Paper presented at the 22nd Asian Conference on Remote Sensing, 5 - 9 November 2001, Singapore.

Semi-Automatic Road Extraction Algorithm from IKONOS Images Using Template Matching

Seung-Ran Park, Taejung Kim
Satellite Technology Research Center, KAIST, 373-1 , Yusung-gu, Taejon, 305-701, ROK
Tel : (82)-42-869-8626 Fax : (82)-42-861-0064
E-mail : {srpark,tjkim}@satrec.kaist.ac.kr, URL:http://satrec.kaist.ac.kr

KEY WORDS: road extraction, template matching, high resolution satellite image

ABSTRACT : This paper describes the development of a semi-automatic road extraction algorithm. The algorithm works as follows: First, a user provides an initial road seed per each road segment in the image. Next, orientation of the road seed is calculated automatically by applying Burns line extraction algorithm. A road template is then formed around the road seed and a road segment is extracted automatically through template matching. For template matching, an adaptive least square matching algorithm was used. This algorithm puts weights on road central line parts and postulates that the relationship between template and target windows can be modeled by similarity transformation. This algorithm also assumes that template and target windows have only small differences in brightness values. A 1m resolution IKONOS image over Seoul area was used for this algorithm test. The algorithm extracted road central lines in any orientation and with moderate curvature successfully, after road seeds were given from a user. Current limitations are that the algorithm may not work on the road cast by shadow and that a user must select valid road seeds on road central lines since the algorithm itself can not judge the validity of input seeds. These limitations are currently being examined. The contribution of this paper is that it showed template matching, instead of the well known "snake", could be used for road extraction.

1. INTRODUCTION

Since the IKONOS has been launched successfully, spaceborne images became available for research and applications where high resolution is required. In high resolution images, identification and/or understanding man-made objects are important. Especially, road network is one of the most important features in GIS layer. Techniques to extract this feature from an urban scene have numerous applications in urban mapping, urban planning, and Geo-Information Engineering.

Until now, various road extraction methods have been proposed. Jedynak et. al.[2] played twenty questions for tracking roads in SPOT images. They applied several tests to remove uncertainties. Their method was applied very well on low-resolution satellite images. Gruen et. al.[3] developed linear feature extraction method using snakes. They combined the characteristics of snakes and Adaptive Least Squares Correlation method. The snakes worked as guides for ALSC, but the method might need large computation time on high-resolution images because of its linear systems.

This paper proposes a road extract algorithm that analyzes high resolution (1m) satellite image and automatically extract road network by using user's initial input point. This algorithm forms a road template around the user input point and extract road centers by template matching. For matching, this algorithm puts weights on road centers and postulates that the relationship between template and target windows can be modeled by similarity transformation. This algorithm also assumes that template and target windows have only small differences in brightness values.

A 1m resolution IKONOS image over Seoul area was used to test this algorithm. The algorithm extracted road central lines in any orientation and with moderate curvature successfully, after road seeds were given from a user.

Current limitations are that the algorithm may not work on the road cast by shadow and that a user must select valid road seeds on road central lines since the algorithm itself can not judge the validity of input seeds. These limitations are currently being examined.

The contribution of this paper is that it showed template matching, instead of the well known "snake", could be used for road extraction. And this algorithm can be utilized in analyzing high resolution satellite images.

In the following section, we explain the principles of this algorithm. Then we present road extraction results from this algorithm with an IKONOS image over Seoul area.

2. ROAD EXTRACTION

Semi-automatic road extraction process in this paper is shown in Figure 1.

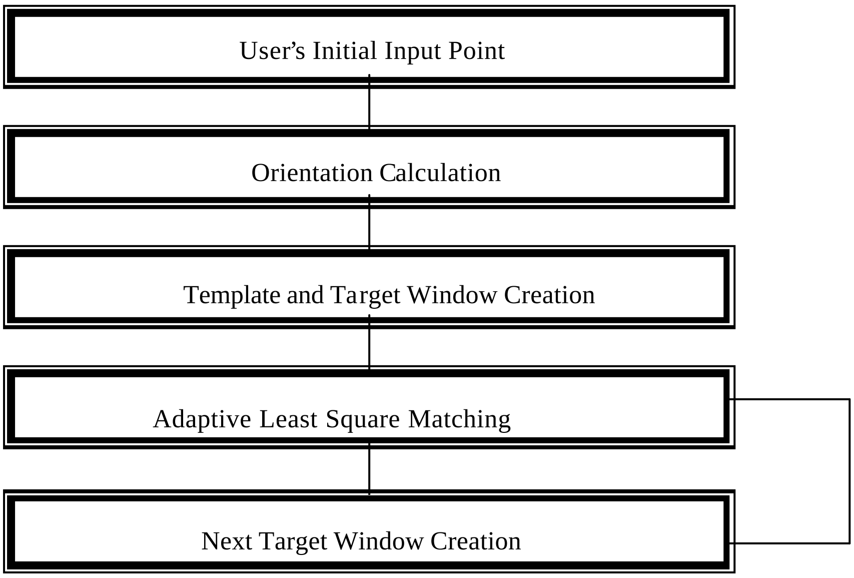

Figure 1. Road extraction flowchart

If a user clicks the initial input point on a road center, line orientation on that point(i.e. orientation of the road center) is calculated automatically by applying Burns line extraction algorithm[4]. Next, a template window is formed around the initial input point. Then template matching based on adaptive least square matching algorithm is performed. After finding a road center adjacent to the initial input point, matching continues along the road orientation. Through iterative adaptive least square matching process, all center points of a road segment are extracted.

2.1 User's Initial Input Point

Because the current algorithm proposed here can not judge by itself the exact location of a road center, a user must click a valid initial input point. The proposed algorithm will be triggered by any user input point, i.e, this algorithm will trace any structure similar to the initial input point through template matching. Therefore, for road extraction, a user must click valid road center. Also user must be aware that this algorithm may not work on the road cast by shadow and hence must avoid clicking such points.

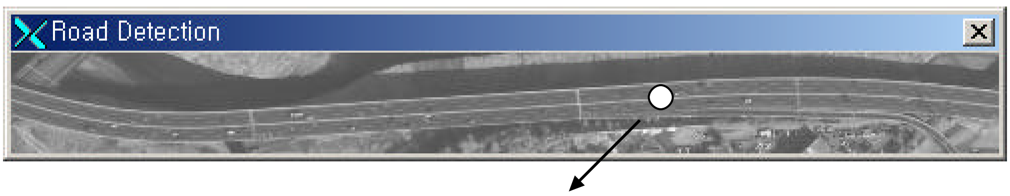

Figure 2. User initial input point

2.2 Orientation Calculation

Once user's initial input point is provided, orientation on the initial input point is calculated. If user has clicked a valid point on road center, the orientation on that point tells road direction. Therefore, we can use the orientation value for template formation and guideline for match direction later on.

For orientation calculation, Burns algorithm of line extraction is applied. Burns algorithm defines a “line-support region” by grouping pixels with similar edge orientation and magnitude together and extract a line from each line support region by planar fitting.

The algorithm proposed here extracts lines around the initial input point based on Burns algorithm. Then orientations of such lines are calculated. The orientation of the line whose length is longest will be regarded as the initial input point’s orientation.

2.3 Template Creation

A template window is formed centered on initial input point and rotated to the orientation of initial input point. Using this template window, road tracking is performed. A target window is also formed by shifting the template window along the direction of road center. This target window will be used as initial approximation for adaptive least square matching.

Figure 3. Template and target window creation

2.4 Adaptive Least Square Matching

A next road center adjacent to the initial point shall be formed through template matching. From the initial target window defined above, true target window is searched for through adaptive least square matching. We assumed the relationship between search window and target window could be modeled by similarity transformation[1]. Our matching algorithm is adaptive in the sense that it can rotate the target window and translate the target window perpendicular to road direction.

This adaptive least square matching can track roads in any orientation given a valid user input on road center. Higher weights were given to central parts of the template (see Figure 4) so that small lines other than center line or bright vehicles could not have significant effect for matching.

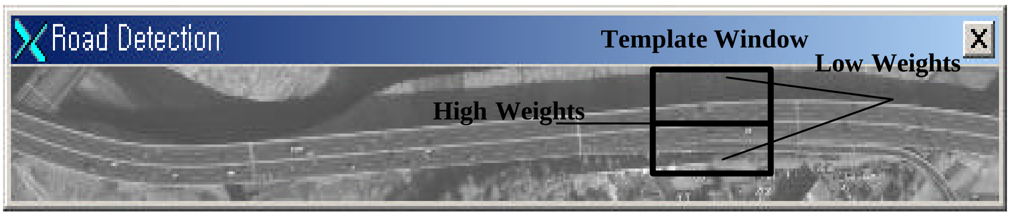

Figure 4. Weights on central line parts of road

2.5 Next Target Window Creation

Through adaptive least square matching, we can obtain a target window whose location and orientation best matches the template window. We can regard the center of this target window as another road center adjacent to the initial point. The orientation of the matched target window is not necessarily the same as that of the template. Hence road direction on the matched window shall be the target window's orientation, not the template's.

In order to track road, we need to apply adaptive least square matching repetitively along with road center lines. Therefore, an initial target window for next match shall be formed by shifting further the target window obtained from the previous match in road direction. As explained above, the road direction in this case shall be the orientation of the matched target window, not that of the template. The template window shall remain unchanged through out the series of adaptive least square matching.

Through search repetitive adaptive least square matching, a series of point on road center lines can be extracted. Matching iteration stops when more than two consecutive matching fail, under the current implementation.

3. ALGORITHM EXPERIMENTS

The algorithm proposed here was tested from IKONOS image over Seoul area. The image contains many different types of road, highways, interchanges, main streets with many vehicles, small roads between houses, etc., at various orientations. There are many high buildings along main streets, which cast shadows (Figure 6, and 7).

Figure 5 shows the results of road extraction. The white array indicates the initial input point user provided. From this point, the adaptive least square matching was carried out leftwards and rightwards. The dark small crosses are the centers of matched target windows, which could be considered as road centers. Figure 5 shows that the road extraction algorithm proposed here has successfully traced road center through the iterative template matching.

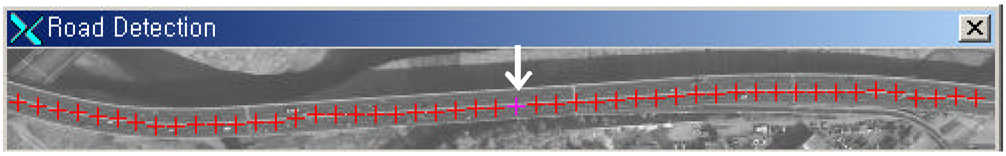

Figure 5. Extracted road segment

Figure 6 shows the results of road extraction over more wide area. In the figure, there are several road segments at various orientations. There are also interchanges where road orientation changes abruptly and intersections where there are no continuing road center lines.

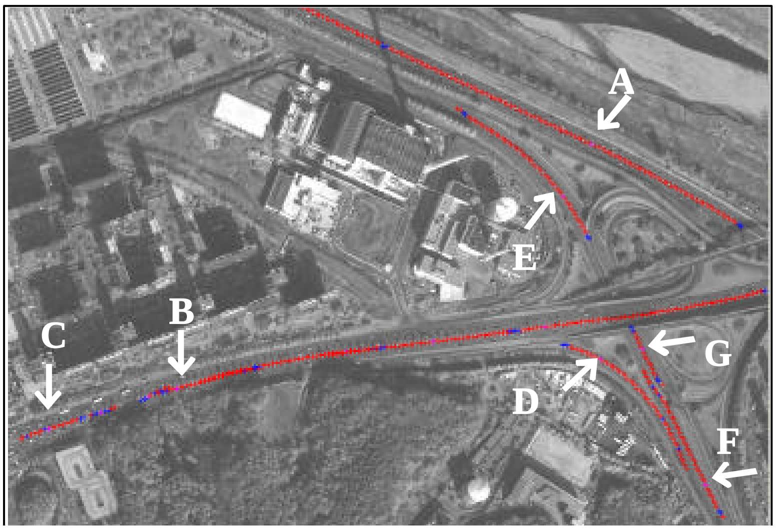

Figure 6. Various road extraction

White arrows are, again, the user initial input points. With the point A, a complete road segment was extracted successfully. This segment "A" stopped only when it met the boundary of the image at the upper end and shadow at the lower end. Segment "B" shows similar results. The initial point C was given to show that when road tracking halted due to shadow, a new initial point at the other side of the segment could be given to continue to track the remaining part of the segment. (And the segment "C" proved). The initial point D was given on the road whose orientation changes abruptly. The segment "D" shows that the algorithm proposed here could successfully extracted such road. It only stopped tracking because center line met a side line of other road (upper end) and because center line vanished. Segment "E" shows similar results. The segment "F", however, was misled by complicated structures of the interchange. In that case, a new input point G could be given to recover correct road segment.

4. CONCLUSIONS

In this paper, we showed semi-automatic road extraction algorithm. From the high resolution satellite image, road segments are extracted by using template matching based on adaptive least square matching algorithm. The experiments with an IKONOS image showed that from a few user input points road segments of highways and major streets were extracted automatically.

One of the major contributions of this paper is that it demonstrated the feasibility of template matching for boundary delineation. So far, many approaches with the "snake" have been proposed for road tracking. Compared to the "snake" template matching may give more flexibility on the road constraints. Our algorithm based on the template matching can track river boundaries, edge of stadiums, etc.. with user input points on such lines. When we clicked a point, which was not on the center line or side line of a road but lied between the center and side lines of the road, our algorithm tracked the road segments keeping the distance between the center and side lines same(see Figure 7). With the "Snake" if one wants to track different features (for example, water boundary), one needs to define a new energy minimization functions accordingly.

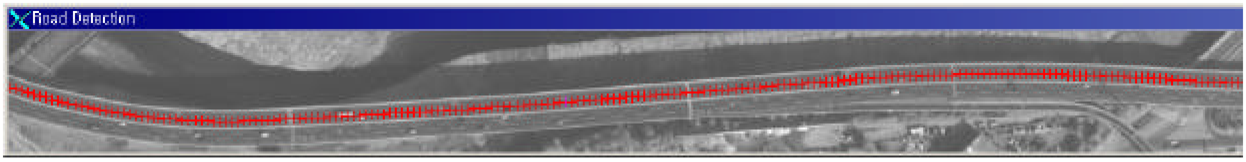

Figure 7. Road tracking along by a point between center and side lines

Current limitations are that the algorithm may not work on the road cast by shadow and that a user must select valid road seeds on road central lines since the algorithm itself can not judge the validity of input seeds. These limitations are currently being examined.

5. REFERENCES

1. Armin Gruen, Peggy Agouris, Haihong Li, 1995. Automatic Extraction of Man-Made Objects from Aerial and Space Images, Monte Verita, pp.83-94.
2. B. Jedynak et. al., 1995. Tracking Roads in Satellite Images by Playing Twenty Questions, Automatic Extraction of Man-made Objects from Aerial and Space Images, vol. 1, pp. 243-254.
3. A.W. Gruen et.al., 1997. Linear Feature Extraction with 3-D LSB-Snakes, Automatic Extraction of Man-made Objects from Aerial and Space Images, vol. 2, pp. 287-298.
4. J.B. Burns, A.R. Hanson and E.M. Riseman, 1986. Extracting Straight Lines, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 8, pp.425-445.