Automatic Taiwanese Municipal Color Map Reading System

Chung-Lin Huang and Yun-Wei Lin

Electrical Engineering Department

National Tsing-Hua University,

Hsin-Chu, Taiwan, ROC

e-mail:clhuang@ee.nthu.edu.tw

Abstract

This paper introduces a map reading system to make fully automatic interpretation of the municipal maps of Taiwan. Here we use a mixed processing algorithm (bottom-up and top-down approaches) to interpret the maps. The bottom-up approach includes the color segregation, symbol recognition, road extraction, and related characters string extraction. The top-down approach guided by a priori knowledge resolves the inconsistency between the extracted objects. In the experiments, we illustrate that our system can interpret the municipal map effectively.

1. Introduction

Recently, automatic map reading has become an important topic in document analysis researches. The subject on map recognition can be mainly divided into several research domains: symbol recognition, road extraction, and character strings identification. Map symbols are difficult to extract. They consist of alphanumeric and various graphic signs. Symbols are positioned arbitrarily and placed with various orientations or sizes on a map. Structure analysis and pattern matching are two basic methods for symbol recognition. The first approach [1] extracts the features of a symbol within the candidates and compares with the features of the symbols to be found. The second approach [2] may be influenced by symbol sizes, orientations and touching noises.

Road extraction is another important task in map reading. Morphological operations are widely used to find the roads. Roads consist of linear features in a map. To extract roads, some researches focus on finding linear objects [3,4] but not on the semantic meaning of a single road. It is especially important for the recognition process of city map rather than the topographical maps. Interactive method can also be used to extract roads and it will improve the accuracy of results. Character string's non-fixed-sized appearance and various directions in the map have created some difficulties for map- readers. Template matching [5] and features points [6] are used to recognize characters.

Many researchers have proposed systems to solve automatic map reading problems [7,8,9]. Their researches focus on their specific domain and proposed many ad hoc ways to solve the map reading problems. We develop a universal system for city map reading. These city maps mainly include information of streets, buildings, and some bureau of government and they are all color-printed. Different colors represent different object categories. Different graphic appearance of symbol means different semantic meaning. Symbol sizes are within a limited scale and printed in the direction of longitude lines. Characters describing an object on the map are arranged in a fixed order.

2. System Architecture

There are many literatures that offer many different approaches to the document processing and interpretation. They can be summarized as two different approaches: data-driven (or bottom-up) and mixed (both bottom-up and top-down) approaches. In the first approach, we start with the low-level analysis of the gray-level or colored images, in which the features are extracted. Generally, these features are grouped together as line segments, symbols, and characters. Associations between these features are detected, and high level graphic entities are constructed, guided by *a priori* knowledge. The main difficulty in our approach is how to obtain significant graphical entities from low-level operators and reliable connections rules among these features in order to have a correct interpretation. In this paper, we are interested in the second approach

We use four-level representations to interpret the color municipal maps (Figure 1): (1) pixel level, (2) vector level, (3) blocks, roads, symbols, character string level, (4) semantic levels. There are two-way inter-relationships (bottom-up and top-down) between two adjacent levels. The top-down approach of the system is based on the so-called "modeling" which is required by an observer to recognize an object with a mental representation of it as *a priori*. To carry out this "modeling", we start from the principal that the entire graphic document relates to a specific organization of all the graphic primitives (features), and it is based on some grouping rules for these graphic entities for object representation. The system administrates the following operations: (a) Building the connections among pixels to form the line vectors and characters. (b) Grouping the line vectors into roads, characters into character strings, and patterns into symbols. (c) Recognizing the symbols and character strings. (d) Finding the semantic links between roads and character strings, or between the symbols and the character strings.

The municipal map is an association of many graphical primitives that are used to build municipal objects. These four kinds of objects in the third level are: (1) roads, (2) blocks that include buildings, schools, and temples, (3) symbols, and (4) character strings. Associated to each kind of object, knowledge can be introduced to construct the object representation and verify if each object is consistent with the other object. For instance, a road is composed of line segments, these line segments are collinear, within a certain distance, and associated to a character strings (the name of the roads). The character string is located on the strip of the road. A building, market, school, or temple is described in terms of blocks. Each block is associated with different color indicating different attributes.

Each municipal map object is made up of the primitives that are locally coherent in some systematical ways. There are two stages in the bottom-up approach: extraction stage and object formation stage. There is only one stage in the top-down approach that is a consistency enforcement stage. The first stage includes the color segregation, grass-land separation, orientation finding. The second stage includes the symbol extraction and recognition, character string extraction, and road extraction. The third stage takes into account the knowledge of each object constructed for intra-object(internal) and inter-object(external) consistency verification and enforcement. For each object, we need to verify its internal consistency and external consistency

At the end of the bottom-up processing, we have a list of consistent and inconsistent objects (correctly or incorrectly interpreted objects). The inconsistencies among objects are examined after the bottom-up feature extraction. Because of the overlapping characteristics of the objects in different levels and the interference between different feature extractors, sometimes, we need user's involvement to cure these inconsistencies. The system may still need user's awareness of its incapacity to solve the problem and ask the user's intervention.

3. The Bottom-up Approach

Each municipal map consists of objects in different levels. In each level, an object is composed of some primitives in the lower level. The bottom-up (data-driven) approach relates to the processing sequence that uses the reliable association rules among lower-level graphic primitives to have a correct interpretation for the higher-level object.

3.1 Color Segregation

In the street maps, each color is used to represent a special kind of information. Different color layers indicate different denotations on the map. Red means the buildings, flesh tints denotes the blocks, green shows the grass plots or parks, yellow illustrates the artery, and blue indicates the streams. To separate each color layer from the original scanned image, we develop a color segregation method, which consists of sharpening, smoothing and splitting.

3.2 Grass-land Image Subtraction

In our maps, the grassland is in green color mixed with dotted black points. These dotted black points create some noise in the black pixel layer processing. It will influence the characters and symbols extraction processes. To remove these noises, we separate the green layer (generated by the K-means color separation) from the color map image. The green layer image has many small holes. To process the green layer image, we use the dilation operation to fill the holes, and then subtract the green layer image from the black pixel layer image.

3.3 Map Orientation Finding

The orientation of a scanned map image is very a important information for characters and symbols extraction and recognition. By finding some line indexes, such as longitudes and latitudes, on a map, we can determine its orientation These lines are straight, running across the entire map, and equidistant. Here, we apply a modified Hough transform to find these straight lines.

3.4 Symbol Extraction and Recognition

Symbols and character strings are important information on the map image. Symbols help map- readers to easily understand the types of objects and their locations, whereas character strings describe the symbols' real meaning on the map. Once the symbols are recognized, we can extract the character strings, and their relative locations on the map. Different kinds of symbols represent different meanings. The symbols used in the city map are shown in Fig. 2. We use model-based matching to find the symbols. The "models" are chosen from the black layer image. Symbols in our map image are very small so that we use Hausdroff distance measurement [11] to compare the symbol image with the pre-stored symbol models. Haudroff distance measure is a nonlinear operator. It measures the extend to which each point of a "model" set lies some point of an "image" set and vice versa.

3.5 The Symbol-Related Character Strings Extraction

With the extracted symbols, we can find the character strings that describe the meaning of the symbols. The character strings can be recognized by OCR software, if they are precisely extracted from the map image. Each symbol-related character string is located very close to the corresponding symbol and aligned in a certain direction (the relationship described in Fig. 3). We can trace the characters (in a certain fixed direction) on the map image. The origin point of Fig. 3 is the central location of a symbol. Gray regions show the possible area for the characters to be located near the symbol. We can locate the symbol-related characters by calculating and investigating the black pixel density around the symbol.

3.6 Roads Extraction

Road data is one of the most important information on the map. The road extraction depends on the skeleton of roads. The connectivity of skeleton lines, the junction, and the noises from unrelated objects through the thinning process may degrade the results of the road extraction. To extract roads, we need to identify the location of roads first and then find the related character strings. Here, we make the following assumptions:

1) Noises from the symbol residue must be removed before applying the road extraction procedure.

2) White and yellow are the specific colors used to represent roads in the maps.

3) The character strings related to the roads are normally located on the road strip.

The road extraction process consists of the following five steps: (1) Color layer extraction. (2) Removing gaps. (3) Finding the skeleton of roads. (4) Finding junctions. (5) Tracing roads.

3.7 Extraction of Road-Related Character String

Here, we define a window of which the size is a little bit larger than the character size and then move this window along the road on the map. We assume that the road sections are straight on the road map. This assumption may not be correct when we deal with a curved road, however, it can be approximated by a sequence of straight road segments which are piece-wisely connected. During the road tracing, we move the window and calculate the black-pixel density. If the black-pixel density inside the window is larger than certain threshold, then there exists a character. Finally, we find that the characters belonging to a single road are found along the road tracing (see Fig 4).

4. The Consistency Enforcement

To interpret the municipal maps, the features extracted by the low-level operator are grouped by the construction of document objects. Here, we use a hierarchical information representation to describe the extracted knowledge. At each level of the bottom-up approach, each object being constructed receives a list of information and attributes. At each stage, each object is analyzed in order to determine whether each object is consistent with other object or not. Similar to [10], we define two kinds of consistency.

4.1 Internal and local consistency

An object is characterized by an internal coherence if most of its components are present. The notion of internal consistency is found at each level of reconstruction of the municipal map objects, i.e., the vector level, the character/symbol level, and semantic level. For instance, in the extraction process of the related character strings in semantic level, there is inconsistency in find character strings. Fig 5 shows that the original "台中十一信" is extracted as "台中十". The reason is that the character "一" is not identified and it has been deleted in the process of the black pixel density thresholding. This consistency is known as internal, since it is only concerned with the considered entity while ignoring the information of neighboring entities. The internal consistency remedy process will find a gap between "十"and "信". The undetected character "一" is back-traced and identified as a character. The second example is the road tracing that can be performed successfully under the internal consistency enforcement based on the slope consistency of the road segments. The road consists of road segments. For a straight road, the road segments are collinear, whereas for a curved road, the slopes of the road segments are coherent.

4.2 External and global consistency

The second type of consistency is defined as external or global consistency that takes into account the neighborhood of the designated objects. If an object has all its components extracted, and it responds to the consistent semantic connection of the considered layer correctly, then it has satisfied the requirement of internal consistency. However, if all the objects adjoining it are all internally consistent, the object will become more reliable through the construction of the externally consistency verification. In black pixel density image, there may be many black strips near a symbol. To distinguish the nearest black strips from others, we have to find the nearest character contained in a black strip. This is based on the assumption that a symbol and its nearest character string are related. In Fig 6(a), we may see that " 西 區 農 會 " is the nearest character string to the symbol "⊖", and " 會 " is the nearest single character to the symbol. The black pixel density image is shown as Fig. 6(b). The character string " 西 區 農 會 " becomes a black strip in black pixel density.

The other example of the symbol-identification error correction is also illustrated. For example, "✱" and "卍" are too similar to be differentiated, however, we may increase the threshold of matching rate to cope with this ambiguity. To minimize the trade-off between recognition rate and miss-recognition, we need to apply the global consistency verification. The extracted symbol-related character strings can be interpreted by our OCR subsystem. Then, by initiating the external consistency verification of the symbols and their neighboring character strings, we may increase the confidence of the recognized symbol.

In some cases, we find that some character strings are overlapped with the background line segments. The incomplete extracted characters are unrecognizable. Therefore, we need to identify the overlapped portion of the characters and the line segments, and then remove the characters from the character string. The incomplete characters can not be identified by OCR software, however, with the other identified characters, the unrecognizable characters can be interpreted by a priori knowledge.

5. Experimental Results

Here, we demonstrate that our map reading system that can effectively recognize Taiwan municipal color maps digitized by 600-dpi resolution scanner (see Fig. 7). We evaluate our experiment results by using the following criteria:

1) **Object extraction evaluation.** The evaluation of the object extraction concerns the color segregation, symbol detection, character string extraction and recognition, road extraction, building extraction and so on. For instance, the character overlapped with the road line segments may deteriorate the character extraction accuracy. Another factor causing object extraction failure is the layer miss-classification. These errors come from particular instance on the drawing. It may also come from the constraints imposed to the feature extractors.

2) **Semantic level evaluation.** After evaluating the low-level operators, we compute, in the semantic level, the number of consistent and inconsistent objects at the end of consistency verification process. To measure the improvements due to the consistency enforcement operation, we compute the number of originally inconsistent objects that become consistent after the analysis.

The low-level color extraction process using K-means algorithm has a good performance in separating colors as shown in Fig. 8. Each color is separated successfully except white and black colors. Fig. 9 shows the recognized symbols. We can see that the symbols can be successfully extracted by using the Hausdroff distance measurement between the symbol model and the image. Fig 10 also shows that some character strings can not be extracted accurately since the characters can not be extracted clearly because of the unsuccessful pre-processing that have some noise left on the characters. By applying the consistency enforcement process, we have some successful character string examples shown in Fig. 11. These strings are not corrupted by the background noises and have no ambiguous characters such as " — ".

The extraction of roads is influenced by noises introduced during the thinning process or by the discontinuity of the skeletons. From Fig. 12(a), we can find that the thinning process may influence the correct identification of the junctions. The road tracing may not extract a complete road-related character string (Fig. 12(b)). However, the local consistency enforcement process may remedy this kind of problem. Since tracing a straight road is easier, the extracted roads and the road-related character strings are accurate (Fig. 12(c)). In Fig. 12(b), we can see that the direction of extracted character string " 大 墩 九 街 " is not accurately extracted either. Since the curved road tracing is not reliable, the following character extraction may not be correct. Road-related characters extraction makes a good performance in processing the map if the characters are printed in the middle of roads. A curved road can be detected after having identified a cross-junction.

It takes about 20 to 30 minutes to process a map with the size of 600×600 square pixel. The quality of the system also relies on the capacities of warning a human operator when an ambiguity about an object is detected. Human correction is time-consuming but necessary. Our system demonstrates a good performance in the extraction of map information. We have developed simple but robust algorithms to achieve our goals. In order to measure the performance of our system which include the extracted objects and the remedy processes after the internal/external consistency enforcement are summarized in Table 1. The results are encouraging since the remedy processes increased the percentage of objects detected as consistent from 70% to 90%. These results demonstrate the value of the consistency analysis.

6. Conclusion

Our contribution includes the construction of reliable low-level tool to extract hierarchical primitive objects from raw image and interpret the interrelation between the objects in different levels. However, the system needs improvement in several aspects. First, we need to generalize the object extraction bottom-up processing system to obtain more reliable objects. Second, we need to add more reasoning processes to remedy the unsuccessfully extracted objects.

Reference

1. K. S. Choi, D. Ahn, S. H. Lee, and Y. H. Kong, "Automatic Recognition of Map Symbols for GIS Input Module," 1997 IEEE Pacific Rim Conf. Communication, Computer, and Signal Processing.

2. J. S. Shieh, "recursive Morphological Sieve Method for Searching Pictorial Point Symbols on Maps," Proc. 3rd Int. Conf. Document Analysis and Recognition, Montreal, Canada, 1995.

3. H. Yamada, K. Yamamoto, and K. Hosokawa, "Directional Mathematical Morphology and Reformalized Hough transform for the Analysis of Topographic Maps," IEEE Trans. PAMI, vol.15, no.4, pp.380-387, 1993.

4. C. Ahn, K. S. Kim, S. B. Rhee, "A Road Extraction Method from Topological Map Image," 1997 IEEE Pacific Rim Conf. Communication, Computer, and Signal Processing.

5. O. Shiku, A. Nakamura, M. Anegawa, H. Takahira, and H. Kuroda, "Extraction of Slant Character Candidates from Maps Using Circular templates." Proc. 3rd ICDAR, Montreal, Canada, 1995.

6. J. Y. Wang, L. H. Chen, K. C. Fan, H. Y. M. Liao, "Separation of Chinese Characters from Graphics," 3rd ICDAR, Montreal, Canada, 1995.

7. S. Suzuki and T. Yamada, "MARIS: Map Recognition Input System," Pattern Recognition, vol.23, no.8, pp.919-933, 1980.

8. N. Ebi, B. Lauterbach, and W. Anheier, "An image Analysis System for Automatic Data Acquisition from Colored Scanned Maps," Machine Vision and Applications, vol.7, pp.148-164, 1994.

9. L. Boatto et al., "An Interpretation System for Land Register Maps," Computer, vol.25, no.7, pp.25-33, 1992.

10. J. M. Ogier, et al."Multilevel Approach and Distributed Consistency for Technical Map Interpretation: Application to Cadastral Maps, " Computer Vision and Image Understanding, vol.70, no.3, pp.438-451, June 1998.

11. D. P. Huttenlocher, G. A. Klanderman and W. J. Rucklidge, "Comparing Images Using the Hausdorff Distance", *IEEE Trans. PAMI*, Vol.15, No.9, pp.850-863, 1993.

Table 1. System performance analysis

	Number of Samples extracted	Int. consistent	Int. consistent after the Enforcement	Ext. consistent	Ext. consistent after the enforcement
Road	122	80	110	77	105
Symbol	143	85	120	81	117
Symbol-related Char.	431	304	443	306	438
Road-related Char.	504	341	443	306	438
Blocks	104	68	93	66	86

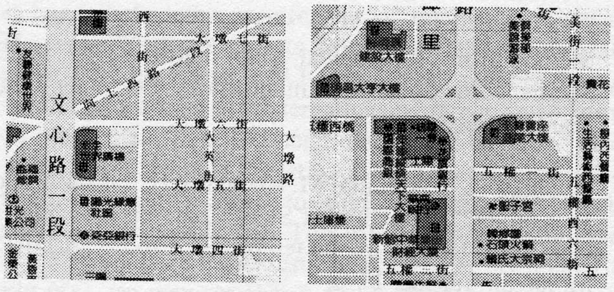

Figure 1. The Taiwanese municipal color maps

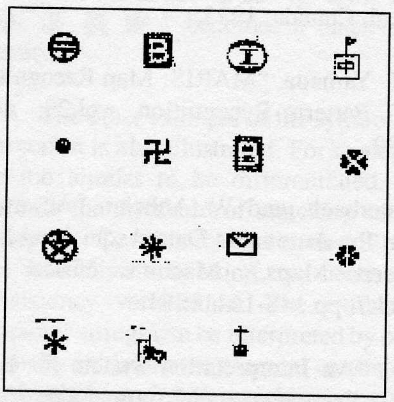

Fig 2. Some symbols appear in our maps.

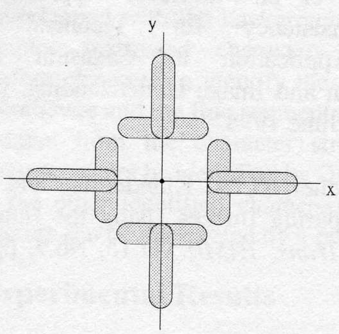

Fig 3 The possible directions to extract characters.

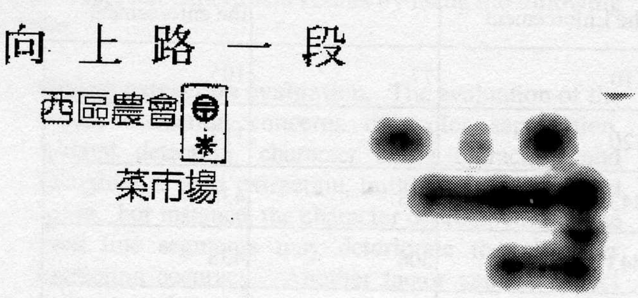

Fig 6 A example of black pixel density.

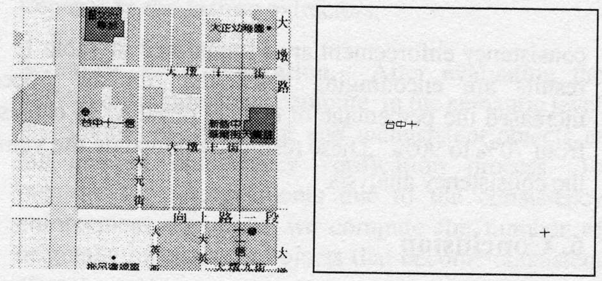

Figure 5. (a) original image (b) incomplete data remedy using internal inconsistency.

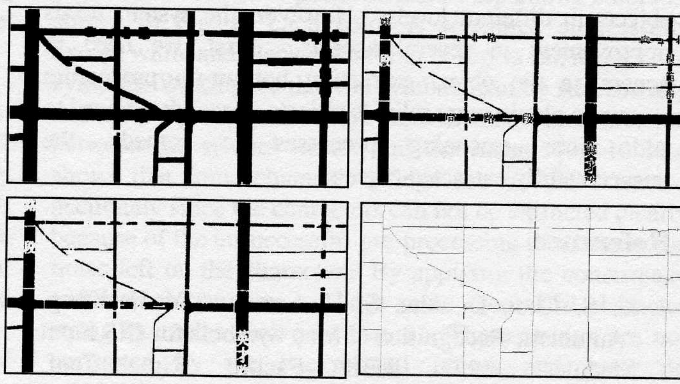

Fig. 4. Road tracing to find the road-related characters.

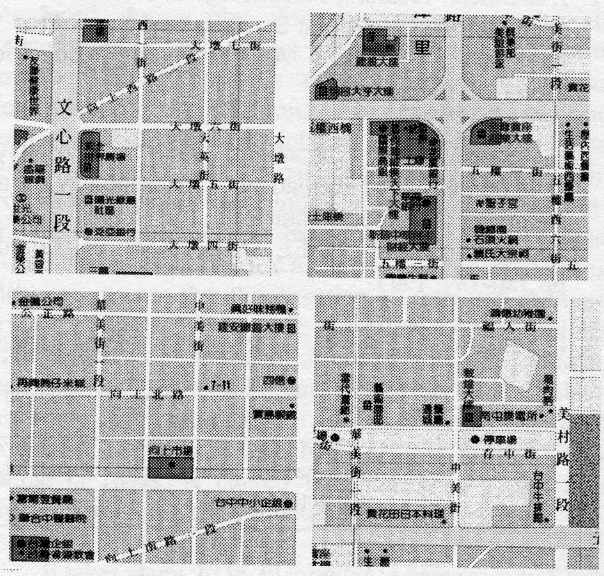

Fig.7 These are the original images of tested maps.

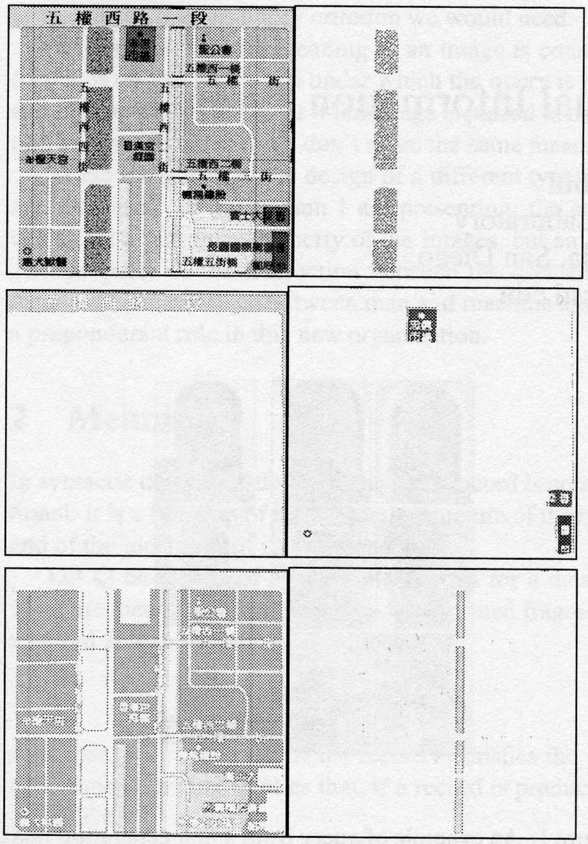

Fig. 8 (a) original image (b) green layer of the image; (c) yellow layer of the image; (d) red layer of the image; (e) flesh tints layer of the image; (f) blue layer of the image. These results show the good performance of K-means algorithm.

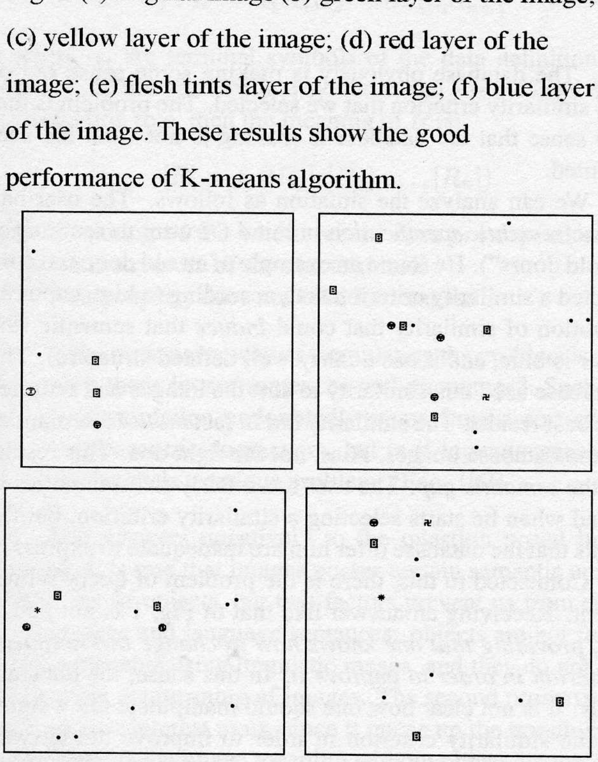

Fig.9 These are the recognized symbols after template matching.

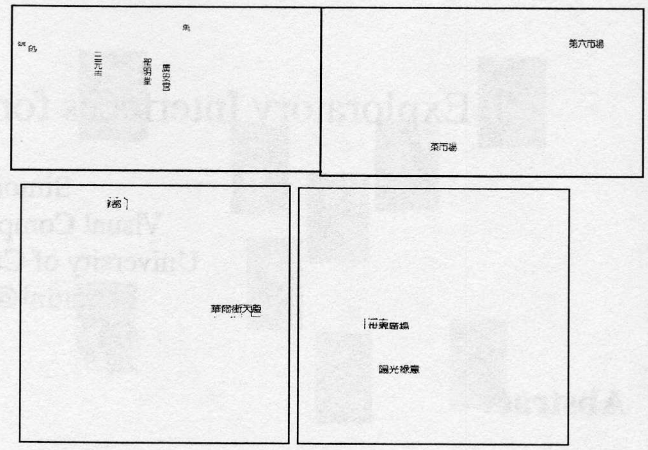

Fig. 10 Some examples of fail extraction of related character strings.

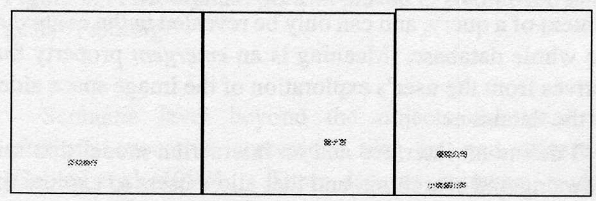

Fig 11 Some examples of successful extraction of related character strings.

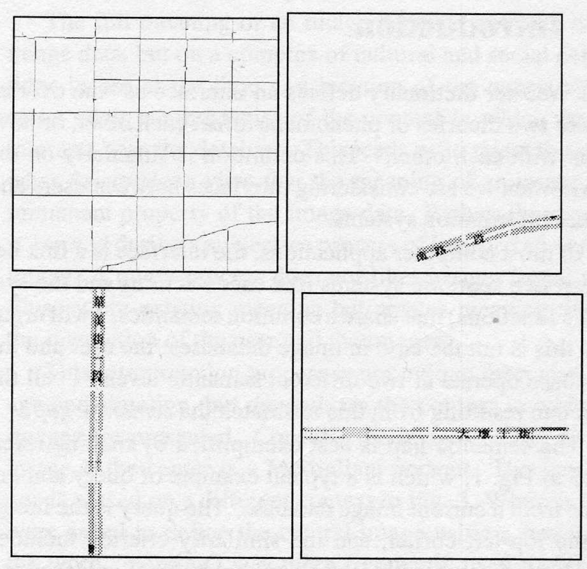

Fig 12 (a)the skeleton lines of roads; (b) curved roads. (c) (d) are two successful results of road extraction.