanani, "Automatic Interpretation of Scanned Topographic Maps : A Raster-Based Approach", In Lecture Notes on Computer Science(Springer-Verlag), Vol. 1389, 1998, pp. 207-220.



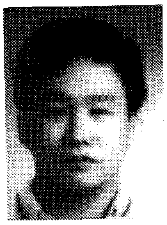
장항배

1997년 중앙대학교 컴퓨터 공학과 졸업 (공학사)

1999년 중앙대학교 컴퓨터 공학과 졸업 (공학 석사)

1999년 현재 중앙대학교 컴퓨터공학과 박사과정 재학중

관심분야 : 패턴인식, 내용기반검색, 컴퓨터 비젼



박종암

1998년 중앙대학교 컴퓨터 공학과 졸업 (공학사)

1999년 현재 중앙대학교 컴퓨터 공학과 석사 과정 재학중

관심분야 : 컴퓨터 비젼, 지도 인식, 의용 영상처리 등

권영빈

1978년 아주대학교 졸업 (공학사, 전교수석졸업)

1981년 한국과학기술원 졸업 (공학 석사)

1986년 프랑스파리 국립고등통신대학원 졸업 (ENST, 공학박사)

1986년~현재 중앙대학교 컴퓨터공학과 교수

1990년~1999년 인공지능 연구센터 운영위원

1996년~1998년 한국정보과학회 이사

1996년~현재 한국인지과학회 이사, 감사

1996년~현재 국제 패턴인식학회(IAPR) 이사

관심분야 : 컴퓨터 비젼, 패턴인식, 한글인식, 지도처리