

Vector map geo-location using GPS tracks

Yuanlu Bao*, Hao Xu, Zhenan Liu

Department of Automation, University of Science & Technology of China, Hefei, China 230026

ABSTRACT

Geographic Information Systems, especially accurate digital traffic maps, have been becoming widely used in the Intelligent Transportation Systems. The GPS data usually are more accurate than the pre-existing base map data. This paper introduces a new cartography procedure for making digital map from a start-up paper map, which has no geography data (longitude and latitude). Contrary to traditional map-making which is first a precise survey-cartography and follow-on digital controlled mapping, the new procedure is first extracting layer from a raster map and vectorization, follow-on geo-adjusting for pinpointing all nodes on the vector map. There are two key processes in the new digital map-making technique. Firstly, the map layer extraction is controlled by output feedback for optimal recognition of road network. Secondly a dynamic topological mapping is developed for geo-location and improving the accuracy of vector traffic map by utilizing some recorded GPS data as the controlled pinpoint samples. Different from those traditional geo-location methods such as affine mapping, and the rubber-streeting with manual point-by-point matching, the new adjusting is decomposed into a series of optimal topological sub-mapping on relative sub-map region. Based on pinpointing one sample point to its accuracy location, the sub-mapping adjusts all nodes within the sub-map region.

Keywords: Raster map, digital map, geo-locating, topological mapping, road network, vector map database

1. INTRODUCTION

Geographic Information Systems (GIS) are becoming more widely used for the mapping and modeling of utility network systems. However, there is lack of topological maps, whether large-scale or even small-scale city and urban maps, in many developing countries including China^[1]. Lack of vector maps and accuracy of vector maps have become the bottleneck of developing digital city transportations and intelligent decisions^[2].

GPS can be used to improve the accuracy of out-of-date inaccurate digital map, and to verify, update, and augment map information. The recorded GPS data will usually be more accurate than the pre-existing base map data. However, existing GIS software has only the commands that perform localized rubber-streeting^[3], so that the base map data can be adjusted or relocated to match the accurate GPS positions if desired. However, the rubber-streeting algorithm is a manual adjustment procedure, which requires a huge human effort.

To do the research on digital traffic map, a simple and effective procedure for building digital map from paper map must be done. The key process in the new digital map-making technique is how to convert a raster map into a vector map and how to get accurate geo-location on the vector map. The two long-standing key technical problems have been investigating for a long time in academy^[4-8].

The road network layer extraction is based on object color, brightness, sharpness and various road sample pattern characteristics^[4,5]. This process usually needs experience to manually adjust thresholds or parameter values of evaluation functions to achieve high quality of pattern recognition. The open-loop recognition process needed human intervention, which is difficult to complete the map conversion rapidly and perfectly. This paper will presents an algorithm for optimal recognition of road network; the extraction process is controlled or deals with output map feedback (Section 2).

Current vector map geo-location is generally accomplished with traditional linear topological mapping such as length-preserving mapping^[6], affine mapping^[2], projecting mapping and interpolation^[7]. These algorithms may correct only linear errors on vector maps, but less effect on nonlinear errors such as paper map warps. It is noticed that traditional linear topological mapping can only reduce or minimize the error summation in the sense of least mean square (LMS). An innovative mapping procedure, which takes different effects on different areas of a vector map, has been introduced in Reference [8]. However, there exist some bugs within the new procedure, which will be improved in Section 4.

*ybao@ustc.edu.cn; phone/fax 86 551 360-1531; gps.ustc.edu.cn. This work was supported by NSFC 60272040 .

The new approach to generate digital maps from printed maps is divided into three processes: Map layers extraction, Road network vectorization and Dynamic topological geo-locating on vector map.

2. MAP LAYERS EXTRACTION

2.1 Main processes for road layer extraction

2.1.1 Program procedure for road layer extraction

There are three main processes for road layer extraction from a raster map: Traffic map normalizing, Noise pixel partition and Feedback on output road check, as shown in Fig. 1.

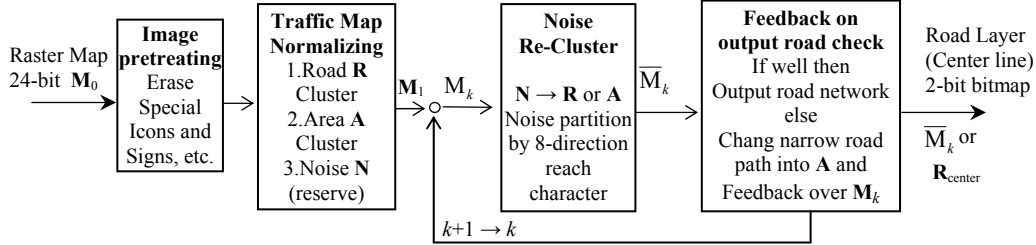


Fig. 1. Program procedure for road layer extraction from a raster map.

2.1.2 Expressions for 24-bit map image and 2-bit map image

The raster traffic map, considering as a sheet of 24-bit color image, which consists of $r + 1$ pixel rows and $l + 1$ pixel columns, is expressed as a set of pixels, M :

$$M = [p_{ij} \mid 0 \leq i \leq l, 0 \leq j \leq r] \quad (1)$$

Where p_{ij} is a pixel at position (i, j) in the pixel map, $0 \leq i \leq l, 0 \leq j \leq r$ as the original point $p_{0,0}$ is at the left down of the map M .

For 24-bit raster map, define the color of pixel p_{ij} as $C(p_{ij}) = [R(p_{ij}), G(p_{ij}), B(p_{ij})]$ which is a 3-dimensional vector with three RGB(red, green, blue) grades elements. Define $D(p_{ij}, C_k)$ as a function of color difference between the pixels p_{ij} and a color C_k :

$$D(p_{ij}, C_k) = \sqrt{[R(p_{ij}) - R_k]^2 + [G(p_{ij}) - G_k]^2 + [B(p_{ij}) - B_k]^2} \quad (2)$$

For 2-bit raster map, as there is only 2 colors, define the color of pixel $p_{ij} = C(p_{ij}) = 1$ or 0 p_{ij} only.

2.2 Traffic map normalizing

People may recognize a physiognomy by its color from the map when they perceive the map first. The map layers extraction procedure deals with the map in a similar way as people do by a color check first. Therefore, the map is first clustered into three layers: a road layer R , an area layer A and a noise layer N , which includes letters, symbol and icons, etc. As the source map image is copied from normative maps on Compact Disk (CD) provided by normal mapping publishing house^[9], it is easy to get the average or center colors of road samples, denoted by set $\bar{C}_R = \{\bar{C}_{R1}, \bar{C}_{R2}, \dots, \bar{C}_{Rj}\}$ and the average or center colors of area samples, denoted by set $\bar{C}_A = \{\bar{C}_{A1}, \bar{C}_{A2}, \dots, \bar{C}_{Aj}\}$. Furthermore, the road center color set \bar{C}_R and \bar{C}_A fit all city maps on the CD.

Define a 3-dimension vector V_R as the swatch threshold, the road layer color set C_R for clustering road layer R is

$$C_R = [C_p \mid D(C_p, \bar{C}_{R1}) \leq V_{R1}, D(C_p, \bar{C}_{R2}) \leq V_{R2}, \dots, D(C_{Rk}, \bar{C}_{Rk}) \leq V_{Rk}] \quad (3)$$

Define a 3-dimension vector V_A as the swatch threshold, the area layer color set C_A for clustering area layer A is

$$C_A = [C_p \mid D(C_p, \bar{C}_{A1}) \leq V_{A1}, D(C_p, \bar{C}_{A2}) \leq V_{A2}, \dots, D(C_p, \bar{C}_{Aj}) \leq V_{Rj}] \quad (4)$$

Simply inspecting the color of every pixel in 24-bit raster map does the traffic map normalizing procedure. For every pixel p_{ij} , if $C(p_{ij}) \in C_R$ then $p_{ij} \in R$ and change its color with normalized road white color $C(p_{ij}) = C^R = [255, 255, 255]$, else if $C(p_{ij}) \in C_A$ then $p_{ij} \in A$ and change its color with normalized road gray color $C(p_{ij}) = C^A = [127, 174, 114]$. For all other pixels, which are considered as noise N , keep its color unchanged.

As an example, a source 24-bit raster map of Hefei city, M_0 , copied from CD^[9] is shown in Fig. 2 (left). The normalized map M_1 is shown in Fig. 2 (right).

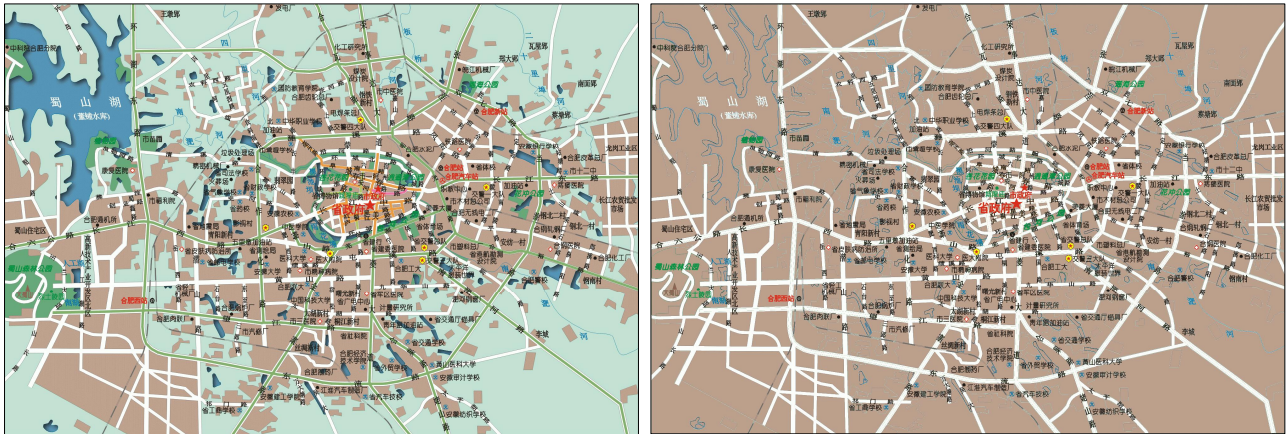


Fig. 2. 24-bit raster map M_0 of city Hefei (left) and its normalized map M_1 (right).

2.3 Noise layer partition

Noise layer partition is done by erasing and replacing color of each pixel of noise. The main function is to partition all pixels at noise region N into two parts, Road-possible N_R and Area-possible N_A . Noise partition is based on inspecting the 8-direction reach characters for every pixel $p_k \in N$ shown in Fig. 3. From point p_k draw 8 radial, check every radial as it just extends out of its noise region and whether it reaches a pixel with area color C^A or not. The number of radials, which reached pixel area, A is denoted as $\alpha(p_k)$, $0 \leq \alpha(p_k) \leq 8$. As shown in Fig. 3. $\alpha(p_1)=0$, $\alpha(p_2)=8$, $\alpha(p_3)=4$, $\alpha(p_4)=8$, $\alpha(p_5)=6$ and $\alpha(p_6)=3$. It is well founded that all points $p_k \in [p \mid \alpha(p) \geq 5]$ should be partitioned into Area-possible N_A and all points $p_k \in [p \mid \alpha(p) \leq 3]$ should be partitioned into Road-possible N_R . It is perplexing with the case of $\alpha=4$. However, needed by feedback control on the next process, all points $p_k \in [p \mid \alpha(p)=4]$ are partitioned into only Road-possible N_R .

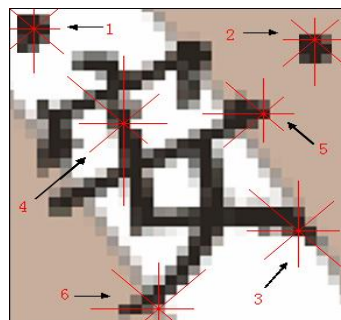


Fig. 3. Reach character of 8-direction from a pixel at noise region.

Then, for every pixel $p_{ij} \in N$, if $p_{ij} \in N_R$ then partitioning $p_{ij} \rightarrow R$ by changing its color with normalized road white color $C(p_{ij}) = C^R = [255, 255, 255]$, else if $p_{ij} \in N_A$ then partitioning $p_{ij} \rightarrow A$ by changing its color with normalized road gray color $C(p_{ij}) = C^A = [127, 174, 114]$. By noise partition, the 24-bit color map M_k becomes 2-bit raster map \bar{M}_k . A normalized map part M_1 is shown in Fig. 4(left) and its noise-classified map \bar{M}_1 (2-bit map) is shown in Fig. 4(right).



Fig. 4. Normalized map M_1 (left) of city Hefei (part) and its noise classified map \bar{M}_1 (right).

2.4 Feedback on output road check

As done before, all points $p_k \in [p | \alpha(p)=4]$ are partitioned into only Road-possible N_R . That means some noise points which possible be at area layer A in fact but now are partitioned into road layer R . Thus there is more "road" on the noise classified 2-bit map \bar{M}_k . So it is needed first to find the possible superfluous or error road path and then feedback the message to its source map M_k by changing the superfluous path into area regions A . Having all new area regions M_k becomes M_{k+1} . By noise layer partition at the new source map M_{k+1} , its noise-classified map \bar{M}_{k+1} should be more credible than \bar{M}_k for road extraction. This feedback recognition process can be down continually, until there exist no more possible superfluous road path. The final 2-bit map \bar{M}_n can be output as road layer extracted from a color raster map M_0 , so finished the process of map layers extraction.

How to find the possible superfluous road path? It is based on the road width at every center point of road to check the superfluous path. Make centerline of road on the noise classified 2-bit map \bar{M}_k , denoted as \tilde{M}_k . Through the road centerline $X = \{x_k | 1 \leq k \leq l\}$ on \tilde{M}_k , for every center point x_k check the path width $d(x_k)$. If $d(x_k) \leq d_{min}$ then change color of point x_k and the corresponding path transverse line with normalized area gray color C^A . After all possible superfluous path are changed into area region A , \tilde{M}_k becomes \tilde{M}_k^* which is used only to show the eliminating process of the possible superfluous road path, does not used as feedback map. A center-lined road map \tilde{M}_1 is shown in Fig. 5(left) and the map \tilde{M}_1^* eliminated superfluous road path is shown in Fig. 5(right).

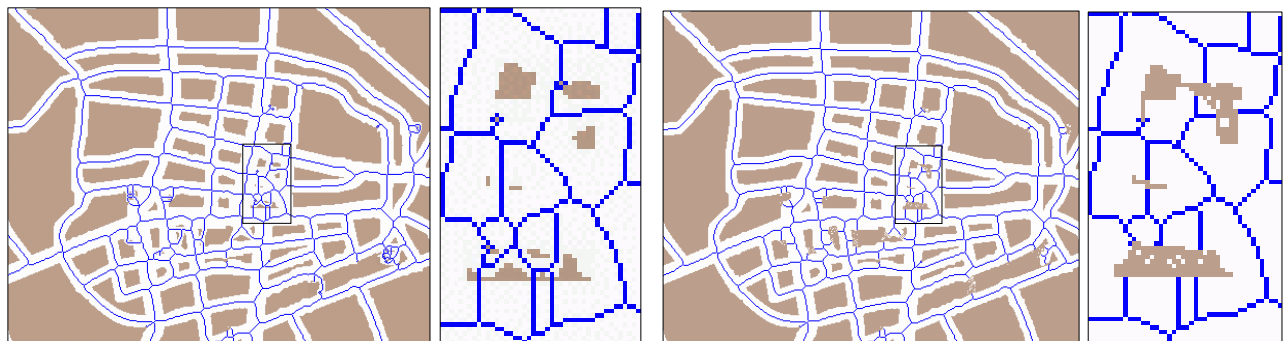


Fig. 5. Center lined road map \tilde{M}_1 (left) and map \tilde{M}_1^* eliminated superfluous road path (right).

Having the new area regions M_1 becomes M_2 by feedback. The 24-bit Feedback modified map M_2 and its noise classified 2-bitmap \bar{M}_2 is shown in Fig. 6(left) and its noise classified 2-bit map \tilde{M}_2 (Center lined) is shown in Fig. 6(right).

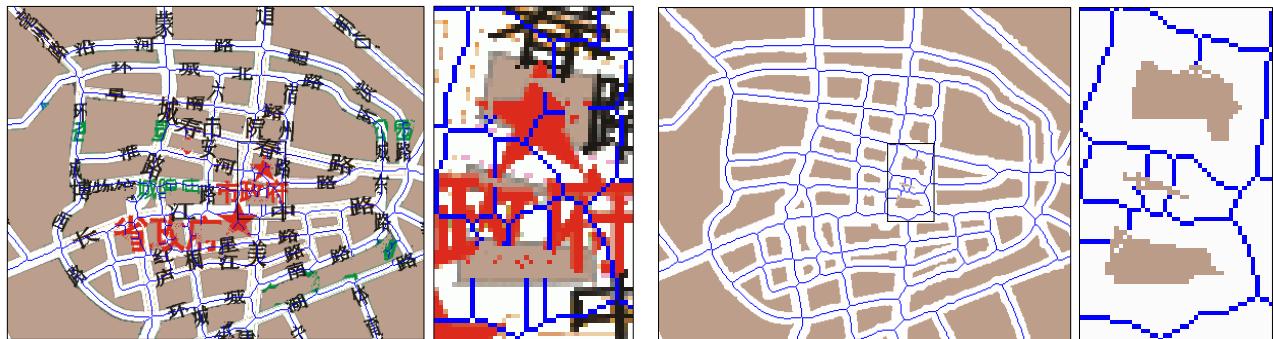


Fig. 6. Feedback modified road map M_2 (left) and its noise classified 2-bit map \tilde{M}_2 (right)

As done continually, the procedure is accomplished at noise-classified map \bar{M}_3 (\tilde{M}_3). The 24-bit feedback modified map M_3 and its noise classified 2-bitmap map \bar{M}_3 (Center lined as \tilde{M}_3) are shown in Fig. 7.

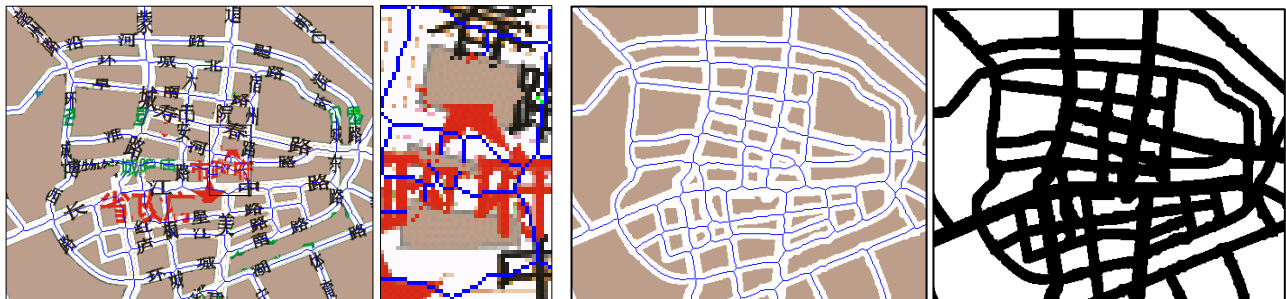


Fig. 7. Feedback modified road map M_3 (left) and its noise classified 2-bit map \tilde{M}_3 (middle), \bar{M}_3 (right).

3. ROAD NETWORK VECTORIZATION

3.1 Road layer vectorization

Vectorization is to convert a black-white bitmap of the line-shape figures into a vector format graph and then greatly reduces the memory space required to save the map geometry information, and makes easy for further operations on the map. This step has a pretreatment process including smoothing, thinning, joining broken lines and deleting spurs. They are briefly as follows. Effective procedures can be finding in details in [2].

- Smoothing: Because of noises and errors, there are some indents and bulges in the line-shape features. This smoothing is to remove these indents and bulges. Thus, it improves the precision of vectorization.
- Thinning: Thinning is a process to erase those points that do not affect the connection of the line shaping figures. Its purpose is to get the one-point-wide line to make the vectorization applicable.
- Joining broken lines and deleting spurs: These treatments of joining broken lines and deleting spurs are developed to fix the errors existed in the original map or entered during the above procedures.

After the above procedures, a bitmap of single-point-wide lines is ready. Now it is for vectorization. The main idea is to determine and store the nodes that represent the road curves by the lines connecting the corresponding nodes. The maximum error among the road curves and their corresponding road lines is controlled within a predefined pixel number. That algorithm is a convergent algorithm because the real maximum error approaches to zero when the number of family nodes increases. Fig. 8 shows the vector map by applying above procedures on an extracted road layer (Fig. 8).

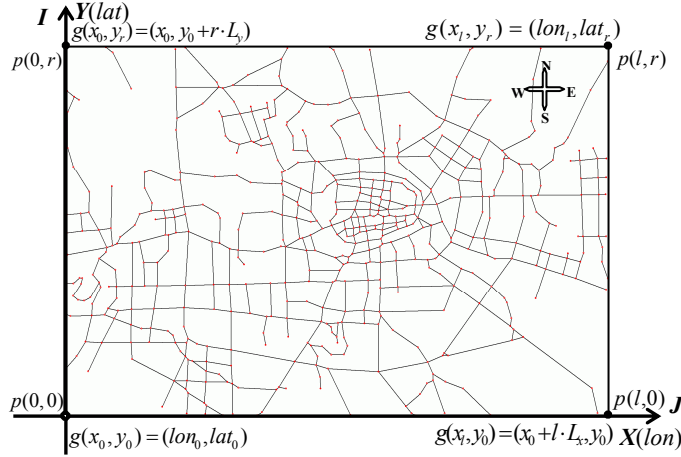


Fig. 8. Vector map based on the extracted road layer.

3.2 Data structure with vector map

3.2.1 Vector map on its node-location space $P(I, J)$

Normally, the size of data storage after vectorization is far less than before, because only the locations of finite nodes and the topological relationship of nodes connected in sequence need to be saved on its node-location integer space $P = \{p(i, j)\}$. Each point $p_k(i_{p_k}, j_{p_k})$ at vector map has its location with the I -coordinate i_k , J -coordinate j_k (Fig. 8).

The database structure for road network is defined with three data chains: Node's chain, Arc's chain, and Road's chain. The Node's chain, denoted as N_R , consists of all N nodes in sequence:

$$N_R = \{n_k \mid n_k = p_{n_k}(i_{n_k}, j_{n_k}), k = 1, 2, \dots, N\} \quad (5)$$

The ordinal number k of k -th node n_k , denoted by $ID(n_k) = k$, is used to identify the node as will be used in the vector database structure. The Arc's chain, denoted as S_R , consists of all m arcs in sequence:

$$S_R = \{S_1, S_2, \dots, S_m \mid S_k = [ID(n_{first}^k), ID(n_2^k), \dots, ID(n_{last}^k)], k = 1, 2, \dots, m\} \quad (6)$$

Each arc (section or path of road) S_k consists of some nodes in sequence, in which only the first node n_{first}^k and the last node n_{last}^k are either an intersection or terminal point of roads. Note that Arc's chain S_R declares the topological relationship of nodes. Within the procedure of topological geo-adjusting the Arc's data chain S_R should not be changed.

3.2.2 Vector map on its geo-located image space $G(X, Y)$

A vector map obtained from raster map, $M = [p(i, j)]$ has no geo-location in longitude and latitude. The geo-located image space $G(X, Y)$, $(-\pi \leq x \leq \pi, -\pi/2 \leq y \leq \pi/2)$, is defined as follow. Suppose the longitude and latitude of bottom left corner $p(0,0)$ and the top right corner $p(l,r)$ have a measurement of $g(x_0, y_0) = (lon_0, lat_0)$ and $g(x_l, y_r) = (lon_l, lat_r)$ respectively, then a fixed linear mapping (Eq. 7) is used to transform the data between node-location space $P(I, J)$ and its geo-located image space $G(X, Y)$.

$$G: p(i, j) \rightarrow g(x, y) = g(lon_0 + i \cdot L_x, lat_0 + j \cdot L_y) \quad (7)$$

$$G^{-1}: g(x, y) \rightarrow p(i, j) = p[(x - x_0)/L_x, (y - y_0)/L_y] \quad (8)$$

Where $L_x = (lon_l - lon_0)/l$ and $L_y = (lat_r - lat_0)/r$ are the scales in two axis directions in pixel. On the following parts of this paper, all discusses and equations will be based on map image space, $M = [g(x, y)]$.

3.3 Map geo-locating by traditional topological mapping

Map geo-adjusting is to improve vector map accuracy. The method is to use the recorded road locus by GPS to adjust the vector traffic map. Select a group of n -sample points set G_n :

$$G_n = \{g_1, g_2, \dots, g_n\} = \{g_1(x_1, y_1), g_2(x_2, y_2), \dots, g_n(x_n, y_n)\} \quad (9)$$

Where the accurate positions (longitudes and latitudes) of the n -sample set G_n , denoted by \hat{G}_n , has known on map image space as

$$\hat{G}_n = \{\hat{g}_1, \hat{g}_2, \dots, \hat{g}_n\} = \{\hat{g}_1(\hat{x}_1, \hat{y}_1), \hat{g}_2(\hat{x}_2, \hat{y}_2), \dots, \hat{g}_n(\hat{x}_n, \hat{y}_n)\} \quad (10)$$

Traditional topological mapping $T(\cdot)$ is used to transform vector map M into $M' = T(M)$, with effect of an n -samples set G_n relocated on $G'_n = T(G_n)$

$$G'_n = \{g'_1(x'_1, y'_1), \dots, g'_n(x'_n, y'_n)\} = T(G_n) = T\{g_1, g_2, \dots, g_n\} \quad (11)$$

Such that G'_n is closed to the exact accurate positions \hat{G}_n as nearly as possible. An optimal performance index is

$$J = \|G'_n - \hat{G}_n\| = \sum_{i=1}^n \sqrt{(x'_i - \hat{x}_i)^2 + (y'_i - \hat{y}_i)^2} \quad (12)$$

The linear topological geo-adjusting can be done by several kinds of mappings such as length-preserving mapping, affine mapping, projecting mapping, etc. Take an affine mapping as an example^[6] where

$$g' = T(g) = (x'_g, y'_g) = (a_1x_g + b_1y_g + c_1, a_2x_g + b_2y_g + c_2) \quad (13)$$

The parameters $V = [a_1, a_2, b_1, b_2, c_1, c_2]$ can be obtained via an optimal searching procedure^[2] to move G_n reach the optimal approximation location G'_n in sense of LMS (Eq.12). To show the effectiveness of the linear topological geo-adjusting, we have got Hefei city vector map as shown in Fig. 8. The adjustment will be done by the directly using some GPS traces of vehicles in the path travel recording to the vector map and select a 34-samples set \hat{G}_n . The GPS location traces are shown as the dotted lines on Hefei city vector map before adjusting (Fig. 9. left). The points marked by red “+” denotes the accurate longitudes and latitudes $\hat{g}_k(x_k, y_k)$ at \hat{G}_n , and the points marked by blue “x” are the corresponding locations g_k at G_n , may be laid on wrong coordinates before adjusting. Via the linear topological geo-locating, Hefei city vector map is improved (Fig. 9. right).

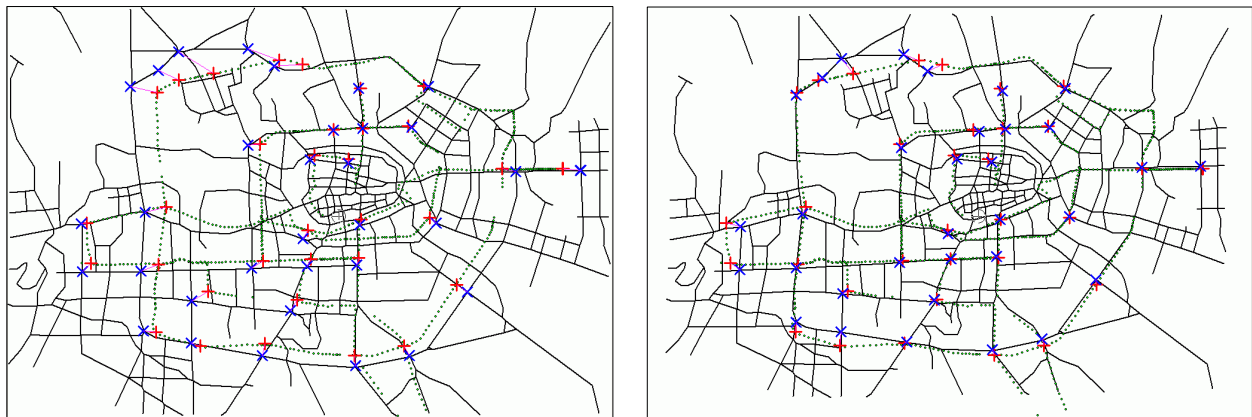


Fig. 9. Map geo-locating using GPS tracks by traditional topological mapping.

The linear errors of vector map rooted in extractions of an accurate printed map with large scale can be revised perfectly by traditional topological mapping. However the source raster maps are often poor in quality in China. Although the small-scale city and urban maps are easy bought but the paper maps have always warp and not up-to-dated, there could

be different scales on the different parts of the traffic map from urban center to the outlying area. There has another data errors problem on the paper maps when which is based on different coordinate system. Thus a new topological geo-locating technique is developed for mapping G_n exactly matching its exact positions \hat{G}_n . Different with linear mapping, the innovative mapping procedure should get different effects on different areas of a vector map. Thus it is a dynamic topological mapping, will be presented in details on the following section.

4. DYNAMIC TOPOLOGICAL GEO-LOCATING ON VECTOR MAP

4.1 Goal of dynamic topological geo-locating on vector map

The vector map accuracy depends only on the finite node point's accuracy. The vector map must keep those nodes topology. The goal of dynamic topological geo-locating on vector map is to find a topological mapping $T(\cdot)$ mapping G exactly onto \hat{G} . The index for dynamic topological geo-locating is

$$T(\cdot): T(G_n) = T\{g_1, g_2, \dots, g_n\} = \hat{G}_n = \{\hat{g}_1, \hat{g}_2, \dots, \hat{g}_n\} \quad (14)$$

With dynamic topological geo-locating $T(\cdot)$ the n -sample-pair $[G_n, \hat{G}_n]$ is exactly geo-located at \hat{G}_n . Using the dynamic topological mapping $T(\cdot)$ on vector map $M = [g(x, y)]$, all N nodes at N_R (Eq. 5) are also mapped onto new location $T(M) = T[g(x, y)] = \hat{M}[T(g(x, y))]$. It is well founded that, as the n -sample-pair $[G_n, \hat{G}_n]$ is exactly geo-located, all neighboring nodes near n -sample points should be changed to reach more accurate locations themselves.

4.2 Structure of dynamic topological mapping

To get n -sample-pair $[G_n, \hat{G}_n]$ exactly geo-located, the dynamic topological mapping $T(\cdot)$ should be disintegrated into n sub-mappings $T_k(\cdot)$, for one sample-pair $[g_k, \hat{g}_k]$ relocated by one sub-mapping $T_k(\cdot)$. Thus the new algorithm adjusts a sample with a sample point adjustment. At the time of adjusting sample-pair $[g_k, \hat{g}_k]$, the algorithm also automatically adjusts nodes on the sample-pair $[g_k, \hat{g}_k]$ neighboring area denoted as dynamic block S_k .

$$T(M) = T_n(T_{n-1}(\dots T_k(\dots T_2(T_1(M))\dots))) \quad (15)$$

The sub-geo-located vector map after k -th step is denoted as $\hat{M}_k = T_k(\dots T_2(T_1(M))\dots)$ and decomposed into two parts: dynamic block S_k and the complementary part region S_k^c .

$$\hat{M}_k = T_k(\hat{M}_{k-1}) = T_k(T_{k-1}(\hat{M}_{k-2})) = T_k(T_{k-1}(\dots T_2(T_1(\hat{M}_0))\dots)) = T_k(S_k \cup S_k^c) = T_k(S_k) \cup T_k(S_k^c) = \hat{S}_k \cup S_k^c \quad (16)$$

Where $\hat{M}_0 = M$ is the source vector map. Denote the k -sample-pair as $[G_k, \hat{G}_k]$, where $G_k = [g_1, g_2, \dots, g_k]$ and $\hat{G}_k = [\hat{g}_1, \hat{g}_2, \dots, \hat{g}_k]$. After k -th step by the sub-mapping $T_k(\cdot)$ the k -sample-pair $[G_k, \hat{G}_k]$ should be geo-located exactly at the accurate locations $\hat{G}_k = [G_k, \hat{G}_k]$, i.e.

$$\hat{G}_k = T_k(\hat{G}_{k-1}, g_k) = T_k(T_{k-1}(\dots T_2(\hat{G}_1, g_2)\dots), g_k) = [T_k(\hat{G}_{k-1}), T_k(g_k)] = [\hat{G}_{k-1}, \hat{g}_k] \quad (17)$$

It enlightens a principle for construction of dynamic block S_k and the complementary part region S_k^c on current vector map \hat{M}_{k-1} . That is at k -th step of dynamic topological mapping, for the sub-mappings $T_k(\cdot)$ there should be $g_k \in S_k \subset \hat{M}_{k-1}$ and $\hat{G}_{k-1} \in S_k^c \subset \hat{M}_{k-1}$. The sub-mappings $T_k(\cdot)$ (Eq. 16) becomes

$$\hat{M}_k = T_k(\hat{M}_{k-1}) = \begin{cases} S_k \Rightarrow \hat{S}_k : & T_k(S_k) = \hat{S}_k \subset \hat{M}_k \text{ with } [g_k, \hat{g}_k] \in S_k \subset \hat{M}_{k-1}, g_k \rightarrow \hat{g}_k \\ S_k^c \Rightarrow S_k^c : & T_k(S_k^c) = S_k^c \subset \hat{M}_k \text{ with } \hat{G}_{k-1} \in S_k^c \subset \hat{M}_{k-1}, \hat{G}_{k-1} \rightarrow \hat{G}_{k-1} \end{cases} \quad (18)$$

There are several methods to construct the dynamic topological block S_k . An effective rule for construction of dynamic block S_k is described bellow.

4.3 Structure of dynamic topological block S_k

4.3.1 Dynamic topological block S_1 and S_2

It is obvious that as $\hat{G}_0 = \Phi$ (for nothing) the first block is taken $S_1 = \hat{M}_0 = M$ as the whole source vector map. As $\hat{G}_1 = \hat{g}_1$, the second block $S_2 = [p \mid p \in M, p \neq \hat{g}_1]$ is the whole map except the point $\hat{G}_1 = \hat{g}_1$.

4.3.2 Dynamic topological block S_k bounded

The k -th block S_k ($k \geq 3$) is determined by the following rules shown in Fig. 10(left) to get $g_k \in S_k$ and $\hat{G}_{k-1} \in S_k^c$. At k step, for every \hat{g}_i ($i=1,2,\dots,k-1$), connect \hat{g}_i and \hat{g}_k (\hat{g}_k is the center point of line $g_k\hat{g}_k$), make a perpendicular line $L_k\hat{g}_i$ to the line $\hat{g}_k\hat{g}_i$ pass point \hat{g}_i . The line $L_k\hat{g}_i$ divides the map into two half planes. Denote the half plane with point \hat{g}_k as $H_k(\hat{g}_i)$. The block S_k should be within $H_k(\hat{g}_i)$, i.e.

$$S_k = \bigcap_{i=1}^{k-1} H_k(\hat{g}_i) \quad k = 3, 4, \dots, n \quad (19)$$

The dynamic topological block S_k may be a limited region as an m -polygon, denoted as $S_k = S_k^{m\Delta} = \bigcup_{j=1}^m \Delta g_k L_k \hat{g}_j$. For example shown in Fig. 10(right), where $k=6$, $S_k = S_6$. The block S_6 is a quadrilateral $ABCD$ with $m=4$. Its boundary m -sides are $L_k \hat{g}_j$, $j = 1, 2, 3, 4$. Note that there is always $m \leq k-1$, for large m , we may have $m \ll k-1$.

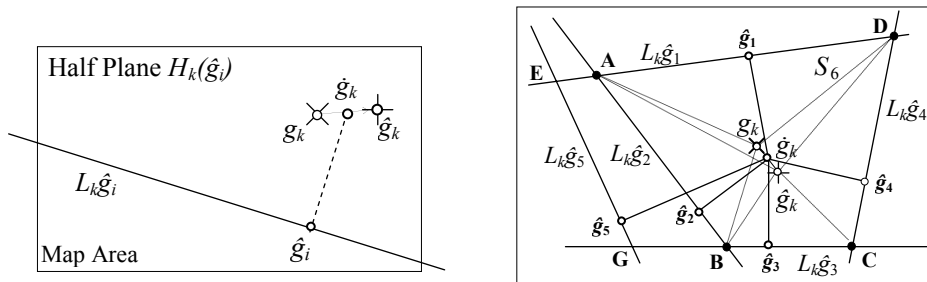


Fig. 10. Block region S_k on the half plane $H_k(\hat{g}_i)$ ($i < k$) and Construction of block S_k ($k=6$).

4.3.3 Dynamic topological block S_k boundless

The dynamic topological block S_k by m -lines $L_k \hat{g}_i$, may not be a bounded region as shown in Fig. 11. Connect point g_k and \hat{g}_k to the vertices of S_k , A and B as in Fig. 11. From g_k draw radial $L_1 g_k$ parallel with $L_k \hat{g}_1$, radial $L_3 g_k$ parallel with $L_k \hat{g}_3$. From \hat{g}_k draw radial $L_1 \hat{g}_k$ parallel with $L_k \hat{g}_1$, radial $L_3 \hat{g}_k$ parallel with $L_k \hat{g}_3$. Then the unbounded region S_k can be considered as an extended m -polygon which consists of one polygon $S_k^{(m-2)\Delta}$, two strip regions, S_k^1 and S_k^2 , and a sector S_k^\vee as shown in Fig. 11 with $m=3$ and its m -sides are $L_k \hat{g}_1$, $L_k \hat{g}_2$ and $L_k \hat{g}_3$.

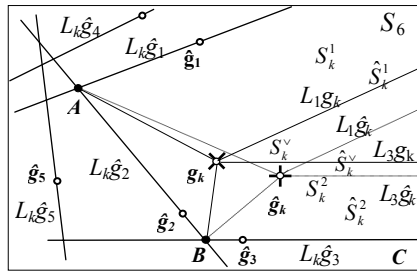


Fig. 11. Constructing a bounded block S_k from an unbounded area.

4.3.4 Exceptive case for constructing topological block S_k

It is very complicated for topological mapping as some adjusting sample-pair $[g_k, \hat{g}_k]$ is not really accurate, or if the source vector map has inaccuracy with its topology. In this exceptive case the topological block S_k may not be constructed. To manage this exceptive case for constructing topological block S_k in this paper the inaccurate sample-pair $[g_k, \hat{g}_k]$ is deleted. More effective method to deal with such exceptive case is in developing.

4.4 Dynamic topological mapping $T_k(\cdot)$

4.4.1 First and second dynamic topological sub-mappings $T_1(\hat{M}_0)$ and $T_2(\hat{M}_1)$

First dynamic topological sub-mapping $T_1(\hat{M}_0)$ is simply moving whole map to get $g_1(x_1, y_1) \rightarrow \hat{g}_1(\hat{x}_1, \hat{y}_1)$ based on vector map \hat{M}_0 . By $T_1(\hat{M}_0)$ all points $s(x, y) \in S_1 = \hat{M}_0 = M$ is mapped into $\hat{s}(x', y')$ as

$$T_1(\hat{M}_0) = \hat{M}_1 : s(x, y) \rightarrow T_1(s) = \hat{s}(x', y') = \hat{s}(x + \hat{x}_1 - x_1, y + \hat{y}_1 - y_1) \quad (20)$$

Second dynamic block S_2 is whole map \hat{M} except the fixed point \hat{g}_1 . By $T_2(\hat{M}_1)$, S_2 is rotated and resized to get $g_2(x_2, y_2) \rightarrow \hat{g}_2(\hat{x}_2, \hat{y}_2)$ based on vector map \hat{M}_1 , i.e. all points $s(x, y) \in S_2$, is mapped into $\hat{s}(x', y')$ with

$$T_2(\hat{M}_1) = \hat{M}_2 : \begin{cases} (y' - \hat{y}_1)/(x' - \hat{x}_1) = \tan\{\alpha + \tan^{-1}[(y - \hat{y}_1)/(x - \hat{x}_1)]\} \\ \sqrt{(x' - \hat{x}_1)^2 + (y' - \hat{y}_1)^2} = \beta \sqrt{(x - \hat{x}_1)^2 + (y - \hat{y}_1)^2} \end{cases} \quad (21)$$

Where the enlargement/reduction factor is $\beta = |g_2 - \hat{g}_2|/|g_2 - \hat{g}_1| = \sqrt{(\hat{x}_2 - \hat{x}_1)^2 + (\hat{y}_2 - \hat{y}_1)^2} / \sqrt{(x_2 - \hat{x}_1)^2 + (y_2 - \hat{y}_1)^2}$ and the rotation angle around point \hat{g}_1 counterclockwise is $\alpha = \angle \overrightarrow{\hat{g}_1 \hat{g}_2} - \angle \overrightarrow{\hat{g}_1 g_2} = \tan^{-1}[(\hat{y}_2 - \hat{y}_1)/(\hat{x}_2 - \hat{x}_1)] - \tan^{-1}[(y_2 - \hat{y}_1)/(x_2 - \hat{x}_1)]$.

4.4.2 K -the dynamic topological sub-mapping $T_k(\hat{M}_{k-1})$ with S_k bounded

When S_k is a convex m -polygon $S_k^{m\Delta}$ (Fig. 10), the block $S_k = S_6^{4\Delta}$ is a convex quadrilateral $ABCD$, g_k and \hat{g}_k are both within the block S_k . Connect g_k to all vertexes of polygon $S_k^{m\Delta}$ making m triangles $\Delta g_k L_k \hat{g}_i$ consisting of side $L_k \hat{g}_i$ and g_k . Connect \hat{g}_k to all vertexes of polygon $S_k^{m\Delta}$ making m triangles $\Delta \hat{g}_k L_k \hat{g}_i$ consisting of side $L_k \hat{g}_i$ and \hat{g}_k . It leads to

$$S_k = S_k^{m\Delta} = \bigcup_{j=1}^m \Delta g_k L_k \hat{g}_j = \bigcup_{j=1}^m \Delta \hat{g}_k L_k \hat{g}_j \quad i_j \in \{1, 2, \dots, k-1\}, j = 1, 2, \dots, m \quad (22)$$

The mapping $T_k(S_k)$ is a mapping S_k on itself. It can be done by m triangle mappings $T_k^\Delta(\Delta)$ on m triangles in S_k :

$$T_k(S_k) = T_k(S_k^{m\Delta}) = T_k\left(\bigcup_{j=1}^m \Delta g_k L_k \hat{g}_j\right) = \bigcup_{j=1}^m T_k^\Delta(\Delta g_k L_k \hat{g}_j) = \bigcup_{j=1}^m \Delta \hat{g}_k L_k \hat{g}_j \quad (23)$$

The triangle mapping $T_k^\Delta(\Delta)$ maps triangle $\Delta g_k L_k \hat{g}_i$ onto the corresponding triangle $\Delta \hat{g}_k L_k \hat{g}_i$, is defined as shown in Fig. 12. Denoted $AB = L_k \hat{g}_i$, for each point $s \in \Delta g_k L_k \hat{g}_i$, extend the line $g_k s$ to cross the line AB at point P and connect \hat{g}_k and P , The mapping rule of $T_k^\Delta(\Delta)$ lets the image point $\hat{s} \in \Delta \hat{g}_k L_k \hat{g}_i$ be on the line segment $g_k P$ and insures

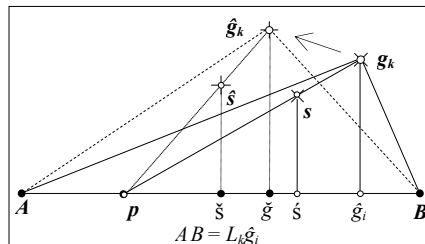


Fig. 12. Triangle mapping $T_k^\Delta(\Delta g_k L_k \hat{g}_i) = \Delta \hat{g}_k L_k \hat{g}_i$ in S_k .

$$\overline{\hat{g}_k \hat{s} / \hat{g}_k p} = \overline{g_k s / g_k p} \quad (24)$$

It uniquely determines the image point \hat{s} of point s . The triangle mapping $T_k^\Delta(\Delta)$ maps the point g_k to point \hat{g}_k ; line AB to itself; line $g_k A$ to $\hat{g}_k A$; line $g_k B$ to $\hat{g}_k B$; and line $g_k P$ to $\hat{g}_k P$. It is easy to see that by $T_k(S_k)$ the nearer a point s to g_k is, the larger adjustment effect on \hat{s} is, thus $T_k(S_k)$ keeps the topology of S_k .

4.4.3 K -the dynamic topological sub-mapping $T_k(S_k)$ with S_k boundless

When S_k is not a bounded region it is treated as an extended m -polygon (Fig. 11) which consists of a polygon $S_k^{(m-2)\Delta}$, two strip regions, S_k^1 and S_k^2 , and a sector S_k^\vee . Define mapping $T_k(S_k)$ still a mapping S_k itself as follow.

$$T_k(S_k) = T_k(S_k^{(m-2)\Delta} + S_k^1 + S_k^2 + S_k^\vee) = T_k(S_k^{(m-2)\Delta}) + T_k(S_k^1) + T_k(S_k^2) + T_k(S_k^\vee) = \hat{S}_k^{(m-2)\Delta} + \hat{S}_k^1 + \hat{S}_k^2 + \hat{S}_k^\vee \quad (25)$$

Where $T_k(S_k^{(m-2)\Delta}) = \hat{S}_k^{(m-2)\Delta}$ have been defined, $T_k(S_k^\vee) = \hat{S}_k^\vee$ is defined as a parallel mapping insuring with S_k^\vee and \hat{S}_k^\vee are exactly same in shape and S_k^\vee shifted to \hat{S}_k^\vee . The strip regions, S_k^1 and S_k^2 , can be considered as an extended triangle which has one vertex at infinite. Then the two extended triangle mappings, $T_k(S_k^1) = \hat{S}_k^1$ and $T_k(S_k^2) = \hat{S}_k^2$ can be easily solved by using the same triangle mapping, $T_k^\Delta(S_k^1) = \hat{S}_k^1$ and $T_k^\Delta(S_k^2) = \hat{S}_k^2$ (Eq.24).

5. EXAMPLES FOR VECTOR MAP GEO-LOCATION USING GPS TRACKS

The effectiveness of the new approaches to accurate geo-locating is still shown by mapping Hefei city vector map with a 34-sample-pair $[G_{34}, \hat{G}_{34}]$ based on some GPS traces on road path (Fig. 9. left). The work is started on the traditional mapped map (Fig. 9. right).

The effectiveness of middle sub-mapping $T_k(\cdot)$ while the case S_k is m -polygon is shown with $k=9$. On this example with $T_9(S_8)$, S_8 is a 5-polygon (Fig. 13. left) while the 8-sample-pair $[G_8, \hat{G}_8]$ has been exactly geo-located at 8 points $\hat{G}_{k-1} = \hat{G}_8$ (Eq.17, Fig. 13. left). Fig. 13 shows the vector map after the 9-th sub-mapping $T_9(\hat{M}_8)$ based on the sample-pair $[g_9, \hat{g}_9]$ re-located.

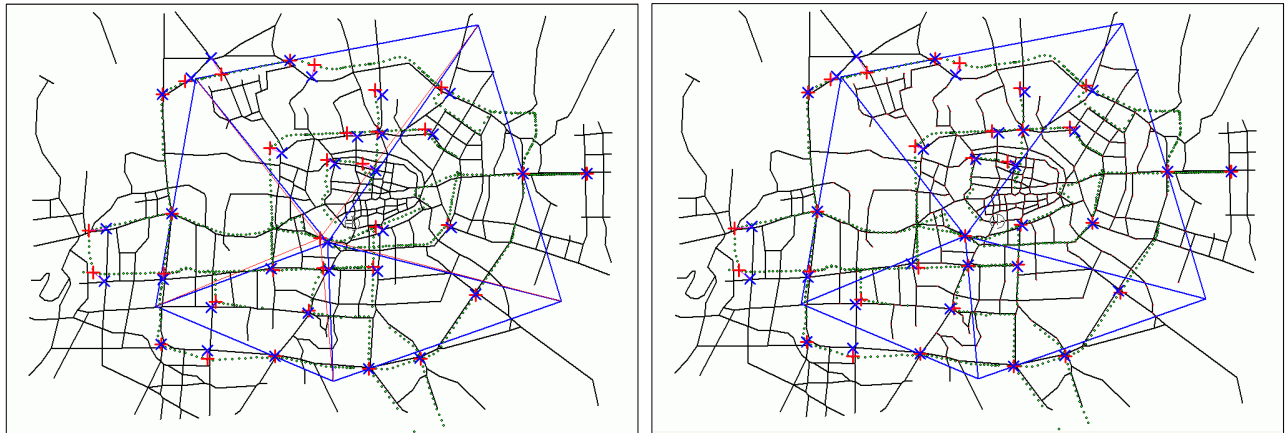


Fig. 13. Hefei city vector map \hat{M}_8 (left) before and after the 9th sub-mapping $\hat{M}_9 = T_9(\hat{M}_8)$ (right).

While S_k is boundless as is treated as an extended m -polygon, the sub-mapping $T_k(\cdot)$ is shown with $k=10$. Where the 9-sample-pair $[G_9, \hat{G}_9]$ has been exactly geo-located at 9 points $\hat{G}_{k-1} = \hat{G}_9$ (Eq. 17, Fig. 14. left). Fig. 14 shows the vector map after the 10-th sub-mapping $T_{10}(\hat{M}_9)$ based on the sample-pair $[g_{10}, \hat{g}_{10}]$ re-located.

Fig. 15 shows the last sub-mapping $T_{34}(\hat{M}_{33})$, which finished this procedure for the dynamic topological geo-locating on the extracted vector map M (Fig. 8.) by $\hat{M}=T(M)=\hat{M}_{34}=T_{34}(\hat{M}_{33})$. The geo-located map \hat{M} is shown at Fig. 15 (right).

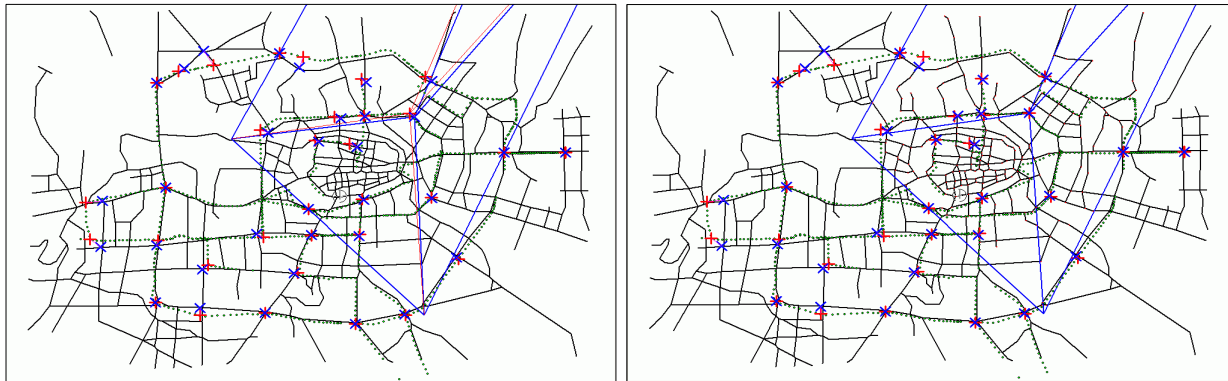


Fig. 14. Hefei city vector map \hat{M}_9 (left) before and after the 10th sub-mapping $\hat{M}_{10}=T_{10}(\hat{M}_9)$ (right).



Fig. 15. Hefei city vector map \hat{M}_{33} (left) and the geo-located map $\hat{M}=T(M)=\hat{M}_{34}=T_{34}(\hat{M}_{33})$

REFERENCES

1. Li Deren, "The digital earth and 3S Technology", Forum for GIS, <http://www.gischina.com>
2. Sheng-Guo Wang and Yuanlu Bao, "New intelligent method to generate vector maps for GPS navigation", IFAC World Congress b'02, Barcelona. Paper 157, July 2002, <http://ifac2002.dit.upm.es/ifac2002/>
3. Noronha V. and M. F. Goodchild, "Map accuracy and location expression in transportation – reality and prospects", *Transportation Research Part C* 8, 53-69 (2000)
4. Ding Bin, Wong Kok Cheong, "A System for automatic extraction of road network from maps", *Proc. IEEE International Joint Symposia on Intelligence and Systems*, Rockville, USA. 359–366 (1998).
5. Wenzhong Shi and Changqing Zhu, "The Line Segment Match Method for Extracting Road Network From High-Resolution Satellite Images", *IEEE Trans. Geoscience and Remote Sensing* 40(2), 511–514 (2002)
6. Han Jing, Li Yingkui, et al. "Algorithms of map data correction with its application and realization in GIS", *Journal of basic science and engineering* 7(4), 366-371 (1999)
7. Jing Li, George Taylor and David B. Kidner, "Accuracy and reliability of map-matched GPS coordinates", *Computers & Geosciences* 31, 241–251 (2005)

8. Yuanlu Bao, Sheng-Guo Wang, Zhenan Liu, "Two new approaches to automatic generation and accurate adjustment of vector maps for AVL" , *Proceedings of ICIEA2006*, Singapore, 1029-1034 (2006)
9. *Atlas of Chinese Road*, Beijing: Geology publishing house, China (2000)